HAL
open science

Eléments méthodologiques pour le développement des systèmes décisionnels dans un contexte de réutilisation

Annoni Estella

To cite this version:

Annoni Estella. Eléments méthodologiques pour le développement des systèmes décisionnels dans un contexte de réutilisation. Informatique [cs]. Université Paul Sabatier - Toulouse III, 2007. Français. NNT : . tel-00549422

HAL Id: tel-00549422

https://theses.hal.science/tel-00549422v1

Submitted on 26 Dec 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.

THÈSE

Présentée devant

l'Université Toulouse 1

en vue de l'obtention du

Doctorat de l'Université Toulouse 1

Spécialité : **INFORMATIQUE**

*P*ar

Estella ANNONI

Eléments méthodologiques pour le développement des systèmes décisionnels dans un contexte de réutilisation

---

Soutenue le 16 juillet 2007, devant le jury composé de :

M. P. BERREBBI	Président Directeur Général du Groupe I-D6	Invité
M. O. BOUSSAID	Maître de Conférences - HDR à l'Université Lyon II	Rapporteur
Mme C. CAUVET	Professeur à l'Université Aix-Marseille III	Rapporteur
M. C. CHRISMENT	Professeur à l'Université Paul Sabatier, Toulouse III	Examinateur
Mme. E. METAIS	Professeur au Conservatoire National des Arts et Métiers Paris	Examinateur
M. F. RAVAT	Maître de Conférences à l'Université Toulouse I	Examinateur
M. O. TESTE	Maître de Conférences à l'Université Paul Sabatier, Toulouse III	Invité
M. G. ZURFLUH	Professeur à l'Université Toulouse I	Directeur de thèse

INSTITUT DE RECHERCHE EN INFORMATIQUE DE TOULOUSE - UMR 5505
Centre National de la Recherche Scientifique - Institut National Polytechnique - Université Paul Sabatier - Université Toulouse I
118 Route de Narbonne, 31062 Toulouse Cedex 09. Tel : 05.61.55.67.65

[A mon père qui nous a quittés]

Comprendre c'est avant tout unifier. Vouloir, c'est susciter les paradoxes
Albert Camus, Le mythe de Sisyphe, p.36, Idées n°1.

Tout d'abord, je voudrais adresser ma profonde reconnaissance à Messieurs Claude Chrisment et Gilles Zurfluh pour m'avoir accueillie au sein de leur équipe. Ils m'ont assurée une compréhension et un soutien de tous les instants, et surtout lors du départ de mon père au cours de mon stage de DEA.

Je tiens à remercier très sincèrement Madame Corine Cauvet, Professeur à l'Université d'Aix Marseille III, pour l'honneur qu'elle me fait en acceptant d'être rapporteur de mon travail et pour ses remarques qui ont permis d'améliorer la qualité de cette thèse. Je tiens à lui exprimer ma reconnaissance pour les orientations relatives au domaine de la réutilisation qu'elle m'a judicieusement conseillée et pour l'honneur qu'elle me fait en participant à ce jury.

Je tiens à remercier très sincèrement Madame Elisabeth Metais, Professeur au Conservatoire National des Arts et Métiers Paris, pour l'honneur qu'elle me fait en acceptant d'être membre de mon jury de thèse.

Je tiens à remercier très sincèrement Monsieur Omar Boussaïd, Maître de Conférences - HDR à l'Université Lyon II, pour l'honneur qu'il me fait en acceptant d'être rapporteur de mon travail et pour ses remarques qui ont permis d'améliorer la qualité de cette thèse. Je tiens à lui exprimer ma reconnaissance pour ses perspectives liées aux entrepôts de contenu XML et pour l'honneur qu'il me fait en participant à ce jury.

Je tiens à exprimer ma profonde gratitude à Monsieur Gilles Zurfluh, Professeur à l'Université Toulouse I pour sa disponibilité, sa rigueur dans le travail et dans l'expression de ce dernier. Il m'a transmis le goût du travail de recherche. J'ai apprécié son écoute attentive et conciliante face aux difficultés de planning d'une thèse CIFRE. Il m'a laissée une grande liberté de travail et il m'a éclairée de ses conseils, de ses suggestions et de son savoir-faire.

Je tiens à exprimer ma profonde gratitude à Monsieur Claude Chrisment, Professeur à l'Université Toulouse III, pour avoir suscité dès ma maîtrise le désir d'entreprendre une thèse. Ses remarques, ses suggestions et les discussions fructueuses que nous avons eues m'ont permis d'améliorer la qualité de cette thèse.

Je tiens à remercier chaleureusement Monsieur Franck Ravat, Maître de Conférences à l'Université Toulouse I, pour le suivi et l'encadrement continu de cette thèse ainsi que pour son aide précieuse. Je lui exprime ici ma reconnaissance pour sa présence, sa patience, sa contribution dans la réalisation de cette thèse et ses efforts considérables pour assurer une activité de recherche intense.

Je tiens à remercier chaleureusement Monsieur Olivier Teste, Maître de Conférences à l'Université Toulouse III, pour le suivi et l'encadrement continu de cette

thèse ainsi que ses discussions qui ont accru ma détermination et ma hardiesse pour faire aboutir mes travaux dans ce contexte CIFRE. Je lui exprime ici ma toute reconnaissance pour son accompagnement, ses judicieux conseils et la motivation qu'il m'a transmise pour une activité de recherche intense.

Je tiens à remercier sincèrement le Groupe I-D6. Cette thèse n'aurait pas pu être menée sans le précieux concours de Monsieur Pierre Berrebbi, PDG du groupe, de Monsieur Cyril Cohen-Solal, DG du groupe, ainsi que de Monsieur Eric Fernandez, Responsable des ressources humaines de l'agence de Toulouse. Je tiens à remercier l'ensemble de mes collègues et particulièrement Assitan, Sylvie, Armelle et Gilles pour les transferts de compétences. Une collaboration au sein d'une société de services ne peut être fructueuse sans la contribution active de ses organisations clientes. Je tiens donc à remercier Mesdames Min-Ying Thang et Elisabeth Schmit des laboratoires MSD-Chibret sans qui cette thèse aurait pu difficilement aboutir.

Je tiens à remercier sincèrement Madame Geneviève Pujolle, Maître de Conférences à l'Université Toulouse I, pour l'intérêt qu'elle a porté à mes travaux en les examinant, pour ses conseils et ses encouragements au cours de ces mois de rédaction.

Je voudrais accorder une mention spéciale aux Professeurs Mohand Boughanem et Bernard Dousset et pour leurs nombreux conseils et recommandations.

Je tiens à remercier l'ensemble des membres de l'équipe SIG et particulièrement Madame Karen Pinel-Sauvagnat, Maître de Conférences à l'Université Toulouse I, pour son soutien, son aide et son amitié au cours de ces années de thèse et particulièrement durant les mois de rédaction.

Je tiens à remercier le Professeur Pierre Bazex pour ses nombreux conseils relatifs à mon orientation. Je remercie aussi chaleureusement les Professeurs Jean-Paul Bashoun et Daniel Hagimont pour leurs sollicitations et soutien amicaux.

Je tiens à remercier sincèrement Madame Christie Ezeife, Professeur à l'Université de Windsor à Ontario au Canada, de m'accueillir en post-doctorat dans son équipe de recherche et pour la confiance qu'elle m'accorde.

Je tiens également à exprimer ma reconnaissance à Madame Sylvie Gaucher-Cazalis, Monsieur Andreï Doncescu et Monsieur Jacky Desachy, membres présents et passés de l'Université des Antilles et de la Guyane qui m'ont aidée et appuyée à poursuivre mes études à l'Université Toulouse III.

Je tiens à remercier l'ensemble des doctorants et post-doctorants présents et passés avec qui j'ai partagé des moments agréables, particulièrement mes collègues de bureau, Nathalie, Désiré et Bachelin. La liste serait longue si je les citais tous !!!

Je tiens à exprimer du fond du coeur ma gratitude et mes remerciements à ma courageuse et forte mère pour son soutien et sa compréhension à toute épreuve. Je souhaite également remercier l'ensemble de mes frères et soeurs qui m'ont supportée et aidée à atteindre l'objectif que je m'étais fixée. Je remercie ma belle-famille et mes amis des Antilles et de l'Hexagone qui m'ont appuyée et réconfortée. Je remercie chaleureusement mes amis correcteurs Domi, Carole et Fidel.

Enfin, je souhaite remercier particulièrement trois êtres qui me sont chers. D'une part, je remercie ma petite soeur, Carine, et mon petit frère, Jimmy, pour leur motivation et leur détermination. D'autre part, je remercie mon tendre et intrépide fiancé, Laurent, pour sa patience et son originalité.

Face au besoin croissant de réactivité et de compétitivité des organisations, les développements de systèmes qui facilitent la prise de décision se sont démocratisés. Ces systèmes sont des « systèmes d'information décisionnels » (SID). Cependant, 80% des projets décisionnels ne parviennent pas à satisfaire les besoins des utilisateurs et 40% d'entre eux échouent à faciliter la prise de décision[1]. Les récentes propositions de méthodes liées aux SID définissent leur schéma à partir des besoins des utilisateurs et des systèmes sources. Néanmoins, elles ne permettent pas de représenter toutes les spécificités des SID. De plus, elles reposent sur des modèles spécifiques ne représentant que les données, l'aspect dynamique des SID n'étant que peu abordé. Aucun de ces modèles n'est donc reconnu par les chercheurs et les industriels.

Dans un premier temps, nous proposons une méthode qui prend en compte les spécificités du SID ainsi que celles des besoins de ces acteurs. Nous analysons l'aspect statique et l'aspect dynamique du SID définis à partir de structures spécifiant les besoins et nous les représentons *via* un modèle proche de la vision multidimensionnelle des données par les utilisateurs. A partir de cette formalisation, nous guidons suivant un processus automatique le choix de l'architecture du SID adaptée à un projet. Cette architecture repose sur plusieurs types de modules décisionnels dont certains sont multidimensionnels. Nous proposons donc un modèle multidimensionnel généralisé à partir des propositions existantes, visant par la même de répondre à ce manque de modèle standard.

Dans un second temps, nous proposons un catalogue de patrons qui capitalise notre méthode de développement. Ce catalogue favorise la réutilisation systématique de notre méthode : d'une part, la formalisation du contexte d'un patron facilite sa recherche car les conditions dans lesquelles il est utilisable et celles dans lesquelles il requiert un autre patron sont formellement spécifiées, d'autre part, la gestion intégrée de la documentation dans sa représentation contribue à améliorer la fiabilité des systèmes développés.

Enfin, pour faciliter le développement rapide de SID par réutilisation de nos patrons, nous avons développé un outil, appelé eBIPAD (Electronic Business Intelligence Patterns for Analysis and Design), de gestion de ces patrons avec des fonctionnalités d'organisation et de réutilisation. Cet outil est dédié aux administrateurs des patrons et aux concepteurs décisionnels. Nous avons pu valider nos propositions au cours des missions que nous avons menées au sein de la société, I-D6[2], spécialisée dans le décisionnel, et ce, dans le cadre d'une collaboration CIFRE.

Mots clés : système d'information décisionnel, méthode, modèle, traitement, analyse, conception.collaboration

[1] [Schiefer et al. 2002]

[2] I-D6 est la société collaboratrice dans le cadre de la thèse CIFRE (Convention Industrielle de Formation par la Recherche). http://www.i-d6.com.

The development of software facilitating decision taking is becoming more frequent, due to the growing need of reactivity and competitiveness within companies. Such software is called Decision Support Systems (DSS). However, 80% of decision making projects fail to satisfy user requirements, and 40% fail to help decision making[3]. The recent works of methods bound to DSS define their schemes from the user requirements and the source systems. However, they can not represent all specificities of the DSS. Moreover, they rely on specific models representing only data, the dynamic aspect of DSS being largely ignored. Thus, none of these models is accepted neither by the researchers nor the practitioners.

In a first time, we will propose a method which takes into account the DSS specificities, including those related to actor requirements. We will analyse the static and dynamic aspects of DSS as defined by structures specifying the requirements. Those will be represented via a model close to the multidimensional vision of the data by users. From this modelling, we will automatically choose the DSS architecture adapted to a project. This architecture relies on many kinds of decisional modules, some of which are multidimensional. Therefore, we propose a multidimensional model generalising the existing models, thus also aiming to fix the lack of an accepted standard model.

In a second time, we propose a catalogue patterns which capitalize upon our development method. This catalogue helps a systematic later reuse of our method. In one part, the pattern context formalisation helps searching for such a pattern since the conditions of its usability and those upon which another pattern is required are formally specified. In a second part, the integrated documentation management into pattern representation helps improving the reliability of the systems being developed. Then, to facilitate the rapid development of DSS by reusing our patterns, we have developed a tool called eBIPAD (Electronic Business Intelligence Patterns for Analysis and Design) to manage, organize and reuse these patterns. This tool is directed to patterns administrators and to the decisional designers. We have been able to validate our proposals through consultancy missions, realized within the ID6[4] company which is specialized in decision taking, through a french corporate PhD grant (CIFRE).

Keyworks : decision support system, method, model, process, analysis, design.

[3] [Schiefer et al. 2002]

[4] I-D6 is a company with which a collaboration have been led. http://www.i-d6.com.

Table des matières

1 Introduction **1**

1.1 Contexte général 2

1.2 Positionnement des SID par rapport aux SI 3

1.2.1 Systèmes d'information (SI) 3

1.2.2 Systèmes d'information décisionnels (SID) 5

1.3 Intérêts des SID 8

1.4 Motivations 8

1.5 Problématiques et objectifs 11

1.6 Organisation de cette thèse 14

I Cadre des travaux **16**

2 Etat de l'art **17**

2.1 Introduction 19

2.2 Les modèles multidimensionnels des SID 19

2.2.1 Les spécificités des SID 21

2.2.2 Niveau conceptuel 27

2.2.3 Bilan et positionnement par rapport aux modèles multidimensionnels 34

2.3 Les méthodes de développement des SID 34

2.3.1 Démarche descendante 36

2.3.2 Démarche ascendante 37

2.3.3 Démarche mixte 40

2.3.4 Bilan et positionnement par rapport aux méthodes de développement 43

2.4 Les méthodes d'analyse des SID 45

2.4.1 Modèles existants 46

2.4.2 Requêtes 48

2.4.3 Modèles de buts 49

2.4.4 Bilan et positionnement par rapport aux méthodes d'analyse . 51

2.5 La capitalisation et la réutilisation dans les SID 54

2.5.1 Les composants réutilisables 54

2.5.2 Ingénierie des systèmes d'information 56

2.5.3 Ingénierie des SID 56
2.5.4 Bilan et positionnement par rapport à la capitalisation et à la réutilisation 58
2.6 Synthèse de l'existant 58
2.6.1 Limites des modèles et méthodes de développement 59
2.6.2 Limites des méthodes d'analyse 60
2.6.3 Limites face au développement rapide et au contexte de réutilisation 60
2.7 Conclusion 61

II Une méthode de développement des SID dans un contexte de réutilisation 62

3 Principes généraux 63
3.1 Introduction 64
3.2 Contextes industriel/scientifique et objectifs 64
3.3 Idées principales de nos propositions 67
3.3.1 Propositions liées à la phase d'analyse 67
3.3.2 Propositions liées à la phase de conception 67
3.3.3 Propositions liées au développement rapide et au contexte de réutilisation 68
3.4 Classification des acteurs et des besoins 68
3.5 Démarche du Trident Décisionnel 70
3.5.1 Ajout de la branche stratégique 71
3.5.2 Modification de l'ordonnancement des tâches 73
3.6 Cycle itératif incrémental et prototypage 73
3.6.1 Description des itérations 74
3.6.2 Nombre variable d'itérations 75
3.7 Conclusion 76

4 Méthode d'analyse des besoins des SID 77
4.1 Introduction 79
4.1.1 Problématique 79
4.1.2 Proposition 80
4.2 Modèles et démarches communs 82
4.2.1 Modèle : tableau croisé 83
4.2.2 Modèle : graphe de propriétés 86
4.2.3 Modèle : diagramme décisionnel 90
4.2.4 Démarche : collecte des besoins utilisateurs 96
4.2.5 Démarche : annotation d'un graphe de propriétés 96
4.2.6 Démarche : formalisation des besoins utilisateurs 97
4.3 Analyse des besoins tactiques 100

4.3.1 Collecte 100
4.3.2 Formalisation 102
4.4 Analyse des besoins stratégiques 106
4.4.1 Collecte 106
4.4.2 Formalisation 107
4.5 Analyse des besoins systèmes 109
4.5.1 Collecte 109
4.5.2 Formalisation 110
4.6 Confrontation des besoins du SID 118
4.7 Conclusion 125

5 Méthode de conception des SID 127
5.1 Introduction 130
5.1.1 Problématique 130
5.1.2 Proposition 131
5.2 Choix de l'architecture décisionnelle 132
5.2.1 Typologie des architectures décisionnelles valides 134
5.2.2 Critères de quantification 135
5.2.3 Automatisation 136
5.3 Modèle multidimensionnel de données et de traitements (MMDT) . . 142
5.3.1 Structure complexe des dimensions et des paramètres : 143
5.3.2 Structure complexe des faits et des mesures : 149
5.3.3 Cohérence de l'interrogation : 155
5.4 Impacts des traitements 157
5.4.1 Impacts des traitements de dérivation des données 158
5.4.2 Impacts des traitements de préparation des données 159
5.4.3 Les traitements par concept multidimensionnel 162
5.5 Conception des modules décisionnels 163
5.5.1 Passage du diagramme décisionnel à notre modèle MMDT . . 164
5.5.2 Passage du diagramme décisionnel au modèle E-A 170
5.6 Conclusion 173

6 Capitalisation dans un catalogue de patrons 176
6.1 Introduction 177
6.1.1 Problématique 178
6.1.2 Proposition 178
6.2 Présentation du catalogue 179
6.2.1 Découpage du catalogue 179
6.2.2 Formalisme P-SIGMA étendu 181
6.3 Ingénierie pour la réutilisation 187
6.3.1 Patrons d'analyse 189
6.3.2 Patron de conception 212

6.4 Ingénierie par la réutilisation : mise en application de notre catalogue 221
6.4.1 Réutilisation des patrons au cours de phase d'analyse 223
6.4.2 Réutilisation des patrons au cours de phase de conception . . 238
6.5 Conclusion 241

III Bilan industriel et développements 243

7 Applications et développements 244
7.1 Introduction 245
7.2 Bilan des projets décisionnels dans l'industrie 245
7.2.1 Montée en compétence sur l'analyse des besoins 246
7.2.2 Montée en compétence sur la conception du SID 247
7.3 Démarche qualité suivant la Roue de Deming 248
7.3.1 Présentation de la Roue de Deming 248
7.3.2 Mise en application de la Roue de Deming 249
7.4 Développement de l'outil eBIPAD 251
7.4.1 Généralités de l'outil 251
7.4.2 Fonctionnalités générales 253
7.4.3 Fonctionnalités pour l'ingénieur de patrons 255
7.4.4 Fonctionnalités pour l'ingénieur d'applications 262
7.4.5 Fonctionnalités pour l'administrateur système 269
7.4.6 Base de patrons de eBIPAD 269
7.5 Evaluation de nos propositions 275
7.5.1 Synopsis de nos propositions 275
7.5.2 Avantages de nos propositions 277
7.5.3 Inconvénients rencontrés de nos propositions 279
7.6 Conclusion 280

8 Conclusion et perspectives 282
8.1 Conclusion 283
8.2 Perspectives 285
8.2.1 Approfondissement et continuité de la recherche réalisée . . . 285
8.2.2 Elargissement du domaine de recherche 286

A Diagrammes d'activités du choix de l'architecture 287

Chapitre 1

Introduction

Table des matières

1.1 Contexte général **2**
1.2 Positionnement des SID par rapport aux SI **3**
1.2.1 Systèmes d'information (SI) 3
1.2.2 Systèmes d'information décisionnels (SID) 5
1.3 Intérêts des SID **8**
1.4 Motivations **8**
1.5 Problématiques et objectifs **11**
1.6 Organisation de cette thèse **14**

1.1 Contexte général

La mondialisation du commerce et particulièrement des services impose une prompte réactivité des organisations. Ce phénomène a induit une croissance du marché des outils liés à l'aide à la prise de décision, soit une hausse annuelle annoncée par le cabinet IDC supérieure à 8 % depuis 2005 et ce jusqu'en 2008 [IDC 2004a]. Par exemple, plus de deux tiers des organismes publics seront dotés d'une application décisionnelle de pilotage ou de gestion de la performance à l'horizon 2008, selon l'enquête « Bilan et Perspectives IT dans l'Administration Française 2006-2008 » publiée par le cabinet Markess International. Plus précisément, les administrations publiques centrales et locales consacreront 6% de leur budget à la mise en place de tels outils [Janiaux 2006].

Le pilotage de la performance représente un moteur de la stratégie décisionnelle comme le confirme une autre étude réalisée par IDG[1] [IDG 2005]. En effet, les 140 organisations contactées affirment que c'est le moteur le plus important de la stratégie décisionnelle à 69%. Pour mener à bien ce pilotage, les données échangées et nécessaires aux organisations sont de plus en plus nombreuses. Ces données proviennent de sources hétérogènes [Gardarin 1999] qu'il convient d'homogénéiser. Malgré l'intégration au sein des organisations d'outils ou de progiciels de gestion intégrée des applications métier, les décideurs ne peuvent pas exploiter seuls et pleinement ces données. De plus, les décideurs sont souvent contraints par l'important taux de charge des services informatiques. Les organisations ont donc besoin d'un système plus complet qui homogénéise, transforme et résume les données des sources afin qu'elles soient compréhensibles et maîtrisables par les décideurs. Elles ont aussi besoin d'un système qui donne plus d'autonomie aux décideurs.

Ainsi, les services informatiques ou les sociétés de services informatiques se retrouvent face à de nombreuses demandes de mise en oeuvre de ces systèmes appelés **systèmes d'information décisionnels**. Tout au long de cette thèse, nous désignons le système d'information décisionnel par le sigle SID.

En raison de la notoriété des systèmes d'information pour la gestion des données échangées au cours des transactions quotidiennes d'une organisation, beaucoup de travaux se sont inspirés des modèles et des méthodes associées.

Ainsi, dans la section 1.2, nous positionnons les systèmes d'information décisionnels par rapport aux systèmes d'information (SI). Puis, dans la section 1.3, nous présentons les intérêts des SID pour les organisations. Dans la section 1.4, nous explicitons les motivations qui ont engagé nos travaux de recherche. Dans la section 1.5, nous présentons les problématiques et les objectifs de nos travaux. Enfin, dans la section 1.6, nous présentons l'organisation de cette thèse.

[1]IDG : Le Groupe IDG Communications est présent en France avec de nombreuses publications spécialisées sur le marché de l'informatique et des technologies associées : Le Monde Informatique, CIO, CSO, Réseaux Télécoms, Distributique, DigitalWorld.

1.2 Positionnement des SID par rapport aux SI

Les premiers **systèmes de gestion** des données développés dans les années 1960 pour gérer les données exploitées par les ordinateurs ont été complétés par de nombreuses briques afin de gérer les données et les traitements liés aux domaines d'activité des organisations. Initialement, ces systèmes, aussi appelés **MIS : Management Information System**, assurent la persistance, la cohérence et l'interrogation des données manipulées.

1.2.1 Systèmes d'information (SI)

Nous considérons la définition du système d'information (SI), une des briques du système de gestion, énoncée par celui qui en a la paternité [Le Moigne 1977]. Par ailleurs, afin d'avoir une vue d'ensemble des problématiques liées aux systèmes d'information, nous nous référons à leur présentation faite par Rolland et Flory [Rolland and Flory 1990]. Ces auteurs y dressent un état de l'art des méthodes de développement de systèmes d'information.

Définition 1.1 (Système d'information) *« Le système d'information est l'ensemble des méthodes et moyens recueillant, contrôlant et distribuant les informations nécessaires à l'exercice de l'activité en tout point de l'organisation. Sa fonction est de produire et de mémoriser les informations, représentation de l'activité du système opérant (système opérationnel), puis de les mettre à disposition du système de décision (système de pilotage) » [Le Moigne 1977].*

Les systèmes d'information implantés dans les organisations permettent de gérer les transactions au quotidien par domaine d'activité. Ils ne préparent pas les données pour la prise de décision. Ces systèmes ne sont pas adaptés pour faire des analyses complexes de données [Codd et al. 1993]. Pour assurer une plus grande réactivité et une plus grande compétitivité, les décideurs requièrent des systèmes qui facilitent leur processus de prise de décision. C'est la raison pour laquelle, nous nous intéressons à la partie du système d'information qui sert de support à l'aide à la prise de décision.

Ainsi, à partir du positionnement du système d'information dans la pyramide du système de gestion [Le Moigne 1977; Rolland and Flory 1990], nous le décomposons dans la figure 1.1.

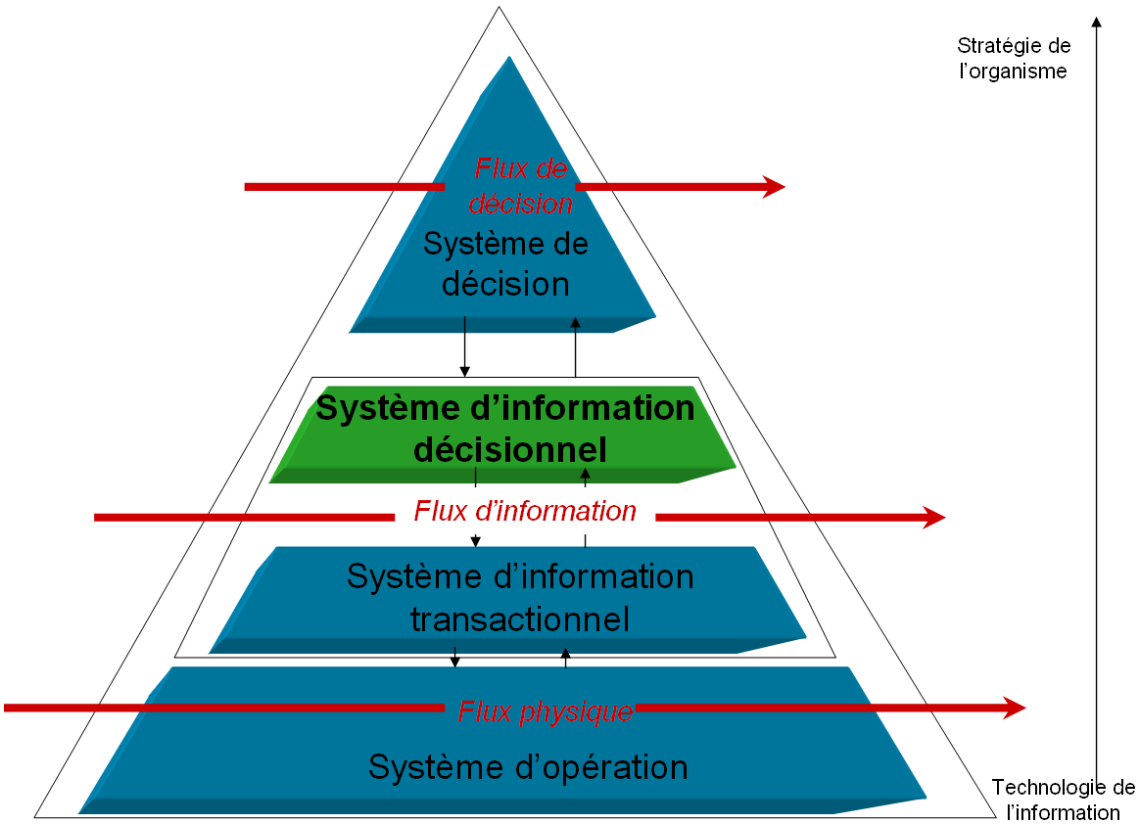


FIG. 1.1 – Les briques de la pyramide du système de gestion

Nous définissons les briques comme suit :

- le système d'opération (ou le système opérant) est constitué d'un ensemble d'humains, de matériels et de logiciels qui permettent d'assurer les traitements de la fonction première de l'organisation. Cette brique initiale assure la gestion technique de l'organisation,
- le système d'information transactionnel est composé de l'ensemble des composants et applications métiers de l'organisation qui gèrent les données au quotidien. Il assure la gestion de toutes les transactions qui ont lieu au sein de l'organisation. Cette deuxième brique intègre la dimension métier de l'organisation. Le système d'information est souvent confondu ou réduit à cette brique. Donc, dans la suite de cette thèse, nous référons à cette brique quand nous désignons les **systèmes d'information classiques**,
- le système d'information décisionnel (SID) est un ensemble organisé d'information, facilement accessible, qui est adapté pour la prise de décision. Il est obtenu à partir de traitements sur les données de systèmes d'information transactionnels internes ou externes à l'organisation. Cette troisième brique prend en compte la dimension stratégique nécessaire à la prise de décision,
- le système de décision est constitué de l'ensemble des moyens humains et matériels permettant d'approfondir l'observation faite depuis le système d'infor-

mation décisionnel. Il comprend les outils de fouille de données, les systèmes experts, ...

Dans la section suivante, nous explicitons la définition du SID car cette brique représente le domaine de nos travaux.

1.2.2 Systèmes d'information décisionnels (SID)

Au début des travaux sur ce domaine, en 1996, le SID a été défini par l'un des pionniers dans ce domaine [Kimball 1996], au travers de l'un de ces modules comme « une copie des données transactionnelles organisées spécialement pour l'interrogation et l'analyse ». Le SID a été progressivement défini comme un système composé de plusieurs modules. Les auteurs [Bouzeghoub et al. 1999] le définissent comme : « une hiérarchie d'espaces de stockage des données comprenant les sources de données jusqu'aux espaces appelés magasins de données contenant des données très agrégées ».

Nous pouvons citer la définition énoncée dans le livre des bases fondamentales des SID [Jarke et al. 2001]. Cette définition caractérise le SID *via* son architecture : « il contient plusieurs couches de données dans lesquelles les données d'une couche sont dérivées à partir d'une couche inférieure. Les sources de données forment la couche la plus base et sont appelées des bases transactionnelles ».

Ces deux définitions mettent en avant les concepts de « source de données » ou de « couche de plus bas niveau », de magasin de données ainsi que celle de « dérivation des données d'une couche de niveau n à partir des données de niveau n-1 ». Nous considérons la définition suivante des SID.

Définition 1.2 (Système d'information décisionnel) *Un système d'information décisionnel, SID, est un système qui réalise la collecte, la transformation des données brutes issues de sources de données et le stockage dans d'autres espaces ainsi que la caractérisation des données résumées en vue de faciliter le processus de prise de décision.*

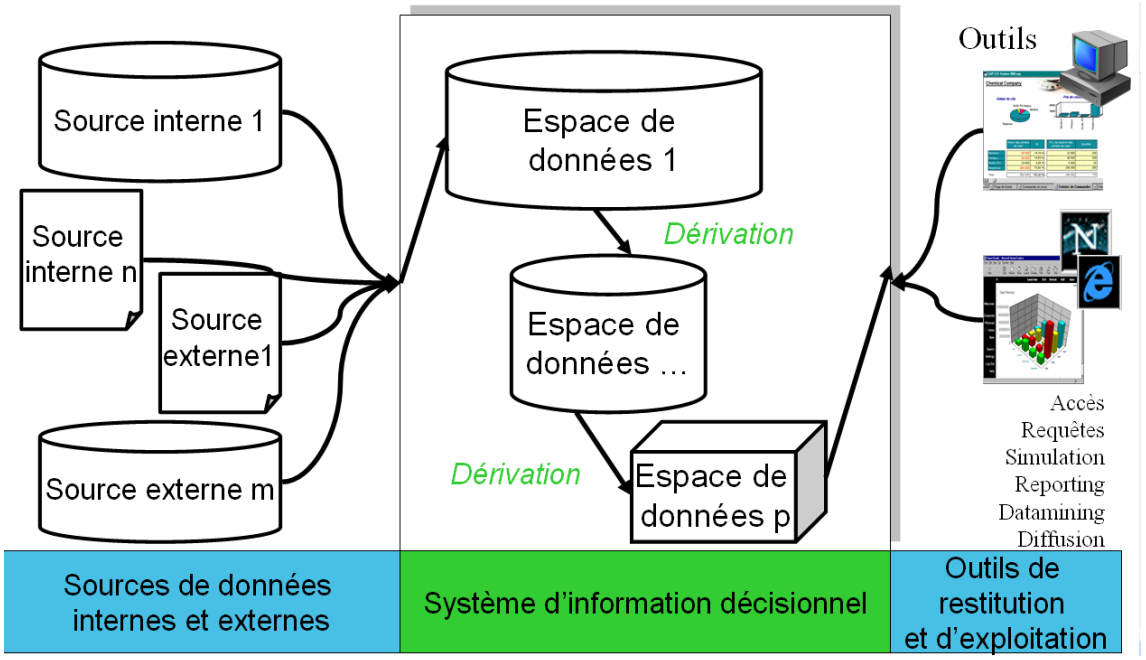


FIG. 1.2 – L'environnement du SID

En résumé, contrairement au système d'information classique, le SID se construit à partir de sources de données existantes. Il est composé de plusieurs modules qui sont le lieu des traitements de dérivation et de présentation des données. Il récupère les données provenant de ces sources et les prépare pour le système de décision. En plus des besoins métiers et techniques déjà présents dans les systèmes d'information classiques, les besoins stratégiques sont de nouveaux besoins à évaluer car le SID se distingue du système d'information par la dimension stratégique. Ces derniers constituent donc la troisième entrée du processus de développement des SID.

Par ailleurs, en raison des analyses complexes qu'il supporte, le SID est souvent associé à un autre concept qui est celui des systèmes OLAP. Plusieurs définitions sont présentées dont la plus connue est celle énoncée dans le rapport technique de référence dans le domaine du décisionnel [Codd et al. 1993]. Cette définition met en avant la représentation multidimensionnelle des données, le caractère rapide et partagé des analyses de données. L'auteur définit 12 règles initiales, appelées FASMI (Fast Analysis of Shared Multidimensional), qui ont été étendues à 18 et qui correspondent aux fonctionnalités que doit vérifier un système OLAP. Les principales caractéristiques énoncées sont :

- la vision multidimensionnelle des données : les données sont représentées comme un point dans un espace à plusieurs dimensions,
- la manipulation intuitive des données : les données sont manipulées directement par des actions sur les cellules de la structure présentant les données,

– la centralisation de données hétérogènes : les applications OLAP se situent entre les sources hétérogènes et les outils qui fournissent des interfaces pour interroger ces données unifiées,
– la transparence entre l'outil de visualisation et l'espace de stockage : les données sources doivent être accessibles depuis les interfaces de restitution des données afin de comprendre les origines de ces dernières.

Définition 1.3 (Système OLAP) *Un système OLAP (On Line Analytical Processing) est un SID qui organise les données dans un espace multidimensionnel de telle sorte que l'organisation des données soit intuitive aux utilisateurs. Il gère l'accès des utilisateurs de manière totalement transparente aux différentes sources de données hétérogènes pour des analyses interactives et dynamiques.*

Le système OLAP, plus généralement le SID, est mis en opposition avec le système d'information transactionnel qualifié de OLTP (On Line Transactional Processing) afin de montrer les spécificités des SID par rapport à ces systèmes. Nous présentons un comparatif de ces deux types de systèmes dans le tableau 1.1.

	OLTP	**OLAP**
Données	Orientées applications	Orientées thèmes et sujets
	courantes, instantanées	Historiques
	Exhaustives, détaillées	Agrégées, résumées
	Changeant constamment, dynamiques	Stables et synchronisées dans le temps
Usages	Nombreux	Peu nombreux
	Variés	Décideurs et analystes
	Mises à jour et interrogations	Interrogations
	Assurent l'activité au quotidien	Facilitent l'analyse et la prise de décision
	Faible volume à chaque transaction	Large volume manipulé
	Temps de réponse immédiat	Temps de réponse moins critique
Processus	Métier	Aide à la prise de décision
Paramètres	Règles de gestion	Règles de gestion métier, stratégies globales, sources de données,
Traitements	Pas de dérivation des données	Dérivation des données
	Pas de préparation des données	Préparation des données
Architecture	Un seul type de module	Plusieurs type de modules

Tab. 1.1 – OLTP versus OLAP

Les SID possèdent donc des spécificités liées aux données, aux usages, aux processus qu'il supporte, aux paramètres d'entrée de son processus de développement, aux traitements de dérivation et de préparation des données ainsi qu'à son architecture.

1.3 Intérêts des SID

La masse de données brutes manipulée au quotidien au sein d'une organisation est difficilement exploitable par les décideurs seuls. De plus, l'ensemble de ces données n'est pas pertinent pour ces derniers. Leurs besoins concernent les données et les traitements nécessaires pour faciliter la prise de décision. Le SID centralise et consolide les données des différents domaines d'activité présents dans les sources internes et externes de l'organisation.

La représentation des données tel que le sujet de l'analyse est un point dans un repère dont les axes sont ceux de l'analyse [Kimball 1996] contribue à faciliter la compréhension et la validation du schéma des SID par les décideurs. De plus, de par cette organisation multidimensionnelle des données et l'accès transparent aux différentes sources de données hétérogènes, le SID tend vers une manipulation intuitive des données par les décideurs. Autrement dit, il a pour objectif d'augmenter l'autonomie des décideurs pour la réalisation de leurs analyses.

Cette structuration proche de leur vision présente un autre intérêt : celui d'offrir des performances satisfaisantes. En fait, les analyses qui sont effectuées étant liées à leur vision des données et l'organisation des données dans le SID se rapprochant de leur vision des données alors, les requêtes liées sont rapidement analysées et traitées.

Cette autonomie des décideurs couplée aux données élaborées du SID favorisent une réactivité efficace et une augmentation de la capacité de l'organisation à s'adapter aux éventuels changements de contextes.

1.4 Motivations

Nos travaux de recherche se sont déroulés dans le cadre industriel de la société I-D6[2]. C'est une société de services dans le développement de systèmes d'information classiques et généralisés. Elle compte une quarantaine de concepteurs de systèmes d'information répartis sur une cinquantaine de projets en moyenne. Les développements réalisés pour les différentes organisations clientes peuvent concerner des domaines d'activités proches.

Tous les concepteurs qui participent à un projet n'ont pas la même expertise du développement des systèmes d'information décisionnels. Ils maîtrisent la conception de systèmes d'information classiques suivant la méthode Merise [Tardieu et al.

[2]I-D6 est une société de services et d'ingénierie informatique spécialisée dans le développement d'outil décisionnel et collaboratrice dans le cadre de la thèse CIFRE (Convention Industrielle de Formation par la Recherche) de numéro de convention 766/2003 préparée par Estella Annoni. http://www.i-d6.com.

1983]. Ainsi, face au manque de méthode reconnue et à la notoriété des outils et des méthodes dans le domaine des systèmes d'information classiques, les concepteurs adaptent les outils, les techniques et les méthodes liés pour mettre en place des SID.

De nombreux projets décisionnels ont été menés aussi bien pour de grands groupes que pour des organisations du marché intermédiaire. La méthode mise en oeuvre pour chaque projet n'est pas capitalisée en vue d'être réutilisée. La société a donc exprimé le besoin d'une méthode de développement de ce type de systèmes :

- exploitable par ses concepteurs décisionnels, toutes expériences confondues,
- qui capitalise la connaissance mise en oeuvre sur les projets antérieurs,
- favorisant voire systématisant la réutilisation.

Ce constat de non capitalisation et non réutilisation est vrai pour un grand nombre d'organisations car elles utilisent des méthodes basées sur l'hypothèse selon laquelle le développement de systèmes d'information se fait à partir de rien comme le précise [Guzelian 2004]. Ces méthodes n'intègrent ni la capitalisation de la connaissance mise en oeuvre sur des projets ni la réutilisation. Par ailleurs, les méthodes de développement des SID dans un contexte de réutilisation ont fait l'objet de peu de travaux [Jones and Song 2005; Feki et al. 2006].

Au cours de cette décennie de travaux dans le domaine du décisionnel, plusieurs propositions de méthodes de développement des SID ont été faites [Kimball 1996; Golfarelli and Rizzi 1998b; Abelló et al. 2002; Luján-Mora and Trujillo 2003]. Ces travaux visent à définir le schéma du SID à partir soit des besoins utilisateurs, soit des sources de données, soit des deux. Cependant, comme le relèvent de nombreux auteurs dont les plus récents [Rizzi et al. 2006], il n'existe pas de méthode de développement des SID reconnue par la communauté de chercheurs et celle d'industriels, la définition d'une méthode de développement de SID est toujours de mise. Cette situation a de multiples causes.

La spécificité des paramètres d'entrée du processus de développement des SID contribue à expliquer ce constat. Les utilisateurs du SID sont des décideurs. En plus de leurs besoins, le processus d'ingénierie des SID évalue un autre paramètre complémentaire qui est l'ensemble des sources de données internes et externes. Ce dernier est nécessaire pour répondre aux besoins des utilisateurs. D'autre part, parmi ces décideurs qui sont de plus en plus nombreux au sein des organisations, il y a les décideurs stratégiques qui représentent les intérêts généraux de l'organisation et les décideurs tactiques qui représentent les intérêts d'un domaine d'activité particulier. L'importance et la portée des besoins exprimés par ces deux types de décideurs ne sont pas les mêmes. Notre point de vue est appuyé par l'étude d'IDG[3] qui précise que l'implication des décideurs stratégiques dans les projets décisionnels est défini comme étant un *facteur important parmi les clés de réussite de ces projets pour 72% des organisations alors que l'adéquation aux objectifs métiers est un facteur important pour 86% des organisations* [IDG 2005]. Cependant, la majorité des méthodes de développement de SID proposées ne distingue pas les deux types de décideurs.

[3]Etude de l'Etat des lieux des projets décisionnels en 2005/2006 faite par IDG.

La faible prise en compte voire l'occultation de la phase d'analyse des besoins est l'un des facteurs de ce constat. Les méthodes de développement passent directement à la modélisation des SID [Mazon et al. 2005]. Ce qui peut être corrélé avec le fait que 1/3 des projets décisionnels ne répondent pas aux besoins des décideurs selon le cabinet Standish group. Des méthodes d'analyse des besoins ont été proposées par [Bruckner et al. 2001; Prakash and Gosain 2003; Gam and Salinesi 2006a]. Certaines définissent le contexte décisionnel en prenant en compte l'importance des décideurs stratégiques mais elles ne se placent pas dans un processus complet de développement de SID. Celles qui se situent en amont d'un processus de développement complet [Bonifati et al. 2001; Mazon et al. 2005; Giorgini et al. 2005; Ghozzi et al. 2005; Soussi et al. 2005] présentent l'inconvénient de représenter les données dans des modèles de besoins spécifiques qui ne se rapprochent pas de la représentation intuitive des données par les décideurs.

A partir de modèles de besoins, les méthodes de développement existantes définissent les modèles conceptuels multidimensionnels. Malgré la profusion de ces modèles au cours de la dernière décennie, il n'y a pas de modèle reconnu [Ravat et al. 2007b]. Ce fait peut s'expliquer d'une part, par le faible nombre de concepts reconnus dans le domaine du décisionnel [Pedersen and Jensen 2005] et d'autre part, parce que ces modèles ne permettent pas de représenter toutes les spécificités présentées dans le tableau 1.1.

De plus, ces méthodes de développement ainsi que les méthodes d'analyses se focalisent sur la modélisation des données contenues dans le SID. Elles n'abordent presque pas les problématiques liées aux traitements de dérivation et de préparation des données alors qu'ils représentent 55% de la charge de développement [Inmon 1997]. Les rares évaluations des traitements au sein des méthodes de développement [Vassiliadis et al. 2002a] ont lieu après la conception du SID, dans un autre schéma que celui du SID ; elles ne sont donc pas validées par les utilisateurs. Elles concernent principalement les traitements liés à la dérivation des données. Ces traitements définissent le contexte technique du SID. Elles concernent aussi des traitements de transformation qui sont liés la préparation des données, autrement dit au contexte décisionnel. Cependant, elles ne modélisent pas les autres traitements liés à ce dernier telle que la validité des données.

Aucune méthode ne permet la mise en place d'un SID basé sur une architecture modulaire et elle ne fournit pas un guidage pour toutes les tâches de l'analyse et de la conception. Les méthodes de développement se concentrent sur un voire deux modules du SID. Ainsi, face à l'architecture multi-modulaire du SID, le concepteur décisionnel est livré à lui-même. Dans [Sen and Sinha 2005], ce dernier est guidé pour déterminer les outils commerciaux les plus adaptés pour une configuration de projet donnée sur la base d'un comparatif, mais le choix des modules de l'architecture à développer n'est pas abordé.

1.5 Problématiques et objectifs

Notre problématique globale est **comment définir une méthode de développement rapide de SID fiables au regard des besoins de tous ses acteurs dans un contexte de réutilisation, utilisable par tous concepteurs décisionnels ?**

La définition d'une méthode qui guide l'analyse et la conception de SID fiables par rapport aux besoins des utilisateurs et aux sources de données dans un contexte de réutilisation aborde des problématiques plus précises relatives à trois domaines de recherches :

- l'**ingénierie des SID** : à l'instar de l'ingénierie des SI qui est le processus par lequel les analystes du système, les ingénieurs informaticiens et les utilisateurs finaux construisent les systèmes d'information [Avison and Hanifa 1997], l'ingénierie des SID est le processus par lequel les concepteurs décisionnels et les décideurs construisent le SID,
- l'**ingénierie des besoins des SID** : l'ingénierie des besoins dans le cadre des systèmes d'information classiques est définie par [Rolland and Prakash 2000] comme le processus qui explore les objectifs des acteurs et les activités pour atteindre leurs objectifs afin de dériver des systèmes qui répondent à leurs besoins. Par extension, l'ingénierie des besoins des SID est le processus qui explore les objectifs des acteurs, les activités et les sources de données qui fournissent les données pour atteindre leurs objectifs afin de dériver des systèmes qui répondent aux besoins des décideurs au regard des sources de données,
- la **réutilisation** : c'est la mise en oeuvre des composants existants capitalisés à partir ou au cours de développements pour la définition de nouveaux systèmes. Ce processus favorise un gain de temps et de fiabilité de ces systèmes.

Nous explicitons nos problématiques par domaine. Puis, nous présentons le fil directeur de nos contributions et les publications associées à la validation scientifique.

L'ingénierie des SID : les SID se distinguent du SI par la dimension stratégique et ils sont basés sur des architectures hétérogènes variées. De plus, les modèles multidimensionnels sur lesquels reposent généralement le SID ne permettent pas de représenter toutes les spécificités multidimensionnelles. De ce fait, la problématique concerne deux thématiques (modèles et démarches) que nous distinguons car tous les modèles multidimensionnels proposés ne sont pas nécessairement associés à une méthode :

- Méthodes de développement des SID :
  - quelles sont les impacts, sur le processus de développement, de cette singularité des SID par rapport aux SI ?
  - comment définir l'architecture du SID ?
  - comment prendre en compte l'hétérogénéité des architectures du SID ?

- Modèles multidimensionnels :
  - comment concevoir les données et les traitements du SID ?
  - comment répondre au manque de modèle reconnu et, par la même, au manque de méthode reconnue ?

L'ingénierie des besoins des SID : les acteurs du SID sont très nombreux et présentent des spécificités qui impactent la satisfaction des besoins des décideurs. La prise en compte des besoins des acteurs est une tâche longue. De plus, le SID à la particularité de dériver de sources de données et de faciliter la prise de décision. En ce sens, la problématique concerne deux points :

- comment analyser rapidement les besoins des différents acteurs en terme de données et de traitements afin de définir un SID fiable par rapport aux besoins de tous les acteurs ?
- comment représenter les besoins de telle sorte qu'ils soient compréhensibles par tous les acteurs et que les décideurs puissent facilement les valider ?

La réutilisation : le développement de SID requiert d'importantes ressources et de longues durées. De plus, plusieurs travaux ont mis en avant la récurrence des tâches au cours des projets décisionnels [Srivastava and Chen 1999; Sen and Sinha 2005]. Cependant, les travaux de méthodes de développement de SID dans un contexte de réutilisation sont à leurs débuts contrairement aux méthodes de développement de SI dans un contexte de réutilisation. Ainsi, la problématique est : comment systématiser la réutilisation afin que tout concepteur décisionnel puisse mettre en place rapidement des SID fiables et pertinents ?

Ainsi, dans cette thèse, nous proposons de **définir des modèles, une démarche et un outil afin d'analyser et de concevoir des SID dans le cadre d'une méthode de développement de SID** qui :

- guide complètement l'analyse et la conception de SID basés sur des architectures variées,
- capitalise la connaissance en terme de démarche et de produit et facilite la réutilisation,
- prend en compte les besoins des décideurs tactiques, des décideurs stratégiques et des sources et les représente de manière intuitive à ces décideurs,
- évalue dès l'analyse l'ensemble des traitements de dérivation et de préparation des données,
- repose sur un modèle conceptuel multidimensionnel permettant de représenter toutes les spécificités des SID en terme de données et de traitements.

Nos propositions ont été définies en respectant le fil conducteur de notre démarche de développement de SID (le Trident décisionnel). Nous avons validé scientifiquement chaque phase de notre méthode par des publications nationales ou internationales comme l'indique la figure 1.3.

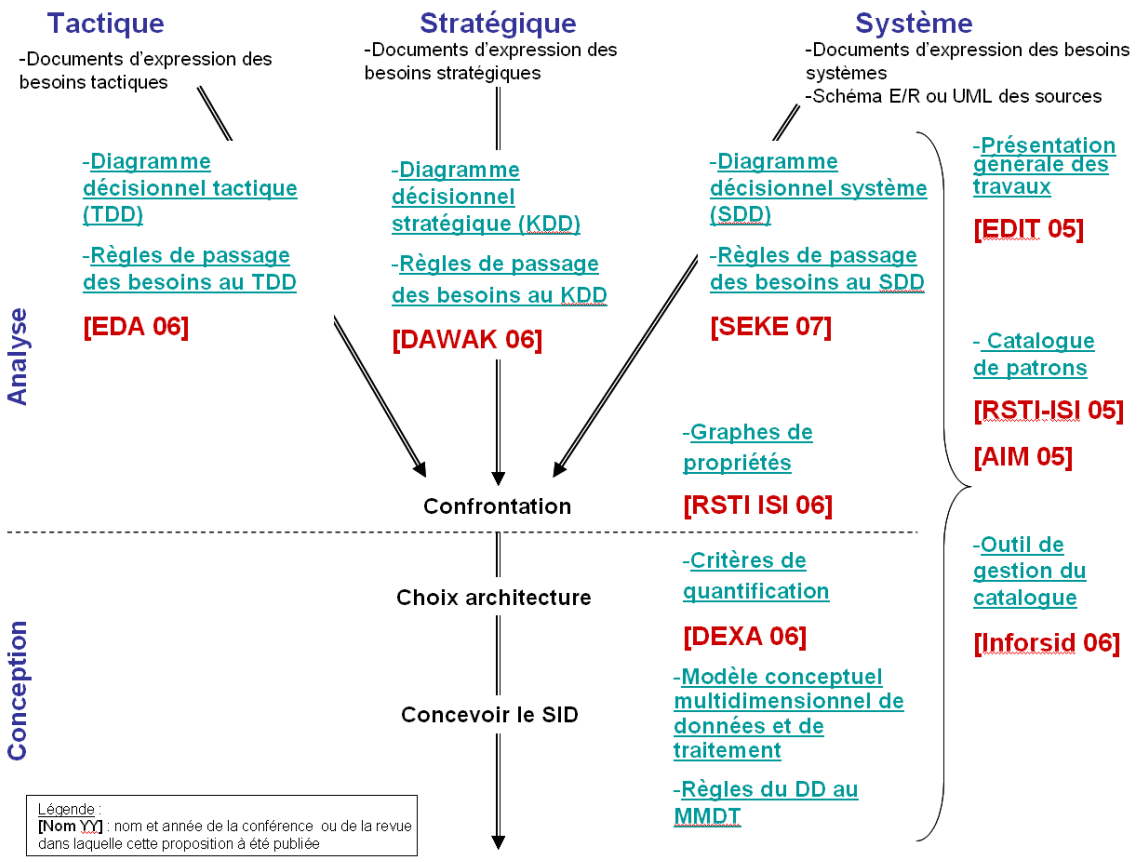


FIG. 1.3 – Nos publications

Les publications sont :

- **[SEKE 07]** : « Data and Process analyses of Data Warehouse Requirements » présentée à la 19ième conférence internationale Software Engineering and Knowledge Engineering [Annoni et al. 2007],
- **[RSTI 06]** : « Traitements à l'origine des systèmes d'information décisionnels » publiée dans la Revue des Sciences et Technologies de l'Information [Annoni et al. 2006a],
- **[DAWAK 06]** : « Towards Multidimensional Requirement Design » présentée à la 8ième conférence internationale Data Warehousing and Knowledge Discovery [Annoni et al. 2006e],
- **[DEXA 06]** : « Automating the Choice of Decision Support System Architecture » présentée à la 17ième conférence internationale Database and Expert Systems [Annoni et al. 2006b],
- **[Inforsid 06]** : « Méthode de Développement des Systèmes d'Information Décisionnels : Roue de Deming » présentée à la conférence nationale Informatique des Organisations et Systèmes d'Information et de Décision,
- **[EDA 06]** : « Modélisation adaptée aux besoins utilisateurs dans le développement des systèmes d'information décisionnels » publiée dans la Revue des Nouvelles Technologies de l'Information [Annoni et al. 2006c],

- **[RSTI 05]** : « Une approche d'analyse et de conception de SID à base de patrons » publiée dans la Revue Ingénierie des Systèmes d'Information [Annoni et al. 2005a],
- **[AIM 05]** : « BIPAD : Une méthode d'analyse et de conception des systèmes d'information décisionnels par réutilisation de patrons » présentée à la conférence nationale AIM Association Information and Management [Annoni et al. 2005b],
- **[EDIT 05]** : « BIPAD : Une méthode d'analyse et de conception des systèmes d'information décisionnels » présentée au colloque des doctorants de l'Ecole Doctorale Informatique et Télécommunications [Annoni 2005].

1.6 Organisation de cette thèse

Notre méthode de développement des SID a pour objectif de prendre en compte toutes leurs spécificités et de permettre à tout concepteur décisionnel, indépendamment de son expérience, de disposer des outils pour mettre en place rapidement des systèmes fiables de ce type. Les travaux réalisés et les propositions émises dans le cadre de la définition de cette dernière sont présentés dans les sept chapitres de cette thèse.

Dans ce chapitre 1, nous avons défini le contexte et la problématique générale de nos travaux de recherche en positionnant le SID par rapport au système d'information.

Dans le chapitre 2, nous dressons un état de l'art des modèles existants dans le domaine du décisionnel ainsi que les concepts qu'ils permettent de représenter. Puis, nous présentons les méthodes de développement de SID suivant leurs démarches et les modèles d'analyses. Enfin, nous abordons les propositions relatives à la réutilisation dans le domaine de l'ingénierie des SID.

Dans le chapitre 3, nous présentons les principes généraux de nos propositions concernant les acteurs, les modules décisionnels et notre démarche de développement du Trident décisionnel.

Dans le chapitre 4, nous explicitons la première phase du Trident décisionnel qui est l'analyse des besoins du SID. Nous présentons les modèles pour la collecte et la formalisation des besoins tactiques, des besoins stratégiques et des sources de données en terme de données et de traitements. Puis, nous présentons notre méthode d'analyse des besoins du SID.

Dans le chapitre 5, nous nous focalisons sur la deuxième phase du Trident décisionnel qui est la conception du SID. Nous dressons une typologie des architectures décisionnelles valides. Puis, nous présentons les critères de quantification des architectures et le processus automatique du choix de l'architecture. Nous continuons

par la définition de notre modèle multidimensionnel conceptuel de données et de traitements. Enfin, nous explicitons la démarche pour le passage de notre modèle de besoins d'une part, à notre modèle conceptuel et d'autre part, au modèle entité-association.

Dans le chapitre 6, nous présentons notre catalogue de composants réutilisables de démarche et de produit en précisant les extensions apportées au formalisme utilisé pour la définition des composants réutilisables.

Dans le chapitre 7, nous dressons une synthèse des applications réalisées au cours des missions dans le cadre de la validation industrielle de notre méthode. Puis, nous explicitons le processus qualité de définition de notre méthode. Enfin, nous présentons l'outil que nous avons développé pour l'implantation du catalogue de composants réutilisables

Première partie

Cadre des travaux

Chapitre 2

Etat de l'art

Table des matières

2.1 Introduction **19**
2.2 Les modèles multidimensionnels des SID **19**
2.2.1 Les spécificités des SID 21
2.2.2 Niveau conceptuel 27
Paradigme entité-association 28
Paradigme Objet 29
Paradigmes spécifiques 30
Synthèse des modèles conceptuels multidimensionnels . . 31
2.2.3 Bilan et positionnement par rapport aux modèles multidimensionnels 34
2.3 Les méthodes de développement des SID **34**
2.3.1 Démarche descendante 36
2.3.2 Démarche ascendante 37
2.3.3 Démarche mixte 40
2.3.4 Bilan et positionnement par rapport aux méthodes de développement 43
2.4 Les méthodes d'analyse des SID **45**
2.4.1 Modèles existants 46
2.4.2 Requêtes 48
2.4.3 Modèles de buts 49
2.4.4 Bilan et positionnement par rapport aux méthodes d'analyse 51
2.5 La capitalisation et la réutilisation dans les SID **54**
2.5.1 Les composants réutilisables 54
2.5.2 Ingénierie des systèmes d'information 56
2.5.3 Ingénierie des SID 56
2.5.4 Bilan et positionnement par rapport à la capitalisation et à la réutilisation 58
2.6 Synthèse de l'existant **58**
2.6.1 Limites des modèles et méthodes de développement . . . 59

2.6.2 Limites des méthodes d'analyse 60
2.6.3 Limites face au développement rapide et au contexte de réutilisation 60
2.7 Conclusion 61

2.1 Introduction

Nos travaux de recherche ont pour objectif de définir une méthode de développement des SID (systèmes d'information décisionnels) qui permet la mise en place rapide de SID fiables dans un contexte de réutilisation.

De nos jours, il est largement accepté que la modélisation de référence pour les SID est la modélisation multidimensionnelle [Inmon 1996; Jarke et al. 2001]. Par ailleurs, il est à noter que 40% des projets décisionnels échouent et 80% ne parviennent pas à répondre aux besoins des utilisateurs [Schiefer et al. 2002] car les concepteurs décisionnels commencent les projets décisionnels directement par la modélisation conceptuelle ou logique du SID [Mazon et al. 2005]. Nous pouvons noter aussi que le domaine de l'ingénierie des systèmes d'information (SI) axée sur la réutilisation a fait l'objet de nombreuses propositions qui ont fait leur preuve en matière de gain de temps au cours des développements et de fiabilité des systèmes développés [Rieu 1999].

Fort de ces constats, nous avons étudié les propositions liées aux quatre principales thématiques de nos trois problématiques qui sont les modèles multidimensionnels, les méthodes de développement des SID, les méthodes d'analyse des SID et la réutilisation.

Ainsi, dans la section 2.2, nous présentons les modèles multidimensionnels afin de mettre en avant les spécificités des SID qui sont peu voire pas prises en compte. Puis, dans la section 2.3, nous étudions les méthodes de développement de SID suivant leur démarche en mettant en avant les tâches qui sont peu guidées ou non abordées. Dans la section 2.4, nous nous focalisons sur la phase d'analyse en mettant en avant les apports et les limites des méthodes d'analyse existantes. Dans la section 2.5, nous continuons par une synthèse des propositions liées à la problématique de l'ingénierie des SI axée sur la réutilisation et des récentes propositions relatives à l'ingénierie des SID axée sur la réutilisation. Dans la section 2.6, nous dressons le bilan des apports de cet existant pour répondre à leurs limites. Enfin, dans la section 2.7, nous clôturons cet état de l'art.

2.2 Les modèles multidimensionnels des SID

Les systèmes d'information décisionnels (SID) ont pour objectif d'assurer le support du processus de prise de décision. Autrement dit, les SID doivent permettre des analyses complexes de données élaborées. Cependant, les caractéristiques des tâches d'analyse de données suivant un grand nombre d'axes ne peuvent pas toutes être représentées avec les modèles de systèmes d'information classiques tels que les agrégations, les calculs complexes, les consolidations. De ce fait, le modèle entité-association est caractérisé tant par la communauté de chercheurs que d'industriels

comme inadéquats pour ces applications [Kimball 1996; Blaschka et al. 1999b]. Pour répondre aux inadéquations de ces modèles, la modélisation multidimensionnelle a été proposée.

Définition 2.1 (Modélisation multidimensionnelle) *La modélisation multidimensionnelle vise à représenter les données sous une forme intuitive aux décideurs (proche de leur vision des données). Elle représente l'information analysée, en l'occurrence les centres d'intérêts, comme un point dans un espace à plusieurs dimensions qui sont les axes de l'analyse [Chrisment et al. 2005b].*

Exemple : afin d'associer un exemple à chaque concept décisionnel, nous nous plaçons dans le cadre d'un projet mené au sein de la société collaboratrice I-D6 pour une organisation qui possède un important parc de biens matériels et immatériels. L'organisation souhaite mieux gérer la disponibilité, la répartition des biens et le renouvellement du parc. Le projet est celui des immobilisations.

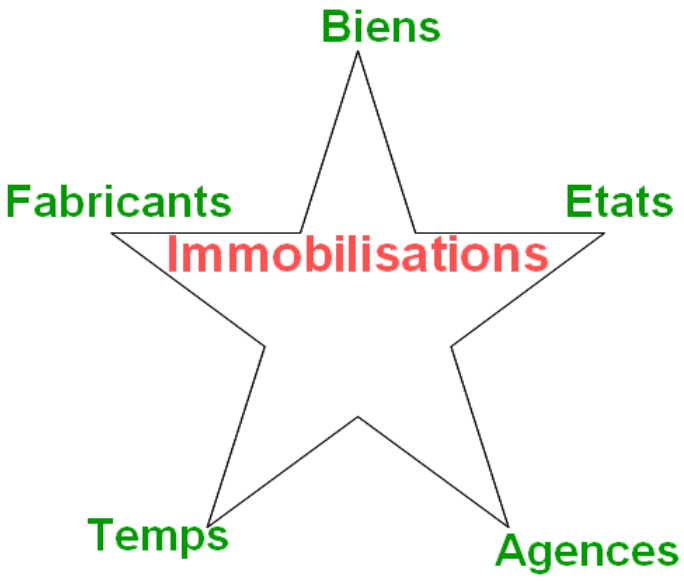


FIG. 2.1 – Une modélisation multidimensionnelle du domaine d'activités immobilisations

Après une décennie de recherche dans ce domaine, les propositions de modèles et de méthodes sont nombreuses mais elles ne reposent pas sur des bases théoriques standards [Rizzi et al. 2006]. Actuellement, il n'y a donc pas de modèle qui soit reconnu comme standard. Cependant, trois concepts sont admis par les communautés de chercheurs et d'industriels comme étant les bases d'une modélisation multidimensionnelle, en l'occurrence le concept de « cube de données » mais aussi ceux de « dimension » et de « fait ».

Définition 2.2 (Cube de données) *Un cube de données représente la structure dont les cellules contiennent des données mesurées et dont les arêtes contiennent les axes d'analyse naturels des données [Kimball et al. 1998]. Ce concept relève de l'organisation des données en vue de l'implantation.*

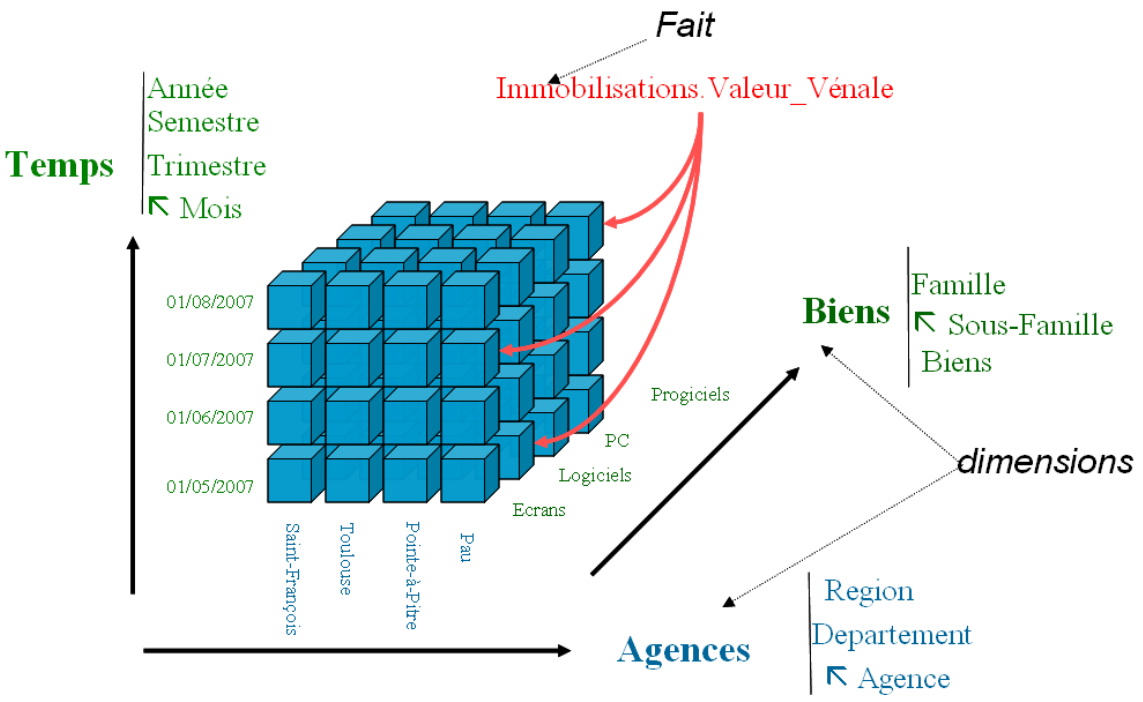


FIG. 2.2 – Un exemple de cube de données avec un fait et trois dimensions

Définition 2.3 (Dimension) *Une dimension représente un axe suivant lequel l'analyse des données sera faite. C'est une arête du cube.*

Exemple : les dimensions présentées dans la figure 2.2 sont « Temps », « Agences » et « Biens ».

Définition 2.4 (Fait) *Un fait représente le sujet, appelé aussi une donnée factuelle ou un centre d'intérêts sur lequel porte l'analyse. Il est défini par une cellule du cube de données pour une combinaison donnée des dimensions.*

Exemple : le fait présenté dans la figure 2.2 est « Immobilisations ».

2.2.1 Les spécificités des SID

Traditionnellement, la modélisation des applications multidimensionnelles était liée à l'implantation comme l'argumente Torlone [Torlone 2003]. Cette singularité est liée à l'importance de la prise en compte des caractéristiques de l'information manipulée afin d'avoir des temps de réponse acceptables par les décideurs. Ainsi, les premiers modèles qui ont été proposés sont basés sur des structures ne représentant pas tous les concepts exprimés par les utilisateurs mais répondant aux caractéristiques de l'implantation. Ces structures sont celles de table ou celles du cube. Pour

répondre à cela et, par la même, définir un modèle multidimensionnel qui représente uniquement les concepts du monde réel indépendamment de toute implantation physique, des listes de propriétés ont été définies dans [Vassiliadis and Sellis 1999; Blaschka et al. 1999b; Abelló et al. 2001a; Rafanelli 2003].

Ces listes comprennent des propriétés dérivées des principes généraux de modélisation des systèmes d'information classiques et des propriétés spécifiques aux applications d'analyses multidimensionnelles. Les propriétés dérivées des principes généraux qui ont fait la renommée du modèle entité-association et du modèle relationnel sont :

- un formalisme indépendant de l'implantation,
- le paradigme de séparation de la structure et des données,
- un langage déclaratif.

Les propriétés spécifiques aux analyses multidimensionnelles requièrent des concepts qui se rapprochent de la vision des données par les décideurs et de la sémantique du décisionnel. Les concepts introduits à cet effet sont les concepts de **mesure**, de **paramètre**, d'**hiérarchie**, d'**attribut faible**, de **schéma en étoile** et de **schéma en constellation**.

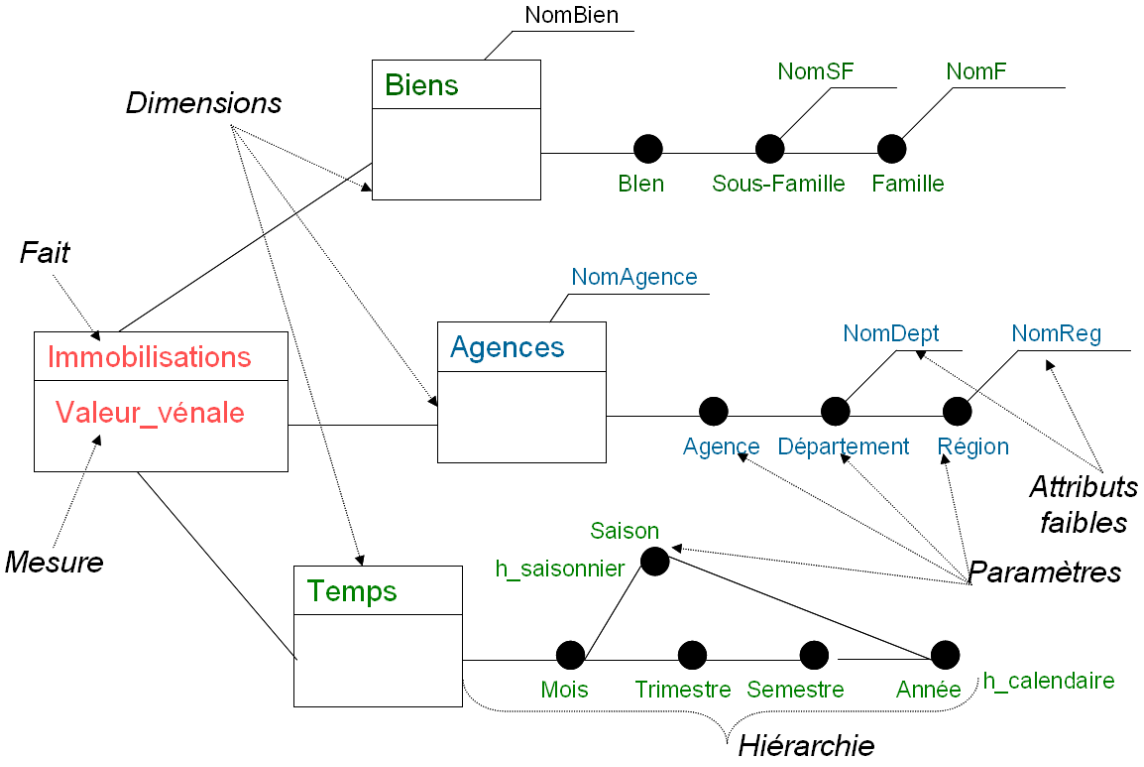


FIG. 2.3 – Les autres concepts multidimensionnels avec notre formalisme inspiré de [Golfarelli et al. 1998b]

Définition 2.5 (Mesure) *Elle représente un indicateur de performance de l'activité de l'organisation. C'est la propriété qui caractérise un fait.*

Exemple : la mesure du fait « Immobilisations » présenté dans la figure 2.3 est la « Valeur_vénale ».

Définition 2.6 (Paramètre) *C'est la propriété constitutive d'une dimension.*

Exemple : les paramètres de la dimension « Biens » présentée dans la figure 2.3 sont « Bien », « Sous-famille » et « Famille ».

Définition 2.7 (Hiérarchie) *C'est une perspective d'analyse des faits. Elle définit un sous ensemble ordonné des paramètres d'une dimension. Les paramètres organisés en hiérarchie sont appelés* ***niveaux****.*

Exemple : la hiérarchie « h_calendaire » de la dimension « Temps » présentée dans la figure 2.3 organise les attributs suivant la réduction du niveau de détail Mois− >Trimestre− >Semestre− >Année.

Définition 2.8 (Attribut faible) *C'est une information généralement textuelle qui caractérise un paramètre.*

Exemple : les attributs faibles des paramètres « Sous-famille » et « Famille » sont respectivement « NomSF » et « NomF ».

Définition 2.9 (Schéma en étoile) *Un schéma en étoile est une représentation multidimensionnelle des données proposée par [Kimball 1996] qui représente le fait au centre et les dimensions qui rayonnent autour du fait. Cette représentation est un standard de fait au sein de la communauté industrielle.*

Définition 2.10 (Schéma en constellation) *Un schéma en constellation est un regroupement de schémas en étoile qui partagent des dimensions communes. Il met en avant les corrélations entre les faits et il évite de définir plusieurs fois la même dimension au sein d'une même organisation.*

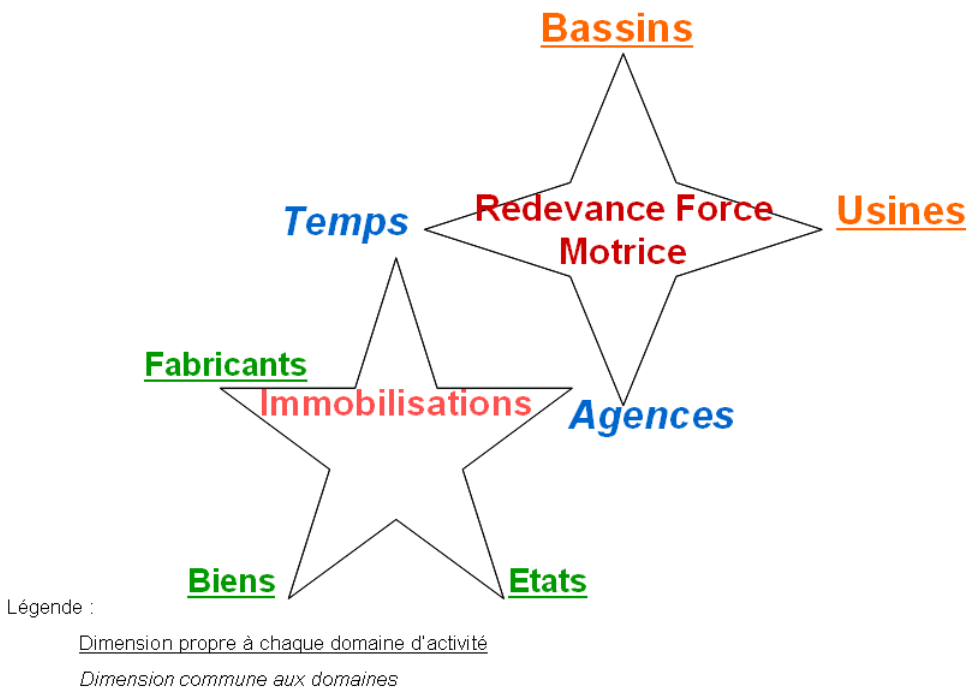


FIG. 2.4 – Un schéma en constellation

Ces applications d'analyses multidimensionnelles requièrent une représentation singulière de l'espace multidimensionnel qui supporte tout croisement pertinent des données. Les propriétés de ces listes spécifiques aux analyses multidimensionnelles sont donc :

- la structuration complexe des dimensions afin d'exprimer les liens qui existent entre les paramètres des dimensions,
- la structuration complexe des faits et des mesures car les faits peuvent être liés, un fait peut être composé de plusieurs mesures et une mesure peut dériver d'une ou de plusieurs autres mesures,
- la cohérence de l'interrogation des données car une requête peut être construite à partir du résultat d'une précédente avec des opérations d'augmentation ou de réduction du niveau de détails des données suivant une hiérarchie,
- le traitement symétrique des faits et des dimensions au cours des manipulations des données. Cette propriété est liée aux algèbres et aux langages de définition et de manipulation des données. Nous n'évaluons donc pas celle-ci.

A cette liste, nous ajoutons la dynamique du SID car comme tout système d'information, les SID ont un aspect dynamique. De plus, ils ont la particularité de dériver de sources de données et de faciliter la prise de décision à partir de données consolidées et élaborées.

Chacune de ces propriétés liées aux analyses multidimensionnelles s'évalue suivant une ou plusieurs spécificités multidimensionnelles. Nous présentons ci-après les spécificités suivant lesquelles nous évaluons les modèles existants.

La *structuration complexe des dimensions* relève de plusieurs concepts tels que les « hiérarchies simples », les « hiérarchies multiples et alternatives », les « hiérarchies non-strictes », les « rôles d'une dimension », les « éléments terminaux multiples » et les « dimensions dégénérées ».

Les **hiérarchies simples** d'une dimension explicitent le chemin entre les paramètres d'une dimension.

Les **hiérarchies multiples et alternatives** sont définies en fonction des liens et des contraintes entre les paramètres d'une dimension. Dans le cas où, une dimension a au moins deux hiérarchies simples dont les niveaux fins ne sont pas les mêmes, elle possède des hiérarchies multiples. Cependant, si les niveaux fins sont les mêmes, ces hiérarchies sont des hiérarchies alternatives.

Les **hiérarchies non-strictes** sont définies quand une instance d'un niveau fin peut appartenir à plusieurs objets du niveau supérieur associé. Ce concept est très présent dans le décisionnel car les perspectives d'analyse peuvent être liées ; il est important de l'identifier afin de traiter correctement au cours du développement.

Exemple : un mois peut appartenir à plusieurs saisons.

Les **éléments terminaux multiples** indiquent qu'une dimension peut être reliée aux faits par plus d'un de ses paramètres. Ce concept est important car tous les faits d'un schéma en constellation ne s'expriment pas nécessairement en fonction du même niveau de granularité d'une dimension. Il importe de le représenter sur le schéma.

Exemple : le fait « Immobilisations » est au niveau de granularité mois alors que le fait « redevance force motrice » est niveau jour.

Les **rôles d'une dimension** permettent de représenter qu'un fait peut s'exprimé plusieurs fois en fonction d'une même dimension. Ces rôles sont pertinents car dans de nombreux projets une dimension intervient plus d'une fois. La conception qui en résulte généralement est la duplication de la dimension. C'est un abus de conception car c'est le même concept mais qui participent à la relation avec deux rôles différents.

Exemple : dans la modélisation d'un SID de l'activité aéroportuaire nationale, la dimension aéroport intervient deux fois ; une première fois en tant que aéroport de départ et une seconde fois en temps que aéroport d'arrivée.

Une **dimension dégénérée** désigne un attribut d'un fait qui permet d'identifier de manière unique ce fait et qui n'est pas une donnée qui caractérise le fait, autrement dit qui n'est pas une mesure.

Exemple : dans le domaine des ventes, le numéro de contrat identifie de manière unique le fait « Ventes » alors que le fait est identifiable de manière unique par

les dimensions en fonction desquelles s'exprime une vente telles que les dimensions « Produits », « Temps » et « Commerciaux ».

La *structuration complexe des faits et des mesures* repose aussi sur de nombreux concepts qui sont : les « mesures multiples », les « faits multiples » ou « schémas en constellation », les « mesures dérivées », les « liens entre les mesures » et les « faits dégénérés ».

Les **mesures multiples** caractérisent un fait composé de plusieurs mesures.

Les **faits multiples** constituent les schémas en constellation. Cette spécificité est de rigueur quand le projet décisionnel est très complexe et qu'il inclut un grand nombre de domaine d'activité.

Les **mesures dérivées** définissent des mesures obtenues par l'application d'un calcul à partir d'une ou de plusieurs mesures d'un fait et d'autres données. Ce calcul peut être basé sur un calcul arithmétique, une fonction d'agrégation ou une fonction analytique.

Exemple : dans le domaine des ventes, la mesure chiffre d'affaire peut être déduite à partir d'un calcul arithmétique simple à partir de la mesure quantité de produits vendus. Nous notons que ce calcul requiert une donnée des sources qui est le prix du produit.

Les **liens entre mesures** explicitent les règles de calcul des mesures dérivées. Le calcul de mesures à partir d'autres mesures ou d'autres données font partie de la tâche des décideurs pour préciser les causes et les tendances. De plus, le fait de définir les mesures dérivées dès le début de l'ingénierie, puis de les calculer permet aussi d'améliorer la qualité et les performances des SID.

Exemple : pour contrôler la fiabilité des données, les concepteurs décisionnels déroulent tout le calcul à l'origine d'une mesure. La connaissance des liens entre les mesures permet de procéder à cette validation plus efficace par étapes et par la même de connaître les mesures qui sont concernées par la modification d'une mesure donnée. Dans le domaine de la force de vente, la prime des commerciaux se calcule à partir des résultats, des objectifs et du nombre d'absences. Si l'une de ces mesures est incorrecte, la validation s'arrête et le contrôle de la prime est caduc.

Un **fait dégénéré** est une mesure telle que à une valeur de celle-ci sont associées plusieurs valeurs d'une dimension pour un même rôle [Kimball 1996; Giovinazzo 2000].

Exemple : dans le cas où une vente peut être réalisée par plusieurs commerciaux, la mesure « Commission » du fait « Ventes » est transformée en fait dégénéré associé au lien entre le fait « Ventes » et la dimension « Commerciaux ».

La *cohérence de l'interrogation* est liée à la fiabilité des données du SID. En effet, une requête construite *via* des opérations de manipulation sur les données suivant une hiérarchie permet de réduire ou d'augmenter le niveau de détail des données. Elle se caractérise par la pertinence des agrégations.

La **pertinence des agrégations** définit les fonctions d'agrégation valides pour le passage d'un niveau de détail n à un niveau moins détaillé m pour une hiérarchie d'une dimension. Cette propriété indique les consolidations de mesures qui ont un sens pour les décideurs [Husemann et al. 2000].

Exemple : la fonction d'agrégation généralement utilisée par défaut est la somme. Ainsi, dans le cas où la fonction somme n'a pas de sens pour une mesure, la fiabilité des données restituées n'est pas garantie.

Les spécificités présentées ci-dessus caractérisent la statique du SID. Cependant, il importe de prendre en compte la dynamique du SID pour des raisons déjà énoncées.

La *dynamique* s'évalue en fonction des traitements que le modèle permet de représenter. Par ailleurs les traitements du SID se rapportent à la dérivation des données et la préparation des données.

La **dérivation des données** regroupe les traitements d'extraction, de chargement, de suivi et de sécurité des données. Les traitements appelés ETL (Extraction-Transformation-Load) sont donc principalement inclus dans cet ensemble.

La **préparation des données** regroupe les traitements liés à la validité, l'historisation, le rafraîchissement, l'archivage, le calcul et la consolidation des données, autrement dit ceux liés à la caractérisation des données pour la prise de décision.

Une vingtaine de modèles ont été proposés dans la littérature. Nous présentons dans la section 2.2.2 une classification des modèles conceptuels. Dans la section 2.2.3, nous dressons le bilan des modèles conceptuels.

Dans cet état de l'art, nous ne traitons pas la modélisation au niveau physique et les langages car nos travaux de recherche ne s'y rapportent pas. En ce qui concerne les langages, c'est un thème de recherche connexe qui fait l'objet de propositions par notre équipe [Ravat and Teste 2001; Ravat et al. 2002, 2006].

2.2.2 Niveau conceptuel

La modélisation au niveau conceptuel répond à la question **« De quoi est-il question ? »** . Elle représente donc les données et les traitements sur les données nécessaires aux analyses. Nous proposons de classifier les modèles conceptuels multidimensionnels existants suivant le paradigme utilisé (entité-association, objet, spé-

cifique). Nous développons un voire deux modèles de référence par paradigme, puis dressons une synthèse des modèles utilisant ce paradigme. Enfin, nous présentons des tableaux récapitulatifs des modèles conceptuels au regard des spécificités liées aux propriétés des listes proposées par [Vassiliadis and Sellis 1999; Blaschka et al. 1999b; Abelló et al. 2001a; Rafanelli 2003].

Paradigme entité-association

Dans [Tryfona et al. 1999], les auteurs proposent un modèle multidimensionnel appelé StarER qui a pour concepts de base des faits, des entités, des relations et des attributs. Le **fait** est un nouveau concept du schéma entité-association étendu. Il correspond à des données factuelles du monde réel qui partagent les mêmes propriétés. Le concept d'**entité** modélise les niveaux hiérarchiques des dimensions. Les **relations** correspondent à des associations entre des entités ou entre des entités et des faits. Elles modélisent les concepts de spécialisation, d'agrégation, de liens entre les entités d'une dimension. Les relations peuvent être de type N-M. Les relations entre les entités forment des hiérarchies. Ces dernières orientent la définition des dimensions. Les attributs sont des propriétés statiques des trois concepts présentés auparavant.

En résumé, les modèles conceptuels multidimensionnels basés sur le paradigme entité-association [Sapia et al. 1999; Tryfona et al. 1999; Sánchez et al. 1999; Franconi and Sattler 1999] représentent les concepts décisionnels *via* des extensions plus ou moins importantes des concepts d'entités et d'associations. Le paramètre est représenté par une entité spécialisée. La dimension est définie comme un ensemble d'entités spécialisées reliées par des associations binaires. Le fait est généralement représenté par une entité ou une association n-aire entre les entités représentant les paramètres de niveau le plus fin de chaque dimension.

La différence fondamentale entre la modélisation des systèmes d'information classiques et les SID est le regroupement et les agrégations de concepts. De ce fait, la représentation des différents concepts décisionnels rend difficile la compréhension du schéma. L'antagonisme de l'expressivité et de la simplicité du modèle prend tout son sens pour les schémas générés à partir de ces modèles. Ainsi, l'expressivité de ces modèles les rend difficilement compréhensibles par les décideurs alors que c'est l'une des caractéristiques de la modélisation multidimensionnelle.

De plus, les concepts utilisés (entités spécialisées, relations, ensemble de domaine) sont propres aux informaticiens mais pas aux décideurs. Ces derniers peuvent donc péniblement valider les schémas associés.

Paradigme Objet

Deux modèles orientés objet sont remarquables de par les concepts et les spécificités du SID qu'ils permettent de représenter.

Dans [Trujillo and Palomar 1998; Trujillo et al. 2002; Luján-Mora et al. 2006], les auteurs proposent un modèle multidimensionnel (GOLD) orienté objet. Ils proposent un profil (ensemble de stéréotypes, de valeurs typées et de contraintes) pour spécialiser UML à la modélisation multidimensionnelle. Le fait est représenté par une **classe avec le stéréotype** <<**Fact**>> contenant des attributs atomiques ou calculés correspondant aux mesures. Un fait est lié à un ensemble de dimensions par des relations d'agrégation. Ces dernières indiquent la granularité du fait. Une dimension est représentée par une **classe avec le stéréotype** <<**Dimension**>> et un graphe acyclique de niveaux représentés par des **classes avec le stéréotype** <<**Base**>> dont la racine est la classe associée à la dimension. Ces classes « Base » sont organisées suivant des hiérarchies. Les agrégations non pertinentes sont indiquées dans des notes UML mais, la structure complexe des mesures n'est pas représentée. Une association avec le stéréotype <<**RollupTo**>> définit les liens entre deux niveaux d'une dimension au sein d'une hiérarchie. Les auteurs proposent aussi des stéréotypes pour les associations afin de modéliser des traitements de dérivation [Luján-Mora et al. 2004b] à partir des sources de données. Les associations sont définies entre les classes du schéma des sources et les classes du schéma conceptuel du SID.

Dans [Abelló et al. 2002, 2006], le modèle multidimensionnel proposé est appelé YAM^2. Il est basé sur la spécialisation du méta-modèle du diagramme de classes UML. Il repose sur la dualité fait-dimension. Une **dimension** est définie comme un graphe connexe direct où chaque noeud est un **niveau**. Un niveau est un ensemble d'instance de même granularité dans une dimension. Le concept de **descripteur** définit des attributs faibles. Un fait est un graphe connexe direct où chaque noeud est une **cellule**. Le concept de cellule définit un ensemble d'instances d'un fait évaluées au même niveau de granularité pour chacune des ses dimensions liées. Le concept de **mesure** représente un attribut d'une cellule. Cette modélisation permet de représenter plusieurs faits et les dimensions partagées entre ces faits ; il est donc possible de définir des schémas en constellation. Le modèle permet de définir plusieurs rôles entre une dimension et un même fait. Ce type de relation entre les classes permet de représenter des hiérarchies non-strictes. Il permet de modéliser les relations N-N entre un fait et les dimensions connectées comme Tryfona en indiquant la cardinalité exacte entre les classes. Il permet aussi de représenter les mesures résultant de la combinaison de un à n dimensions connectées au fait ainsi que les agrégations pertinentes pour ce fait mais, il n'indique pas sur le schéma les liens entre les mesures. Néanmoins, il représente aussi beaucoup d'informations inutiles ; ce qui rend difficile la conception même par un concepteur décisionnel.

En résumé, les modèles conceptuels multidimensionnels orientés objet [Luján-Mora et al. 2006; Abelló et al. 2006; Prat and Akoka 2002] profitent de la popularité de la modélisation UML. Ils représentent les concepts multidimensionnels par des

classes avec des stéréotypes liés au domaine du décisionnel. Ces modèles sont des spécialisations d'UML au domaine du multidimensionnel par la définition de stéréotypes, de valeurs types et de contraintes. Ils héritent donc de la propriété des modèles orientés objets de favoriser la capitalisation de la connaissance en vue de la réutilisation.

Cependant, certains stéréotypes, tels que ceux pour désigner les niveaux, s'éloignent quelque peu de la sémantique multidimensionnelle avec les concepts de Base [Luján-Mora et al. 2006]. Ces modèles ont la particularité de représenter les niveaux comme une classe à part entière afin de représenter les hiérarchies qui les organisent.

De plus, ces modèles n'exploitent pas l'avantage du diagramme de classe UML qui permet de représenter les traitements. En effet, [Abelló et al. 2002] occultent la spécification des traitements et [Prat and Akoka 2002] transforment les méthodes en des attributs. Le seul de ces modèles [Luján-Mora et al. 2006] qui aborde les problématiques liées à la dynamique du SID étend le concept d'association au lieu d'utiliser la partie opération des classes UML ; ce qui introduit de nouveaux concepts et complexifie le schéma. De plus, les traitements modélisés sont liés uniquement à la dérivation des données. Ils ne sont pas liés à la préparation des données ; ils ne tiennent donc pas compte du contexte décisionnel des SID.

Paradigmes spécifiques

Nous présentons le modèle de [Golfarelli et al. 1998b] car nous utilisons au sein de l'équipe un formalisme qui s'en inspire et le modèle de [Vassiliadis et al. 2002a] car c'est le premier modèle qui a représenté certains traitements des SID.

Dans [Golfarelli et al. 1998b], les auteurs proposent un modèle appelé « Dimensional Fact Model » (DFM). Cette modélisation multidimensionnelle graphique distingue clairement les concepts tels que **les faits, les dimensions, les mesures, les attributs faibles et les hiérarchies**. Ce modèle propose un formalisme simple pour la représentation des données. Les agrégations pertinentes définies pour une mesure ne tiennent pas compte des dimensions liées car elles sont représentées à côté du nom de la mesure dans le fait.

Dans [Vassiliadis et al. 2002a, 2005], les auteurs proposent un modèle conceptuel qui permet de représenter les traitements d'extraction, de transformation et de chargement. Ils indiquent les concepts des sources de données qui participent à l'alimentation d'un concept du SID cible. Ce modèle est défini après la conception du schéma du SID car il repose sur les correspondances entre les sources et le SID. Il est compréhensible par les informaticiens, mais il l'est difficilement par les décideurs car il utilise de nombreux symboles pour représenter les spécificités de ces traitements. Il ne se rapproche pas de représentation multidimensionnelle des données par les utilisateurs car il définit des correspondances entre les données sources et les données cibles. De plus, il semble difficile de représenter sur un même schéma

tous les concepts du SID et celles des sources ainsi que leurs correspondances. Les schémas définis à partir de ce modèle sont donc généralement la représentation d'un sous-ensemble du SID.

En résumé, les modèles conceptuels multidimensionnels spécifiques [Golfarelli et al. 1998b; Cabibbo and Torlone 1998; Moody and Kortink 2000; Husemann et al. 2000; Tsois et al. 2001; Vassiliadis et al. 2002a, 2005] utilisent des paradigmes complètement nouveaux pour représenter les concepts du domaine décisionnel. L'avantage de ces modèles est qu'ils utilisent exclusivement des concepts liés au décisionnel sauf [Tsois et al. 2001], en l'occurrence les concepts de faits, dimensions, mesures, paramètres et hiérarchies.

Cependant, ces paradigmes constituent un frein pour les concepteurs décisionnels qui doivent les appréhender alors qu'ils maîtrisent des paradigmes très connus pour leur expressivité et la grande sémantique qu'ils permettent d'exprimer.

Parmi ces modèles, un seul [Vassiliadis et al. 2002a] permet de représenter à la fois les données et les traitements. Les traitements représentés sont liés uniquement à la dérivation des données ; les traitements liés à la préparation des données, soit au contexte décisionnel ne sont pas abordés.

Synthèse des modèles conceptuels multidimensionnels

Les modèles conceptuels multidimensionnels proposés reposent sur deux paradigmes existants (entité-association et objet) et sur des paradigmes totalement nouveaux. Ces modèles se concentrent principalement sur la modélisation des données. Ils permettent de représenter l'ensemble des concepts multidimensionnels avec des terminologies plus ou moins proches de la sémantique décisionnelle. De même, les spécificités générales liées aux listes de propriétés, autrement dit liées à la définition d'un modèle conceptuel, telles que « un formalisme indépendant de l'implantation » et « le paradigme de séparation de la structure et des données » sont supportées par ces modèles.

Cependant, les spécificités liées aux propriétés caractérisant le domaine multidimensionnel ne sont prises en compte que partiellement voire non traitées comme indiqué dans le tableau de la figure 2.5.

En ce qui concerne, la *structuration complexe des dimensions*, l'ensemble des modèles conceptuels permettent de représenter les **hiérarchies simples**. De même, les **hiérarchies multiples et alternatives** sont définies par la majorité des modèles conceptuels. Cependant, seuls [Tryfona et al. 1999; Luján-Mora et al. 2006; Abelló et al. 2006] permettent de représenter les **hiérarchies non-strictes** en spécifiant les cardinalités ou multiplicités N. Par ailleurs, aucun de ces modèles ne permet de représenter les **éléments terminaux multiples** sur le schéma. Les **rôles d'une dimension** sont modélisés uniquement par [Abelló et al. 2006]. Cependant, ces derniers

ne permettent pas de représenter le concept de dimension dégénérée contrairement à [Luján-Mora et al. 2006] qui le représentent comme un attribut de la classe fait.

La *structuration complexe des faits et des mesures* est aussi prise en compte partiellement. Plus précisément, la majorité des modèles conceptuels de données permettent de représenter les **mesures multiples** mais, ils ne permettent pas toujours de représenter les **faits multiples** [Prat and Akoka 2002; Moody and Kortink 2000; Husemann et al. 2000; Tsois et al. 2001; Vassiliadis et al. 2002a, 2005]. Les **mesures dérivées** sont évoquées par [Golfarelli et al. 1998b; Tryfona et al. 1999] mais ils ne permettent pas de les représenter sur le schéma du SID. [Abelló et al. 2006] permettent de représenter toutes les mesures au croisement des combinaisons de une à n dimensions et des mesures dérivées. Cependant, les mesures résultants des différentes combinaisons ne relèvent pas de la structure complexe des mesures mais elles relèvent de la matérialisation des vues de données les plus sollicitées par les décideurs. Cette thématique n'est pas traitée dans cette thèse. [Abelló et al. 2006; Sánchez et al. 1999; Franconi and Sattler 1999] présentent les mesures dérivés à partir de l'application de fonctions d'agrégation mais, ils ne présentent pas les **liens entre les mesures** sur le schéma. Aucun, modèle ne permet de représenter ces derniers. Par ailleurs, seul [Luján-Mora et al. 2006] permet de représenter **fait dégénéré**.

Une autre propriété que doivent garantir les modèles conceptuels multidimensionnels est la *cohérence de l'interrogation* des données *via* la définition de la **pertinence des agrégations**. Cependant, seuls [Luján-Mora et al. 2006; Abelló et al. 2006; Golfarelli et al. 1998b] permettent de spécifier sur le schéma les fonctions d'agrégation qui peuvent être appliquées à une mesure. [Golfarelli et al. 1998b] ne tiennent pas compte des dimensions connectées au fait.

La *dynamique du SID* est représentée par deux modèles et elle est limitée aux traitements de dérivation du SID à partir des sources de données. Plus précisément, [Vassiliadis et al. 2002a, 2005; Luján-Mora et al. 2006] proposent de modéliser les traitements ETL après la conception du schéma des SID. De plus, ils les représentent dans un autre schéma que celui du SID. Ces travaux n'abordent donc pas les traitements de préparation des données liés au contexte décisionnel.

Propriétés / Paradigmes	Modèles conceptuels multidimensionnels		Structure complexe des dimensions						Structure complexe des faits et des mesures					Cohérence de l'interrogation	Aspect dynamique	
			hiérarchies simples	hiérarchies multiples et alternatives	hiérarchies non-strictes	dimension dégénérée	rôles multiples des dimensions	éléments terminaux multiples	faits multiples	mesures multiples	mesure dérivée	liens entre mesures	faits dégénérés	agrégations pertinentes des mesures	traitements de dérivation des données	traitements de préparation des données
Entité-Association	[Sapia et al., 1999]	M/ER	X	X						X						
	[Tryfona et al., 1999]	StarER	X	X	X					X						
	[Sanchez et al., 1999]	IDEA	X	X					X	X	X					
	[Franconi and Sattler, 1999, Franconi and Kamble, 2004]	GMD	X	X					X		X					
Objet	[Trujillo and Palomar, 1998, Trujillo et al., 2002, Lujan-Mora et al., 2006]	GOLD	X	X	X	X			X	X			X	X	X	
	[Prat and Akoka, 2002]	UMM	X													
	[Abello et al., 2002]	YAM²	X	X	X		X		X		X			X		
Spécifique	[Golfarelli et al., 1998]	DFM	X	X						X				X		
	[Cabibbo and Torlone, 1998]	MD	X	X					X	X	X					
	[Pedersen and Jensen, 1999]		X	X												
	[Moody and Kortink, 2000]		X	X												
	[Husemann et al., 2000]		X	X												
	[Tsois et al., 2001]	MAC	X	X												
	[Vassiliadis et al., 2002a , Vassiliadis et al., 2005]		X	X											X	

FIG. 2.5 – Comparatif des modèles conceptuels multidimensionnels

2.2.3 Bilan et positionnement par rapport aux modèles multidimensionnels

Les nombreuses propositions de modèles conceptuels, à l'unisson, ne permettent pas de représenter toutes les spécificités caractérisant la structure complexe des dimensions, des faits et des mesures ainsi que la pertinence des agrégations. De plus, elles ne représentent que les aspects statiques du SID. Ce constat est vrai pour les modèles conceptuels basés sur le paradigme objet, bien qu'ils permettent de représenter le plus grand nombre de concepts et de spécificités multidimensionnels.

Nous nous positionnons par rapport au modèle orienté objet proposé par [Luján-Mora et al. 2006] car il permet de représenter le plus grand nombre de spécificités parmi les modèles orientés objet. Cependant, il présente les inconvénients suivants :

- représentation partielle de la structure complexe des dimensions : les rôles multiples ainsi que les éléments terminaux multiples ne sont pas définis,
- représentation partielle de la structure complexe des faits et des mesures : les mesures dérivées et les liens entre les mesures d'un même fait ne sont pas abordés,
- modélisation non systématique des agrégations pertinentes : seuls les agrégations non pertinentes sont représentées et les fonctions d'agrégation valables par mesure ne sont pas définies,
- prise en compte partielle des traitements du SID : les traitements liés à la préparation des données ne sont pas abordés.

2.3 Les méthodes de développement des SID

La mise en place d'un SID requiert une méthode spécifique en raison des singularités des SID par rapport aux systèmes d'information classiques [Golfarelli and Rizzi 1998a]. Une dizaine de méthodes de développement de SID a été proposée associée à certains modèles présentés dans la section 2.2.

Dans cette section, nous présentons les méthodes de développement de SID en fonction de leur démarche en vue de dresser un bilan comparatif de leurs caractéristiques. Les chercheurs et les industriels reconnaissent trois types de démarches.

- la démarche **descendante**, orientée utilisateurs, est la première qui a fait l'objet de travaux. Elle définit le schéma conceptuel à partir des besoins des utilisateurs du SID,
- la démarche **ascendante**, orientée données, définit le schéma du SID à partir des schémas des sources de données,
- la démarche **mixte** prend en entrée du processus de développement les besoins des utilisateurs et les sources de données afin de produire le schéma du SID.

Suivant les travaux de [Srivastava and Chen 1999], le processus de développement est basé sur 5 tâches : **le pré-développement des métiers, la sélection de l'architecture, la création du schéma, l'alimentation du système et la maintenance**. Nous nous focalisons sur les phases d'analyse et de conception, en l'occurrence sur les trois premières tâches. La tâche de pré-développement des métiers correspond à l'analyse des besoins utilisateurs et à l'analyse des sources de données. Les méthodes existantes proposent des définir 1 à 2 modules décisionnels de l'architecture qui sont l'entrepôt de données et le magasin de données (cf. définitions ci-après).

Définition 2.11 (Entrepôt de données) *Il est présenté par l'un des pionniers, Bill Inmon [Inmon 1996], comme un ensemble de données orientées sujet, intégrées, non volatiles, historisées, résumées et disponibles pour l'interrogation et l'analyse.*

- **orientées sujet** : seules les données utiles au processus décisionnel des différents domaines d'activités de l'organisation sont définies,
- **intégrées** : les données sources proviennent de sources de données multiples et hétérogènes. Elles doivent être nettoyées, unifiées, converties et consolidées,
- **non volatiles** : contrairement aux données des systèmes transactionnels qui sont modifiées lors de chaque transaction, les données de l'entrepôt ne sont généralement pas modifiées et supprimées,
- **historisées** : les données sont conservées au cours du temps afin d'évaluer des tendances et de faire des corrélations sur des périodes de temps,
- **disponibles** : les décideurs doivent accéder aux données de différentes manières (requête, rapport, analyse, fouille de données),
- **résumées** : les données manipulées par les systèmes transactionnels sont très nombreuses et très détaillées. Cependant, tout ce détail n'est pas pertinent pour les décideurs. Les données sont donc agrégées, calculées et transformées.

Définition 2.12 (Magasin de données) *Un ensemble cohérent de données répondant aux besoins spécifiques d'une classe de décideurs. C'est un sous ensemble de l'entrepôt de données propre à un groupe de décideurs.*

En raison de la spécificité des SID de dériver de sources de données et de préparer les données pour faciliter la prise de décision, nous ajoutons de manière transversale au processus de développement la prise en compte des traitements du SID. Nous évaluons donc les méthodes de développement en fonction des trois premières tâches et de la prise en compte des traitements du SID.

De plus, au regard du temps requis pour la mise en place d'un SID, nous évaluons aussi les propositions d'automatisation, de parallélisation des tâches ainsi que de capitalisation et de réutilisation de la connaissance liées à ces méthodes.

2.3.1 Démarche descendante

Dans cette section, nous présentons les principes des méthodes basées sur une démarche descendante afin de présenter leurs avantages et leurs inconvénients.

Dans [Kimball 1996; Kimball et al. 1998], les auteurs fournissent une véritable discussion sur la modélisation multidimensionnelle à partir d'une compilation d'expériences. Les auteurs présentent les problèmes rencontrés lors de la mise en place de SID dans différents domaines fonctionnels. La méthode proposée a deux caractéristiques principales. La première est l'**architecture de bus**. Elle consiste à créer une série de magasins de données respectant le modèle logique du schéma en étoile à partir desquels, l'entrepôt de données fédérées de l'organisation est créé. La deuxième caractéristique est le **cycle de vie multidimensionnel** qui précise la gestion d'un projet décisionnel à partir des besoins métier. La méthode est basée sur une démarche itérative composée de cinq étapes principales : « Planification du projet », « Définition de besoins métiers », « Choix de la technologie », « Modélisation des données » et « Spécification et développement des applications ». Les auteurs définissent deux tâches finales liées au déploiement et à la maintenance du SID.

Dans [Tsois et al. 2001], la méthode proposée permet de construire un schéma conceptuel basé sur le modèle de données MAC défini par ces auteurs. Les besoins sont exprimés sous forme de requêtes. Le résultat des requêtes des utilisateurs est présenté sous forme tabulaire où les colonnes et les lignes sont composées des paramètres des dimensions. A partir de ces tableaux, les auteurs définissent un ensemble minimal de concepts (niveau, hiérarchie, dimensions, cubes, attributs et des relations de manipulation de données).

Dans [Prat and Akoka 2002], la méthode est basée sur le modèle multidimensionnel unifié. Les auteurs élaborent des diagrammes de classes UML à partir de l'expression informelle non structurée des besoins par les décideurs. Ils utilisent des règles pour transformer et enrichir les diagrammes de classes afin de prendre en compte la modélisation multidimensionnelle. Les classes simples, les classes d'association et les associations 1-1 et 1-N sont évaluées au cours de deux étapes de simplification pour obtenir un schéma pivot. La première étape consiste à enlever la dynamique du modèle en substituant les méthodes des classes du diagramme de classes UML par des attributs. La seconde étape consiste à transformer les liens N-M en des liens 1-N. La transformation du schéma pivot en un modèle multidimensionnel unifié est réalisée au niveau logique.

Synthèse des méthodes basées sur une démarche descendante

La démarche descendante vise à satisfaire les utilisateurs en prenant en compte uniquement ce dont ils ont besoin. Cependant, l'inconvénient majeur est l'évolution permanente des besoins des utilisateurs voire la non-stabilité des besoins comme précisé par [Moody and Kortink 2000]. Ces besoins peuvent donc difficilement être définis. De plus, la prise en compte des schémas des sources de données est très

tardive car les systèmes sources sont évalués uniquement après l'obtention du schéma conceptuel du SID. La démarche travaille sur un périmètre d'analyse restreint aux besoins des utilisateurs. Ce dernier point présente un inconvénient majeur ; le schéma conceptuel produit peut parfois s'avérer impossible à mettre en oeuvre. Cela peut être dû à la difficulté voire à l'impossibilité de disposer des données ou encore à des problèmes de coûts.

Qui plus est, ces trois méthodes ne définissent que deux des tâches de référence car elles ne guident pas le choix de l'architecture et elles ne proposent pas de modéliser les traitements du SID. Une de ces méthodes, soit [Kimball 1996], aborde les problèmes liés à l'alimentation du SID mais elle ne l'explicite pas. Il manque aussi une démarche, dont toutes les tâches sont détaillées, aux méthodes [Kimball 1996; Tsois et al. 2001].

2.3.2 Démarche ascendante

Dans cette section, nous présentons les méthodes basées sur une démarche ascendante en expliquant comment elles décrivent un schéma multidimensionnel à partir d'un schéma entité-association ou d'un schéma UML.

Dans [Golfarelli and Rizzi 1998b], les auteurs proposent de dériver à partir des sources de données entité-association ou relationnelle, le schéma du SID qui respecte leur modèle conceptuel de données DFM. La méthode proposée repose en partie sur un algorithme. Les concepts de faits, de dimensions, d'attributs et de hiérarchies sont définis. Le fait est défini comme une entité ou une association entre plusieurs entités fréquemment mises à jour et représentant un intérêt premier de l'organisation. Chaque fait devient la racine d'un arbre dont les noeuds sont les attributs de l'entité (fait candidat) et des entités liées suivant l'algorithme. Ce dernier tient compte uniquement des relations 1-N car les relations N-M ne sont pas évaluées. Sur cet arbre, des élagages et des greffes des branches non guidés sont réalisés afin que les données contenues dans l'arbre soient pertinentes par rapport à l'application. Puis, les dimensions sont définies en considérant les noeuds fils, de la racine de l'arbre raffiné. Par la suite, les mesures des faits sont définies à partir des attributs numériques de l'arbre. Un fait peut ne pas avoir d'attribut. Enfin, les attributs qui composent une hiérarchie sont définis s'il existe une relation 1-N entre un noeud et ses descendants. A cette étape de la méthode, il est encore possible de raffiner l'arbre pour ajouter des niveaux. La méthode suppose que les concepteurs décisionnels connaissent très bien le domaine d'activité pour réaliser les élagages et les greffes.

Dans [Cabibbo and Torlone 1998, 2000], la méthode vise à construire le modèle multidimensionnel MD défini par ces auteurs à partir d'un schéma entité-association présentant une vue intégrée des bases de données transactionnelles. La méthode est composée de quatre étapes :

1. la définition des faits et des dimensions suite à une analyse globale et manuelle des schémas sources,

2. la restructuration du schéma entité-association des sources pour faire apparaître les faits et les dimensions, et les niveaux hiérarchiques,
3. la dérivation du graphe dimensionnel à partir du schéma entité-association restructuré. Le graphe dimensionnel est un schéma entité-association étendu,
4. la transformation du graphe dimensionnel dans un schéma multidimensionnel MD.

Dans [Husemann et al. 2000], les auteurs proposent de définir le schéma du SID suivant le modèle multidimensionnel proposé par ces mêmes auteurs. Cette méthode utilise un acquis des méthodes de conception des systèmes d'information classiques qui est la définition des dépendances fonctionnelles pour déterminer les concepts multidimensionnels. Cette méthode comprend les étapes :

- la spécification et l'analyse des besoins : après le choix des sources de données pertinentes par les utilisateurs, les concepteurs décisionnels accompagnés de ces derniers doivent déterminer le caractère optionnel et le niveau d'importance métier des attributs. En sortie de cette étape, il y a le schéma semi-formel des concepts multidimensionnels et une sorte de dictionnaire listant les caractéristiques des attributs. Le schéma semi-formel est défini à partir des dépendances fonctionnelles entre les mesures définies par les décideurs et les attributs définis dans le dictionnaire. Pour chaque mesure un sous-ensemble minimal d'attributs est défini. Ce sous-ensemble permet de définir les niveaux les plus fins des hiérarchies. A chaque élément du sous-ensemble est associé une dimension. Les dépendances fonctionnelles qui ont le même ensemble de départ définissent les mesures qui composent un fait. Pour déterminer les hiérarchies au sein d'une même dimension, il faut utiliser les dépendances fonctionnelles entre les attributs des dimensions et construire un graphe direct. Les attributs faibles des dimensions sont les attributs qui ont été définis comme optionnels.
- la modélisation conceptuelle : le schéma semi-formel est transformé en un schéma conceptuel multidimensionnel. Ce schéma présente chaque fait avec leurs mesures et les dimensions liées.

Cette méthode utilise les dépendances fonctionnelles pour déterminer les faits et les dimensions ; ce qui permet une définition formelle de ces concepts. Cependant, elle suggère la participation des utilisateurs pour la désignation des sources de données pertinentes alors que les utilisateurs, par définition, ne maîtrisent pas les schémas des sources de données.

Dans [Moody and Kortink 2000], les auteurs proposent de construire un schéma en étoile ou ses différentes variantes (flocon, constellation) en utilisant la démarche utilisée pour la conception d'un modèle entité-association telle que Merise. Les auteurs martèlent que le paradigme entité-association peut être utilisé aussi bien pour la construction des systèmes d'information classiques que des SID. Cependant, la méthode repose sur trois étapes (« la classification des entités », « l'identification des hiérarchies » et « la production des modèles multidimensionnels des magasins de données ») qui évoquent les sources transactionnelles, le processus d'extraction des données, l'entrepôt de données, les processus de chargement des données de l'entrepôt vers les magasins, les magasins et les utilisateurs finaux. Les différentes étapes pour la mise en place complète du SID ne sont pas explicitées. Par ailleurs,

les auteurs présentent la construction de schémas de magasins de données en quatre étapes :

- classification des entités en trois catégories entités de transaction, entités composantes, entités de classification,
- identification des hiérarchies : une hiérarchie est définie comme une séquence d'entités reliées par des associations 1-N, toujours ordonnées dans le même sens. L'entité liée au niveau le plus fin de la hiérarchie est caractérisée par une absence de relation 1-N avec une autre entité. L'entité associée au niveau le plus élevé est caractérisée par une absence de relations N-1 avec d'autres entités,
- définition du schéma dimensionnel : les différents schémas dérivés du schéma en étoile obtenu sont construits par l'application de deux opérations. La première opération est la fusion entre deux niveaux d'une hiérarchie n_f (niveau fin) et n_a (niveau agrégé). Elle consiste à inclure n_a dans n_f. La seconde opération est l'agrégation. Elle consiste à agréger une entité pour obtenir des données agrégées,
- évaluation et raffinement du schéma dimensionnel : cette dernière étape a pour objectif d'améliorer le modèle. Par exemple, en regroupant les faits qui ont les mêmes clés.

Cependant, seules les relations 1-N sont prises en compte.

Synthèse des méthodes basées sur une démarche ascendante

La démarche ascendante a pour objectif de pouvoir répondre à l'ensemble des requêtes et des analyses possibles sur les données des sources et ainsi d'anticiper les besoins futurs. L'avantage de cette démarche est qu'elle représente exactement la sémantique sous-jacente des sources de données et elle facilite la définition des hiérarchies. Cependant, le périmètre des données des sources est parfois trop large et l'étude des schémas des sources requiert du temps et des ressources humaines importantes. Cette démarche s'avère lourde lorsque le nombre de sources ou la complexité des sources sont importants. Elle se base sur l'assertion idéaliste que toutes les données sont contenues dans le schéma des sources alors que les données des sources peuvent être insuffisantes pour répondre aux besoins utilisateurs [Gardner 1998; List et al. 2000]. Elle présente l'inconvénient majeur de ne pas prendre en compte les besoins des décideurs car le schéma du SID défini ne répond pas forcément aux besoins des utilisateurs.

De plus, ces méthodes permettent la mise en place d'un seul module qui est l'entrepôt de données ou le magasin de données et elles ne guident pas les concepteurs dans le choix de l'architecture décisionnelle. Elles ne couvrent donc pas tout le processus de développement des SID. Par ailleurs, elles débutent par une définition qualitative des mesures ou encore par la définition des mesures par les décideurs. La première procédure n'est pas précise. La seconde rencontre des limites car elle requiert que les utilisateurs se réfèrent aussi aux schémas entité-association alors que ces derniers n'ont pas pour objectif d'être compréhensibles par les utilisateurs [Diviné 1982].

Qui plus est, ces méthodes se focalisent sur la modélisation des données et elles n'abordent pas la dynamique du SID.

2.3.3 Démarche mixte

Dans cette section, nous présentons les méthodes basées sur une démarche mixte qui est caractérisée par l'analyse des besoins des utilisateurs (démarche descendante produits des schémas idéaux) et l'analyse des sources (démarche ascendante produits des schémas candidats). Cette démarche inclut une tâche de confrontation généralement entre un « schéma idéal » produit à partir des besoins des utilisateurs et des « schémas candidats » produits à partir des schémas des sources de données. De cette tâche dépend la fiabilité du SID par rapport aux besoins des utilisateurs et aux sources. Cependant, l'inconvénient principal est que la tâche de confrontation est longue, proportionnellement au nombre de schémas comparés. Ainsi, nous évaluons aussi ces méthodes en fonction du nombre de schémas idéaux et candidats. Le nombre et le type de schéma en entrée de la phase de confrontation permet d'évaluer la fiabilité du SID et le gain de temps.

Dans [Bonifati et al. 2001], la méthode proposée définit le schéma du SID suivant le modèle en étoile. Elle repose sur cinq étapes. Au cours de la démarche descendante, les auteurs utilisent le paradigme GQM (Goal/Question/Metric) [Basili et al. 1994] afin de générer manuellement un schéma idéal à partir des besoins utilisateurs. Au cours de la démarche ascendante, les schémas candidats sont générés automatiquement suivant un algorithme. Puis, ces n schémas candidats sont confrontés à un seul schéma idéal.

Dans [Cavero et al. 2001], les auteurs suggèrent de ne pas définir une méthode de développement des SID distincte de celle des SI classiques mais de l'intégrer dans celle des SI afin de ne pas rendre plus complexe le processus. La méthode qu'ils proposent, appelée MIDEA, représente les besoins des utilisateurs *via* leur modèle conceptuel multidimensionnel IDEA suivant une démarche analogue à celle relative à la conception d'un modèle entité-association, selon les auteurs. Cette démarche n'est pas explicitée. La méthode autorise la création de nouveaux schémas entité-association et la modification des schémas existants afin de satisfaire les besoins utilisateurs. Ce dernier point relève d'un problème très complexe (non traité dans l'article) lié à la modification et la vérification de schémas de bases de données en production qui est celui des évolutions de schémas [Blaschka et al. 1999a]. De plus, la fonction du SID n'est pas de modifier les schémas des sources mais de faire des accès en lecture pour des analyses et des recopies pour les calculs.

Cette méthode présente d'autres inconvénients car la phase d'analyse des besoins utilisateurs est minimisée. Elle se focalise sur la démarche ascendante ; la phase descendante est réduite à la validation des schémas candidats par les utilisateurs.

Dans [Carneiro and Brayner 2002], les auteurs proposent une méthode mixte qui capitalise la connaissance dès le début du projet *via* des méta données pour pallier au problème de changement continu des employés d'un poste. Les auteurs proposent une méthode, X_META, basée sur un cycle de vie itératif incrémental pour concevoir des magasins mais également l'entrepôt. Cette méthode comprend plusieurs itérations décomposées en phase, sous-phase, groupe, module et activité dont la première est un prototypage du SID. Le SID résulte des trois principales phases appelées le prototypage, le développement et la production.

Cette méthode diffère des autres méthodes car elle suggère de gérer les méta données en vue d'une réutilisation. Elle propose de construire plusieurs schémas idéaux, mais le nombre de schéma candidat n'est évoqué. De plus, aucune des tâches n'est explicitée et elle ne repose pas sur un modèle.

Dans [Phipps and Davis 2002], les auteurs proposent d'obtenir un schéma conceptuel multidimensionnel M/ER en dérivant le schéma entité-association des sources de données suivant cinq étapes explicites. Ils utilisent un algorithme qui identifie les faits candidats en listant toutes les entités qui ont des attributs numériques. Pour chaque entité, un schéma candidat sera créé. Les attributs du fait sont ceux de l'entité associée qui sont clés primaires ou numériques. Les éventuels attributs non numériques sont regroupés en une dimension. Enfin, toutes les entités connectées à l'entité associée au fait sont transformées en une dimension avec tous les attributs de l'entité connectée. Une évaluation des schémas candidats est faite suivant qu'ils répondent dans l'ordre à la clause « From » et à la clause « Select » d'une dizaine de requêtes des utilisateurs.

Les tâches qui précèdent celles d'évaluation et de raffinement des schémas candidats sont automatiques mais, celles-ci même ne le sont pas. Le principe du décompte du nombre d'attributs numériques pour définir de manière quantitative le fait est pertinent.

Dans [Luján-Mora and Trujillo 2003], la méthode proposée est appelée « Data warehouse Engineering Process ». Elle est basée sur le paradigme objet avec le langage de modélisation UML et le processus de développement unifié UP. Les auteurs utilisent quatre types de schémas au cours du processus afin de modéliser les sources de données, les objets multidimensionnels, le stockage physique et les accès aux données en fonction du schéma physique. Cette méthode s'articule autour de trois grandes étapes qui sont l'analyse, la conception, l'implantation et les tests.

Elle définit un schéma idéal et un schéma candidat. Elle est basée sur un modèle relativement complet défini suivant le profil UML adapté au multidimensionnel que proposent les auteurs, mais le guidage méthodologique n'est pas explicité.

Dans [Ghozzi et al. 2005], les auteurs proposent une méthode qui déroule complètement les deux démarches (ascendante et descendante).

La démarche descendante repose sur trois étapes. Elle définit un schéma idéal à partir des besoins utilisateurs exprimés sous forme de requêtes-types. La démarche ascendante définit un seul schéma candidat en six étapes (détermination des faits, détermination des dimensions, définition de la dimension temporelle, définition des granularités, organisation des paramètres des dimensions, expression des contraintes).

Dans [Soussi et al. 2005], la méthode proposée définit le schéma du SID à partir des besoins utilisateurs exprimés sous forme de tableaux n-dimensionnels et les schémas entité-association des sources. A partir des besoins utilisateurs, plusieurs schémas idéaux. De même, plusieurs schémas en étoile sont dérivés des sources afin de constituer la base de données de l'outil utilisé pour l'expression des besoins et de confronter les besoins.

Synthèse des méthodes basées sur une démarche mixte

La démarche mixte cumule les avantages des démarches précédentes. Néanmoins, elle peut facilement en multiplier les inconvénients au regard de l'avantage très important qu'elle procure. Les méthodes qui proposent de confronter un schéma résultant de chacune des démarches limitent cet inconvénient et elles contribuent à garantir la fiabilité du SID en évaluant tant les besoins des utilisateurs que les schémas des sources. [Ghozzi et al. 2005; Luján-Mora and Trujillo 2003] ne produisent qu'un schéma idéal et un schéma candidat ; cela implique que le nombre de comparaisons est invariable indépendamment de la taille du projet. Cependant, les méthodes [Bonifati et al. 2001; Phipps and Davis 2002; Soussi et al. 2005] produisent n schémas idéaux ou n schémas candidats, ils tendent vers une complexité croissante du processus de développement.

Un autre inconvénient de ces méthodes est qu'elles se focalisent sur la modélisation des données. De plus, la majorité de ces méthodes minimise la phase d'analyse des besoins des utilisateurs voire l'occultent [Cavero et al. 2001; Phipps and Davis 2002; Carneiro and Brayner 2002; Luján-Mora and Trujillo 2003] car elle passe directement à la modélisation conceptuelle du SID. De même, elle ne guide pas le choix de l'architecture du SID car elle définisse un [Bonifati et al. 2001; Cavero et al. 2001; Ghozzi et al. 2005; Soussi et al. 2005] voire deux modules du SID [Luján-Mora and Trujillo 2003]. Elle ne couvre donc pas le tout le processus de développement.

Comme la tâche du choix de l'architecture, les problématiques liées à l'alimentation du SID et plus généralement à la dynamique du SID ne sont pas abordées. Seule, la méthode proposée par [Luján-Mora and Trujillo 2003] permet de définir des traitements d'alimentation, en l'occurrence les traitements ETL.

Tous ces travaux, sauf [Carneiro and Brayner 2002], n'abordent pas la problématique de capitalisation et encore moins de réutilisation de la connaissance. Ces derniers évoquent la capitalisation de la connaissance, mais ils ne proposent ni de modèle ni d'outil. Cependant la complexité et le temps important requis pour la mise en place des SID sollicitent de tels mécanismes.

2.3.4 Bilan et positionnement par rapport aux méthodes de développement

De nos jours, les propositions de méthodes prennent en compte les besoins des utilisateurs et les sources de données existantes pour la définition du schéma du SID afin de définir un SID fiable par rapport aux besoins des utilisateurs et aux sources.

Cependant, parmi ces méthodes, aucune méthode n'est largement reconnue car toutes les tâches précédant la création du schéma ne sont pas traitées complètement par les méthodes existantes. Plus précisément, elles présentent les inconvénients suivants :

- la phase d'analyse des besoins est réduite ou occultée : les méthodes de développement passent directement à la modélisation des données du SID. Les besoins des utilisateurs peuvent donc ne pas être satisfaits par la mise en place du SID. De plus, la confrontation est alourdie suivant le nombre de schémas résultant de l'analyse des besoins utilisateurs et sources,
- le choix de l'architecture n'est pas guidée : les méthodes de développement permettent la mise en place d'un voire deux modules architecturaux du SID,
- la prise en compte des traitements du SID est partielle et faible : seuls les traitements ETL sont analysés et ce par une seule méthode [Luján-Mora and Trujillo 2003],
- la capitalisation et la réutilisation ne sont pas favorisées : les méthodes de développement ne capitalisent pas la connaissance. Cette capitalisation permettrait d'assurer les transferts de connaissances entre les différents concepteurs décisionnels et aussi de favoriser un gain de temps lors de la mise en place des SID.

De plus, la phase de confrontation située après la conception du schéma conceptuel du SID rend le développement complexe car toute erreur implique la modification de la modélisation conceptuelle. De même, elle est empirique car d'une part, elle ne repose pas sur un support formel et d'autre part, les concepteurs ne sont pas guidés. Qui plus est, aucune de ces méthodes ne propose une phase de prototypage explicite, ce qui implique que les insatisfactions des utilisateurs sont mises en lumière qu'après la réalisation complète du SID.

Afin de mieux comprendre les inconvénients liés à la phase d'analyse ainsi qu'à la capitalisation et à la réutilisation, dans les deux sections suivantes, nous présentons l'état de l'art des méthodes propres à la phase d'analyse. Puis, nous présentons les propositions relatives à la réutilisation dans le domaine des SID et certains apports de la réutilisation dans le domaine des systèmes d'information que nous utilisons pour la définition de notre méthode de développement dans un contexte de réutilisation.

Méthodes de développement de SID	Approche	Modèle	Démarche globale/ tâche par tâche	Démarche explicite/implicite	Automatisation Parallélisation	Prototypage/ Cycle de vie	Analyse des besoins utilisateurs	Choix de l'architecture	schémas idéaux	schémas candidats	Modules décisionnels	Prise en compte des traitements	Réutilisation
[Kimball, 1996, Kimball et al., 1998]	D	Etoile	tâche	implicite		itératif	réduite	abordée	1		entrepôt		
[Tsois et al., 2001]	D	MAC	globale	implicite			réduite		1		entrepôt		
[Prat and Akoka, 2002]	D	UMM	tâche	explicite			réduite		1		entrepôt		
[Golfarelli and Rizzi, 1998b]	A	DFM	tâche	explicite	semi-auto				1		entrepôt		
[Cabibbo and Torlone, 1998,Cabibbo and Torlone, 2000]	A	MD	tâche	explicite	1er et 2ième tâches parallèles					1	entrepôt		
[Husemann et al., 2000]	A	Huseman et al	tâche	explicite						1	entrepôt		
[Moody and Kortink, 2000]	A	Etoile	globale	implicite		itératif				1	entrepôt		
[Bonifati et al., 2001]	M	Etoile	tâche	explicite	semi-auto	itératif	développée		1	n	entrepôt		
[Cavero et al., 2001]	M	IDEA	globale	implicite			réduite		1	n	entrepôt		
[Carneiro and Brayner, 2002]	M		globale	implicite		itératif prototype	occultée		n		entrepôt, magasin		
[Phipps and Davis, 2002]	M	M/ER	tâche	explicite	auto		occultée			n	entrepôt		
[Lujan-Mora and Trujillo, 2003]	M	GOLD	tâche	implicite			réduite		1	1	entrepôt, magasin	ETL	
[Ghozzi et al., 2005] MMC (Modèle Multidimensionnel Contraint)	M	MMC	tâche	explicite			développée		1	1	magasin		
[Soussi et al., 2005]	M	Etoile	tâche	explicite	semi-auto		développée		n	n	magasin		

D/A/M : descendante / ascendante / mixte
DC : diagramme de classes UML

FIG. 2.6 – Comparatif des méthodes de développement de SID

2.4 Les méthodes d'analyse des SID

Dans cette section, nous présentons les méthodes d'analyse du SID en mettant en avant les apports et les inconvénients de ces propositions pour la mise en place rapide d'un SID pertinent.

La tâche d'analyse située en amont du processus de développement a pour objectif de définir l'environnement du projet et la majorité des données et des traitements nécessaires au processus de développement. Elle impacte donc tout le processus de développement. De plus, le SID à pour fonction de centraliser plusieurs domaines d'activités ; il est donc transversal à l'organisation. Il vise donc à répondre aux besoins des décideurs stratégiques et des décideurs tactiques qui ont parfois des intérêts divergents. Les spécificités du SID et l'importance de la tâche d'analyse au sein du processus de développement justifient une méthode d'analyse des besoins spécifique au SID [Winter and Strauch 2003].

De plus, 80% des projets ne répondent pas aux besoins des décideurs [Schiefer et al. 2002] car cette étape est occultée ou pas développée dans des projets réels [Giorgini et al. 2005]. Ces chiffres, s'expliquent par le fait de nombreuses méthodes de développement de SID commencent directement par la conception du schéma conceptuel et elles ne proposent pas une phase d'analyse spécifique au SID comme mis en avant dans le bilan 2.3.4. Ils sont aussi le fruit d'une mauvaise communication entre les utilisateurs et les informaticiens [Mazon et al. 2005]

Dans [Winter and Strauch 2003], les auteurs définissent un processus d'analyse des besoins composé des 5 tâches suivantes que nous renommons :

1. collecte les besoins des utilisateurs correspond à la tâche « déterminer les besoins d'information des utilisateurs »,
2. prise en compte des sources correspond à la tâche « comparer les besoins d'information avec les sources de données »,
3. confrontation des besoins correspond à la tâche « évaluer et homogénéiser les besoins d'information »,
4. définition de priorités correspond à la tâche « établir des priorités parmi les besoins à satisfaire »,
5. formalisation des besoins correspond à la tâche « spécifier formellement les besoins manière à faciliter la validation des besoins par les utilisateurs ».

Ils indiquent les listes des tâches associées à chaque étape suivant les recommandations du responsable de projet. Cependant, ils n'indiquent pas comment réaliser chaque tâche. De plus, ils ne fournissent pas de modèle pour la collecte et la formalisation des besoins.

Néanmoins, ces travaux énoncent trois principes importants pour l'analyse du SID qui sont l'analyse complémentaire des besoins des utilisateurs et des sources de

données, la définition de priorités parmi les besoins utilisateurs et la spécification formelle des besoins.

En raison de l'importance de l'analyse complémentaire des besoins des décideurs et des sources de données et la définition de priorités parmi les besoins, nous nous intéressons à la phase d'analyse des méthodes de développement mixtes. Par ailleurs, des travaux dédiés à l'ingénierie des besoins ont fait des propositions dans le cadre des SID. Ainsi, dans les sections suivantes, nous dressons un état de l'art de la phase d'analyse des méthodes mixtes et de ces travaux émanant de l'ingénierie des besoins dans le cadre des SID. Nous évaluons ces méthodes d'analyse en fonction des modèles utilisés pour la collecte et la formalisation des besoins, mais aussi par rapport aux cinq tâches du processus d'analyse défini par [Winter and Strauch 2003].

Les utilisateurs du SID sont des décideurs, mais ces derniers n'ont pas le même rôle au sein de l'organisation car leurs intérêts peuvent porter sur toute l'organisation (décideurs stratégiques) ou sur un domaine d'activité en particulier (décideurs tactiques). De plus, comme expliqué dans le chapitre 1 à partir de l'étude d'IDG référencée dans la note de bas de page 1, l'impact de ces deux types de décideurs ne sont pas les mêmes. De même, la prise en compte des traitements est une tâche transversale au développement du SID. Il importe donc d'étudier aussi ces méthodes d'analyse en fonction de la distinction des décideurs et de la prise en compte des traitements dès la phase d'analyse. Par ailleurs, face au temps et aux ressources importantes requis pour la mise en place du SID, nous évaluons aussi ces méthodes d'analyse en fonction des propositions de parallélisation, d'automatisation et de capitalisation de la connaissance dans l'optique d'une réutilisation.

Nous classifions ces méthodes d'analyse en trois groupes en fonction du modèle de formalisation qu'elles utilisent :
- **modèles existants** : elles utilisent des modèles existants tels que le modèle entité-association, les diagrammes de classes ou les cas d'utilisation,
- **requêtes** : elles utilisent des requêtes pour représenter les besoins,
- **modèles de buts** : elles représentent les besoins suivant les buts qu'ils visent. Les buts peuvent être représentés suivant différentes approches.

Nous définissons les besoins du SID comme l'ensemble des besoins des utilisateurs et des sources de données.

2.4.1 Modèles existants

Certaines méthodes procèdent de manière analogue à la tâche d'analyse des besoins utilisateurs dans le cadre de l'ingénierie des SI classiques. Elles interrogent de manière empirique les utilisateurs *via* des interviews, des questionnaires, puis elles formalisent les besoins des utilisateurs exprimés en langage *via* un modèle tel que le

modèle entité-association [Cavero et al. 2001] et les cas d'utilisation [Prat and Akoka 2002; Luján-Mora and Trujillo 2003].

Dans [Bruckner et al. 2001; List et al. 2002], les auteurs utilisent des cas d'utilisation pour formaliser les besoins des utilisateurs avec un niveau de détail très fin. Ils dressent une analyse comparative des trois démarches de développement de SID suivant de nombreux paramètres tels que les domaines d'application, les méthodes de gestion, niveaux d'organisation, évolution du taux d'implication des utilisateurs. Ils distinguent trois groupes de besoins qui sont :

- le groupe décision : il regroupe tous les objectifs de haut-niveau que doit satisfaire le SID. La mise en place du SID doit induire dans un premier temps ces objectifs. Ils sont définis lors de la définition du cadre du projet.
- le groupe utilisateur : il représente les tâches que les utilisateurs doivent être capables d'accomplir à l'aide du SID. Les auteurs suggèrent de collecter ces besoins fonctionnels et non fonctionnels auprès des utilisateurs directs du SID en ne favorisant pas l'apparition de nouveaux besoins,
- le groupe système : il comprend des besoins fonctionnels et non-fonctionnels. Les besoins fonctionnels sont relatifs aux fonctionnalités que doit remplir le SID afin que les utilisateurs accomplissent leurs tâches.

Les auteurs recommandent une analyse itérative et incrémentale des besoins. La classification des besoins est judicieuse et pertinente en vue de définir des priorités parmi ces derniers. Nous nous basons sur cette classification des besoins pour notre classification des acteurs du SID. Cependant, ces travaux ne se placent pas dans le cadre d'un processus de développement de SID.

Dans [Soussi et al. 2005], les auteurs utilisent une structure pour collecter les besoins des décideurs. La structure est appelée tableau n-dimensionnel. Elle représente aussi bien les concepts de faits, de dimensions, de mesures et de paramètres que d'hiérarchies et d'attributs faibles. Cette structure est très expressive mais trop complexe pour qu'elle soit définie directement par les utilisateurs. De ce fait, pour la définition des besoins, les auteurs ont développé un outil reposant sur une base contenant des concepts multidimensionnels définis à partir de la base des applications métiers. Les décideurs doivent donc se familiariser à un outil au début du projet qui ne sera d'aucune utilité par la suite. Ceci représente une charge inutile pour la mise en place du SID.

Ces tableaux n-dimensionnels sont formalisés en des diagrammes de classes pour créer un référentiel. A partir de ces diagrammes, ils dérivent plusieurs schémas en étoile et ils les confrontent à tous les schémas en étoile obtenus à partir des sources de données. La création d'un référentiel composé de diagrammes de classes des besoins constitue une capitalisation de la connaissance au cours du processus de développement des SID.

Synthèse des méthodes d'analyse utilisant des modèles existants

Les méthodes d'analyse des besoins du SID qui reposent sur des modèles existants ne couvrent pas tout le processus d'analyse du SID car elles ne définissent pas de priorités parmi les besoins, sauf [Bruckner et al. 2001]. Elles ne proposent pas un guidage pour la confrontation des besoins, sauf [Soussi et al. 2005; Luján-Mora and Trujillo 2003]. De même, elles utilisent des modèles qui ne sont pas familiers aux décideurs, ce qui peut compromettre la validation des schémas par les décideurs [Winter and Strauch 2003] et de facto l'acceptation du SID par ces derniers. De plus, la plupart de ces méthodes ne distinguent pas les décideurs stratégiques des décideurs tactiques.

La méthode de [Soussi et al. 2005] présente l'avantage d'utiliser une structure de collecte des besoins que les décideurs utilisent au quotidien [Fernandez 2003]. Cette dernière rend possible une automatisation du processus de l'analyse. De plus, les utilisateurs, maîtrisant les tableaux, peuvent donc exprimer seuls leurs besoins dans des tableaux.

Cependant, comme l'ensemble de ces méthodes, elle ne favorise pas la parallélisation des tâches. De plus, elle présente un autre inconvénient qui est l'adaptation requise à l'outil pour l'expression des besoins.

La nécessité d'un cycle itératif et incrémental pour l'analyse pointée par [Bruckner et al. 2001] contribue à assurer une fiabilité et une bonne qualité des SID.

2.4.2 Requêtes

D'autres méthodes représentent les besoins de utilisateurs sous forme des **requêtes**.

[Phipps and Davis 2002] utilisent ces requêtes pour valider les schémas candidats définis à partir des sources en vérifiant que les requêtes satisfont les clauses « Select » et « From ».

Dans [Ghozzi et al. 2005], les besoins exprimés en langage naturel lors de la collecte des besoins des utilisateurs sont représentés *via* des requêtes. La structure de ces requêtes se rapproche de la structure des besoins exprimés en langage naturel. Elles sont définies suivant les clauses :

- Analyser : elle définit les données que les décideurs souhaitent analyser en répondant à la question « Quoi »,
- En fonction : elle détermine les paramètres de l'analyse des données en répondant aux questions « Qui, Où, Quand »,
- Pour : elle définit les restrictions sur les données analysées en répondant aux questions « Pour qui, Quelles données ».

A partir de ces requêtes, les auteurs définissent une matrice des besoins pour déterminer les concepts multidimensionnels. Puis, ils dérivent un schéma en étoile

idéal. Parallèlement, ces auteurs définissent un schéma en étoile candidat à partir du schéma des sources de données.

Synthèse des méthodes d'analyse utilisant des requêtes

Représenter les besoins des utilisateurs sous forme de requêtes est une approche intuitive pour les concepteurs décisionnels car ils sont familiers au langage d'interrogation. Cependant, la structuration sous forme de requête n'est pas familière aux non-informaticiens. Les décideurs valideront avec difficulté la spécification de leurs besoins. Ainsi, comme pour les méthodes d'analyse basée sur des formalismes existants, la seule validation possible sera celle des schémas conceptuels multidimensionnels des données.

De plus, ces méthodes ne guident donc pas la collecte et la confrontation des besoins. De même, elles ne distinguent pas les décideurs et elles ne définissent pas de priorités parmi les besoins des décideurs.

L'idée d'utiliser une structure pour la formalisations des besoins est judicieuse dans le sens où il sera possible d'automatiser cette phase. Cependant, il importe que cette structure soit maîtrisée par les décideurs.

2.4.3 Modèles de buts

Dans cette section, nous présentons les méthodes d'analyse des besoins du SID qui se basent sur des modèles de buts.

Dans [Bonifati et al. 2001], les auteurs utilisent un modèle de buts nommé GQM, pour représenter les besoins des utilisateurs. Plus précisément, ils utilisent deux formulaires avec des champs qui caractérisent les buts des décideurs. Ces caractéristiques sont relatives à l'aspect statique du domaine d'activité mais pas à l'aspect dynamique du processus de prise de décision. Ces formulaires sont remplis de manière interactive avec les décideurs après qu'ils aient présenté les tâches liées à leur fonction dans l'organisation. Les auteurs précisent que la définition de chaque but prend environ 2 heures. Ils ont présenté un cas qu'ils ont qualifié de simple composé de 16 buts. Cela veut dire que les utilisateurs sont mobilisés pendant au moins 4 jours uniquement pour la définition de leurs besoins dans le cadre d'un projet simple. Il faut ajouter à ce volume horaire, le temps nécessaire à la confrontation des besoins pour la validation des besoins. Cette caractéristique représente un inconvénient car il ne faut pas monopoliser les utilisateurs sachant que l'analyse des besoins n'est pas leur activité première.

A partir d'un schéma des sources, les auteurs obtiennent n schémas conceptuels multidimensionnels. Afin que le schéma issu des sources et celui issu des besoins soient comparables, le modèle GQM est transformé en modèle multidimensionnel.

Ainsi la spécification des besoins n'est pas validée avant la conception du schéma du SID.

Dans [Prakash and Gosain 2003], les auteurs collectent les besoins sous forme de buts et de décisions. Ils définissent un but comme un concept passif et la décision comme un concept actif tel qu'un processus. Ils utilisent un modèle de buts propriétaire GDI pour les représenter. Ils évoquent des processus liés au contexte décisionnel qui sont sous-jacents aux stratégies pour atteindre les buts.

Dans [Giorgini et al. 2005], les auteurs définissent une méthode d'analyse des besoins des décideurs qui intègre des concepts organisationnels liés au processus métier pour lequel le SID sera défini. Les auteurs distinguent les besoins organisationnels des besoins décisionnels. Ils définissent les **besoins organisationnels** comme étant l'ensemble des besoins relatifs aux acteurs physiques et logiciels et les **besoins décisionnels** comme étant l'ensemble des besoins relatifs aux décideurs. Ils utilisent le modèle de buts « i* » pour représenter les besoins évalués en deux analyses des besoins. L'analyse des besoins organisationnels est composée de trois étapes. Elle consiste à analyser les buts afin de déterminer les faits. L'analyse des besoins décisionnels commence par l'identification de tous les décideurs. Elle se déroule en quatre étapes et elle consiste à analyser les buts afin de déterminer les dimensions et les mesures associées aux faits. Ils définissent le passage de l'analyse des besoins à la conception suivant une démarche mixte incluant la démarche ascendante de [Golfarelli et al. 1998b].

Dans [Mazon et al. 2005], les auteurs utilisent le modèle de buts « i* ». Ils proposent de formaliser les besoins suivant les buts qu'ils définissent en distinguant trois types de buts (stratégiques, décision, information). Ils définissent une hiérarchie parmi ces buts tels les buts stratégiques sont ceux de plus haut niveau puis, les buts décision indiquent comment atteindre les buts stratégiques et enfin, les buts information définissent les informations des sources nécessaires pour atteindre ces deux précédents types de buts. La méthode inclut un ensemble de règle pour guider la formalisation des besoins *via* le modèle « i* ».

Dans [Gam and Salinesi 2006a,b], les auteurs proposent une méthode d'analyse des besoins des décideurs. Les auteurs utilisent le modèle de buts MAP pour représenter les intentions et les stratégies mises en place pour parvenir à un but. Les besoins sont regroupés en quatre niveaux : « niveau organisation métier », « niveau général du processus de décision », « niveau détaillé du processus de décision » et « niveau action ». Les auteurs définissent des modèles de buts MAP pour les trois premiers niveaux. Les besoins du « niveau action » sont représentés *via* un pseudo langage (dont la structure est <VerbeAction> <Cible> [<Parameter>]*). A partir d'heuristiques basées sur le modèle en étoile, les auteurs caractérisent les concepts multidimensionnels. Puis, ils définissent des règles pour passer du niveau action à un modèle multidimensionnel. Ils enrichissent leurs modèles en réutilisant les modèles multidimensionnels.

Synthèse des méthodes d'analyse utilisant des modèles de buts

Les méthodes d'analyse des besoins qui utilisent les modèles de buts proposent de collecter les besoins exprimés en langage naturel et de les formaliser suivant un guidage partiel lié à la mise en oeuvre du modèle de buts [Bonifati et al. 2001]. Cependant, elles ne distinguent pas les décideurs et elles ne définissent pas de priorité parmi les besoins.

Elles utilisent aussi des modèles de formalisation des besoins qui ne se rapprochent pas de la représentation multidimensionnelle des données par les décideurs. Ce dernier point représente un inconvénient car les décideurs pourront difficilement valider la formalisation des besoins avant la conception du schéma du SID. De ce fait, la seule validation sera celle du schéma multidimensionnel.

De plus, elles ne prennent pas nécessairement en compte les sources de données et de ce fait, elles ne n'abordent pas la confrontation des besoins, sauf [Bonifati et al. 2001; Giorgini et al. 2005; Mazon et al. 2005]. Ces derniers sont aussi les seules méthodes d'analyse utilisant des modèles de buts qui se placent dans un processus de développement.

[Giorgini et al. 2005] suggère la parallélisation des tâches de l'analyse afin de gagner du temps mais, aucune ne propose un mécanisme pour la capitalisation ou la réutilisation au cours de l'analyse.

2.4.4 Bilan et positionnement par rapport aux méthodes d'analyse

Toutes les méthodes d'analyse des besoins reposant soit sur le paradigme entité-association, soit sur le paradigme objet ou encore les modèles de buts présentent des inconvénients dans le sens où la structure n'est pas intuitive aux décideurs.

Parmi les modèles utilisées pour la formalisation des besoins, aucune ne se rapproche de la représentation des données par les décideurs qui est multidimensionnelle.

La structure tabulaire de collecte des besoins permet aux décideurs d'exprimer de manière structurée leurs besoins sans l'intervention des concepteurs décisionnels. Cela évite, par la même, une mauvaise interprétation en raison de la distance entre le domaine d'activité des décideurs et celui des concepteurs décisionnels. De plus, les tableaux ayant une structure connue, l'automatisation du processus d'analyse est envisageable. Cependant, elle ne permet de représenter que l'aspect statique des besoins.

De manière plus générale, les méthodes d'analyse se focalisent sur la statique du SID. [Prakash and Gosain 2003] abordent les processus liés au contexte décisionnel

mais le passage au niveau conceptuel n'est pas explicité. De plus, ces méthodes ne guident pas nécessairement la collecte et la confrontation des besoins du SID. Par ailleurs, rares sont les méthodes proposées qui définissent des priorités parmi les besoins comme le préconisent [Winter and Strauch 2003; Mazon et al. 2005] alors que cette hiérarchisation des besoins contribue à faciliter l'analyse.

De plus, les méthodes qui prennent en compte les sources suggèrent une confrontation des besoins utilisateurs et de sources des données après la réalisation du schéma conceptuel.

Il manque donc un modèle intuitif aux décideurs qui permette de représenter la statique et la dynamique du SID ainsi qu'une démarche pour la spécification, la formalisation et la confrontation des besoins du SID. Cette démarche doit paralléliser l'analyse des besoins utilisateurs et l'analyse des sources comme l'argumente [List et al. 2002] afin de gagner du temps. De même, elle doit favoriser la fiabilité et la réutilisation pour augmenter la fiabilité des SID.

Méthodes d'analyse de SID	Structure de collecte des besoins	Structure de formalisation des besoins	Démarche globale / tâche par tâche	Démarche explicite / implicite	Prise en compte des sources	Confrontation	Priorité dans les besoins	Distinction des décideurs	Parallélisation / Automatisation	Prise en compte de la dynamique	Réutilisation
[Winter and Strauch, 2003].	L N		tâche	implicite	oui	oui	oui	oui			
[Cavero et al., 2001]	L N	E-A	globale	implicite	oui						
[Lujan-Mora and Trujillo, 2003]	L N	CU	tâche	implicite	oui	oui					
[Bruckner et al., 1999]	L N	CU	globale	implicite				oui			
[Soussi et al., 2005]	tableau	DC	tâche	explicite	oui	oui					DC
[Phipps and Davis, 2002]	L N	requête	globale	explicite	oui						
[Carneiro and Brayner, 2002]	L N	requête	globale	implicite	oui						
[Ghozzi et al., 2005]	L N	requête	tâche	explicite	oui	oui					
[Bonifati et al., 2001]	L N	GQM	tâche	explicite	oui	oui					
[Giorgini et al., 2005]	L N	i*	tâche	explicite	oui			oui	oui		
[Prakash and Gosain, 2003]	L N	GDI	globale	explicite						abordée	
Gam and Salinesi, 2006a, Gam and Salinesi, 2006b]	L N	MAP	globale	explicite							
[Mazon et al., 2005]	LN	i*	tâche	explicite	oui		oui	oui			

L N : Langage naturel
DC : diagramme de classes UML
CU : Cas d'utilisation UML

FIG. 2.7 – Comparatif des méthodes d'analyse de SID

2.5 La capitalisation et la réutilisation dans les SID

Dans cette section, nous présentons d'abord les propositions de capitalisation et de réutilisation existantes pour favoriser un gain de temps et de la fiabilité au cours des projets. Puis, nous dressons un état de l'art des méthodes de développement de SID dans un contexte de réutilisation.

La mise en place d'un SID est une activité complexe et coûteuse en ressources et en temps [Carneiro and Brayner 2002]. Elle requiert aussi des développements spécifiques aux caractéristiques et aux besoins de l'organisation. Cependant, des projets décisionnels relatifs à un même domaine d'activité voire à des domaines d'activité différents présentent de nombreuses tâches communes. De manière plus générale, comme le précisent [Sen and Sinha 2005] cinq tâches sont récurrentes au cours des développements : l'analyse des besoins, la conception des données, la conception de l'architecture, l'implantation du système et le déploiement du système.

Après plus d'une décennie de travaux dans le domaine du décisionnel, bien qu'il n'y ait pas de modèles et de méthodes reconnus, les propositions ont atteint une maturité de par les standards de fait. Ainsi, à l'instar de l'ingénierie des systèmes d'information classiques, nous jugeons donc intéressant et pertinent de capitaliser les points communs et la variabilité de l'ingénierie des SID dans des composants réutilisables. Nous dressons donc un état de l'art des composants réutilisables définis en réponse aux problématiques de réutilisation dans le domaine des systèmes d'information classiques. Nous nous focalisons sur un composant réutilisable très utilisé.

2.5.1 Les composants réutilisables

Définition 2.13 (Composant réutilisable) *Un composant réutilisable est un fragment de produit ou de processus utilisable dans le processus de développement de plusieurs systèmes [Pujalte et al. 2004]. L'intérêt des composants réside dans le fait qu'ils permettent aux expériences passées d'être disponibles facilement [Deneckère 2001].*

Au regard des autres composants réutilisables qui ont été proposés les frameworks [Johnson 1992], les cas (Riesbeck et al, 1989), les objets métiers [Semmak 1998], le patron possède un bon niveau pour les trois critères, reconnus par l'OMG, qui caractérisent un composant réutilisable :

- la granularité : le nombre de problèmes que le composant résout et de solutions alternatives que propose le composant,

- la variabilité : les possibilités offertes à l'utilisateur pour adapter le composant à ses besoins,
- l'articulation : le degré de complétude des artefacts pour aboutir à une ou plusieurs solutions.

Dans le cadre du décisionnel, les objets métiers ne sont pas satisfaisants car ils sont spécifiques à un métier alors que le décisionnel est une activité transversale à de nombreux domaines (les ressources humaines, le contrôle de gestion, la gestion financière, la logistique et d'autres). De plus, ils ne permettent de capitaliser que les informations de type produit au niveau de la phase d'analyse. Les frameworks sont quant à eux des composants trop complexes, implantés au travers d'un grand nombre de classes. Contrairement aux patrons qui sont des directives pour résoudre un problème donné, ils sont des environnements complets pour le développement d'applications pour un domaine particulier. L'adaptation de ces composants complexes aux différents domaines liés au décisionnel s'avère conséquente. Le patron est une boîte blanche dont le degré de variabilité favorise une adaptation simple de notre méthode aux nombreux métiers et domaines d'activité liés au décisionnel.

Définition 2.14 (Patron) *Un patron est une base de savoir et de savoir-faire pour résoudre un problème récurrent dans un domaine particulier.*

Les premiers patrons dans le domaine de l'informatique sont apparus avec [Beck and Cunningham 1987] pour la conception et la programmation objet. Dans la littérature, les patrons sont généralement classifiés en fonction de l'étape d'ingénierie cible, appelée aussi portée (analyse, conception, implantation).

Définition 2.15 (Patron d'analyse) *Les patrons d'analyse aident le concepteur dans la construction de modèles représentant au mieux les besoins de son système [Coad 1992; Rolland 1993; Fowler 2004; Cauvet et al. 1998; Rieu 1999; Cauvet et al. 2000; Gzara 2000].*

Définition 2.16 (Patron de conception) *Les patrons de conception permettent d'identifier, de nommer et d'abstraire des thèmes communs du domaine de la conception [Gamma et al. 1995; Bushmann et al. 1996].*

Définition 2.17 (Patron d'implantation) *Les patrons d'implantation, aussi appelés idiomes [Coplien 1992] sont généralement spécifiques à un langage de programmation.*

Un autre avantage des patrons est celui d'instaurer un vocabulaire commun au sein de l'équipe de développement car la base de connaissance minimale est à disposition de tous et elle forme un référentiel auquel chacun peut se rapporter.

2.5.2 Ingénierie des systèmes d'information

La recherche dans le cadre de la réutilisation se réalise suivant deux axes : la conception des composants et l'exploitation des composants. Les deux principales problématiques sous-jacentes sont :

- l'**ingénierie pour la réutilisation** (« Design for reuse ») : la définition, la conception de composants réutilisables généralement associée de la spécification du processus qui en assure la réutilisation,
- l'**ingénierie par réutilisation** (« Design by reuse ») : le développement de méthodes et d'outils pour exploiter les composants réutilisables. Selon [Rieu 1999], il privilégie la mise en oeuvre de nouveaux environnements de développement fournissant des fonctionnalités concernant la gestion des catalogues, la spécification des systèmes et l'implantation par génération de codes.

L'ingénierie des systèmes d'information axée sur la réutilisation a engendré la définition de nombreux patrons afin de couvrir le processus de développement et ses produits.

Définition 2.18 (Processus) *Un processus est un ensemble d'actions, avec un séquencement, auquel on a assigné un objectif [Gzara 2000].*

Définition 2.19 (Produit) *Un produit est le résultat d'un processus [Gzara 2000].*

Les patrons associés à ces derniers sont définis comme suit [Gzara 2000] :

- patrons produits : patrons qui capitalisent des fragments de modèles, plus généralement les résultats du développement,
- patrons processus : patrons qui capitalisent des fragments de processus liés au développement.

Cette réutilisation peut être faite *a priori* au cours de la mise en place du système ou *à posteriori* à partir de la connaissance déployée sur des projets antérieurs.

2.5.3 Ingénierie des SID

Dans le domaine du décisionnel, la réutilisation reste empirique car peu de travaux abordent cette problématique. Dans cette section, nous présentons les deux propositions de méthodes de développement de SID axée sur la réutilisation.

Dans [Jones and Song 2005], les auteurs proposent de mettre en place un SID *via* l'application de patrons depuis la phase d'analyse des besoins du SID. Ce sont les premiers travaux relatifs au développement de SID pour la réutilisation. Ils définissent des patrons de domaines à partir de la récurrence des dimensions et de leurs paramètres dans les modèles en étoile proposés dans la littérature relatifs à différents domaines tels que ceux présentés dans l'ouvrage [Kimball et al. 1998]. Pour chaque

question qu'il convient de se poser lorsqu'une histoire est contée à un auditoire, en l'occurrence « Quand, Où, Qui, Qu'est-ce qui est fait ou accompli, Comment et Pourquoi », ils définissent un patron domaine. Cette démarche est dérivée du principe de qualité QQOQCP (Qui ?, Quoi ?, Ou ?, Quand ?, Comment ?, Pourquoi ?). Les patrons de domaine sont représentés par des diagrammes de classes UML. Ils permettent de définir les dimensions du SID comme suit :

- le patron de domaine « Quand » permet de définir la dimension Temps,
- le patron de domaine« Où » permet de définir la dimension Géographique,
- le patron de domaine « Qui » permet de définir la dimension Acteurs,
- le patron de domaine « Qu'est-ce qui est fait ou accompli » permet de définir les dimensions Comportement et Tâche,
- le patron de domaine « Comment » permet de définir la dimension des Objets physiques qui existent réellement et celle des Objets conceptuels,
- le patron de domaine « Pourquoi » permet de définir quatre dimensions qui qualifient l'histoire.

Les auteurs définissent un processus de réutilisation des patrons de domaine lors de l'analyse des besoins du SID en deux étapes pour chaque processus métier lié au projet. Lors de la première étape, il convient de poser des questions relatives aux six patrons de domaines pour déterminer les dimensions associées. Lors de la seconde étape, il convient de poser aussi des questions relatives aux attributs des classes de chacun des six patrons pour définir les attributs associés, sachant que chaque classe est composée d'un dizaines d'attributs. Ces deux étapes peuvent être itérées plusieurs fois. Ils ont mis en avant l'intérêt de l'utilisation des patrons de domaines dans le cadre d'un cours de conception de bases de données décisionnelles. Le gain apporté par l'utilisation des patrons est la conception du schéma des SID en 12.4 minutes au lieu de 18.5 minutes.

Cependant, cette méthode permet de déterminer uniquement les dimensions et leurs paramètres. Les concepts de faits et de mesures ne sont pas abordés. De plus, les auteurs ne proposent pas des patrons processus exhaustifs et ils ne fournissent pas d'outils pour concevoir un SID par réutilisation.

Dans [Feki et al. 2006], les auteurs proposent de réutiliser des patrons en étoile. Ces patrons sont des schémas en étoile par domaine d'activité. A partir de la documentation standard des processus métiers et de la fréquence d'apparition des objets, ils définissent des schémas en étoile qui modélisent l'ensemble des concepts de l'activité. Les auteurs réutilisent les patrons lors de la conception logique du SID. Ils instancient le patron en supprimant, modifiant ou ajoutant des dimensions, mesures, paramètres et hiérarchies afin de l'adapter aux besoins particuliers des décideurs d'une organisation.

Dans l'optique d'être générique, les patrons en étoile sont surchargés ; ce qui rend difficile leur compréhension. De plus, ces patrons ne guident pas les concepteurs décisionnels dès la phase d'analyse car ce sont des patrons produits de la phase

de conception au niveau logique. Comme la méthode précédente, les auteurs ne proposent pas un outil pour concevoir un SID par réutilisation.

2.5.4 Bilan et positionnement par rapport à la capitalisation et à la réutilisation

Les travaux liés aux problématiques de l'ingénierie pour la réutilisation dans les systèmes d'information classiques ont proposé de nombreux composants. Face au constat d'un grand nombre de points récurrents dans les étapes d'analyse et de conception des SID et au regard des nombreux domaines auxquels s'applique le décisionnel, nous retenons le modèle de composant patron [Alexander. 1977; Gamma et al. 1995] pour la capitalisation de notre démarche et de ses produits. De plus, afin de favoriser progressivement la fiabilité des SID déployés, nous privilégions une réutilisation à posteriori.

Dans le domaine de l'ingénierie des SID dans un cadre de réutilisation, seules deux méthodes proposent de capitaliser la connaissance. Ces deux propositions reposent sur le modèle de composants de patrons. Elles définissent des patrons pour respectivement l'analyse des besoins du SID [Jones and Song 2005] et la conception au niveau logique du SID [Feki et al. 2006]. Plus exactement, ces patrons sont des patrons produits. Ils ne guident donc pas les concepteurs décisionnels dans la mise en place du SID tout au long du processus d'ingénierie. De ce fait, les auteurs ne proposent pas de patrons processus exhaustifs et d'outils pour concevoir un SID par réutilisation.

Par conséquent, nous remarquons une absence des composants produits qui couvrent tout le processus d'ingénierie ainsi que des composants processus pour capitaliser le développement des SID et pour faciliter la réutilisation de ces composants. De même, le besoin d'outils pour la mise en oeuvre de ces systèmes par réutilisation se fait ressentir.

2.6 Synthèse de l'existant

De nombreuses méthodes de développement des SID ont été proposées. Cependant, aucune méthode et aucun modèle n'est reconnu par l'ensemble de la communauté industrielle et de la communauté de chercheurs [Rizzi et al. 2006].

2.6.1 Limites des modèles et méthodes de développement

Les méthodes se focalisent en grande partie sur la modélisation des données et sur la structure dans laquelle seront stockées les données. Elles reposent sur des modèles qui ne représentent que l'aspect statique des SID [Tsois et al. 2001]. L'évaluation statique du SID ne tient pas compte du contexte décisionnel telles que la durée et les conditions de validité, de disponibilité des données.

Néanmoins, la méthode [Luján-Mora and Trujillo 2003] traite de la dynamique des données à l'origine du SID. Les modèles intégrant la dynamique du SID [Vassiliadis et al. 2002a; Luján-Mora et al. 2006] représentent les traitements ETL entre le schéma conceptuel du SID et le schéma des sources, soit les traitements de dérivation. Cependant, ils ne tiennent pas compte des traitements de préparation liés au contexte décisionnel. L'analyse de la dynamique de SID, quand elle est évaluée, reste très technique.

De plus, les méthodes reposent souvent sur des démarches dont les étapes ne sont pas explicitées [Kimball 1996], [Carneiro and Brayner 2002]. Elles ne définissent pas le schéma du SID à partir des sources de données et des besoins des décideurs et elles occultent la phase d'analyse des besoins. Seules les méthodes mixtes prennent en compte les besoins des utilisateurs et les sources. Parmi les méthodes mixtes, l'analyse des sources des données est parfois privilégiée par rapport à celle des besoins des décideurs [Phipps and Davis 2002] car elle facilite la définition des processus ETL [Giorgini et al. 2005] et les hiérarchies. Cependant, les sources de données ne servent qu'à compléter et à valider la faisabilité du schéma des besoins des décideurs au regard de la disponibilité des données dans celles-ci et la complexité des transformations [Luján-Mora 2005].

Toutes ces méthodes de développement définissent au plus deux modules de l'architecture décisionnelle et elles ne guident pas les concepteurs dans le choix de l'architecture adéquate en fonction des besoins des utilisateurs et des sources. Elles ne permettent pas la mise en place complète d'un SID car elles ne couvrent pas toutes les tâches du processus de développement dont celles de l'analyse. De plus, elles ne favorisent pas une capitalisation et une réutilisation de la connaissance dans l'optique d'un gain de fiabilité et de temps.

Les modèles de données associés à ces méthodes permettent de représenter tous les concepts multidimensionnels, mais ils ne permettent pas de représenter toutes les spécificités des SID liées à la structure complexe des dimensions, des faits et des mesures, à la cohérence des interrogations ainsi qu'à la prise en compte des traitements.

2.6.2 Limites des méthodes d'analyse

Ces méthodes d'analyse des besoins formalisent les besoins exprimés généralement de manière informelle, en langage naturel, (cf. figure présentant le tableau comparatif des méthodes d'analyse 2.7). De plus, lors de la collecte de ces besoins les concepteurs décisionnels ne sont pas guidés ; ils utilisent des questionnaires ou autre support. Cette forme de collecte introduit un biais dans le sens où les concepteurs décisionnels ne maîtrisent pas les domaines d'activité des décideurs [Bruckner et al. 2001]. Cependant, une particularité du SID est que ces utilisateurs sont des décideurs. Ils utilisent au quotidien des tableaux de chiffres plus au moins complexes. Ils maîtrisent donc la représentation tabulaire des données. De ce fait, ils sont capables d'exprimer seuls leurs besoins sous forme de tableaux. L'expression des besoins sous la forme intuitive tabulaire peut ainsi être faite sans l'intervention des concepteurs décisionnels ou sans l'utilisation d'un logiciel dédié.

La fiabilité du SID par rapport à l'ensemble des besoins et des sources résulte de la tâche de confrontation à laquelle participe l'ensemble des acteurs (ou leurs représentants). De ce fait, afin de favoriser la participation et la validation des utilisateurs, les schémas résultant de l'analyse des besoins des décideurs doivent être proche de la représentation multidimensionnelle des données par les décideurs. Cependant, les méthodes d'analyse des besoins du SID représentent les besoins *via* des modèles qui ne se rapprochent pas de la représentation des données par les utilisateurs et qui ne tiennent pas compte des traitements du SID. Elles utilisent des modèles de buts, des requêtes ou des formalismes existants qui n'ont pas été adaptés au décisionnel pour représenter les besoins. Par ailleurs, les modèles de but représentent le contexte décisionnel [Bonifati et al. 2001; Prakash and Gosain 2003; Gam and Salinesi 2006a; Mazon et al. 2005] voire les contextes organisationnel et décisionnel [Giorgini et al. 2005] afin de garantir l'objectif principal du SID qui est de faciliter la prise de décision.

De plus, toutes ces méthodes d'analyses ne distinguent pas les décideurs et elles ne définissent pas de priorités entre les besoins, sauf [Giorgini et al. 2005; Mazon et al. 2005].

2.6.3 Limites face au développement rapide et au contexte de réutilisation

En dépit du temps et des ressources importantes nécessaires à la mise en place des SID, rares sont les méthodes d'analyse et de développement qui parallélisent, qui automatisent les tâches de l'ingénierie des SID ou qui capitalisent la connaissance développée au cours des projets.

Une seule méthode d'analyse proposent de paralléliser les tâches [Giorgini et al. 2005] mais, comme l'ensemble des méthode, elle ne favorise pas l'automatisation.

Dans [Jones and Song 2005; Feki et al. 2006], des patrons sont définis à posteriori. La sélection et l'utilisation de ces patrons sont relatives à l'appartenance au même domaine d'activité. Les composants réutilisables proposés capitalisent les produits du processus de développement mais, toute la démarche de développement n'est pas capitalisée. De plus, ces patrons ne sont pas liés entre eux ou organisés afin de faciliter et de systématiser la réutilisation.

2.7 Conclusion

A partir de ce bilan, nous notons que la prise en compte parallèle et complémentaire des sources de données pertinentes et des besoins utilisateurs est indispensable dès l'analyse pour la conception d'un schéma qui soit possible à implanter.

Il importe aussi de distinguer les décideurs stratégiques des décideurs tactiques et de définir des priorités parmi leurs besoins.

Nous avons dénoté l'importance de la prise en compte commune de l'aspect statique et de l'aspect dynamique afin de définir les données mais aussi les traitements relatifs au contexte technique et au contexte décisionnel. Plus précisément, outre les traitements de dérivation, les traitements de préparation des données sont importants.

De plus, il importe de faciliter la phase de confrontation en représentant les besoins des décideurs et les sources de données dans des modèles proches de la vision des données par les décideurs. Cette confrontation doit précéder la conception du schéma conceptuel du SID. De ce fait, les besoins des utilisateurs et les sources doivent donc être représentés dans des modèles comparables.

De même, les concepteurs ont besoin d'un guidage pour toutes les tâches de l'ingénierie des SID qui sont encore empiriques telles que la collecte des besoins, la formalisation de besoins, la confrontation des besoins, le choix de l'architecture décisionnelle et la formalisation des données et des traitements du SID.

Ces concepteurs ont aussi besoin d'un modèle qui permet de représenter l'ensemble des spécificités du SID dont celles supportées par peu ou pas de modèles actuels les hiérarchies non-strictes, les éléments terminaux multiples, les rôles des dimensions, les mesures dérivées, les liens entre ces mesures, les agrégations pertinentes, les traitements de dérivation et les traitements de préparation des données.

Deuxième partie

Une méthode de développement des SID dans un contexte de réutilisation

Chapitre 3

Principes généraux

Table des matières

3.1 Introduction **64**
3.2 Contextes industriel/scientifique et objectifs **64**
3.3 Idées principales de nos propositions **67**
3.3.1 Propositions liées à la phase d'analyse 67
3.3.2 Propositions liées à la phase de conception 67
3.3.3 Propositions liées au développement rapide et au contexte de réutilisation 68
3.4 Classification des acteurs et des besoins **68**
3.5 Démarche du Trident Décisionnel **70**
3.5.1 Ajout de la branche stratégique 71
3.5.2 Modification de l'ordonnancement des tâches 73
3.6 Cycle itératif incrémental et prototypage **73**
3.6.1 Description des itérations 74
3.6.2 Nombre variable d'itérations 75
3.7 Conclusion **76**

3.1 Introduction

Le turnover des informaticiens au sein des sociétés de services en ingénierie informatique (SSII) reste toujours visé autour des 20% comme le montre deux études réalisées avec 6 ans d'écart, soit 2001 [Zerbib and Delsol 2001] et 2007 [Rouzeré 2007]. Les concepteurs décisionnels font partis des informaticiens qui ne reste pas plus 2 à 3 voire 4 ans dans une même SSSI. Cette rotation continue constitue une faiblesse dans la mesure où les échanges et les transferts de connaissance ne sont toujours menés à leur terme et la connaissance des projets décisionnels antérieurs n'est pas capitalisée. De plus, le vocabulaire n'est pas univoque car les collaborateurs viennent d'écoles de pensée distinctes (gestion, informatique).

Dans le domaine de la recherche une dizaine de méthodes ont vu le jour. Ces dernières tendent vers une prise en compte des besoins des utilisateurs qui sont à l'origine des projets décisionnels et des sources qui contiennent les données de l'organisation. Cependant, ces méthodes ne reposent pas sur des bases théoriques stables. De ce fait, dans ce chapitre nous présentons les principes généraux liés à toutes nos propositions.

Ainsi, dans la section 3.2, nous présentons le contexte industriel et le contexte scientifique par rapport aux objectifs de nos travaux de thèse. Puis, dans la section 3.3, nous listons les idées principales de nos propositions. Dans la section 3.4, nous posons les bases de nos propositions en termes d'acteurs et de modules de l'architectures décisionnelle. Ainsi, dans la section 3.5, nous esquissons notre démarche de développement. Dans la section 3.6, nous développons le cycle de vie itératif incrémental incluant une phase de prototypage de notre méthode. Enfin, dans la section 3.7, nous dressons la conclusion des principes généraux de notre méthode.

3.2 Contextes industriel/scientifique et objectifs

Dans cette section, nous explicitons les objectifs industriels et scientifiques qui ont motivé nos travaux de recherche.

Le phénomène mondial de démocratisation des systèmes d'information décisionnels (SID) à l'ensemble des PME-PMI génère un important marché pour les sociétés spécialisées dans le décisionnel [IDC 2004c]. Ce contexte économique d'externalisation est connu par toutes les sociétés de services qui ont la responsabilité opérationnelle de l'informatique d'organisations clientes. Ce contexte est décrit par le taux de 7,6% de croissance en 2003 pour ce marché en France selon une étude réalisée par IDC [IDC 2003]. En focalisant sur l'infogérance des applications liées aux systèmes d'informations qui inclut les applications décisionnelles, l'étude d'IDG [Rosé 2005] montre qu'en 2005, elle représente 37% des 47% de projets d'infogérance en cours ou à venir.

Au regard des propositions existantes et de notre contexte industriel, notre méthode doit atteindre les objectifs suivants :

- liés à la définition de la méthode :
  - augmenter la fiabilité des SID par rapport aux besoins des utilisateurs et aux sources,
  - améliorer le taux d'acceptation du SID par les décideurs,
  - prendre en compte l'aspect statique et l'aspect dynamique du SID,
  - permettre la mise en place d'un SID complet,
  - définir un modèle conceptuel généralisé qui vise à répondre au manque de modèle standard,
  - faciliter l'intégration des collaborateurs débutants,
  - harmoniser les terminologies utilisées.
- liés à la mise en oeuvre de la méthode :
  - réduire le temps nécessaire à la mise en place du SID,
  - améliorer l'échange des informations entre les collaborateurs,
  - capitaliser la connaissance mise en place,
  - réutiliser la connaissance.

Pour atteindre ces objectifs, au regard des apports et des limites des méthodes existantes, notre méthode doit présenter les propriétés suivantes :

- liées à la définition de la méthode :
  - prise en compte des besoins utilisateurs et des sources,
  - distinction les utilisateurs et définir des priorités parmi leurs besoins,
  - unification des spécificités que permettent de représenter les modèles conceptuels existants et généralisation de ces derniers,
  - prise en compte de toutes les spécificités des SID,
  - développement de tous les modules de l'architecture du SID,
  - guidage explicite pour toutes les tâches.
- liées à la mise en oeuvre de la méthode :
  - mise en oeuvre facile et rapide du SID,
  - capitalisation de la connaissance au cours des projets,
  - gestion systématique de la documentation projet,
  - réutilisation de la connaissance,
  - mise à disposition d'un outil qui guide le développement des SID par réutilisation.

Nos travaux menés dans le cadre industriel bénéficient des propositions faites auparavant au sein de notre équipe. Nous considérons une architecture décisionnelle qui respecte la dichotomie des espaces de stockage. Cette dichotomie consiste à séparer l'entrepôt de données des magasins de données comme présenté dans la figure 3.1 mais, surtout à définir des modèles de données différents pour ces deux espaces de stockage afin d'assurer une gestion efficace des données en fonction des caractéristiques de ces espaces. Ainsi, généralement, l'entrepôt de données repose sur

un schéma de données qui ne préjuge pas de l'exploitation et de la restitution des données [Teste 2000] tel que le schéma entité-association. Les magasins de données reposent sur des schémas multidimensionnels pour répondre aux besoins d'analyse de cette classe de décideurs. Nos propositions s'appliquent aussi aux entrepôts basés sur un schéma multidimensionnel.

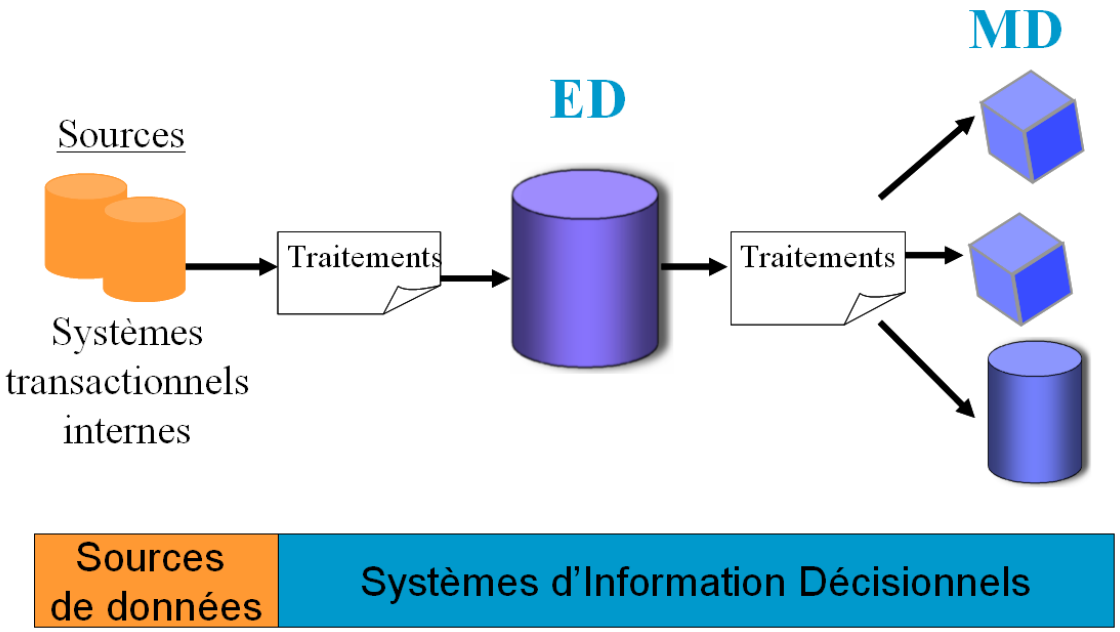


FIG. 3.1 – Dichotomie des espaces de stockage

Une méthode de développement de magasins de données a déjà été définie [Ghozzi 2004] reposant sur cette architecture. Elle ne traite que l'aspect statique d'un seul module du SID et elle ne guide pas toutes les tâches du processus de développement. De plus, elle ne gère pas la capitalisation et la réutilisation de la connaissance. Un résumé de cette méthode est présenté dans l'état de l'art.

Nos travaux ont donc pour objectif de proposer **une méthode de développement rapide de SID complets répondant aux besoins de tous les acteurs du SID au regard des sources de données, utilisable par des concepteurs décisionnels de tous niveaux d'expérience, dans un contexte de réutilisation avec une documentation systématique**. En raison du grand nombre d'outils décisionnels disponibles pour l'implantation des SID et dans un souci de proposer une méthode générique, nous nous sommes focalisés sur les phases de l'analyse et de la conception du SID.

3.3 Idées principales de nos propositions

Pour atteindre les objectifs de nos travaux de recherche, nous définissons d'une part, **des modèles, une démarche et un outil pour l'analyse et la conception des SID** dans le cadre d'une méthode de développement de SID complets dans un cadre de réutilisation. D'autre part, fort des constats des projets développés au sein de la société I-D6 avant et après la définition de notre méthode, nous proposons **un catalogue de composants réutilisables et un outil pour la mise en place de SID**.

3.3.1 Propositions liées à la phase d'analyse

Nous proposons de prendre en compte les sources de données et les besoins des utilisateurs dès la phase d'analyse pour l'évaluation de la statique du SID à partir respectivement de schémas conceptuels de données tels que le schéma entité-association et d'expressions structurées des besoins. Puis nous proposons de guider les concepteurs décisionnels pour l'évaluation de la dynamique du SID exprimée lors des interviews des acteurs du SID *via* des graphes orientés. L'évaluation de l'aspect statique et de l'aspect dynamique tient compte non seulement des données mais aussi des traitements sur celles-ci en définissant le contexte technique et le contexte décision. Nous distinguons les utilisateurs afin de définir des priorités parmi leurs besoins. Ces derniers et les sources sont représentés dans des modèles comparables et compréhensibles par tous les acteurs du SID.

Enfin, nous définissons la phase d'analyse suivant un cycle itératif incrémental avec un nombre d'itérations fonction de la tâche de confrontation des besoins. Cette tâche de confrontation est facilitée par l'annotation et la représentation des besoins du SID proche de la vision des utilisateurs. De plus, nous proposons un système de règles pour guider cette tâche.

3.3.2 Propositions liées à la phase de conception

Pour la phase de conception, nous proposons de guider le concepteur décisionnel dans le choix de l'architecture adaptée en fonction des besoins des décideurs et des sources de données. Pour cela, nous définissons une typologie unifiée des architectures de SID valides et des critères de quantification pour le choix de l'architecture. Enfin, nous définissons un modèle conceptuel multidimensionnel de données et de traitements du SID qui est une généralisation des propositions existantes. Il représente tous les concepts multidimensionnels et les spécificités liées à leur structure complexe et leur dynamique. Il permet de représenter les données et les traitements des modules reposant sur une architecture multidimensionnelle.

3.3.3 Propositions liées au développement rapide et au contexte de réutilisation

Pour répondre aux critères de rapidité de développement et de fiabilité des SID énoncés dans nos objectifs, nous proposons de :

- paralléliser l'analyse des besoins utilisateurs et l'analyse des sources,
- automatiser les tâches d'analyse du SID et le choix de l'architecture décisionnelle,
- capitaliser l'expérience et l'expertise mises en oeuvre au cours des projets par la définition d'un catalogue de patrons processus relatifs à l'analyse et à la conception de SID,
- capitaliser l'information technique et décisionnelle déployée au sein des projets dans un catalogue de composant produit.

Dans l'optique de valider nos travaux, nous présentons un bilan de la mise en oeuvre de nos propositions dans des contextes industriels différents et un guide méthodologique pour l'analyse et la conception de SID par réutilisation.

3.4 Classification des acteurs et des besoins

La prise en compte des spécificités des acteurs du SID nécessite de classer ces derniers et leurs besoins. Les acteurs d'un projet décisionnel ont des fonctions et des profils différents qu'il importe d'analyser séparément.

Les concepteurs décisionnels interviennent au cours du projet décisionnel pour satisfaire les besoins utilisateurs à partir des sources, durant la mise en place du SID mais aussi durant sa maintenance. Ils maîtrisent les caractéristiques de ces systèmes sur lesquels reposent le SID. La définition d'une méthode de développement requiert la prise en compte de ces acteurs et de leurs rôles. Ainsi, nous considérons les concepteurs décisionnels comme des acteurs du SID, mais pas comme des utilisateurs.

Les utilisateurs du SID sont des décideurs et des analystes. Nous les avons distingués dans le diagramme de classes présenté dans la figure 3.2 afin de refléter au mieux la réalité industrielle. Comme nous ne traitons pas le processus de décision dans cette thèse, mais de l'aide pour la mise en oeuvre de ce processus, nous utilisons l'unique terme et le plus commun « décideurs » pour désigner ces deux acteurs. Ces décideurs n'ont pas la même couverture métier et leurs décisions n'ont pas la même portée comme indiqué par l'étude d'IDG qui précise l'impact des acteurs des projets décisionnels (présentée au chapitre1).

Nous classons les besoins exprimés par les différents acteurs d'un projet SID en quatre types :

- *analytique* : besoins liés aux analyses de données, à la qualité des données et aux processus de consolidation, d'historisation, d'archivage et de rafraîchissement des données,
- *fonctionnel* : besoins liés aux trois principales tâches de la chaîne décisionnelle, soit l'alimentation (règles de gestion,...), le stockage (volume et format des données) et la présentation (restitution et diffusion),
- *non fonctionnel* : besoins liés à la sécurité des données, aux performances d'interrogation et de restitution de l'information,
- *stratégique* : besoins liés aux différentes politiques de l'organisation.

Ainsi, à l'instar de la classification des besoins de [Bruckner et al. 2001] qui définit des besoins utilisateurs, des besoins pilotage, des besoins systèmes où il y a un raffinement entre les besoins tel que un besoin de pilotage est décomposé en besoins utilisateurs et un besoin utilisateur regroupe plusieurs besoins systèmes, nous identifions trois types d'acteurs. La différence entre cette proposition est la nôtre réside dans le fait qu'entre les besoins exprimés par ces acteurs, il n'existe pas nécessairement de relations de raffinement entre les besoins car les intérêts d'un domaine d'activités peuvent être distincts de ceux des intérêts généraux d'une organisation. La figure 3.2 indique les acteurs formant les trois groupes : tactique, stratégique et système.

Le **groupe tactique** comprend les décideurs liés à un métier spécifique . Il exprime principalement des besoins analytiques, fonctionnels (règles de gestion, règles de calcul) ainsi que non fonctionnels (ergonomie d'outil).

Le **groupe stratégique** comprend les décideurs qui ont une vision transversale des métiers, autrement dit, les décideurs liés à la direction de l'organisation. Il traite des problèmes liés à la gestion du projet et à la coordination des différentes équipes. Il exprime aussi des besoins analytiques, fonctionnels, non fonctionnels (sécurité, performance) mais surtout des besoins liés aux orientations stratégiques de l'organisation.

Le **groupe système** est lié aux sources de données existantes et aux équipements décisionnels qui sont et qui seront déployés. Il exprime les besoins fonctionnels liés au stockage et à la disponibilité ainsi que des besoins non fonctionnels liés à la sécurité des données et à la maintenance. Les concepteurs décisionnels et les administrateurs des systèmes d'information font partie de ce groupe. Il importe d'intégrer car ils gèrent le SID et respectivement les sources. De plus, seuls ces derniers maîtrisent la disponibilité des données des sources.

Peu de méthodes [Giorgini et al. 2005; Mazon et al. 2005] distinguent les décideurs au sein d'une même organisation afin de définir des priorités parmi leurs besoins.

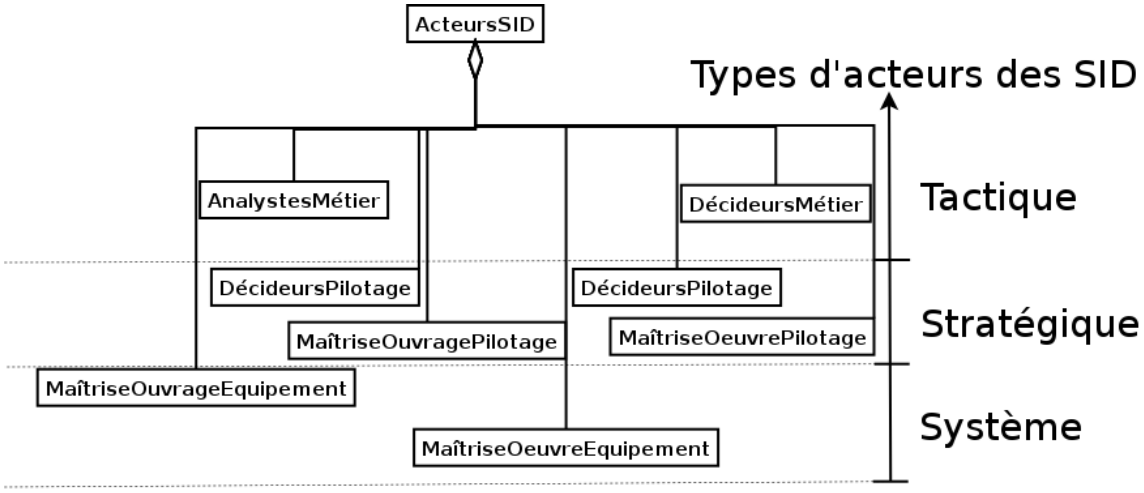


FIG. 3.2 – Le diagramme de classes UML des groupes d'acteurs décisionnels

La prise en compte des besoins de ces trois groupes d'acteurs est un élément capital pour le succès de l'intégration du SID dans l'organisation car elle implique :
- la prise en compte des spécificités de chaque groupe,
- la définition simplifiée des priorités inter-groupes,
- l'évaluation d'une partie de l'environnement des données, en l'occurrence de la portée des décisions et la couverture métier des données.

La confrontation des besoins de ces trois groupes est indispensable pour assurer, dans un premier temps, une analyse complète, rigoureuse et satisfaisante des SID et, dans un second temps, un développement d'un SID fiable par rapport aux besoins des utilisateurs et aux sources. Ainsi, à partir d'un consensus des besoins issus de cette confrontation, nous proposons de définir les modules qui composent l'architecture du SID.

3.5 Démarche du Trident Décisionnel

Notre démarche de développement réalise l'analyse des besoins des trois groupes d'acteurs et la conception de modules décisionnels. En raison du manque de méthode standard dans les SID, de la renommée et de la maîtrise des systèmes d'information classiques, elle est basée sur une démarche reconnue dans le domaine de ces systèmes appelée le diagramme en Y, aussi appelée 2TUP : « Two Track Unified Process ». Elle a été proposé par [Roques 2003]. Elle présente les avantages suivants : elle est associée à la modélisation UML et elle parallélise l'analyse des besoins fonctionnels et des besoins techniques.

3.5.1 Ajout de la branche stratégique

Les SID se distinguent des SI par la prise en compte de la dimension stratégique. Pour prendre en compte cette singularité, nous avons étendu le diagramme en Y par l'ajout de la « branche stratégique ». Notre extension est appelée le **Trident décisionnel**. Elle est présentée dans la figure 3.3.

Notre démarche de développement est composée de deux étapes qui sont « Mettre en place un SID » et « Maintenir un SID ». Dans le cadre de cette thèse, nous nous focalisons sur la première étape. Il est composé de trois phases : « Analyser un SID », « Concevoir un SID » et « Implanter un SID ». Ces tâches sont représentées dans le diagramme d'activités explicitant sa logique procédurale 3.3.

La première phase « Analyser un SID » est composée de trois branches reposant sur notre classification des acteurs. Ces branches permettent l'évaluation des spécificités du groupe tactique, des spécificités du groupe système et des spécificités du groupe stratégique. Elles favorisent le traitement parallèle des tâches d'analyse. Cette phase permet de définir le contexte d'exploitation du SID et le modèle de besoins de chaque groupe d'acteurs.

La deuxième phase « Concevoir un SID » est représentée par la partie supérieure du manche du trident décisionnel. Au cours de celle-ci, nous procédons à la définition et à la conception des modules de l'architecture décisionnelle adaptée en terme de données et de traitements du SID.

La troisième phase « Implanter le SID » est représentée à la base du manche du trident. Au cours de cette phase, les modules du SID et les traitements du SID sont implantés. Elle se termine par le déploiement du SID au sein de l'organisation. Le premier déploiement correspond à la mise en oeuvre du prototype. Les différents acteurs peuvent ainsi valider la fiabilité du SID à partir de jeux de tests (de données et d'exploitation).

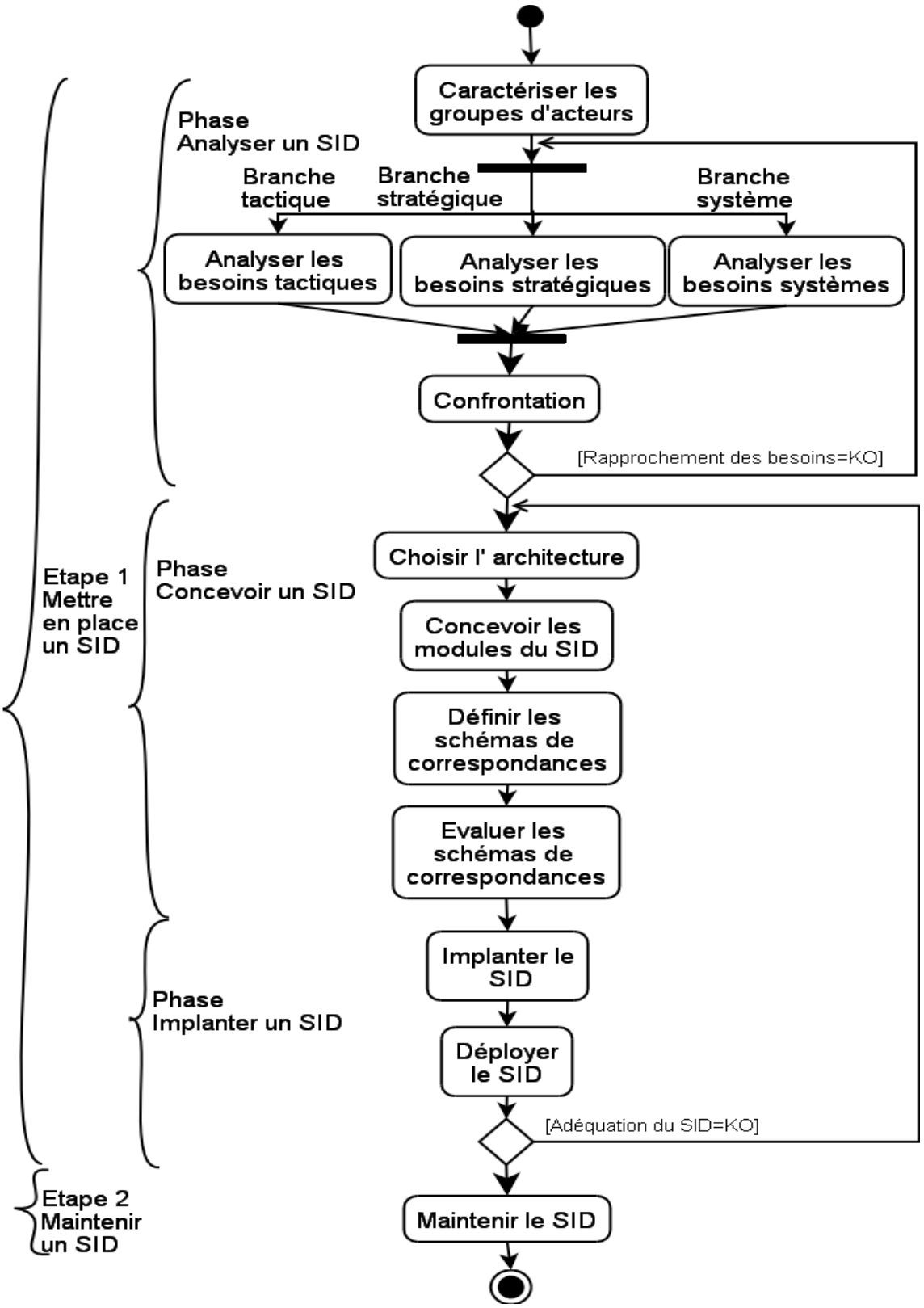


Fig. 3.3 – Vue d'ensemble de la démarche du Trident décisionnel

3.5.2 Modification de l'ordonnancement des tâches

Notre proposition de méthode est basée sur une démarche mixte qui requiert le rapprochement des besoins des trois groupes d'acteurs au cours du processus de développement. Cette démarche inclut la tâche décisive pour la fiabilité du SID qui est la confrontation des besoins et des sources.

Les méthodes de développement de SID existantes réalisent la confrontation, après la conception, entre le schéma conceptuel idéal défini à partir des besoins des décideurs et le ou les schémas candidats construits à partir des sources. Cette évaluation tardive implique une augmentation du temps de mise en place du SID car les incohérences sont détectées uniquement après la conception. Il faut donc parfois remonter à des tâches très antérieures qui relèvent de l'analyse des besoins ; le fait de replonger dans les développements et dans la documentation requiert du temps. La confrontation au plus tôt des besoins des différents groupes s'avère incontournable pour une conception efficace. Il importe donc qu'il y ait adéquation entre les différents besoins avant de commencer la conception du SID. De ce fait, nous avons déplacé cette tâche avant la conception du SID comme indiqué dans la figure 3.3.

3.6 Cycle itératif incrémental et prototypage

Cette section présente les modifications apportées au cycle itératif incrémental sur lequel repose la démarche du diagramme en Y pour répondre aux objectifs de nos travaux. Le cycle itératif incrémental du diagramme en Y inclut un seul point de décision. Il est défini après l'implantation du SID.

Dans le cadre de notre méthode, la tâche de confrontation est décisive car tant que les besoins et les sources ne peuvent pas être synchronisés, le SID résultant ne répond pas aux besoins des utilisateurs. Nous proposons donc de réitérer la phase d'analyse autant de fois que nécessaire afin de rapprocher les besoins avant la conception. Ainsi, nous proposons de définir un premier point de décision après la tâche de confrontation. Le Trident décisionnel comprend donc deux points de décision.

Son cycle de vie s'organise autour d'itérations de taille croissante au fil du déroulement du projet car au début les itérations sont réduites à la phase d'analyse. Puis, elles s'étendent à la phase de conception. Ceci favorise une communication plus régulière entre les différents acteurs au début du projet, sachant que c'est la période la plus difficile à appréhender sur un projet.

3.6.1 Description des itérations

Nous proposons un cycle de vie du développement du SID composé de trois types d'itérations associées sur les trois principales parties des projets décisionnels : cadrage, prototypage, mise en production (cf. figure 3.4). Le **cadrage** correspond à la définition les besoins des acteurs du projet décisionnel. Le **prototypage** permet de présenter un SID répondant à une partie ou à l'ensemble des besoins confrontés suivant les priorités de ces derniers. La **mise en production** correspond à la mise en place d'un SID répondant à tous les besoins des acteurs.

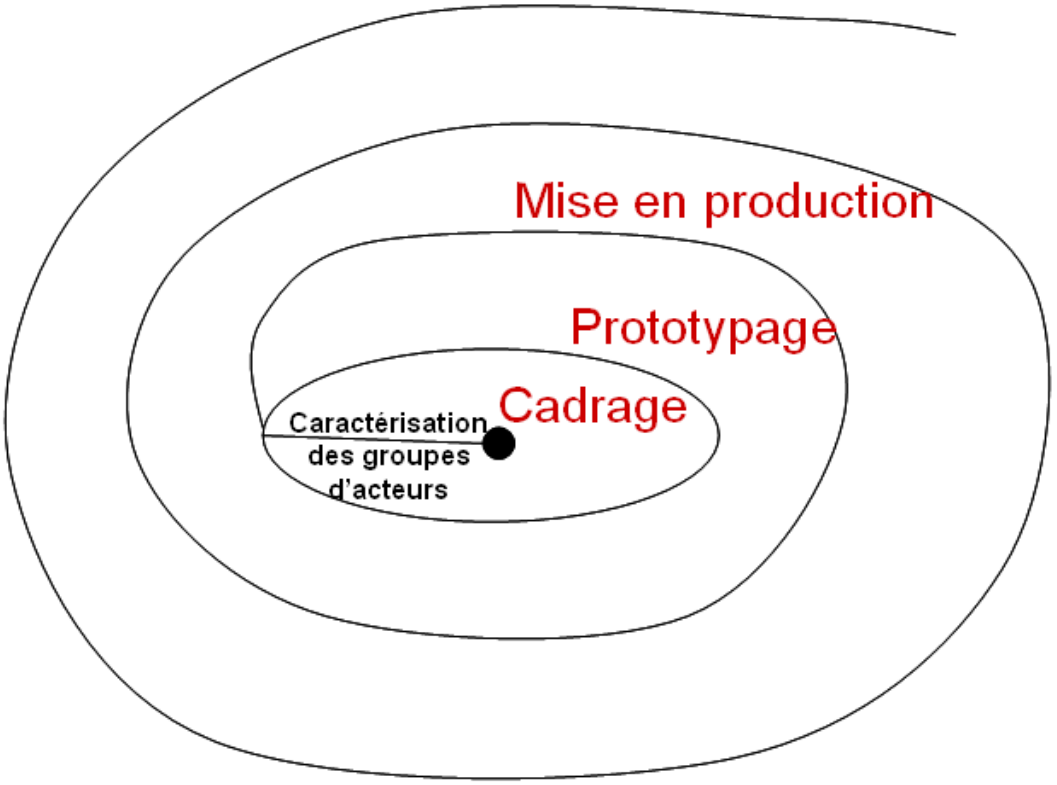


FIG. 3.4 – Les trois principaux types itérations

Le « Cadrage » prend en entrée le cahier des charges du projet et les documents d'expression des besoins de chaque groupe et produit en sortie le document de cadrage du projet et les documents de spécifications détaillées de chaque groupe d'acteurs. Elle se déroule généralement au cours de la première itération mais, suivant la taille du projet elle peut couvrir plusieurs itérations. Ces itérations sont limitées à la phase d'analyse afin de préciser et de rapprocher les besoins en fonction des priorités parmi ces derniers.

Le « Prototypage » commence par l'analyse des besoins qui vise à les rapprocher en fonction des priorités des besoins car nous avons constaté au sein de la société I-D6 qu'il y a rarement de rapprochement total au cours du cadrage. Elle se déroule

toujours en une itération où la tâche de caractérisation n'est pas réitérée. Les tâches de l'analyse seront moins longues car seule une partie des besoins sera évaluée. Les besoins sont rapprochés et nous pouvons définir l'architecture du SID, concevoir chaque module, définir et évaluer les schémas de correspondances. Puis, nous passons à l'implantation du SID. Enfin, nous déployons le SID. Ce premier SID déployé est un prototype. Il servira à valider la fiabilité des données et des traitements du SID. Cette itération permet d'évaluer la valeur ajoutée, la faisabilité du SID et de générer un prototype.

En raison de délais de mise à disposition du SID très court, le cadrage et le prototypage peuvent être fusionnés. Dans ce cas, le projet décisionnel se déroule en une phase complète d'analyse, une phase complète de conception et une phase complète d'implantation afin de générer un prototype dans des délais très brefs.

La mise en production commence par des développements complémentaires réalisés sur le prototype afin de concevoir et d'implanter toutes les données et tous les traitements nécessaires pour atteindre les objectifs fixés en début de projet. Elle se déroule généralement en une itération mais, plusieurs itérations peuvent s'avérer utiles.

Comme notre démarche impose la validation des besoins avant la conception, nous avons remarqué que les retours des qualifications concernaient la conception et plus précisément les correspondances entre les sources et les modules architecturaux du SID. Afin d'être efficace, la phase d'analyse n'est pas effectuée au cours des itérations associées à la mise en production. Cependant en cas de nouveau besoin, la phase d'analyse est réitérée sans la tâche de caractérisation des groupes d'acteurs. Ceci correspond à un nouveau prototypage. Puis, le SID passe en maintenance.

3.6.2 Nombre variable d'itérations

Le Trident décisionnel est un processus itératif et incrémental reposant sur trois itérations dans le cas où il n'y a pas d'anomalies de la première itération de cadrage et que les fonctionnalités sont rajoutées au cours d'une seule itération de mise en production. Cependant, dans le cas où le rapprochement entre les besoins du SID n'est pas possible, la phase d'analyse en cours est terminée et une nouvelle phase d'analyse des besoins sans la tâche de caractérisation des besoins commence. Ainsi, tant qu'il n'y pas de rapprochement entre les besoins du SID, l'itération en cours est clôturée et une nouvelle itération commence. En somme, dans le cas d'un projet SID incluant n itérations de cadrage, un prototypage et une itération de mise en production, le nombre d'itérations sera n+2.

De plus, dans le cas où la mise en production met en avant des inadéquations du SID par rapport à des besoins déjà exprimés par les décideurs, seules les phases de conception et d'implantation sont réitérées. Cependant, dans le cas de nouveaux

besoins, la phase d'analyse est aussi réitérée. Tant que le SID mis en exploitation présente des problèmes, ces phases sont réitérées. En somme, dans le cas d'un projet avec n tâches d'évaluation de la connaissance de l'environnement, un prototypage et m mises en production, le nombre d'itérations sera n+1+m. Pour les projets de très grosse taille menés au sein de la société I-D6, le nombre d'itérations de mise en production s'est limité à deux car ils passent ensuite à l'étape de maintenance.

3.7 Conclusion

Au cours des projets décisionnels de nombreux acteurs interviennent dont les concepteurs décisionnels et les responsables des systèmes d'informations. Nous avons donc proposé une classification des acteurs et une classification de leurs besoins les incluant afin de prendre en compte les spécificités de tous les acteurs et ainsi de définir un SID fiable et pertinent. Les besoins des acteurs déterminent les modules décisionnels qui composent le SID.

Ainsi, pour définir des SID fiables basés sur une architecture modulaire et prenant en compte la dimension stratégique qui distingue les SID des systèmes d'information, nous avons étendu le processus de développement appelé diagramme en Y ou encore 2TUP [Roques 2003]. Cette extension appelée Trident décisionnel inclut une phase d'analyse spécifique aux SID et une tâche pour le choix de l'architecture multi-modulaire. Notre Trident décisionnel repose sur un cycle itératif incrémental avec un cadrage réduit à la phase d'analyse et un prototypage en fonction des priorités des besoins. Le nombre d'itérations varie fortement en fonction de la possibilité des sources à fournir les données nécessaires pour la dérivation du SID et la préparation des données pour faciliter la prise de décision.

Chapitre 4

Méthode d'analyse des besoins des SID

Table des matières

4.1 Introduction **79**
4.1.1 Problématique 79
4.1.2 Proposition 80
4.2 Modèles et démarches communs **82**
4.2.1 Modèle : tableau croisé 83
4.2.2 Modèle : graphe de propriétés 86
4.2.3 Modèle : diagramme décisionnel 90
4.2.4 Démarche : collecte des besoins utilisateurs 96
4.2.5 Démarche : annotation d'un graphe de propriétés 96
4.2.6 Démarche : formalisation des besoins utilisateurs 97
4.3 Analyse des besoins tactiques **100**
4.3.1 Collecte 100
4.3.2 Formalisation 102
4.4 Analyse des besoins stratégiques **106**
4.4.1 Collecte 106
4.4.2 Formalisation 107
4.5 Analyse des besoins systèmes **109**
4.5.1 Collecte 109
4.5.2 Formalisation 110
Règles de sélection des concepts multidimensionnels . . . 111
Règles de structuration liées au groupe système 114
Règles syntaxiques liées au groupe système 114
4.6 Confrontation des besoins du SID **118**
4.7 Conclusion **125**

Notre démarche de développement du Trident décisionnel définit le schéma du SID à partir des besoins du groupe tactique, des besoins du groupe stratégique et des besoins du groupe système. De ce fait, elle commence par la caractérisation de ces groupes. Dans ce chapitre, nous faisons un zoom sur la phase de l'analyse de notre démarche afin de présenter l'analyse des besoins de ces trois groupes comme indiqué dans la figure 4.1. L'objet de ce chapitre est de définir les concepts, les modèles et la démarche que nous proposons pour l'analyse de ces besoins.

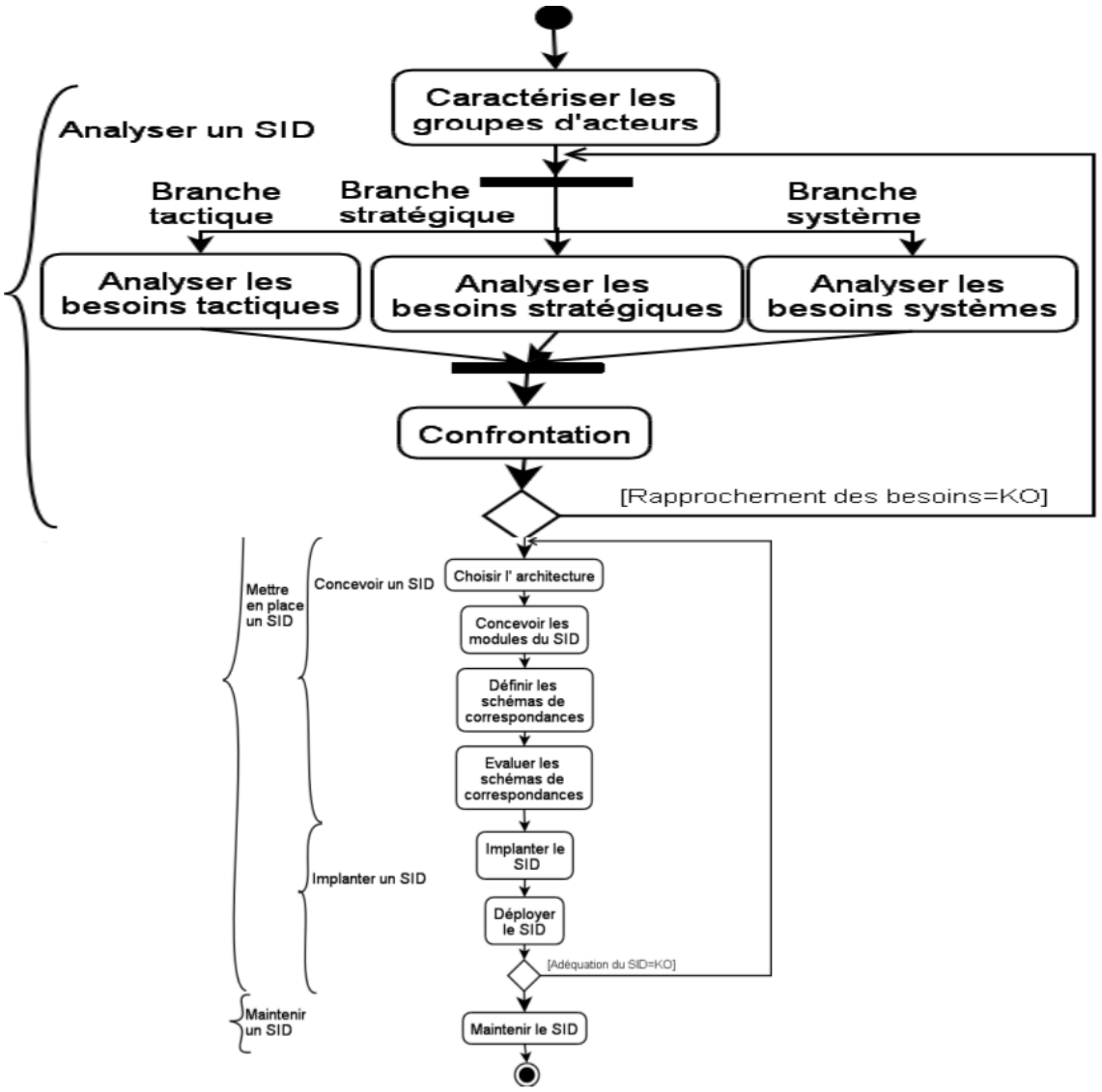


FIG. 4.1 – Zoom sur la phase d'analyse de notre Trident décisionnel

Ainsi, dans la section 4.1, nous introduisons notre problématique pour répondre aux limites de l'existant. Puis, dans la section 4.2, nous développons nos propositions communes à l'analyse des besoins de ces groupes. Nous poursuivons par la présentation des propositions spécifiques à chaque groupe en déroulant un exemple d'application suivant notre démarche dans trois sections dédiées (groupe tactique

dans la section 4.3, groupe stratégique dans la section 4.4 et groupe système dans la section 4.5). Dans la section 4.6, nous explicitons le déroulement de la tâche de confrontation des besoins. Enfin, dans la section 4.7, nous faisons le bilan de nos propositions liées de la phase d'analyse.

4.1 Introduction

Parmi les différentes phases de l'ingénierie des SID, la phase d'analyse est celle qui a fait l'objet du plus faible nombre de propositions. Ce manque de proposition et le faible intérêt accordé à cette phase au cours de projets réels expliquent l'échec de 80% des projets SID comme le précisent [Giorgini et al. 2005].

4.1.1 Problématique

Certaines méthodes de développement des SID accordent peu d'intérêts à l'analyse des besoins des décideurs [Golfarelli and Rizzi 1998a; Husemann et al. 2000] ou aux sources de données [Tsois et al. 2001; Prat and Akoka 2002] alors que ces deux paramètres impactent la définition du SID. Comme l'argumentent les auteurs de [Winter and Strauch 2003], une méthode d'analyse des besoins du SID requiert une prise en compte complémentaire de ces deux paramètres. Les méthodes mixtes sont les seules qui définissent le SID à partir des deux types de besoins. Nous nous intéressons donc à ces dernières. Cependant, elles ne distinguent pas les décideurs, elles ne définissent pas de priorités parmi leurs besoins et elles ne prennent pas en compte l'aspect dynamique des SID. De plus, elles ne confrontent pas les besoins et les sources avant la conception du SID *via* un modèle intuitif aux décideurs. De ce fait, ces derniers ne sont pas validés avant la conception du SID.

Des travaux relevant de l'ingénierie des besoins des SID ont été proposés. Ces propositions captent les objectifs et les stratégies *via* des modèles de buts [Bonifati et al. 2001; Prakash and Gosain 2003; Giorgini et al. 2005; Mazon et al. 2005; Gam and Salinesi 2006a]. Ces modèles représentent les données et les processus liés aux stratégies des décideurs. Ils abordent donc l'aspect dynamique du SID contrairement aux autres méthodes d'analyse qui se focalisent sur la statique du SID. Cependant, cette modélisation présente quatre inconvénients :

- le modèle inadapté : dans le cas de SID, la modélisation doit être compréhensible par les utilisateurs. Elle doit être proche de la manière dont les décideurs représentent les données car ils valident la pertinence et la fiabilité du SID. Les différents modèles de buts tels que MAP [Gam and Salinesi 2006a], GDI [Prakash and Gosain 2003], i* [Giorgini et al. 2005] et GQM [Bonifati et al. 2001] ne se rapprochent pas de la représentation multidimensionnelle des données par les décideurs,

- la modélisation partielle ou inexistante des processus : la modélisation des processus n'est pas conservée de l'analyse à la conception du SID parce que les propositions ne se placent pas dans le cadre d'une méthode complète de développement ou parce que la démarche de la conception du SID tient compte uniquement des données,
- deux modèles distincts pour la confrontation : les besoins des décideurs sont représentés *via* un modèle de buts alors que les sources de données sont représentés *via* des modèles multidimensionnels. Ces modèles ne sont pas proches et ils sont donc difficilement comparables,
- couverture incomplète du processus d'analyse : ces méthodes n'incluent pas nécessairement les tâches associées à l'analyse des sources et donc à la confrontation des besoins. De plus, elles ne sont pas toujours incluent dans un processus complet de développement.

De plus, une seule méthode d'analyse propose la parallélisation des tâches de l'analyse des besoins des décideurs et l'analyse des sources de données [Giorgini et al. 2005] bien que la mise en place des SID est une tâche gourmande en temps et en ressources. De même, face à ces problèmes, les méthodes d'analyse ne guident pas toutes les tâches associées et elles ne favorisent pas la réutilisation de la connaissance, sauf [Soussi et al. 2005] qui capitalisent les besoins dans des diagrammes de classes UML.

La problématique est donc **« comment analyser les besoins du SID de manière pertinente et fiable pour les décideurs mais aussi de manière rapide et fiable pour les concepteurs décisionnels ? »**

4.1.2 Proposition

L'**aspect statique** est relatif aux données et l'**aspect dynamique** est relatif au traitement sur ces données. Afin de prendre en compte toutes les spécificités des besoins du SID, nous proposons de distinguer deux composantes : technique et décision. Nous appelons l'ensemble des propriétés liées à ces deux composantes et à ces deux aspects, les propriétés CSD (contexte statique et dynamique) comme indiqué dans la figure 4.2. Les méthodes d'analyse existantes ne couvrent que la zone inférieure de cette figure car elles se focalisent sur la modélisation des données des sources pour alimenter le SID.

La **composante technique** est définie par les besoins exprimant l'aspect statique et l'aspect dynamique (les données et les traitements) liés à la dérivation des données des SID à partir des sources. Elle est relative à l'exploitation des sources de données. Les traitements ETL sont principalement liés à cette composante du SID.

La **composante décision** est définie par les besoins exprimant l'aspect statique et l'aspect dynamique (les données et les traitements) liés à la préparation des don-

nées pour faciliter l'aide à la prise de décision. Par exemple, ces besoins sont la validité des données ou encore les modalités de restitution.

La prise en compte de la composante « décision » impacte le schéma conceptuel du SID et surtout le niveau de satisfaction et le taux d'acceptation du SID car elle définit la valeur ajoutée des données fournies aux décideurs par rapport aux données des sources. De ce fait, en plus des deux aspects traditionnels statique et dynamique, il faut évaluer aussi bien la composante technique que la composante décision pour la conception d'un SID répondant aux besoins des décideurs et possible à mettre en place.

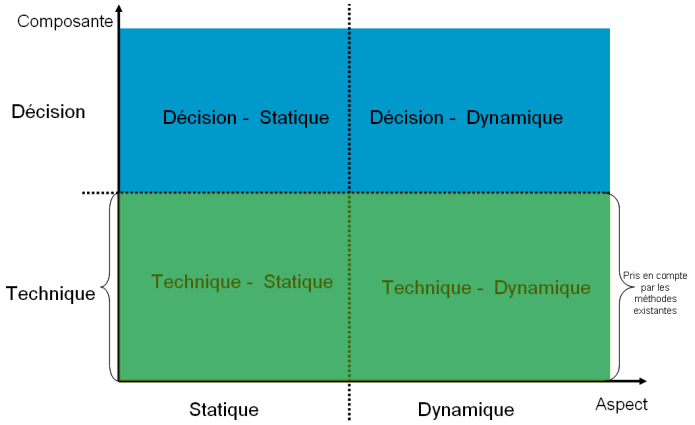


FIG. 4.2 – Catégorisation des propriétés CSD des SID

Pour atteindre les objectifs de nos travaux, nous proposons une méthode d'analyse des besoins de tous les acteurs du SID qui parallélise et qui traite séparément les tâches liées à chaque groupe suivant notre démarche du Trident décisionnel. Elle repose sur :

- une structure d'expression des besoins des décideurs liés à la statique du SID qui permet à ces derniers de les exprimer seuls. Les éventuels biais techniques dûs à l'interprétation des concepteurs décisionnels sont donc faibles.
- un graphe de propriétés du SID qui guide le concepteur décisionnel dans l'interview des acteurs au cours de laquelle ces derniers expriment leurs besoins liés à la dynamique du SID,
- un modèle de données et de traitements représentant les besoins des acteurs du SID de manière intuitive aux décideurs afin de faciliter la validation du SID,
- une confrontation simplifiée des besoins basée sur la représentation des besoins dans le même référentiel, soit le même modèle, et la définition de priorités parmi les besoins intra-groupe et inter-groupe d'acteurs.

Pour prendre en compte les spécificités des différents groupes d'acteurs et faciliter la confrontation de leurs besoins, nous analysons de manière séparée les besoins de chaque groupe et nous avons défini des modèles et une démarche communs aux groupes. Ces derniers sont présentés dans la section suivante. Les modèles et les démarches spécifiques à chaque groupe sont présentés dans les sections suivantes

dédiées. Le diagramme d'activités détaillé de notre méthode d'analyse est présenté dans la figure 4.3.

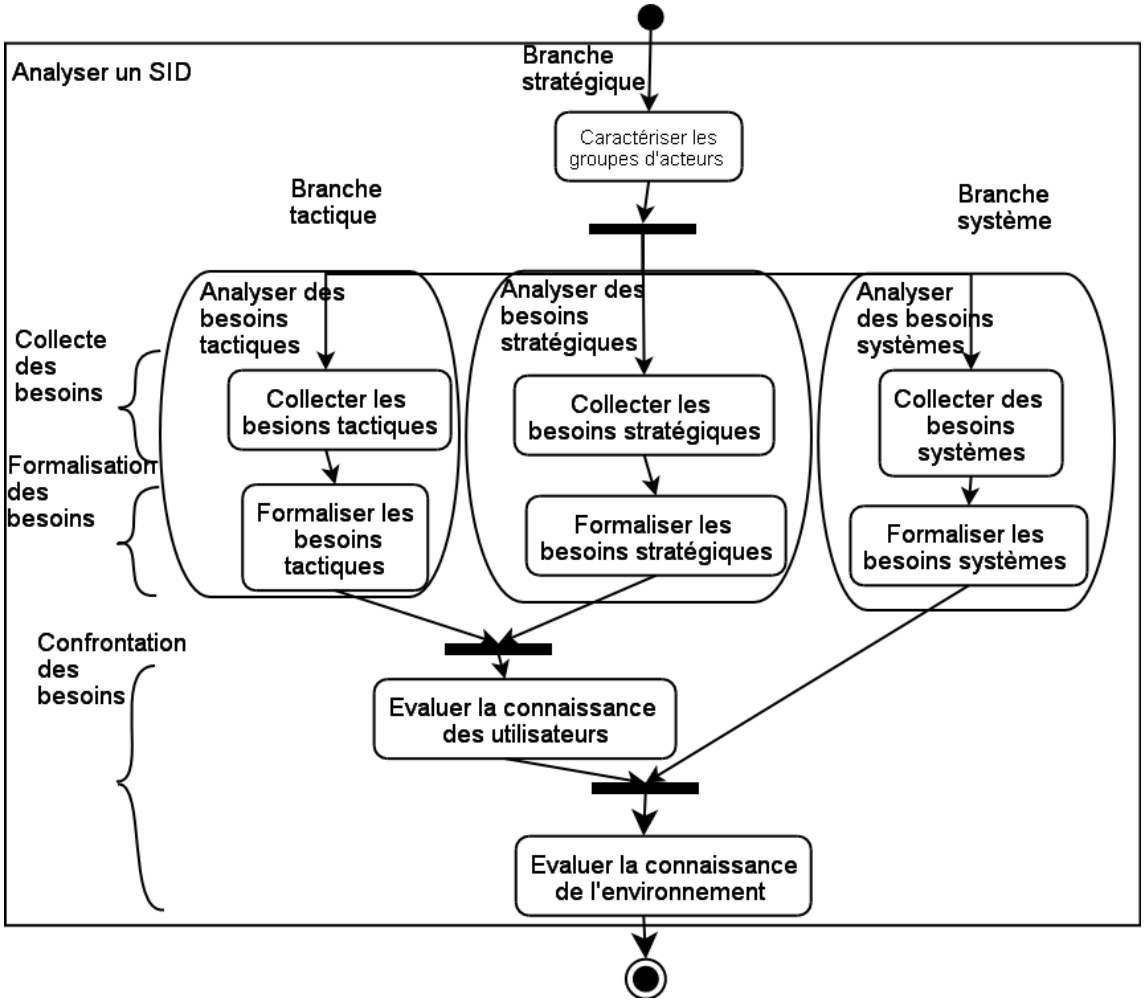


FIG. 4.3 – Diagramme d'activités UML de la phase d'analyse du Trident décisionnel

4.2 Modèles et démarches communs

Nous proposons un modèle commun au groupe tactique et au groupe stratégique (**tableau croisé**) et deux modèles communs à l'ensemble des trois groupes (**graphe de propriétés**,**diagramme décisionnel**). Le tableau croisé et le graphe de propriétés permettent de réaliser la collecte des propriétés CSD (contexte statique et dynamique). Le diagramme décisionnel permet de formaliser les besoins des acteurs en terme de données et de traitements. Nous proposons une démarche pour la collecte et la formalisation des besoins commune au groupe tactique et au groupe stratégique.

4.2.1 Modèle : tableau croisé

Nous rappelons que nous désignons par besoins utilisateurs : les besoins exprimés par tous les décideurs aussi bien du groupe tactique que du groupe stratégique. Nous désignons par besoins du SID : les besoins utilisateurs et ceux du groupe système. Lorsque nous parlons des sources de données, nous faisons référence aux besoins du groupe système.

Définition 4.1 (Exigence analytique) *Par analogie aux exigences fonctionnelles qui doivent être garanties par les systèmes d'information, les exigences analytiques représentent les manipulations de données qui doivent être possibles via le SID.*

Les décideurs expriment des exigences analytiques qu'ils représentent sous forme de tableaux de chiffres. Ils manipulent au quotidien ces tableaux plus au moins complexes comme le précise [Fernandez 2003]. Ces acteurs sont donc familiers à ces tableaux. Ils maîtrisent la représentation des données dans un tableau à une où deux dimensions qui définit les indicateurs du fait en fonction de ces derniers. Comme les décideurs sont demandeurs de plus d'autonomie, cette représentation tabulaire des données est un moyen pour exprimer leurs besoins sans l'intervention de concepteurs décisionnels.

Cette spécificité des utilisateurs permet de ne pas manipuler l'expression informelle des besoins des décideurs et d'éviter des interprétations trop techniques par les concepteurs décisionnels. Afin que cette tâche ne monopolise pas les décideurs, nous limitons le tableau à un ou à deux dimensions. L'expression structurée des besoins utilisateurs favorise une automatisation de leur formalisation.

Définition 4.2 (Tableau croisé) *Un tableau croisé est un tableau bidimensionnel qui représente le fait de l'analyse en fonction de deux dimensions. L'extension de chaque dimension est définie alors que les autres dimensions suivant lesquelles le fait peut être analysé ne sont pas représentées et sont positionnées à une valeur donnée définie éventuellement dans le nom du tableau.*

Un intérêt du tableau croisé (cf. tableau 4.1) est la facilité de l'expression des restrictions de domaines sur les dimensions et les faits de l'analyse. De même, il met en avant le type des données qui sont des informations difficilement exprimables par les décideurs car ils ne sont pas des informaticiens. Nous utilisons le terme de mesure au lieu d'indicateur afin de ne pas manipuler plusieurs termes.

S.$\{C_{r_S}\}^+$	$A_j.\{E_l\}^+$		
	$Val1_{E_{l_{A_j}}}$	...	$Valm_{E_{l_{A_j}}}$
$A_i.\{E_k\}^+$			
$Val1_{E_{k_{A_i}}}$	$ValC_{r_S}(Val1_{E_{k_{A_i}}}, Val1_{E_{l_{A_j}}})$	...	$ValC_{r_S}(Val1_{E_{k_{A_i}}}, Valm_{E_{l_{A_j}}})$
$Val2_{E_{k_{A_i}}}$	$ValC_{r_S}$ $(Val2_{E_{k_{A_i}}}, Val1_{E_{l_{A_j}}})$	...	...
$Valn_{E_{k_{A_i}}}$	$ValC_{r_S}(Valn_{E_{k_{A_i}}}, Val1_{E_{l_{A_j}}})$	...	$ValC_{r_S}$ $(Valn_{E_{k_{A_i}}}, Valm_{E_{l_{A_j}}})$

TAB. 4.1 – Tableau croisé type

Les concepts du tableau croisé type où les variables i, x, r, n, p, q sont des entiers correspondent à :

- A_i est une dimension suivant laquelle le fait est analysé. Elle est composée de paramètres $E_{k_{A_i}}$ ($0 \leq k \leq n$). Le k-ième paramètre a plusieurs valeurs : $Valx_{E_{k_{A_i}}}$ ($1 \leq x \leq p$). L'explication est la même pour la dimension A_j,
- S est le fait présenté dans ce tableau croisé. Il est composé de mesures C_{r_S} ($1 \leq r \leq q$). La mesure a plusieurs valeurs $ValC_{r_S}(Valx_{E_{k_{A_i}}}, Valw_{E_{l_{A_j}}})$,
- $ValC_{r_S}(Valx_{E_{k_{A_i}}}, Valw_{E_{l_{A_j}}})$ est la valeur de la r-ième mesure du fait S pour la x-ième valeur de $E_{k_{A_i}}$ et la w-ième valeur de $E_{l_{A_j}}$.

Exemple : dans le cas de notre projet immobilisations, un tableau croisé exprimant l'évolution de la valeur vénale des immobilisations par catégorie au cours des trois dernières années est présentée dans le tableau 4.2.

Immobilisations. Valeur_vénale		Temps.Année		
		2004	2005	2006
Catégories.Catégorie	Catégories.Sous_catégorie			
Immatériel	Logiciels	173 170,09	115 999,26	69 059,42
	Progiciels	23 769,32	11 429,62	8 958,61
Postes utilisateurs	Ecrans	4 624,12	3 445,14	1 147,43
	PC	7 420,95	5 319,81	3 601,85
	Terminaux	1 985,47	1 798,06	1 291,34

TAB. 4.2 – Tableau croisé de la valeur vénale des immobilisations par catégorie au cours des trois dernières années

Dans le cas où les tableaux croisés ne sont pas directement disponibles car les utilisateurs n'ont pas de chiffres ou que le besoin n'est pas précis, nous proposons de spécifier les besoins des décideurs avec des requêtes-types définies suivant notre pseudo-langage *BIQuery*. Nous avons étendu le pseudo-langage proposé au sein de notre équipe [Ghozzi 2004] afin d'exprimer explicitement les restrictions sur les faits et celles sur les mesures. La clause « QUAND » que nous proposons d'ajouter permet de distinguer les mesures des paramètres. Le langage définit l'ordre suivant :

ANALYSER $\{S.\{C_{r_s}\}^+\}^+$	– le(s) fait(s) et le ou les mesures $S.C_{r_S}$,
QUAND $\{Cond(S.C_{r_s})\}^+$	– les restrictions sur la(les) mesure(s) du (des) fait(s),
EN FONCTION $\{A_i.\{E_{k_{A_i}}\}^+\}^+$	– la ou les dimensions et le(s) paramètre(s) $A_i.E_{k_{A_i}}$,
POUR $\{Cond(A_i.E_{k_{A_i}})\}^+$	– les restrictions sur le(s) paramètres(s) de(s) dimension(s) tel que $Cond(X)$ défini par X op Expression(Val(X)) avec op un opérateur de comparaison, Expression(Val(X)) : une expression sur la valeur de X.

Cette requête est transformée en tableau croisé pour l'analyse des besoins suivant notre démarche du Trident décisionnel. Les faits, les dimensions, les mesures et les paramètres sont automatiquement identifiés car les symboles sont identiques à ceux du tableau croisé type.

Exemple : l'expression en langage naturel de l'exigence analytique « Analyser l'évolution de la valeur vénale des immobilisations par catégorie et sous-catégorie pour les trois dernières années» se traduit sous forme de requête-type comme suit :

ANALYSER	
Immobilisations.Valeur_vénale	– fait Immobilisations avec sa mesure
QUAND Valeur_vénale > 1000	– restriction sur la mesure Valeur_vénale
EN FONCTION Catégories.Catégorie	– dimension Catégories avec ses paramètres
, Catégories.Sous_catégorie	– dimension Catégories avec ses paramètres
, Temps.Année	– dimension Temps avec son paramètre Année
POUR	
Temps.Année DANS (2004,2005,2006)	– restriction sur le paramètre Année

La transformation de cette requête suivant le tableau-type 4.1 est le tableau croisé 4.3.

Immobilisations. Valeur_vénale		Temps.Année		
		2004	2005	2006
Catégories.Catégorie	Catégories.Sous_catégorie			

TAB. 4.3 – Tableau croisé de la valeur vénale des immobilisations supérieure 1000 par catégorie au cours des trois dernières années

Le tableau croisé exprime principalement les données, soit l'aspect statique. Le SID ayant aussi un aspect dynamique, nous proposons de guider la collecte des besoins relatifs à cet aspect *via* un graphe orienté.

4.2.2 Modèle : graphe de propriétés

Les besoins exprimés via le tableau croisé sont des besoins principalement liés à l'aspect statique des SID. Les besoins liés à l'aspect dynamique des SID sont principalement exprimés lors des interviews des utilisateurs par les concepteurs décisionnels. Cependant, cette tâche est généralement occultée au cours du processus de développement associé aux méthodes existantes. De plus, les méthodes, qui abordent cette tâche, ne fournissent pas de support pour la formalisation des besoins. L'objectif est donc de proposer un modèle pour la formalisation de l'aspect dynamique qui :
- soit compréhensible par les utilisateurs dans l'optique d'une validation,
- guide les concepteurs décisionnels au cours de cette tâche.

Ainsi, pour définir ce modèle, nous avons défini la liste des propriétés liées à l'aspect dynamique des SID, les plus récurrentes, que nous analysons au cours des projets décisionnels menés au sein de la société I-D6. Nous les avons regroupés en catégorie suivant les rapprochements sémantiques entre les propriétés.

Afin de guider les concepteurs décisionnels, nous avons défini les catégories de propriétés et les propriétés suivant une structure de graphe orienté. De plus, les noeuds d'un graphe peuvent porter des informations pertinentes qui lui sont liées. Nous proposons donc d'annoter le graphe avec des contraintes de données ou de poids exprimées en langage naturel liées à la propriété. Nous avons fait le choix de ne pas utiliser de langage de spécification tel que le langage Z et ses dérivés pour l'annotation des noeuds car 2/3 des groupes d'acteurs (groupe tactique et groupe stratégique) du projet SID sont des non-informaticiens.

Définition 4.3 (Graphe de propriétés) *Un graphe de propriétés est un graphe connexe orienté dont les noeuds sont les catégories de propriétés et les propriétés et dont les arêtes représentent les liens d'appartenance des propriétés aux catégories de propriétés du SID. L'annotation de ce graphe permet de spécifier les besoins liés à ces propriétés.*

Le modèle de graphe de propriétés est valable pour tous les acteurs contrairement au tableau croisé. Il permet d'évaluer les besoins suivant les deux composantes technique et décision. Nous avons constaté que le contexte décisionnel exprimé par les acteurs du SID est caractérisé par sept catégories qui sont « le niveau décisionnel existant », « l'exploitation », « la restitution », « la validité », « la transformation » et « la pérennité du SID ». De même, le contexte technique du SID est caractérisé par quatre catégories qui sont « le profil informatique », « la gestion des erreurs », « la sécurité » et « l'extraction - chargement des données ».

Nous avons défini les propriétés qui caractérisent le contexte technique et le contexte décision comme indiqué respectivement dans le tableau 4.4 et le tableau 4.5.

Catégorie	**Propriété**	**Description**
Profil informatique	Habitudes informatiques	Configurations informatiques avec lesquelles travaillent ces acteurs
	Aptitudes informatiques	Compétences et montées en compétences logicielles de ces acteurs
Gestion des erreurs	Suivi	Suivi des erreurs au cours de l'alimentation du SID
	Remontée	Remontée des erreurs au cours de l'alimentation du SID
Sécurité	Criticité	Importance de l'information au sein de l'organisation
	Ouverture	Accès au SID suivant le niveau hiérarchique, plus généralement suivant l'organisation interne de la société
Extraction - Chargement	Disponibilité	Plages horaires durant lesquelles le SID est exploitable
	Hétérogénéité	Diversité des sources de données en termes de structure, d'emplacement, de répartition des données
	Couverture	Domaines d'activités auxquels se rapportent les sources de données
	Complexité	Niveau de complexité sur la source en vue d'exploiter les données. Des sources sont qualifiées de complexes s'il faut les homogénéiser ou les centraliser afin de ne pas pénaliser l'activité quotidienne des sources et de faciliter la gestion des données

TAB. 4.4 – Propriétés « technique » du SID

Catégorie	Propriété	Description
Niveau décisionnel existant	Niveau d'informatisation décisionnelle	Outils décisionnels présents au sein de l'organisation
	Niveau d'urbanisation décisionnelle	Systèmes d'information décisionnels existants au sein du SID
Exploitation	Accès	Caractéristiques nécessaires du mode d'exploitation par accès ou outils d'accès au SID
	Interrogation	Caractéristiques nécessaires du mode d'exploitation par interrogation ou outils d'interrogation du SID
	Reporting	Caractéristiques nécessaires du mode d'exploitation par reporting ou outils de reporting du SID
	Analyse	Caractéristiques nécessaires du mode d'exploitation par analyse ou outils d'analyse du SID
Restitution	Modalités	Caractéristiques nécessaires de la restitution des informations
	Fragmentation	Découpage pour la restitution des objets du SID
	Réactivité	Gestion des accès aux objets du SID
Validité	Rafraîchissement	Caractéristiques de la conservation à court terme des informations
	Historisation	Caractéristiques de la conservation à moyen terme des informations dans le SID
	Archivage	Caractéristiques de la conservation à long terme des informations dans le SID
Transformation	Consolidation	Aggrégation nécessaires pour exploiter les informations
	Calcul	Type de calcul à appliquer sur les données (numériques, regroupement, analytiques)
Niveau décisionnel	Mode de pilotage	Type de gestion de l'organisation (coût, rentabilité)
	Périmètre	Environnement métier du SID
Pérennité	Dichotomie des espaces de stockage	Modèle conceptuel différent suivant le modèle décisionnel
	Normalisation du schéma du SID	Niveau de normalisation du schéma du SID indiquant le respect des architectures valides

TAB. 4.5 – Propriétés « décision » du SID

L'intérêt de l'utilisation d'un graphe orienté de concepts est la modélisation claire et précise des différentes propriétés du SID [Sowa 1998] et le guidage au cours de cette tâche. Nous définissons un graphe par groupe d'acteurs qui ne contient que les

noeuds liés aux propriétés exprimées par ce groupe. Nous représentons ces graphes de propriétés par groupe d'acteurs dans les sections dédiées par groupe. Le graphe de propriétés, tout groupe d'acteurs confondu, est présenté dans la figure 4.4.

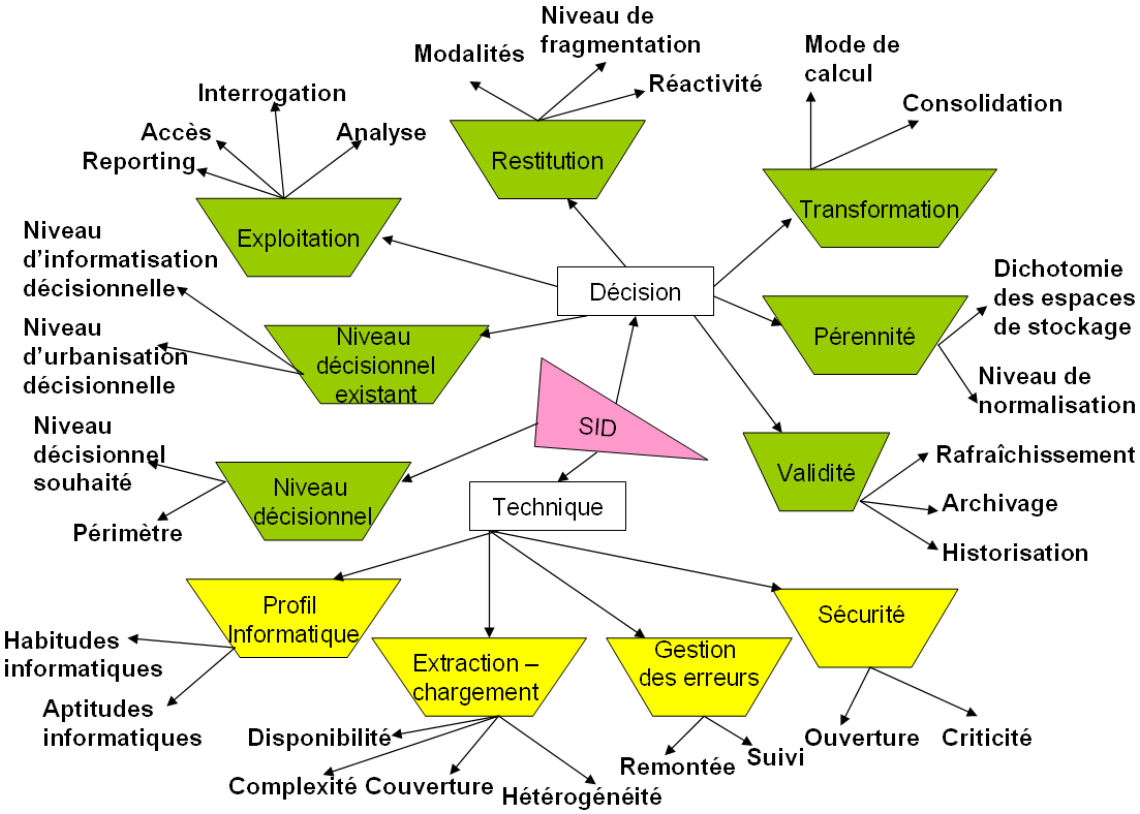


FIG. 4.4 – Graphe général de propriétés

Un graphe de propriétés est composé comme suit :
- noeud racine : groupe pour lequel les propriétés sont valables ou SID,
- noeud non-feuille de rang 1 : composante du SID,
- noeud non-feuille de rang 2 : catégorie de propriétés,
- noeud feuille : propriété du SID exprimé par le groupe.

Pour chaque groupe d'acteurs, nous définissons un graphe dont la racine (représentée par un triangle) porte le nom du groupe d'acteurs ou SID (dans le cas du graphe de propriétés du SID indépendamment des groupes). Un graphe est composé de deux sous-graphes principaux associés aux composantes du SID qui sont le sous-graphe technique et le sous-graphe décision (dont la racine est représentée par des rectangles). Ces sous-graphes sont composés de noeuds « non-feuille » qui représentent les catégories des propriétés CSD (représentés par des trapèzes).

Un avantage du graphe de propriétés est que sa taille est invariable. Autrement dit, la taille du projet décisionnel n'influe pas sur la taille du graphe de propriétés.

De plus, le graphe de propriétés présente aussi l'avantage de fournir une structure pour spécifier les propriétés CSD. L'annotation du graphe simplifie et favorise la validation des besoins exprimant l'aspect dynamique du SID par les décideurs. Dans la sous-section 4.2.5, nous présentons la démarche d'annotation des graphes.

4.2.3 Modèle : diagramme décisionnel

Dans les sous-sections précédentes, nous avons défini deux modèles pour la collecte des besoins suivant l'aspect statique et l'aspect dynamique ainsi que suivant la composante technique et la composante décision. Maintenant, il importe de formaliser ces besoins.

Nous avons pour objectif de proposer une formalisation qui a les propriétés suivantes :
- elle est proche de la vision multidimensionnelle des utilisateurs en s'inspirant du schéma en étoile et en constellation,
- elle permet d'exprimer les traitements liés au contexte technique et au contexte décisionnel,
- elle représente dans un seul schéma les données et les traitements de par la modélisation objet,
- elle représente les besoins des trois groupes d'acteurs,
- elle favorise la réutilisation et l'évolutivité des besoins.

Nous proposons le diagramme décisionnel présenté dans la figure 4.5. C'est une extension du diagramme de classes UML qui repose sur trois points :
- la définition de deux types de classes avec des stéréotypes basés sur les deux principaux concepts décisionnels,
- la spécification d'opérations au niveau de l'attribut car des traitements tels que le calcul qui définit un attribut est spécifique à un attribut et pas à toute la classe,
- le lien d'association 1- N entre les deux types de classes se lit « En fonction ».

Le diagramme décisionnel ne contient que deux types de classes afin de faciliter la compréhension des décideurs. La définition de ces deux types de classes vise à représenter les deux concepts reconnus par l'ensemble de chercheurs et des industriels. Les mesures des faits et les paramètres des dimensions sont représentés par les attributs des classes multidimensionnelles. Les traitements associés aux faits et aux dimensions sont représentés par des opérations. Le diagramme décisionnel répond au besoin de structuration multidimensionnelle des données suivant le schéma en étoile et plus globalement, le schéma en constellation [Kimball 1996] car il représente les classes-dimensions partagées autour des classes-faits. La modélisation orientée objet nous permet d'une part, de traiter séparément les données et les traitements bien qu'étant dans le même modèle et d'autre part, de favoriser la réutilisation au sein de notre méthode.

Définition 4.4 (Diagramme décisionnel) *Un diagramme décisionnel est une extension du diagramme de classes UML au multidimensionnel composée de deux classes multidimensionnelles définies par les stéréotypes << Fait >> et << Dimension >>. Il représente les classes-dimensions qui rayonnent autour des classes-faits.*

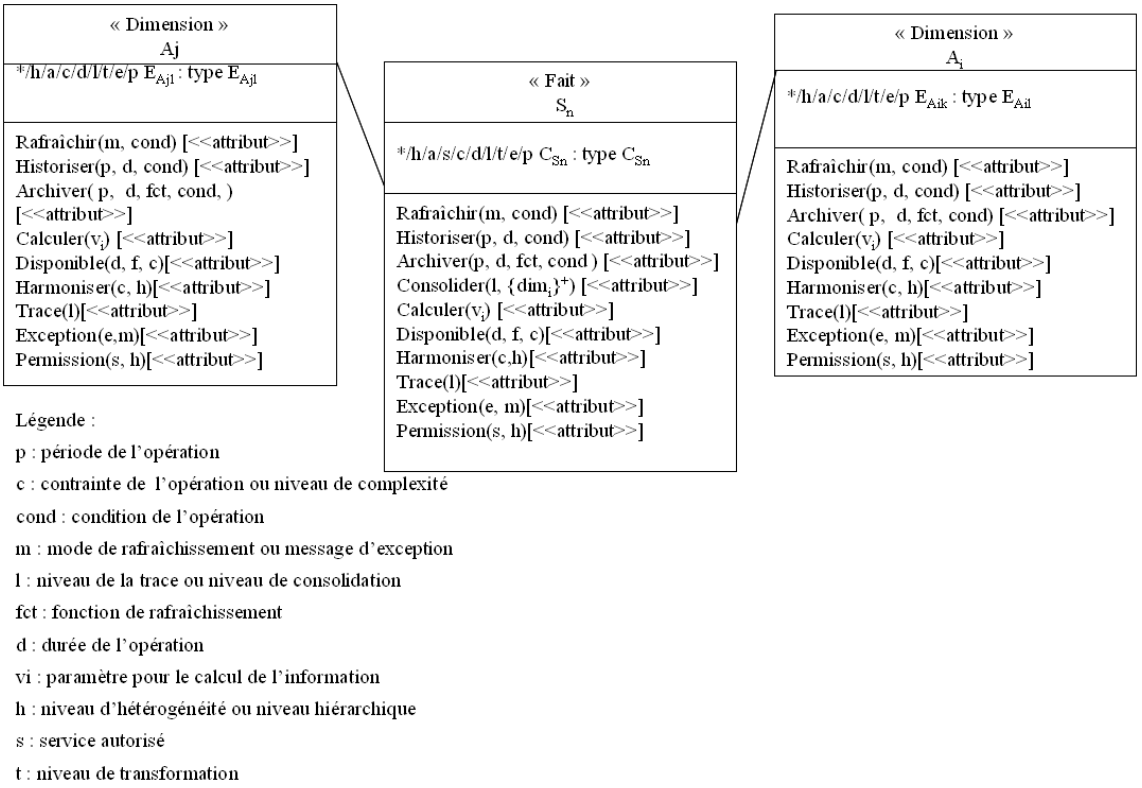


FIG. 4.5 – Diagramme décisionnel type

L'aspect statique du diagramme décisionnel type présenté dans la figure 4.5 est défini par :

- un fait S est représenté par une classe avec le stéréotype << Fait >>, appelée « classe-fait ». La classe-fait est composée d'attributs qui correspondent à ses mesures. La r-ième mesure C_{r_S} est de type $TypeC_{r_S}$,
- une dimension A_i est représentée par une classe avec un stéréotype << Dimension >>, appelée « classe-dimension ». La classe-dimension est composée d'attributs qui correspondent à ses paramètres. Le k-ième paramètre est $E_{k_{A_i}}$ de type $TypeE_{k_{A_i}}$,
- le lien entre une classe-fait et une classe-dimension est une association 1-N. Le fait peut être analysé uniquement suivant les dimensions qui lui sont liées. La connaissance de toutes les dimensions connectées au fait permet de déterminer ce dernier. Les multiplicités ne sont pas représentées.

L'aspect dynamique est défini à partir des propriétés annotées dans le graphe de propriétés. Les traitements liés au contexte technique du SID et ceux liés au contexte

décisionnel sont représentés suivant les signatures des opérations associées. Nous définissons cinq catégories de traitements à partir des onze catégories de propriétés car nous avons constaté que seules ces catégories sont nécessaires pour la formalisation des besoins. Les catégories de propriétés sont aussi utilisées pour d'autres tâches au cours du processus d'ingénierie que nous présentons dans les chapitres suivants. Les traitements que nous avons définis sont :

- disponibilité : l'alimentation du SID à partir des sources durant les périodes de forte sollicitation des sources dégrade les performances de l'activité quotidienne. De plus, le SID doit être disponible quand les décideurs en ont besoin. La gestion de la disponibilité des sources et du SID permet de réaliser la création et la mise à jour du SID durant les périodes de faible activité ou d'inactivité des sources sans perturber l'activité de l'organisation,
- complexité : les données restituées aux décideurs ont généralement subi des transformations telles que des calculs numériques simples, des agrégations ou des calculs analytiques. Cette complexité est aussi fonction de l'hétérogénéité des sources de données qu'il faut regrouper et harmoniser,
- suivi des erreurs : les traitements de création et de mise à jour du SID peuvent générer des erreurs qui indiquent des incohérences dans la conception ou l'implantation du SID. Ces erreurs peuvent se produire à plusieurs niveaux qui sont au niveau donnée, au niveau traitement ou encore au niveau global,
- remontée des erreurs : les erreurs générées au cours de la création et de la mise à jour du SID impactent la qualité des données restituées aux décideurs. Les concepteurs décisionnels doivent être informés de ces erreurs,
- criticité : les informations contenues dans le SID sont critiques et ne doivent généralement pas être consultables par tout le personnel de l'organisation,
- consolidation : les agrégations de toutes les mesures du fait ne sont pas toujours pertinentes [Husemann et al. 2000]. Il faut donc préciser quelles fonctions d'agrégations peuvent être appliquées. Nous utilisons les niveaux de pertinence des agrégations définis par [Pedersen and Jensen 1999],
- calcul : les données restituées aux décideurs résultent de calculs qu'il faut spécifier dès la conception afin de déceler des incohérences voire de déterminer les éventuelles données manquantes,
- rafraîchissement : les données contenues dans le SID doivent être rafraîchies en fonction du cycle de décision afin que les données restituées soient toujours fiables pour les décideurs,
- historisation : les décideurs souhaitent conserver les données afin que des requêtes soumises sur les données à des intervalles de temps fournissent le même résultat. Lors des rafraîchissements du SID, il faut donc déterminer la période durant laquelle les données doivent être conservées,
- archivage : l'historisation des données correspondant à une sauvegarde dans leur niveau de détail initial, mais cela génère d'importants volumes de données qui dégradent les performances du SID. Pour pallier à cela, les données doivent être historisées de manière agrégée suivant une fonction d'agrégation.

Le tableau 4.6 présente les opérations associées aux traitements par catégorie.

Catégorie	Traitement	Signature de l'opération et description
		Dérivation
Extraction - Chargement	Harmonisation des données	**Harmoniser(c, h)** : harmonisation des sources de données à partir desquels le concept est dérivé avec un niveau de complexité c et un niveau d'hétérogénéité h
Gestion des erreurs	Suivi des erreurs	**Trace(l)** : suivi des erreurs avec un niveau de suivi l (valeur du niveau 1 : données, 2 : traitements, 3 : global)
Gestion des erreurs	Remontée des erreurs	**Exception(e, m)** : remontée des erreurs e avec le message m
Accès	Permission d'accès	**Permission(s, h)** : l'accès au concept est autorisé au service s avec le niveau hiérarchique h
		Préparation
Validité des données	Rafraîchissement	**Rafraîchir(f, m, cond)** : rafraîchissement du concept décisionnel avec la fréquence de rafraîchissement f, de condition de rafraîchissement cond et de mode de rafraîchissement m
Validité des données	Historisation	**Historiser(p, d, cond)** : historisation du concept décisionnel de période de l'historisation p, de contrainte d'historisation c et de condition de l'historisation cond
Validité des données	Archivage	**Archiver(p, d, fct, cond,)** : archivage du concept décisionnel de période de l'archivage p, de contrainte d'archivage c, de condition de l'archivage cond et suivant la fonction d'agrégation fct
Transformation	Consolidation	**Consolider(l,**$\{dim_i\}^+$**)** : consolidation au niveau de consolidation l (cf. tableau 4.7) défini par [Pedersen and Jensen 1999] afin de prendre en compte les agrégations pertinentes. Par défaut, les agrégations sont pertinentes suivant toutes les dimensions connectées au fait. $\{dim_i\}^+$ sont les dimensions suivant lesquelles les agrégations ne sont pas pertinentes.
Transformation	Calcul	**Calculer(**$\{v_i\}^+$**)** : calcul de l'information à partir des paramètres v_i
Restitution	Disponibilité des données	**Disponible(d, f, c)** : le SID est disponible pendant la durée d, avec la fréquence f et la contrainte c en fonction de disponibilité des sources et de la réactivité nécessaire aux décideurs

TAB. 4.6 – Signatures des opérations associées aux traitements du SID

Niveau de consolidation	Fonctions d'agrégation applicables
1	sum, avg, min, max, stdev, var, count
2	avg, min, max, stdev, var, count
3	count
4	pas de fonction applicable

TAB. 4.7 – Niveaux de consolidation et fonctions d'agrégation associées

Afin d'exprimer qu'un attribut fait l'objet d'un traitement, nous définissons le concept d'informativité.

Définition 4.5 (Concept d'informativité) *Un concept d'informativité est un symbole associé à un traitement qui indique qu'un attribut fait l'objet de ce traitement. Il se place à côté des symboles de visibilité de l'attribut.*

Le concept d'informativité et les propriétés nécessaires pour la définition des traitements sont définis dans le tableau 4.8. La colonne « Groupe » indique le groupe pour lequel le traitement est pertinent (T : Tactique, St : Stratégique et Sy : Système).

Traitement	**Propriétés**	**Informativité**	**Groupes**
	Dérivation		
Harmonisation des données **Harmoniser(c,h)**	Complexité, Hétérogénéité	l	Sy
Suivi des erreurs **Trace(l)**	Suivi	t	T, Sy
Remontée des erreurs **Exception(e,m)**	Remontée	e	T, Sy
Permission d'accès **Permission(e,m)**	Ouverture, Criticité	p	St
	Préparation		
Rafraîchissement **Rafraîchir(m,cond.)**	Rafraîchissement	*	T, St, Sy
Historisation **Historier(p,d,cond.)**	Historisation	h	T, St, Sys
Archivage **Archiver(p,d,t,cond.)**	Archivage	a	T, St, Sys
Consolidation **Consolider(l,{dim$_i$})**	Consolidation	s	T, St, See
Calcul **Calculer({v$_i$}$^+$)**	Calcul	c	T, St, Sy
Disponibilité des données **Disponible (d,f,c)**	Disponibilité, Réactivité	d	T,Sy

TAB. 4.8 – Concept d'informativité et propriétés associés aux traitements du SID par groupe

Dans la même optique, les valeurs de paramètres d'une opération définie pour une classe multidimensionnelle ne sont pas nécessairement valables pour tous les attributs de cette classe. Pour cela, il importe de définir un comportement relatif aux attributs des classes multidimensionnelles. Cependant, UML ne permet pas de définir des opérations au niveau de l'attribut d'une classe. Ainsi, nous proposons des opérations d'attribut en ajoutant le stéréotype attribut << attribut >> à la signature de l'opération définie avec les valeurs de paramètres valables pour l'attribut. Pour indiquer l'attribut auquel s'applique l'opération, il est défini comme le premier paramètre de l'opération. Il est donc possible de définir et de distinguer les opérations de classe et les opérations d'attribut.

Ainsi, avec le concept d'informativité et celui de opération d'attribut, il est possible d'exprimer qu'un attribut fait l'objet d'un processus d'historisation comme le suggère les différents niveaux d'historisation définis par [Teste 2000].

Afin de distinguer le diagramme décisionnel associé à chaque groupe et en raison des propriétés et traitements spécifiques à chaque groupe d'acteurs, nous utilisons trois concepts :

- diagramme décisionnel tactique (TDD) : diagramme décisionnel du groupe tactique,
- diagramme décisionnel stratégique (KDD avec K pour « kernel ») : diagramme décisionnel du groupe stratégique,
- diagramme décisionnel système (SDD) : diagramme décisionnel du groupe système.

4.2.4 Démarche : collecte des besoins utilisateurs

Les documents d'expression des besoins des groupes tactique et stratégique contiennent les tableaux de chiffres qu'ils manipulent au quotidien. Ces tableaux croisés peuvent contenir des informations identiques ; ce qui est redondant et qui augmente la charge de travail ainsi que les problèmes de cohérence des données. En ce sens, suivant une comparaison syntaxique, nous préconisons de ne pas évaluer les tableaux croisés qui ont les mêmes faits caractérisés par les mêmes mesures et les mêmes dimensions composées des mêmes paramètres comme le définit la règle suivante :

- soit TM l'ensemble des tableaux croisés TM_i d'un groupe d'acteurs (tactique ou stratégique),
- soit TM_A l'ensemble des tableaux croisés des besoins d'un groupe d'acteurs,
- soit F_{TM_i}l'ensemble des faits analysés f_i dans le tableau croisé TM_i avec i $\in \mathbb{N}^*_+$,
- soit D_{TM_i} l'ensemble des dimensions d_i du tableau croisé TM_i avec i $\in \mathbb{N}^*_+$.

$$\forall\ \mathrm{TM}_j,\ \mathrm{TM}_i \in TM,\ (F_{TM_i} = F_{TM_j} \wedge D_{TM_i} = D_{TM_j}) \\ \Rightarrow (TM_i \notin TM_A, TM_j \in TM_A)$$

4.2.5 Démarche : annotation d'un graphe de propriétés

Après la collecte de la statique du SID par l'étude des tableaux croisés au regard du tableau croisé type 4.1, nous proposons de collecter l'aspect dynamique des besoins de manière semi-automatique. Cette tâche est partagée par les trois groupes d'acteurs. Pour cela, nous interviewons les acteurs suivant un parcours en profondeur du graphe de propriétés de ce groupe d'acteurs. A chaque noeud non-feuille de rang 2 et noeud feuille nous interrogeons le groupe d'acteur à propos de la catégorie de propriétés ou de la propriété comme suit :

- les noeuds non-feuilles de rang 2 : nous interrogeons le groupe d'acteurs sur l'importance afin de déterminer le poids de la catégorie de propriétés, soit p_i. Le coefficient de pondération (p) de la catégorie peut prendre des valeurs entre 1 et 4, par défaut il est égal à 1. Nous définissons une assertion pour

chaque catégorie p=p_i. Ces assertions constituent des contraintes de poids. Elles définissent des priorités parmi les besoins du groupe,
- les noeuds feuilles : nous posons une question relative à la propriété. A partir de la réponse des décideurs du groupe, nous définissons une contrainte de valeurs en langage naturel. Ces contraintes de valeurs doivent être vérifiées lors de la conception, et plus généralement, lors du développement sur le principe de la conception par contrat de Meyer [Meyer 2000].

Ces contraintes de valeurs permettent de définir les paramètres des opérations associées aux traitements lors de la formalisation des besoins de chaque groupe d'acteurs. Les contraintes de poids permettent la confrontation entre les besoins du SID.

4.2.6 Démarche : formalisation des besoins utilisateurs

Notre méthode d'analyse des besoins des acteurs du SID repose sur trois principales tâches qui sont la collecte des besoins, la formalisation des besoins et la confrontation des besoins. La figure 4.3 présente l'ordonnancement séquentiel de ces dernières.

La tâche de formalisation des besoins bien que reposant sur le même modèle n'est pas commune à tous les trois groupes car les besoins sont exprimés *via* des modèles différents par les décideurs et par acteurs systèmes. Elle est commune uniquement aux groupes tactique et stratégique.

La formalisation des besoins correspond à la représentation *via* le diagramme décisionnel. Pour passer des tableaux croisés et du graphe de propriétés au diagramme décisionnel d'un groupe de décideurs, nous proposons un processus automatique de structuration des données et des traitements basé sur trois types de règles :
- les règles de transformation : elles permettent de définir les diagrammes décisionnels à partir des tableaux croisés,
- les règles syntaxiques : elles permettent de vérifier que les diagrammes décisionnels dérivés sont cohérents par rapport à la définition d'un diagramme décisionnel,
- les règles de fusion : elles permettent de fusionner deux diagrammes décisionnels d'un même groupe d'acteurs ou de groupes d'acteurs différents.

Le concepteur applique d'abord les règles de transformation des tableaux croisés en diagrammes décisionnels, puis les règles syntaxiques et enfin les règles de fusion. La définition de ces règles suppose que la sémantique est fiable.

Pour chaque tableau croisé type, il applique les règles de transformation relatives aux données, puis celles relatives aux traitements.

- Règles de transformation de l'environnement du projet Ex :
  - Information EI :
    - **EI1** : un diagramme décisionnel est associé à tout tableau croisé type,
    - **EI2** : tout tableau croisé spécifiant les besoins utilisateurs doit avoir une structure proche de celle du tableau croisé type(cf. tableau 4.1). Dans le cas contraire, il est transformé en un tableau à deux dimensions qui sont la dimension « Temps » et une des dimensions principales du domaine d'application lié. Cette dernière est positionnée à la valeur « All ».

- Règles de transformation du fait Sx :
  - Information SI :
    - **SI1** : une classe-fait avec le stéréotype << Fait >> est associée à tout fait,
    - **SI2** : un attribut est associé à toute mesure.
  - Traitements SP :
    - **SP1** : une opération est définie pour chaque traitement (applicable à ce groupe en fonction du tableau 4.8) dont les propriétés associées sont annotées dans le graphe de propriétés,
    - **SP2** : une opération d'attribut est associée à toute mesure qui possède des arguments différents de ceux de la classe pour un traitement donné,
    - **SP3** : les concepts d'informativité sont associés aux attributs de la classe-fait en fonction des traitements déclarés.

- Règles de transformation des dimensions Ai :
  - Information AI :
    - **AI1** : une classe-dimension avec le stéréotype << Dimension >> est associée à toute dimension,
    - **AI2** : un attribut « Id» est associé à chaque classe-dimension,
    - **AI3** : un attribut est associé à tout paramètre,
  - Traitements AP :
    - **AP1** : une opération est définie pour chaque traitement (applicable à ce groupe en fonction du tableau 4.8) dont les propriétés associées sont annotées dans le graphe de propriétés,
    - **AP2** : une opération d'attribut est associée à tout paramètre qui possède des arguments différents pour un traitement donné. La classe-dimension « Temps » n'est généralement pas rafraîchie,
    - **AP3** : les concepts d'informativité sont associés aux attributs de la classe-dimension en fonction des traitements déclarés,

Puis, les règles syntaxiques sont appliquées :
- Information SDI :
  - **SDI1** : une classe-dimension ne peut pas être reliée une autre classe-dimension,
  - **SDI2** : une classe-fait ne peut pas être reliée à une autre classe-fait,
  - **SDI3** : à toute mesure est associée le concept d'informativité et le traitement d'historisation sur l'exercice précédent pour l'analyse des tendances,

  - **SDI4** : à tout paramètre est associé le concept d'informativité d'historisation et le traitement sur l'exercice précédent pour l'analyse des tendances.
- Traitements SDP :
  - **SDP1** : si le concept d'informativité porte sur tous les attributs de la classe et avec les mêmes paramètres alors l'opération liée est spécifiée au niveau de la classe,
  - **SDP2** : si une des mesures du fait de l'analyse possède les concepts d'informativité liée à l'historisation « h» ou à l'archivage « a» alors toutes les dimensions liées doivent posséder aussi cette propriété. De plus, la période et la condition de l'opération associée doivent être au moins égales à celles de la mesure,
  - **SDP3** : l'opération « Rafraîchir() » ne doit pas être appliquée à la dimension temps car elle est initialisée pour assurer tout le cycle de vie du SID.

A partir de ces diagrammes décisionnels, le concepteur décisionnel applique les règles de fusion afin de simplifier la confrontation des besoins en manipulant le moins de diagrammes possible, soit un diagramme par groupe d'acteurs, il convient de les fusionner à l'aide des règles associées. Le diagramme décisionnel obtenu (étoile ou constellation) varie suivant les possibilités de mise en commun des classes-dimensions et des classes-faits, autrement dit suivant la logique de l'organisation.

- FUS : regrouper les diagrammes décisionnels ayant les mêmes classes-faits et des classes-dimensions en commun,
  - **FUS1** : fusionner les classes-dimensions partagées par ajout des attributs et des opérations,
  - **FUS2** : fusionner les classes-faits par ajout des attributs et des opérations,
  - **FUS3** : ajouter les classes-dimensions propres à chaque diagramme.
- FDS : regrouper les diagrammes décisionnels ayant des classes-faits différentes et des classes-dimensions en commun,
  - **FDS1** : fusionner les classes-dimensions par ajout des attributs et des opérations.
- FDR : regrouper les classes-dimensions qui ont des attributs en commun,
  - **FDR1** : pour les classes-dimensions de noms différents ayant peu d'attributs en commun, définir les relations de dépendance fonctionnelle entre les attributs des classes-dimensions et les attributs communs. Puis, il faut supprimer les attributs communs des classes-dimensions dont les attributs communs et les attributs de ces classes-dimensions ne définissent pas des relations de dépendance fonctionnelle directes. Enfin, il faut définir les relations de dépendance fonctionnelle entre ces classes-dimensions,
  - **FDR2** : les classes-dimensions de noms différents mais, dont les attributs sont identiques à l'identifiant près sont fusionnés,
  - **FRD3** : définir des n relations « En Fonction » entre la classe-fait et la classe-dimension résultant de la fusion de n classes-dimensions,
  - **FRD4** : définir n rôles différents exprimant la sémantique du lien entre la classe-fait et cette classe-dimension.
- FRC : fusion des liens entre classes multidimensionnelles,
  - **FRC1** : reporter les liens entre les différents classes multidimensionnelles,

– **FRC2** : pour la cohérence des données, une classe-dimension ne peut pas être cible d'une relation de dépendance fonctionnelle et participer à des liens d'association avec des classes-faits qui sont liés à la classe-dimension qui est source de la relation de dépendance fonctionnelle. Le lien de dépendance fonctionnelle est conservé et les liens d'association avec les classes-faits sont supprimés.

Il y a toujours au moins un élément en commun, en l'occurrence la dimension Temps. La confrontation des besoins prend toujours en paramètre deux diagrammes.

4.3 Analyse des besoins tactiques

Dans cette section, nous présentons la méthode d'analyse des besoins du groupe tactique. La particularité de ce groupe est que les acteurs sont liés à des métiers spécifiques. Ces acteurs ont donc une vision verticale de la prise de décision au sein de l'organisation.

4.3.1 Collecte

Au début de l'analyse des besoins du groupe tactique, les tableaux croisés ou les requêtes des exigences analytiques définis dans le document d'expression des besoins tactiques sont analysés. Nous évaluons la redondance de ces tableaux suivant la démarche de collecte des besoins utilisateurs.

Exemple : après l'étude du document d'expression des besoins du groupe tactique, les tableaux croisés définis sont les tableaux 4.2, 4.9, 4.10.

Immobilisations. Valeur vénale			Temps.Année		
			2003	2004	2005
Bien.Date achat	Bien.id	Bien.no immo			
14/03/2005	125486	AB4582	523,14	424.25	409.21
25/03/2005	125495	AB5512	715,45	698,14	654,78
05/04/2005	137258	AC2451	8 245,85	8 019,24	7 985,23
31/09/2005	145326	AB6981	2 452,64	2 425,54	2 425,54

TAB. 4.9 – Tableau croisé de la valeur vénale des biens du fabricant "BELL" et d'état "utilisé" pour les trois dernières années

Immobilisations. Valeur vénale		Etats.Etat_description		
		Disponible	Utilisé	Hors usage
Catégories.Catégorie	Catégories.Sous_catégorie			
Immateriel	Logiciels	142 542,25	24 425,14	6202,70
	Progiciels	24 527	59 785,26	1 531,31
Postes utilisateurs	Ecrans	5 36,14	3 945,21	142,77
	PC	1 452,12	5 024,53	944,30
	Terminaux	224,12	1 067,22	0,00

TAB. 4.10 – Tableau de bord de la valeur vénale par catégorie et par état au cours l'année 2005

Ces tableaux croisés partagent des dimensions et des faits mais ils ne se recouvrent pas totalement, les trois sont donc évalués pour l'analyse.

Après la collecte de la statique du SID, nous interviewons les décideurs en parcourant le graphe de propriétés du groupe tactique suivant la démarche d'annotation des graphes. Nous obtenons le graphe de propriétés représenté dans la figure 4.6.

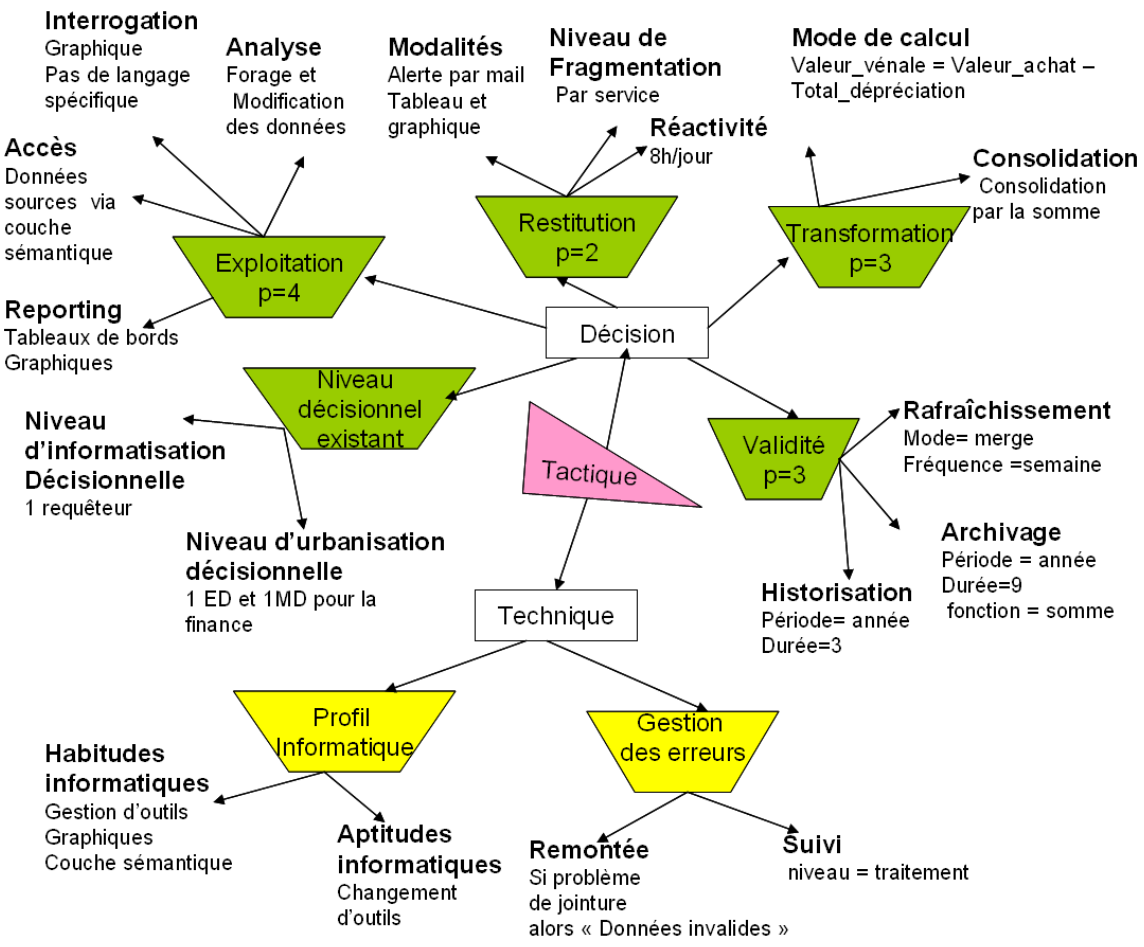


FIG. 4.6 – Graphe de propriétés CSD du groupe tactique du projet immobilisations

4.3.2 Formalisation

Chaque tableau croisé est transformé en un diagramme décisionnel en fonction du graphe de propriétés tactiques. La transformation est faite à partir des concepts des tableaux croisés et du diagramme décisionnel. La mise en oeuvre de la démarche de formalisation des besoins pour le tableau 4.2 est la suivante :

- EI1 : il y a 3 tableaux croisés donc 3 diagrammes décisionnels seront crées,
- EI2 : les tableaux croisés sont bien structurés.

- SI1 : la classe-fait « Immobilisations » est associée au fait « Immobilisations »,
- SI2 : l'attribut « Valeur_vénale » est associé à la mesure « Valeur_vénale »,
- SP1 : les propriétés « suivi », « remontée », « rafraîchissement », « historisation », « archivage », « consolidation », « calcul », « niveau d'informatisation décisionnelle », « niveau d'urbanisation décisionnelle » sont annotées alors nous spécifions les opérations associées,

- SP2 : toutes les opérations sont au niveau de la classe multidimensionnelle car elles ne sont pas spécifiques à au moins une mesure du fait, sauf l'opération « Calculer() ». Cette dernière est déclarée au niveau de l'attribut car elle indique que l'attribut est calculée à partir de la « Valeur_achat » et de la « Dépréciation », soit Calculer(Valeur_vénale, Valeur_achat, Dépréciation) << attribut >>,
- SP3 : les concepts d'informativité « c » et « s » sont associés à la mesure « Valeur_vénale » car elle est calculée et consolidée en plus de ceux liés aux traitements définis au niveau de la classse-fait.

- AI1 : les classes-dimensions « Catégories » et « Temps » sont associées aux dimensions « Catégories » et « Temps »,
- AI2 : l'attribut « Id » est ajouté à chaque classe-dimension « Temps » et « Catégories »,
- AI3 : les attributs de la classe-dimension « Temps » sont « Mois », « Trimestre », « Semestre » et « Année ». Les attributs de la classe-dimension « Catégories » sont « Catégorie » et « Sous_catégorie »,
- AP1 : les propriétés « suivi », « remontée », « rafraîchissement », « historisation », « archivage », « consolidation », « calcul », « niveau d'informatisation décisionnelle », « niveau d'urbanisation décisionnelle » sont annotées alors nous spécifions les opérations associées,
- AP2 : les opérations sont définies au niveau des classes-dimensions car il n'y a pas de spécificités pour un paramètre donné. La classe-dimension « Temps » n'est pas rafraîchie dans ce projet,
- AP3 : les concepts d'informativité « h », « * », « a », « t », « e » sont appliqués aux attributs des classes-dimensions car les opérations associées sont définies.

On obtient alors le diagramme décisionnel associé au tableau 4.2.

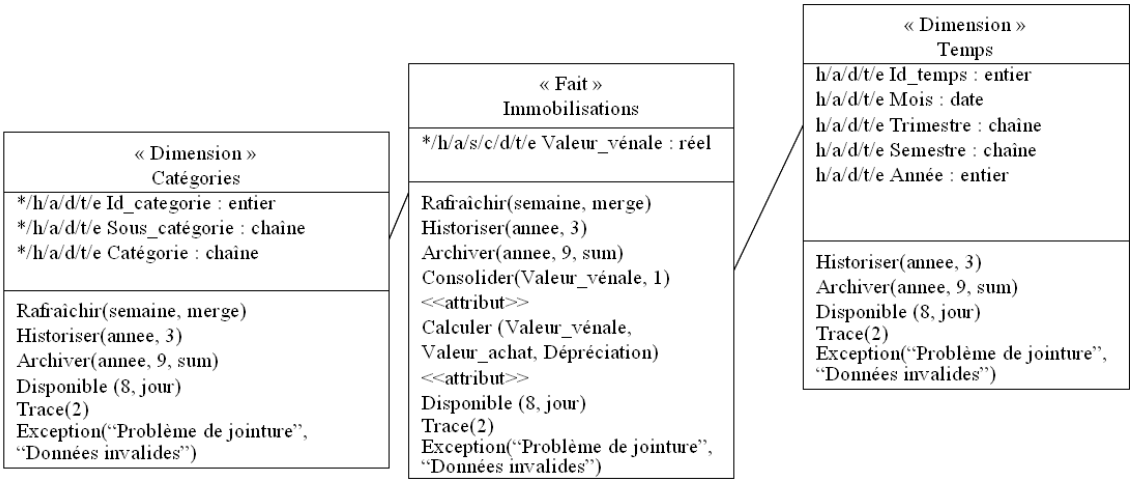


FIG. 4.7 – Diagramme décisionnel associé au tableau 4.2

En appliquant les règles de transformation aux tableaux croisés 4.9 et 4.10 comme ci-dessus, on obtient les diagrammes décisionnels représentés dans les figures 4.8 et 4.9.

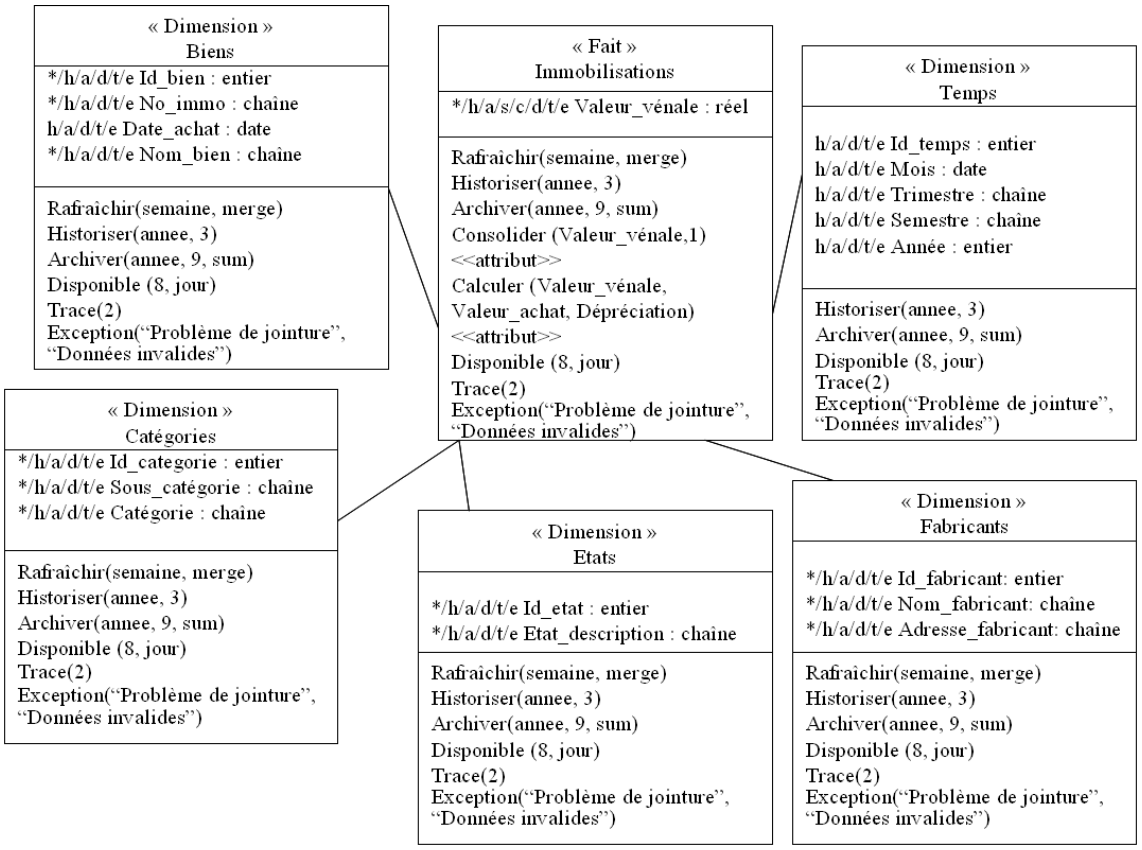


FIG. 4.8 – Diagramme décisionnel associé au tableau 4.9

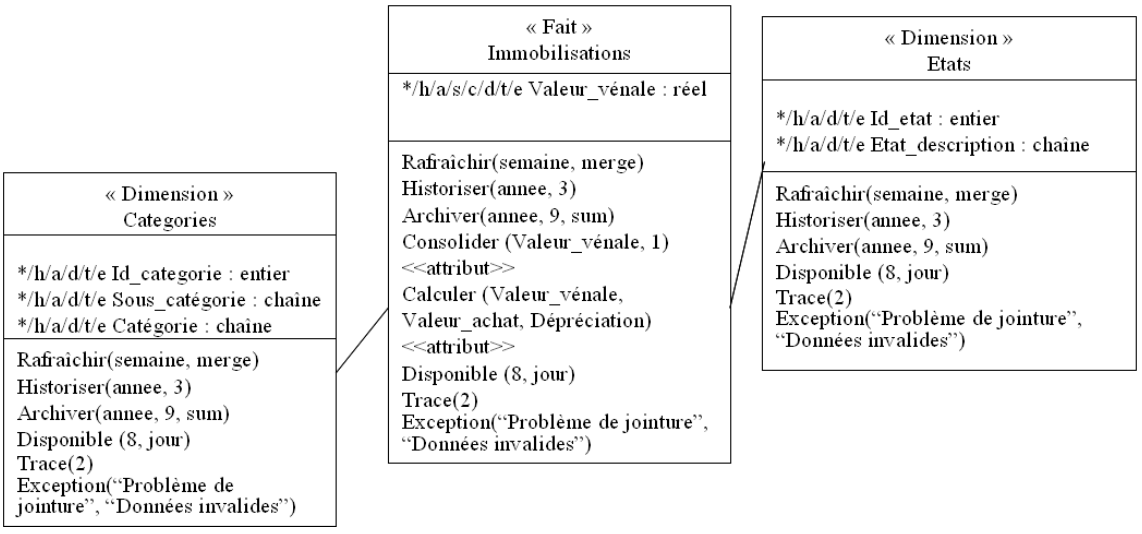


FIG. 4.9 – Diagramme décisionnel associé au tableau 4.10

Lors de la vérification des règles syntaxiques à la fin de la conception des diagrammes décisionnels, il n'y a pas d'incohérence détectée. Nous fusionnons les diagrammes du groupe tactique dans l'optique de la confrontation. Tous les diagrammes décisionnels ont la même classe-fait : « Immobilisations ». On applique donc uniquement les règles FUS car les diagrammes ont la même classe-fait « Immobilisations » et des classes-dimensions en commun,

- FUS1 : la fusion des classes-dimensions par ajout des attributs laisse les classes-dimensions inchangées car les attributs étaient les mêmes pour chaque classe-dimension. Les classes-faits « Immobilisations » des diagrammes ont en commun les classes-dimensions « Temps », « Catégories », « Etats »,
- FUS2 : la fusion des classes-faits n'a pas lieu d'être car elles ont les mêmes données et les mêmes opérations,
- FUS3 : la classe-dimension « Fabricants » et « Biens » sont ajoutées.

Les trois diagrammes décisionnels sont fusionnés en un seul composé d'une classe-fait et de cinq classes-dimensions. Le diagramme décisionnel du groupe tactique résultant est présenté dans la figure 4.10.

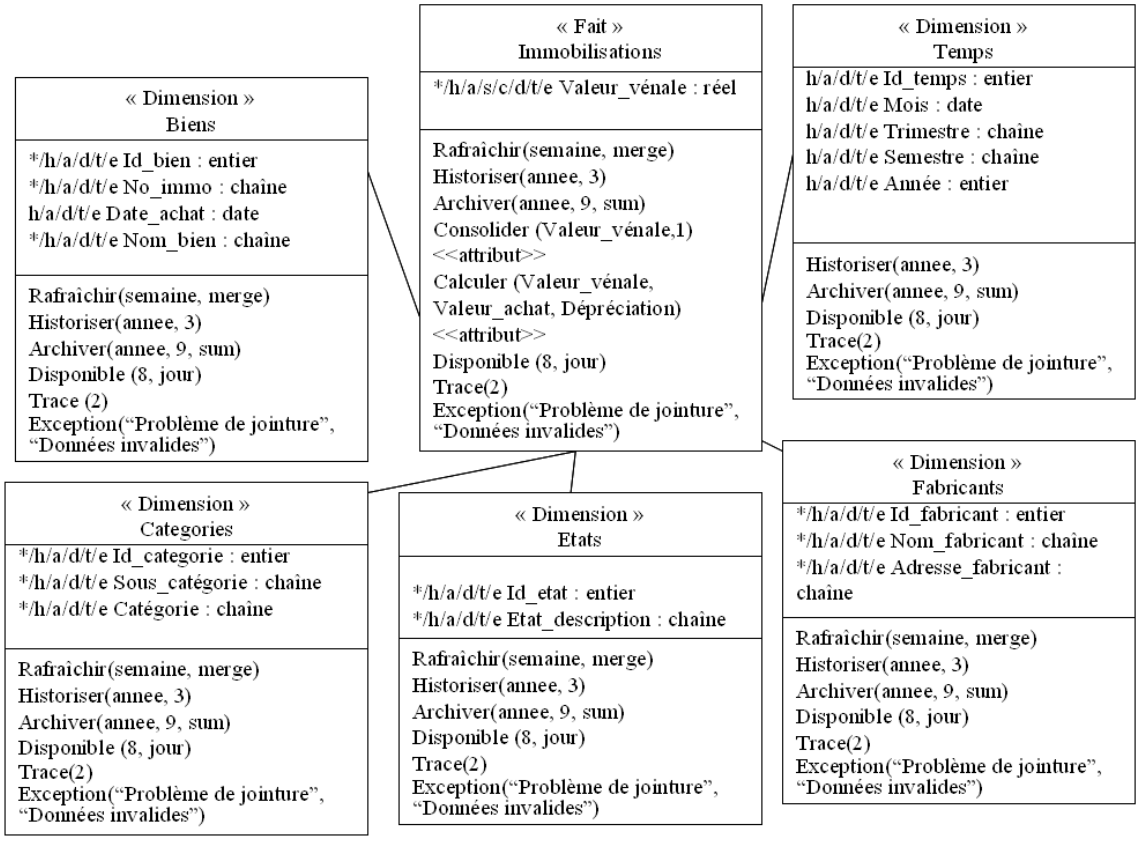


FIG. 4.10 – Diagramme décisionnel du groupe tactique du projet immobilisations

4.4 Analyse des besoins stratégiques

Le groupe stratégique est composé des « hauts » décideurs. Il exprime des besoins liés à la gouvernance de l'organisation. Ces acteurs ont une vue synthétique et globale de celle-ci. En plus, de ces besoins qui définissent en grande partie le contexte décisionnel, ces acteurs expriment des besoins liés à la sécurité de l'information.

Au début du projet, ces acteurs définissent le périmètre du projet dont les domaines d'activités concernés. Nous définissons un domaine d'activités comme un secteur d'activité, un ensemble de métiers conjointement liés ou une classe d'utilisateurs d'une organisation.

4.4.1 Collecte

Contrairement aux décideurs du groupe tactique qui ont une vision détaillée de leur métier, les acteurs stratégiques ont une vision synthétique de l'organisation. Ces derniers expriment donc principalement leurs besoins sous forme de tableaux qui présentent les tendances des faits au cours du temps *via* une liste d'indicateurs de performance. Ces indicateurs de performance sont les mesures du sujet d'analyse, soit le fait. Ces tableaux qui sont des variantes des tableaux croisés sont appelés des **tableaux croisés synthétiques**.

Exemple : le tableau croisé synthétique défini dans le document d'expression des besoins stratégiques est présenté dans le tableau 4.11.

	Temps.Mois		
	Octobre	Novembre	Décembre
Valeur vénale	19 452 422,15	19 985 62,10	21 310 771,72
Amortissement	950 758,19	1 095 058,19	1 116 726,79
Dépréciation	9 757 215,25	9 895 456,38	10 823 340,82

Tab. 4.11 – Tableau croisé synthétique des immobilisations sur le dernier trimestre

Nous définissons les tableaux croisés valides pour l'analyse des besoins stratégiques.

Exemple : Dans notre exemple, il y a un seul tableau croisé synthétique donc nous n'appliquons pas les règles de sélection des tableaux.

Les mesures des tableaux croisés synthétiques sont souvent liées car les décideurs croisent les chiffres afin d'évaluer de manière plus précise les tendances. Ces mesures dérivées sont pertinentes pour les décideurs mais elles impactent la fiabilité des

données. Il importe donc de définir les règles de calcul associées à chaque mesure et les liens entre ces mesures lors de la collecte de l'aspect dynamique des besoins.

Exemple : les mesures du tableau 4.11 sont liées. Plus précisément, la mesure « Valeur vénale » est une mesure dérivée à partir de la soustraction de la mesure « Dépréciation » à la « Valeur_achat » du bien. La mesure « Amortissement » est une mesure fournie directement par les sources.

Après l'analyse de la statique des besoins stratégiques, nous procédons à l'interview des acteurs stratégiques suivant la démarche d'annotation du graphe de propriétés CSD. Le résultat de collecte des besoins du groupe stratégique exprimant l'aspect dynamique est présenté dans la figure 4.11.

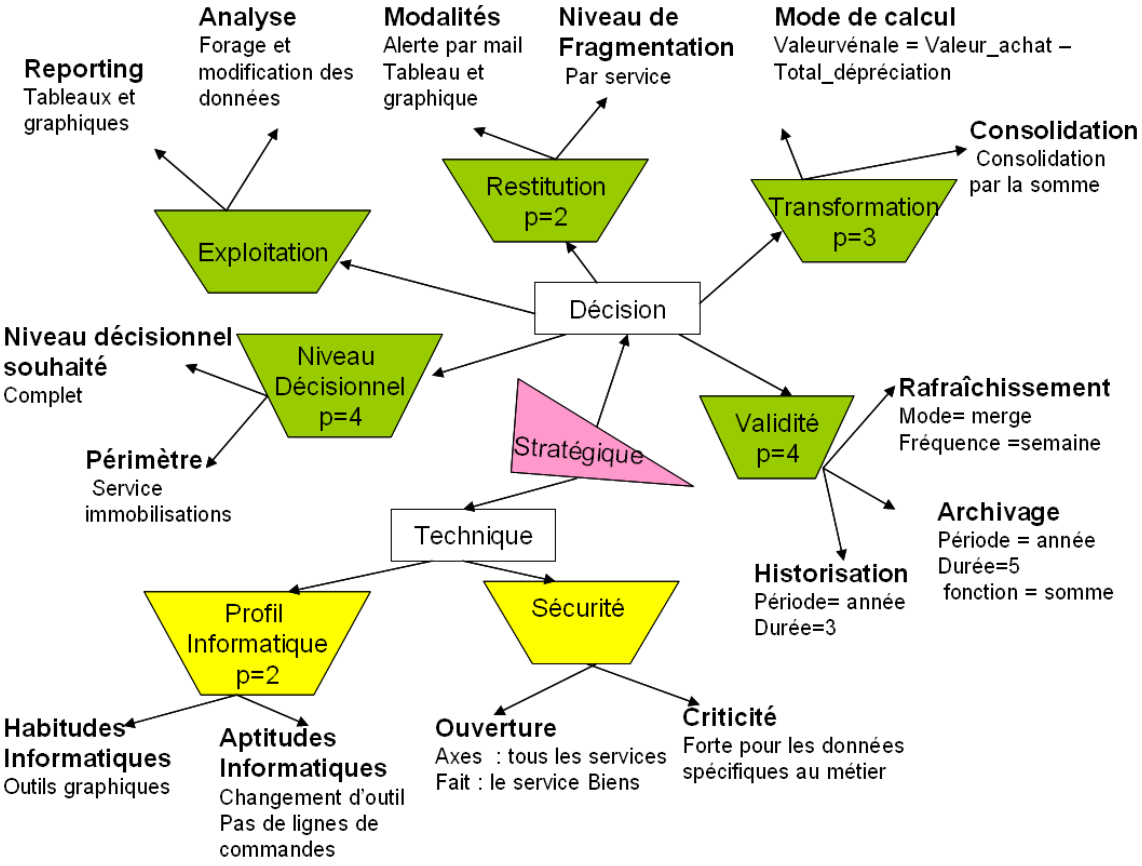


FIG. 4.11 – Graphe de propriétés CSD du groupe stratégique du projet immobilisations

4.4.2 Formalisation

Les besoins du groupe stratégique sont exprimés sous forme synthétique du tableau croisé. Pour la formalisation des besoins, il est nécessaire de transformer les tableaux croisés synthétiques en des tableaux croisés.

- EI1 : il y a un tableau croisé synthétique donc un diagramme décisionnel est crée,
- EI2 : le tableau est restructuré en un tableau à deux dimensions, soit les dimensions « Biens » et « Temps », présentant le fait « Immobilisations » caractérisé par trois mesures qui sont : « Valeur_vénale », « Amortissement », « Dépréciation ».

- SI1 : la classe-fait « Immobilisations » est associée au fait « Immobilisations »,
- SI2 : les attributs « Valeur_vénale », « Amortissement », « Dépréciation » sont associés aux mesures « Valeur_vénale », « Amortissement », « Dépréciation »,
- SP1 : les propriétés « criticité », « ouverture », « rafraîchissement », « historisation », « archivage », « consolidation », « calcul » sont annotées alors nous spécifions les opérations associées,
- SP2 : toutes les opérations sont au niveau de la classe car elles ne sont pas spécifiques à au moins une mesure du fait, sauf l'opération « Calculer() » ». Elle est déclarée au niveau de l'attribut car elle indique que l'attribut est calculée à partir de la « Valeur_achat » et du « Dépréciation », soit Calculer(Valeur_vénale, Valeur_achat, Dépréciation) << attribut >>,
- SP3 : les concepts d'informativité « p », « h », « * », « a », « s », « c » sont associés aux attributs en raison des traitements définis au niveau de classe-fait.

- AI1 : les classes-dimensions « Biens » et « Temps » sont associées aux dimensions « Biens » et « Temps »,
- AI2 : l'attribut « Id » est ajouté à chaque classe-dimension « Biens » et « Temps »,
- AI3 : les attributs de la classe-dimension « Temps » sont « Mois », « Trimestre », « Semestre » et « Année ». Les attributs de la classe-dimension « Biens » ne sont pas définis,
- AP1 : les propriétés « criticité », « ouverture », « rafraîchissement », « historisation », « archivage », « consolidation », « calcul » sont annotées alors nous spécifions les opérations associées,
- AP2 : les opérations sont définies au niveau des classes-dimensions car il n'y a pas de spécificités pour un paramètre donné,
- AP3 : les concepts d'informativité « p », « h », « * », « a » sont appliqués aux attributs des classes-dimensions car les traitements associées y sont définis.

La formalisation du tableau croisé 4.11 est le diagramme décisionnel représenté dans la figure 4.12.

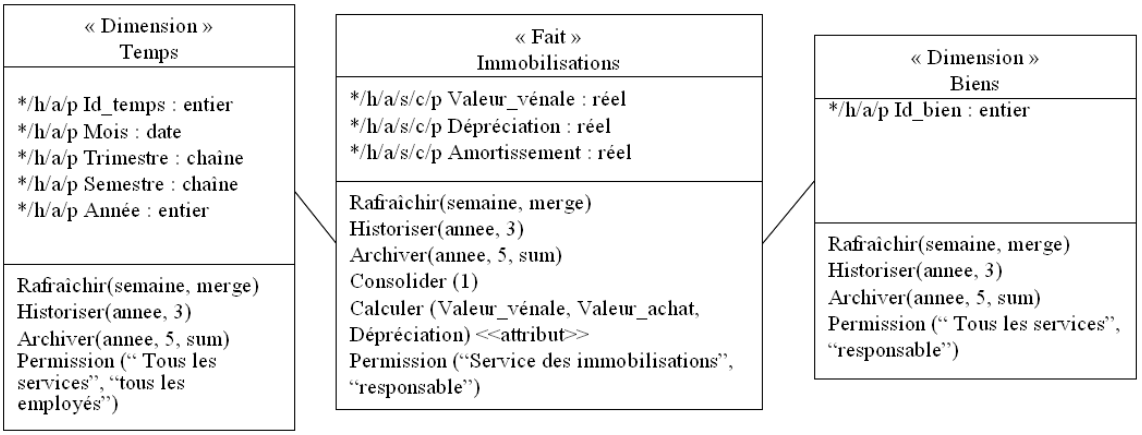


FIG. 4.12 – Diagramme décisionnel associé au tableau 4.11

Lors de la vérification des règles syntaxiques à la fin de la conception des diagrammes décisionnels, une incohérence est détectée suivant la règle SDP3 car l'opération « Rafraîchir() » est définie pour la dimension « Temps ». Pour garantir la cohérence du KDD, il convient de supprimer l'opération « Rafraîchir() » et le concept d'informativité « * » associé aux attributs de cette classe-dimension. La fusion n'a pas lieu d'y être dans notre exemple d'analyse des besoins stratégiques car il y a un seul diagramme décisionnel KDD.

4.5 Analyse des besoins systèmes

Le groupe système exprime des besoins liés aux sources de données internes ou externes de l'organisation. Leurs besoins principalement liés à la composante technique du SID abordent aussi des problèmes liés à la composante décision car ces acteurs doivent prendre en compte l'urbanisation du SID dans le temps (passé, présent et futur du SID).

4.5.1 Collecte

La collecte des besoins systèmes correspond à la récupération des schémas entité-association des sources. Dans le cas où les schémas ne sont pas disponibles, nous les collectons avec un outil commercial de reverse engineering. Afin de prendre en compte que les sources pertinentes pour le projet, nous considérons les sources des applications transactionnelles ou des progiciels liés aux domaines d'activités concernant le projet.

Dans notre exemple, nous disposons du schéma des sources 4.13.

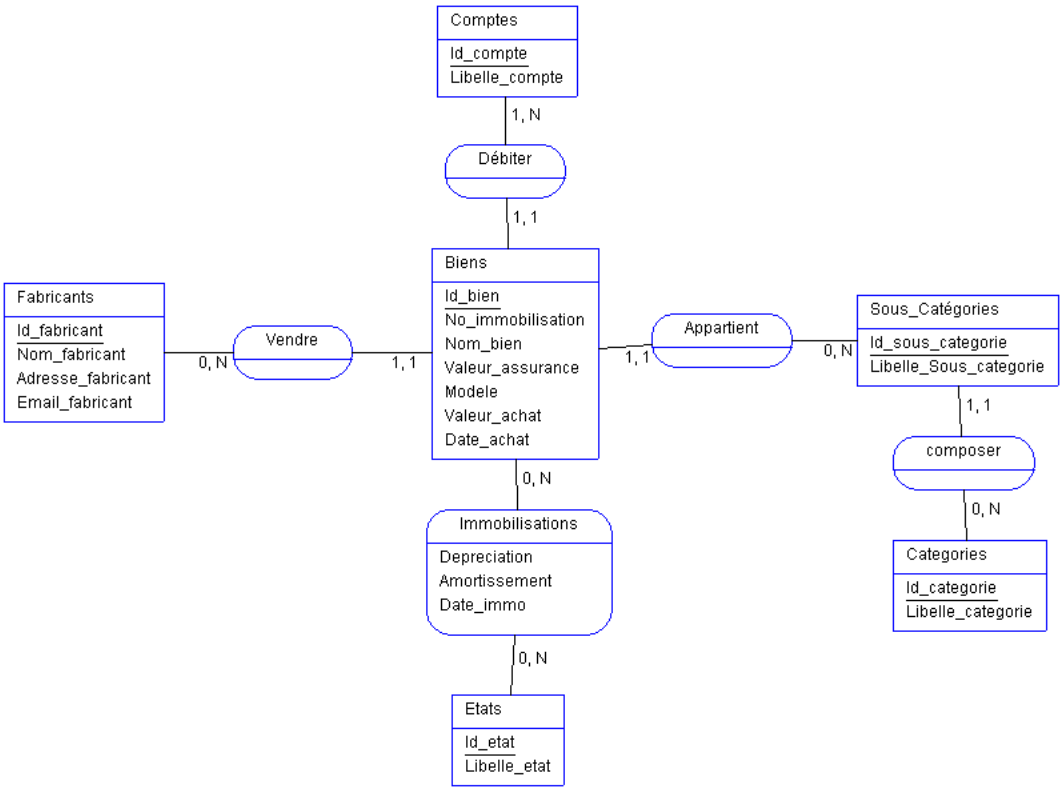


FIG. 4.13 – Modèle entité-association de l'application des immobilisations

Après l'étude des schémas des sources qui informe sur les données disponibles, nous procédons aux interviews suivant la démarche d'annotation du graphe de propriétés CSD. Dans notre exemple, nous obtenons le graphe de propriétés du groupe système présenté dans la figure 4.14.

4.5.2 Formalisation

La formalisation des besoins du groupe système consiste à dériver un diagramme décisionnel système (SDD) à partir d'un schéma entité-association. Nous proposons quatre types de règles pour cette dérivation : les règles de sélection des concepts multidimensionnels contenus dans les sources, les règles de transformation, des règles syntaxiques et des règles de fusion. Nous appliquons dans l'ordre les règles de sélection, les règles de transformation, les règles syntaxiques et enfin les règles de fusion.

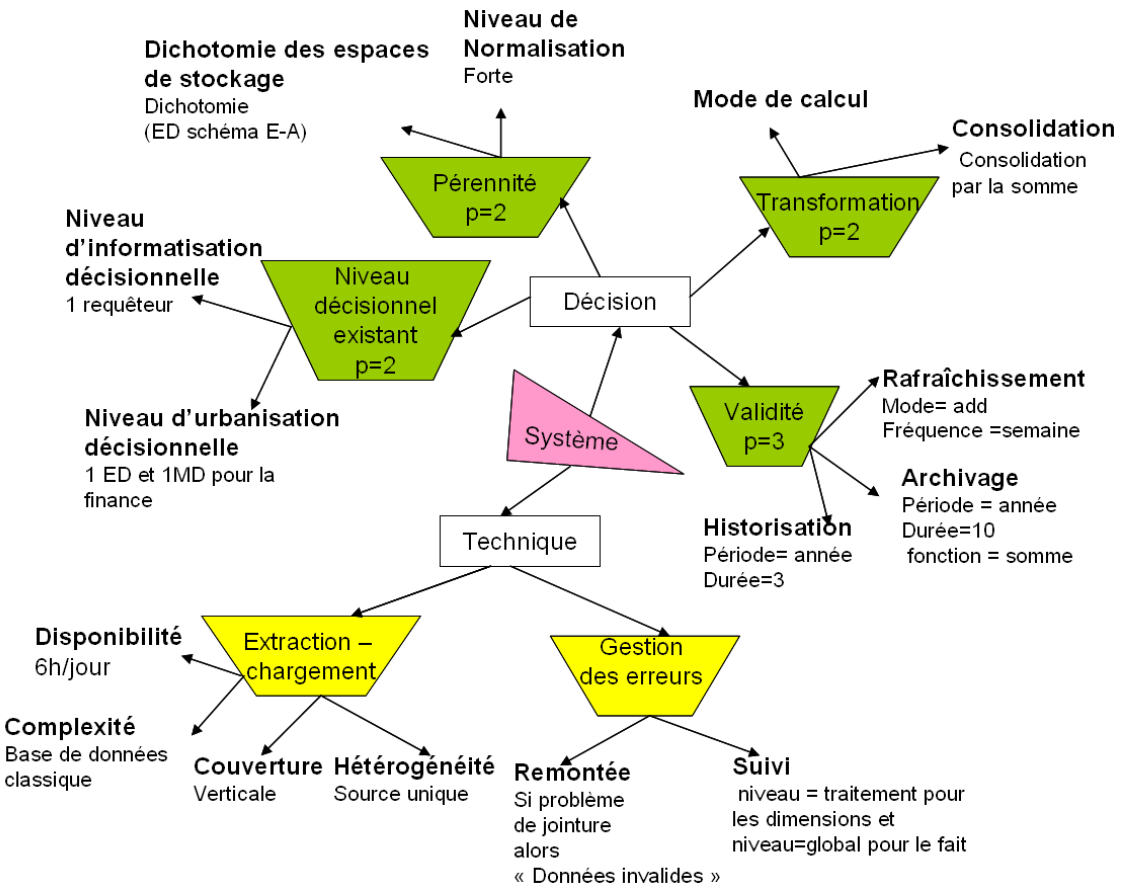


FIG. 4.14 – Graphe de propriétés CSD du groupe système du projet immobilisations

Règles de sélection des concepts multidimensionnels

Les règles de sélection permettent de déterminer dans un premier temps les faits candidats et dans un second temps les dimensions candidates. Enfin, les concepts multidimensionnels valides sont déterminés afin de formaliser les besoins *via* le diagramme décisionnel.

Les méthodes existantes définissent généralement le fait soit de manière qualitative, soit *via* l'intervention des décideurs [Golfarelli and Rizzi 1998a; Cabibbo and Torlone 1998; Husemann et al. 2000; Luján-Mora and Trujillo 2003]. [Kimball 1996; Phipps and Davis 2002] proposent d'évaluer les attributs numériques, mais ils ne tiennent pas compte du fait que ces attributs soient des clés primaires (PK). Afin de déterminer avec plus de précision les faits candidats, nous proposons de tenir compte non seulement du nombre d'attributs numériques mais aussi de la contrainte portée par l'attribut et de l'arité de l'entité ou de l'association. L'arité d'une entité ou une association est le nombre de liens définis entre cette entité ou cette association avec d'autres associations respectivement d'autres entités.

Nous proposons de définir les faits candidats CF_k à partir d'une matrice. Cette matrice évalue le nombre de liens des entités et des associations ainsi que le nombre d'attributs numériques qui ne sont pas clé primaire. Le nombre de lignes (Nb_att) de la matrice est égal au nombre maximum d'attributs numériques qui ne participent pas à une clé primaire et qui ne sont pas uniques (max(Nb_att)) d'une entité ou d'une association du schéma. Les entêtes des lignes prennent leurs valeurs dans l'intervalle [1 ;max(Nb_att)]. Le nombre de colonnes (Nb_liens) est égal au plus grand nombre de liens d'une entité ou d'une association du schéma. Les entêtes des colonnes prennent leurs valeurs dans l'intervalle [1 ;max(Nb_liens)]. La cellule(n, p) contient les noms des entités ou des associations qui ont n attributs numériques qui ne participent pas à une clé primaire, non-uniques et qui ont une arité égale à p.

Exemple : La matrice obtenue à partir du schéma conceptuel de notre source de données est présentée dans le tableau 4.12. « Biens » est un fait candidat car cette entité contient deux attributs numériques qui ne sont pas des clés primaires et qui ne sont pas uniques, soit « Valeur_achat » et « Valeur_assurance ». « Immobilisations » est un fait candidat car cette association est contient deux attributs numériques qui ne sont pas des clés primaires et qui ne sont pas uniques, soit « Amortissement » et « Dépréciation ».

		N_liens			
		4	3	2	1
N_att	4				
	3				
	2	Biens		Immobilisations	
	1				

TAB. 4.12 – Matrice des faits candidats

Soit CF_k le fait candidat caractérisé par la mesure candidate CM_{CF_k}. Il est connecté à la dimension candidate CD_{CF_k}. Soit f_k le fait valide caractérisé par la mesure m_{f_k}. Il est connecté à la dimension valide d_{f_k}.

Nous définissons la fonction $ArityR : CF \rightarrow CF$ qui retourne les entités ou les associations qui sont liées à une association respectivement à une entité fait candidat CF_k. Nous avons constaté que les faits valides sélectionnés dans les sources sont situés dans la partie supérieure du tableau mais, nous définissons les règles suivantes pour les déterminer formellement.

Règles de sélection des faits (SF) et des mesures (SM) :

- **SF1** : toute entité ou association contenue dans la matrice est un fait candidat CF_k,
- **SF2** : si $CF_i \in ArityR(CF_k)$ alors CF_k est un fait $f \in F$ et $CF_i \notin F$. Dans le cas où $CF_i \in ArityR(CF_k)$ et $CF_k \in ArityR(CF_i)$, les faits candidats qui sont des associations sont définis comme faits valides et les faits candidats qui sont des entités ne sont pas définis comme valides,

- **SF3** : si $\mathrm{CF}_k -> CF_i$ (dépendance fonctionnelle) alors CF_k est un fait $\in F$, $\mathrm{CF}_i \notin F$,
- **SM1** : tous les attributs numériques des CF_k qui ne participent pas à une clé primaire sont des mesures candidates CM_{CF_k},
- **SM2** : tous les attributs numériques des faits valides f_k qui ne participent pas à une clé primaire sont des mesures.

Règles de sélection des dimensions (SD) :

- **SD1** : toute entité en relation directe avec un fait candidat CF_k qui n'est pas un fait candidat ou un fait est une dimension candidate CD_{CF_k},
- **SD2** : toute entité $\in ArityR(f_k)$ qui n'est pas un fait est une dimension $\mathrm{d}_{f_k} \in D$,
- **SD3** : pour f_k, les attributs $\notin M_{f_k}$ qui ne sont pas de type date forment une nouvelle dimension $\mathrm{d}_{f_k} \in D$,
- **SD4** : toute entité en relation directe avec une dimension d_k qui n'est pas un fait f_k est une dimension valide $\mathrm{d}_{d_k} \in D$. Cette règle s'applique de manière récursive afin de définir les niveaux des hiérarchies,
- **SD5** : les attributs de type date de f_k sont extraits afin de créer la dimension Temps. Le fait est relié à la dimension n fois avec n=nombre de d'attributs de type date. Dans le cas où n>1, alors il faut ajouter un rôle à l'association,
- **SD6** : dans le cas où la dimension « Temps » n'a pas été créée auparavant, elle est rajoutée parmi les concepts multidimensionnels valides,
- **SD7** : les liens avec une notion de temps entre un fait f_k et une entité ou une association $\in ArityR(CF_k)$ sont transformés en plusieurs relations « En fonction » avec des rôles différents,
- **SD8** : vérifier la non-existence de dépendances fonctionnelles entre les attributs d'une classe-fait et ceux d'une classe-dimension. Si une dépendance fonctionnelle existe, alors les attributs de la classe-fait qui dépendent fonctionnellement d'un ou de plusieurs attributs de la classe-dimension sont transférés dans cette dernière,
- **SD9** : vérifier la non-existence de dépendances fonctionnelles entre les attributs des classes-dimensions. Si une dépendance fonctionnelle existe, elle est représentée par un trait discontinu entre les classes-dimensions.

Règles de sélection des paramètres (SP) et des attributs faibles (SW) :

- **SP1** : tout attribut numérique d'une dimension d_{f_k} est transformé en un paramètre $\mathrm{p}_k \in P$. Les attributs qui participent à la clé primaire sont traités suivant cette règle,
- **SP2** : les dimensions d_{f_k} ne peuvent pas contenir des attributs qui participent à la définition des attributs des faits f_k,
- **SW1** : tout attribut non-numérique dune dimension d_{f_k} est transformé en attribut faible $\mathrm{a}_k \in A$.

Règles de structuration liées au groupe système

Les règles de structuration permettent de représenter *via* le diagramme décisionnel les concepts multidimensionnels définis à partir des règles de sélection.

Règles de structuration des données :

- **SRD1** : tout fait f_k est transformé en une classe-fait avec des attributs correspondant à ses mesures m_{CF_k},
- **SRD2** : toute dimension d_{f_k} est transformée en une classe-dimension avec des attributs correspondant à ses paramètres p_k et ses attributs faibles a_k,
- **SRD3** : tout classe-fait f_k est associée à chacune de ses classes-dimensions d_{f_k} par une association représentée un trait continu qui signifie « f_k analysé EN FONCTION de d_{f_k} ». Ce lien est précisé avec les rôles définis à partir des règles de sélection,
- **SRD4** : toute classe-dimension est reliée à ses classes-dimensions par des flèches discontinues représentant un lien hiérarchique,
- **SRD5** : toute règle de calcul définie dans le graphe de propriétés *via* la propriété « Calcul » de la catégorie « Transformation » est spécifiée par une opération attribut dans la classe-fait ou la classe-dimension liée.

Règles de structuration des processus :

- **SRP1** : les opérations **Disponible(d, f, c)**, **Harmoniser(c, h)**, **Trace(l)**, **Exception(e,m)**, **Calculer**($\{v_i\}^+$), **Historiser(p, d, cond)**, **Rafraîchir(f, m)**, **Archiver(p, d, fct, cond)** sont définies en fonction des annotations du graphe de propriétés du groupe système,
- **SRP2** : une opération d'attribut est associée à tout paramètre qui possède des arguments différents pour un traitement donné,
- **SRP3** : les concepts d'informativité sont associés aux attributs des classes multidimensionnelles en fonction des traitements déclarés.

Règles syntaxiques liées au groupe système

Les règles syntaxiques permettent de contrôler la consistance des SDD dérivés à partir des sources.

- **WR1** : une classe-fait peut être reliée à une autre classe-fait.
- **WR2** : une classe-dimension peut être reliée à une autre classe classe-dimension si et seulement s'il existe une dépendance fonctionnelle entre les deux entités.
- **WR3** : une classe-dimension peut être reliée à une classe-fait par un trait continu et à une autre classe-dimension par un trait discontinu.

Les règles de fusion des SDD sont identiques à celles des diagrammes décisionnels des groupes tactique et stratégique.

Exemple : l'application des règles de sélection, de structuration et syntaxiques pour notre source de données est la suivante :

- **SF1** : l'association « Immobilisations » et l'entité « Biens » sont des faits candidats CF_k,
- **SF2** : « Biens » $\in ArityR(Immobilisations)$ et Immobilisations est une association alors « Immobilisations » est un fait et « Biens » n'est pas un,
- **SF3** : Il n'existe pas de dépendance fonctionnelle entre le fait « Immobilisations » et « Biens »,
- **SM1** : les attributs numériques, soit « Valeur_assurance », « valeur_achat » sont des mesures candidates du fait candidat Biens,
- **SM2** : les attributs numériques, soit « Valeur_assurance », « Valeur_achat », « Dépréciation », « Amortissement » ne participent pas à la clé primaire de « Immobilisations », ils sont donc des mesures candidates CM_{CF_k} sont des mesures.

- **SD1** : les entités « Fabricants », « Comptes », « Sous_catégories » sont en relation direct avec « Biens » alors elles sont des dimensions CD_{f_k},
- **SD2** : les entités « Biens » et « Etats » sont en relation directe avec « Immobilisations » et elles ne sont pas des faits, alors « Biens » et « Etats » sont des deux dimension D_{f_k},
- **SD3** : pour le fait « Immobilisations », il n'y a pas d'attribut autre que les mesures et Date_immo de type date,
- **SD4** : l'entité « Catégories » n'est pas un fait et elle est en relation direct avec la dimension « Sous_catégorie », elle est donc une dimension valide. Elles forment la hiérarchie « h_catégorie »,
- **SD5** : la dimension « Temps » est crée à partir de l'attribut « Date_immo »,
- **SD6** : la dimension « Temps » existe déjà,
- **SD7** : il n'y pas d'autre notion de date entre l'association « Immobilisations » et d'autres entités,
- **SD8** : il y n'a pas de dépendance fonctionnelle entre le fait et les dimensions « Biens » et « Etats »,
- **SD9** : il y a des dépendances fonctionnelles entre la dimension « Biens » et la dimension « Sous_catégories », le lien entre ces dimensions est donc représenté par un trait discontinu. Une dépendance fonctionnelle existe aussi entre la dimension « Biens » et la dimension « Fabricants », le lien entre ces dimensions est donc représenté par un trait discontinu.

- **SP1** : les attributs numériques des dimensions « Biens », « Sous_catégories », « Catégories », « Comptes », « Etats » et « Fabricants » sont transformés en paramètres de ces derniers,
- **SP2** : les dimensions ne contiennent pas d'attributs participant à la définition des attributs du fait,

- **SW1** : les attributs non-numériques des dimensions « Biens », « Sous_catégories », « Catégories », « Comptes », « Etats » et « Fabricants » sont transformés en attributs faibles de ces derniers.

Les concepts multidimensionnels de notre source de données sont structurés comme ci-dessous :

- **SRD1** : le fait « Immobilisations » est transformé en une classe-fait avec les attributs « Dépréciation », « Amortissement » correspondant à ses mesures,
- **SRD2** : les dimensions « Biens », « Sous_catégories », « Catégories », « Comptes », « Etats », « Fabricants » et « Temps » sont transformées en des classes-dimensions avec des attributs correspondant à leurs paramètres p_k et leurs attributs faibles a_k,
- **SRD3** : la classe-fait « Immobilisations » est associée à ses classes-dimensions « Biens », « Etats », et « Temps » par des traits continus,
- **SRD4** : la classe-dimension « Biens » est reliée à ses classes-dimensions « Sous_catégories », « Catégories », « Comptes » et « Fabricants » par des flèches discontinues représentant un lien hiérarchique,
- **SRD5** : la propriété « Calcul » de la catégorie « Transformation » du graphe de propriété ne spécifie aucune règle de calcul.

- **SRP1** : les opérations sont définies en fonction des annotations du graphe de propriétés du groupe système comme suit :
  - **SRP1** : **Disponible(6, jour)** pour la classe-fait « Immobilisations » et **Disponible(6, semaine)** pour les classes-dimensions suivant la propriété « Disponibilité » de la catégorie « Extraction-chargement »,
  - **SRP2** : **Harmoniser(1, 1)** pour toutes les classes multidimensionnelles car la source est unique et elle est une base classique et l'extraction des données qui requiert des transformations simples comme indiqué dans les propriétés « Complexité » et « Hétérogénéité » de la catégorie « Extraction-chargement »,
  - **SRP3** : **Trace(2)** pour toutes les classes multidimensionnelles car les acteurs souhaitent suivre les traitements suivant la propriété « Suivi » de la catégorie « gestion des erreurs »,
  - **SRP4** : **Exception("Problème de jointure", "Données invalides")** pour toutes les classes multidimensionnelles car les acteurs souhaitent contrôler la cohérence des données suivant la propriété « Remontée » de la catégorie « gestion des erreurs »,
  - **SRP6** : il n'y a pas d'opération « Calculer() » car la propriété « Calcul » de la catégorie « Transformation » n'est pas annotée,
  - **SRP7** : **Historiser(année, 3)** pour toutes les classes multidimensionnelles car les décideurs souhaitent revenir sur des données détaillées datant au plus de trois ans selon la propriété « Historisation » de la catégorie « Validité »,
  - **SRP8** : **Rafraîchir(semaine, merge)** suivant la propriété « Rafraîchissement » de la catégorie « Validité »,
  - **SRP9** : **Archiver(année, 10, sum)** suivant la propriété « Archivage » de la catégorie « Validité ».

– **SRP2** : il n'y a pas d'opération d'attribut car tous les paramètres sont définis au niveau de la classe.
– **SRP3** : les concepts d'informativité « * », « h », « a », « d », « l », « t », « e », « p » sont associés aux attributs des classes multidimensionnelles en fonction des traitements déclarés,

L'application des règles permet de définir le diagramme décisionnel des besoins du groupe d'acteurs présenté dans la figure 4.15.

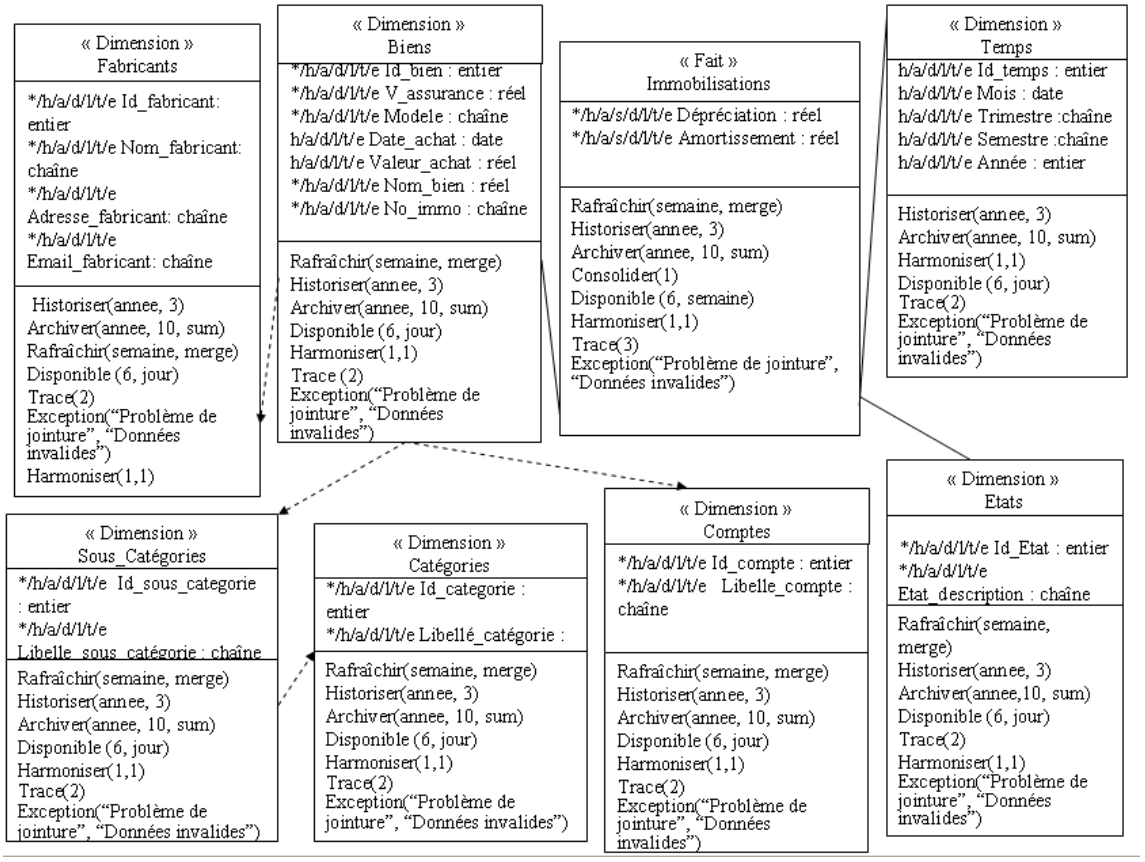


FIG. 4.15 – Diagramme décisionnel du groupe système du projet immobilisations

Le contrôle de la consistance de notre diagramme décisionnel SDD suivant les règles syntaxiques, il n'y pas de modifications à faire sur le diagramme. De plus, il y n'a qu'une source de données donc l'application des règles de fusion n'est pas requise.

4.6 Confrontation des besoins du SID

La tâche de confrontation est la dernière tâche de la phase d'analyse. En entrée de cette tâche, il y a un graphe de propriétés et un diagramme décisionnel par groupe d'acteurs. Tous les acteurs du SID y participent jusqu'au rapprochement de leurs besoins. Comme nous distinguons les décideurs, nous définissons deux confrontations afin de prendre en compte les priorités propres aux besoins utilisateurs et les priorités entre les besoins des utilisateurs et les sources de données. Les deux confrontations sont appelées "Evaluer la connaissance utilisateur" et "Evaluer la connaissance de l'environnement" comme indiqué dans la figure 4.3.

- "Evaluer la connaissance utilisateur" est la confrontation entre le diagramme décisionnel du groupe tactique (TDD) et celui du groupe stratégique (KDD). Nous réalisons cette tâche afin de synchroniser les intérêts des domaines d'activités et les orientations stratégiques de l'organisation avant la confrontation avec les sources. Nous obtenons le diagramme décisionnel utilisateur (UDD),
- "Evaluer la connaissance de l'environnement" est la confrontation entre le diagramme décisionnel utilisateur (UDD) et celui du groupe système (SDD). Elle correspond à la tâche de confrontation évoquée par les méthodes mixtes.

Pour la confrontation des besoins utilisateurs, nous considérons le KDD comme le noyau car il importe de fédérer les besoins des décideurs autour des grandes orientations de l'organisation. Ainsi, toutes les classes multidimensionnelles du KDD sont conservées ; seuls des ajouts et des modifications sont réalisables. Ces modifications sont faites en fonction des priorités définies par catégories de propriétés dans le graphe de propriétés.

Par ailleurs, les besoins du groupe tactique impactent les performances de l'organisation. Ils doivent prévaloir comme le confirme l'étude de l'Etat des lieux des projets décisionnels en 2005/2006 faite par IDG (cf. note de bas de page 1. Autrement dit, les catégories de propriétés du graphe de ce groupe ayant un poids important doivent être traitées et satisfaites en priorité pour assurer une bonne gestion de la performance de l'organisation.

Ainsi, pour la confrontation des besoins du groupe tactique et du groupe stratégique, nous parcourons les graphes de propriétés annotées suivant une recherche en profondeur. La confrontation consiste à comparer les propriétés suivant les règles définies ci-après afin d'obtenir le diagramme décisionnel utilisateur. Nous commençons par le sous-graphe « Décision » puis nous finissons par le sous-graphe « Technique ». Nous appliquons les règles UCONF suivantes :

- **UCONF1** : appliquer les règles de fusion pour les classes multidimensionnelles de même nom,
- **UCONF2** : pour la confrontation des attributs de même nom appartenant à deux classes multidimensionnelles différentes,

  - **UCONF2A** : si les attributs sont partagés par des classes-faits distinctes alors il convient de préciser fonctionnellement l'environnement afin de déterminer si ces données représentent un ou plusieurs concepts,
  - **UCONF2B** : si les attributs sont partagés par des classes-dimensions distinctes alors il faut vérifier les dépendances fonctionnelles entre ces classes,
- **UCONF3** : pour la confrontation des opérations de même nom, ayant des paramètres différents, appartenant à deux classes multidimensionnelles de même nom, nous évaluons les propriétés associées à ces traitements :
  - **UCONF3A** : si les contraintes de valeurs des propriétés sont proches alors nous considérons la moins restrictive,
  - **UCONF3B** : si les contraintes de poids sont égales et les contraintes de valeurs sont contraires alors nous considérons les contraintes de valeurs du graphe de propriétés du groupe tactique car ces acteurs maîtrisent le domaine d'activité,
  - **UCONF3C** : si les contraintes de poids ne sont pas égales, les contraintes de valeurs sont contraires et la contrainte de poids la plus importante est celle du groupe stratégique alors nous considérons la contrainte de valeurs définie par ce groupe,
  - **UCONF3D** : si les contraintes de poids ne sont pas égales, les contraintes de valeurs sont contraires et la contrainte de poids la plus importante est celle du groupe tactique alors nous considérons que l'itération de l'analyse en cours est clôturée et qu'une nouvelle itération est définie.

Exemple : l'application des règles de la confrontation entre le diagramme décisionnel du groupe tactique (TDD) et celui du groupe stratégique (KDD) est présentée ci-dessous. Nous obtenons le diagramme décisionnel utilisateur (UDD) présenté dans la figure 4.16.

- **UCONF1** : application des règles de fusion pour les classes multidimensionnelles de même nom,
  - FUS : regroupement du diagramme décisionnel tactique et du diagramme décisionnel stratégique car ils ont en commun la classe-fait « Immobilisations » et les classes-dimensions « Biens » et « Temps »,
    - **FUS1** : fusion respective des classes-dimensions partagées « Biens », « Temps ». La classe-dimension « Biens » possède 4 attributs et 7 opérations par ajout des données et des opérations. La classe-dimension « Temps » possède 5 attributs et 6 opérations par ajout des données et des opérations. Cependant, il y a un conflit entre les paramètres de l'opération « Archiver » des classes-dimensions fusionnées,
    - **FUS2** : fusion des classes-faits « Immobilisations ». La classe-fait « Immobilisations » obtenue possède 3 attributs et 9 opérations par ajout des données et des opérations. Cependant, il y a un conflit entre les paramètres des opérations « Archiver » et « Consolider »,
    - **FUS3** : ajout des classes-dimensions « Catégories », « Etats » et « Fabricants » propres au diagramme décisionnel tactique ;
  - FDS : les diagrammes décisionnels n'ont pas de classes-faits différents,
  - FDR : les classes-dimensions n'ont pas des attributs communs,

- **UCONF2** : il n'y a pas d'attribut de même nom appartenant à deux classes multidimensionnelles différentes,
- **UCONF3** : les opérations « Archiver » et « Consolider » ont des paramètres différents pour des classes multidimensionnelles de même nom :
  - **UCONF3A** : pour le conflit concernant l'opération « Archiver », nous évaluons la propriété archivage des graphes de propriétés des groupes tactique et stratégique. Ce conflit est valable pour toutes les classes multidimensionnelles du diagramme décisionnel utilisateur, nous évaluons donc une seule fois ce conflit. Les contraintes de valeurs de la propriété archivage sont proches car seules les durées d'archivage sont différentes. Nous considérons la moins restrictive, soit la durée de 9 ans. L'opération « Archiver » du diagramme décisionnel est donc « Archiver(annee, 9, sum) ». Pour le conflit concernant l'opération « Consolider », nous évaluons la propriété consolidation des graphes de propriétés des groupes tactique et stratégique. Ce conflit est valable que pour la classe-fait « Immobilisations ». Les contraintes de valeurs de la propriété consolidation sont identiques, l'opération « Consolider » du diagramme décisionnel stratégique est définie pour les trois attributs dont l'attribut Valeur_vénale (attribut pour lequel est définie l'opération attribut du diagramme décisionnel tactique). Nous considérons la moins restrictive, soit la consolidation de tous les attributs de la classe-fait. L'opération « Consolider » de la classe-fait du diagramme décisionnel utilisateur est donc « Consolider(1) ».

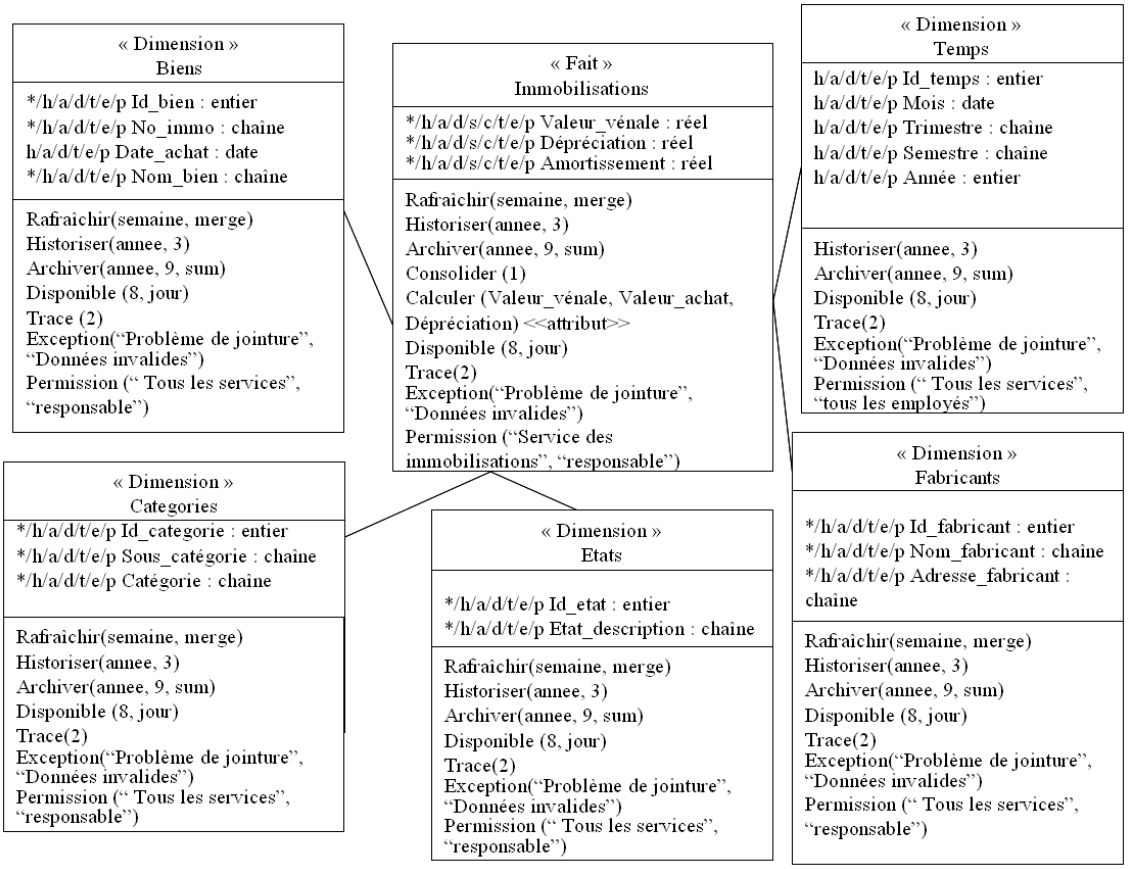


FIG. 4.16 – Diagramme décisionnel utilisateur du projet immobilisations

Pour la confrontation des besoins utilisateurs et des besoins systèmes, nous devons tenir compte de la disponibilité des données dans les sources. La confrontation consiste à comparer les propriétés suivant les règles définies ci-après afin d'obtenir le diagramme décisionnel du SID. Ces propriétés sont celles du graphe de propriétés du groupe système et du graphe de propriétés du groupe tactique ou stratégique suivant la valeur qui a été retenue pour la définition des diagrammes décisionnels. Dans le cas où les sources de données sont insuffisantes par rapport aux besoins des utilisateurs, la mise à disposition de ces données relève de la budgétisation du projet. Cette problématique étant spécifique à chaque projet, elle n'est pas traitée dans cette thèse. Nous utilisons les règles suivantes pour cette confrontation :

- **SCONF1** : les classes multidimensionnelles définies dans le diagramme utilisateur (UDD) qui ne sont pas définies dans le diagramme décisionnel système (SDD) doivent être évaluées et budgétisées,
- **SCONF2** : les classes multidimensionnelles définies dans le diagramme système (SDD) qui ne sont pas définies dans le diagramme décisionnel utilisateur (UDD) sont supprimées dans le cas où ces données sont des données supplémentaires non nécessaires à la cohérence des données sinon, elles sont conservées,
- **SCONF3** : appliquer les règles de fusion pour les classes multidimensionnelles de même nom,

- **SCONF4** : pour la confrontation des attributs de même nom appartenant à deux classes multidimensionnelles différentes,
  - **SCONF4A** : si les attributs sont partagés par des classes-faits distinctes alors il convient de préciser fonctionnellement l'environnement afin de déterminer si ces données représentent un ou plusieurs concepts,
  - **SCONF4B** : si les attributs sont partagés par des classes-dimensions distinctes alors il faut vérifier les dépendances fonctionnelles entre ces classes,
- **SCONF5** : pour la confrontation des opérations de même nom, ayant des paramètres différents, appartenant à deux classes multidimensionnelles de même nom, nous évaluons les propriétés associées à ces traitements :
  - **SCONF5A** : si la contrainte de valeurs retenue pour la définition du diagramme décisionnel utilisateur est proche de celle du groupe système sont proches alors nous considérons celle retenue pour la définition du diagramme décisionnel utilisateur,
  - **UCONF5B** : si la contrainte de poids retenue pour la définition du diagramme décisionnel utilisateur est égale à celle du groupe système et si la contrainte de valeurs retenue pour la définition du diagramme décisionnel utilisateur est contraire à celle du groupe système alors nous considérons la contrainte de valeurs retenue pour la définition du diagramme décisionnel utilisateur,
  - **SCONF5C** : si la contrainte de poids retenue pour la définition du diagramme décisionnel utilisateur n'est pas égale à celle du groupe système et si la contrainte de valeurs retenue pour la définition du diagramme décisionnel utilisateur est contraire à celle du groupe système alors nous considérons la contrainte de valeurs associée à la plus grande contrainte de poids.

Exemple : l'application des règles de la confrontation entre le diagramme décisionnel utilisateur (UDD) et celui du groupe système (SDD) est présentée ci-dessous. Nous obtenons le diagramme décisionnel du SID présenté dans la figure 4.17.

- **SCONF1** : toutes les classes multidimensionnelles définies dans le diagramme utilisateur (UDD) sont définies dans le diagramme décisionnel système (SDD). Le projet ne requiert donc pas une nouvelle budgétisation,
- **SCONF2** : les classes multidimensionnelles « Sous_catégories » et « Comptes » du diagramme système (SDD) ne sont pas définies dans le diagramme décisionnel utilisateur (UDD). La classe-dimension « Comptes » est supprimée car la notion de compte n'a pas été exprimée par les décideurs. La classe-dimension « Sous_catégories » est conservée car les données sont nécessaires à la cohérence des données entre les classes-dimensions « Biens » et « Fabricants ». Le diagramme décisionnel du SID contient donc la classe-fait « Immobilisations » et les classes-dimensions suivantes : « Biens », « Fabricants », « Sous_catégories », « Catégories », « Etats » et « Temps »,
- **SCONF3** : application des règles de fusion pour les classes multidimensionnelles « Immobilisations », « Temps », « Biens », « Fabricants », « Etats » et « Catégories »,
  - FUS : regroupement du diagramme décisionnel utilisateur et du diagramme décisionnel système car ils ont en commun la classe-fait « Immobilisations »

et les classes-dimensions « Temps », « Biens », « Fabricants », « Etats » et « Catégories »,
  - **FUS1** : nous précisons les conflits résultant des fusions. Pour la classe-dimension « Temps », l'ajout des attributs et des opérations ne génère pas des conflits des données mais il génère des conflits pour les opérations « Archiver » et « Disponible ». Ces conflits sont valables pour toutes les classes multidimensionnelles. Pour la classe-dimension « Fabricant », l'attribut Email_fabricant n'est pas conservé car il n'est pas exprimé parmi les besoins des utilisateurs. Pour la classe-dimension « Catégories »,
  - **FUS2** : la fusion des classes-faits « Immobilisations » par ajout des attributs et des opérations ne génère pas de conflit de données mais elle génère des conflits concernant les opérations « Archiver » et « Trace »,
  - **FUS3** : seule la classe-dimension « Sous_catégories » est ajoutée car la classe-dimension « Compte » n'est pas un besoin des décideurs et n'est pas nécessaire pour la cohérence des données suivant l'application de la règle SCONF2,
- FDS : les diagrammes décisionnels n'ont pas de classes-faits différentes,
- FDR : les classes-dimensions « Sous_catégories » et « Catégories » partagent l'attribut « Libelle_sous_catégorie ». Dans le diagramme utilisateur, il y a l'attribut nommé « Sous_catégorie » au lieu de « Libelle_sous_catégorie » car généralement le concept est substitué à son libellé ou sa description.
  - **FDR1** : l'attribut « Libelle_sous_catégorie » dépendant directement de l'attribut « Id_sous_catégorie ». L'attribut « Libelle_sous_catégorie » est donc supprimé de la classe-dimension « Catégories ». Il y a une relation de dépendance fonctionnelle entre la classe-dimension « Sous_catégories » et « Catégories » qui a pour cible la classe-dimension « Catégories »,
  - **FDR2** : il n'y a pas de classes-dimensions de noms différents qui ont presque tous les attributs en commun à l'identifiant près,
  - **FRD3** : il n'y a pas eu fusion de classes multidimensionnelles,
  - **FRD4** : il n'y a pas de définition de rôles car il n' y a pas eu fusion.

- FRC : fusion des liens entre les classes multidimensionnelles,
  - **FRC1** : la classe-fait « Immobilisations » est liée aux classes-dimensions « Temps », « Fabricants », « Etats », « Biens », « Catégories ». Les classes-dimensions « Fabricants », « Sous_catégories » sont cibles de relations de dépendance fonctionnelle avec la classe-dimension « Biens » et ,
  - **FRC2** : la classe-dimension « Catégories » est cible d'une relation de dépendance fonctionnelle et elle est liée à la classe-fait « Immobilisations ». Le lien d'association est supprimé et la relation de dépendance fonctionnelle est conservée,
- **SCONF4** : la confrontation des attributs de même nom appartement à deux classes multidimensionnelles différentes,
  - **SCONF4A** : il y a une seule classe-fait. Il n'y donc pas de partage d'attributs,
  - **SCONF4B** : il n'y plus d'attributs communs entre deux classes-dimensions,

- **SCONF5** : la confrontation des opérations de même nom, ayant des paramètres différents, appartement à deux classes multidimensionnelles de même nom :
  - **SCONF5A** : pour le conflit concernant la méthode « Archiver », nous évaluons la propriété « Archivage » du graphe de propriétés du groupe système et celle du graphe de propriétés du groupe tactique car elle a été retenue pour la définition du diagramme décisionnel utilisateur. Les contraintes de valeurs de la propriété « Archivage » sont proches car seule les durées d'archivage sont différentes. Ce conflit est valable pour toutes les classes multidimensionnelles du diagramme, nous évaluons donc une seule fois ce conflit. Nous considérons celle retenue pour la définition du diagramme décisionnel utilisateur, soit la durée de 9 ans,
  - **UCONF5B** : pour le conflit concernant « Trace » de la classe-fait « Immobilisations », nous évaluons la propriété « Suivi » du graphe de propriétés des groupes tactique et système. Les contraintes de poids associées à ces propriétés sont égales et les contraintes de valeurs sont contraires alors nous considérons la contrainte de valeurs retenue pour la définition du diagramme décisionnel utilisateur, soit le niveau de trace=2,
  - **SCONF5C** : pour le conflit concernant « Disponible », nous évaluons la propriété « Réactivité » du graphe de propriétés du groupe tactique et la propriété « Disponibilité » du graphe de propriétés du groupe système. Les systèmes sources sont disponibles 6 heures par jour et les décideurs ont exprimé le besoin que le SID soit indisponible pendant 8 heures par jour. Nous considérons donc la contrainte de valeurs associée à la plus grande contrainte de poids. La contrainte de poids associée à la propriété « Réactivité » définie dans le graphe de propriétés tactique est égale à 2 et la contrainte de poids de la propriété « Disponibilité » est égale à 1. Nous considérons donc la contrainte de valeurs de 8 heures par jour pour le diagramme décisionnel du SID.

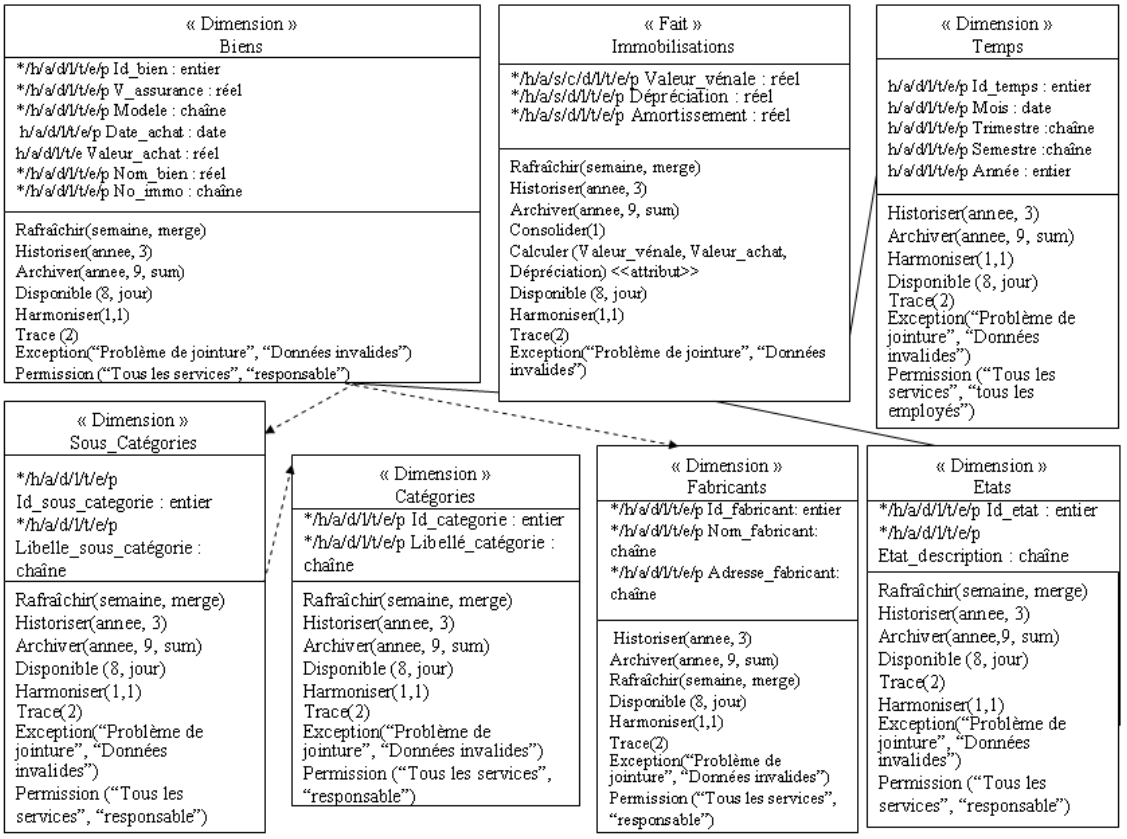


FIG. 4.17 – Diagramme décisionnel du SID du projet immobilisations

4.7 Conclusion

En entrée de notre processus d'analyse, nous préconisons de caractériser les groupes d'acteurs et les métiers auxquels ils sont liés. La documentation suivante doit être disponible :

- le groupe tactique : le document d'expression des besoins tactiques comprenant les tableaux croisés de leurs domaines d'activités,
- le groupe stratégique : le document d'expression des besoins stratégiques comprenant les tableaux croisés synthétiques,
- le groupe système : le document d'expression des besoins systèmes comprenant les schémas entité-association des sources.

Nous guidons les trois tâches de l'analyse des besoins du SID (la collecte, la formalisation et la confrontation des besoins) de manière semi-automatique ou automatique. Nous utilisons les graphes de propriétés, les tableaux croisés, les diagrammes décisionnels et des ensembles de règles (règles de transformation, règles de sélection, règles de structuration, règles syntaxiques et règles de fusion).

L'expression structurée sous forme de tableau croisé des besoins utilisateurs contribue à la fiabilité du SID développé car le biais dû à l'interprétation technique est faible. De plus, l'annotation du graphe de propriétés CSD permet d'évaluer la dynamique du SID relative à la composante technique et à la composante décision.

Nous parallélisons l'analyse des groupes d'acteurs avec des points de synchronisation associés aux deux confrontations des besoins des acteurs afin de modéliser un SID fiable par rapport aux besoins utilisateurs et aux sources. Nous considérons deux confrontations car nous distinguons les décideurs et nous définissons des priorités parmi leurs besoins. Cette tâche de confrontation est accessible à tous les acteurs car les besoins sont représentés *via* le même modèle.

Notre proposition se distingue des méthodes d'analyse existantes car :
- elle représente les besoins des trois groupes d'acteurs *via* le même modèle multidimensionnel orienté réutilisation au cours de processus parallèles,
- elle préconise l'expression structurée des besoins utilisateurs,
- elle guide les concepteurs décisionnels *via* des graphes orientés lors de l'interview des acteurs pour la collecte de l'aspect dynamique du SID,
- elle définit des priorités parmi les besoins du SID *via* l'annotation des graphes,
- elle confronte deux diagrammes décisionnels indépendamment de la taille du projet.

Par ailleurs, le graphe de propriétés que nous proposons par groupe d'acteurs permet de définir les traitements du SID. Cependant, toutes les propriétés ne sont pas utilisées pour la formalisation des besoins. Elles sont aussi utilisées pour la phase de conception et la phase d'implantation. Plus précisément, elles serviront pour le choix de l'architecture (cf. chapitre 5) et le choix des outils lors de l'implantation. A partir de cette modélisation des besoins confrontés, nous passons à la phase de conception du SID.

Ainsi, les besoins liés à l'aspect statique, exprimés sous forme de tableaux croisés synthétiques ou simples, de schémas entité-association et ceux liés à l'aspect dynamique, exprimés sous forme de graphes de propriétés, sont formalisés en des diagrammes proche de la vision des données par les utilisateurs. Cette formalisation est assurée par des ensembles de règles de transformation des besoins. Ces règles assurent la traçabilité de notre processus d'analyse des besoins des SID car elles permettent d'associer les données et les opérations de chaque besoin des acteurs. Ces données et ces opérations associées aux besoins sont définies dans les classes multidimensionnelles d'analyse au terme de l'analyse. Afin d'assurer la traçabilité de la prise en compte des besoins des acteurs, dans le chapitre suivant dédié à la conception des SID, nous explicitons les données et les opérations de notre modèle conceptuel associés à ceux de notre modèle de besoins.

Chapitre 5

Méthode de conception des SID

Table des matières

5.1 Introduction 130
5.1.1 Problématique 130
5.1.2 Proposition 131
5.2 Choix de l'architecture décisionnelle 132
5.2.1 Typologie des architectures décisionnelles valides 134
5.2.2 Critères de quantification 135
5.2.3 Automatisation 136
5.3 Modèle multidimensionnel de données et de traitements (MMDT) 142
5.3.1 Structure complexe des dimensions et des paramètres : . 143
Concepts de dimension et de niveau 145
Exemple 146
Positionnement par rapport à CWM 148
5.3.2 Structure complexe des faits et des mesures : 149
Concepts de fait et de mesure 150
Exemple 151
Positionnement par rapport à CWM 154
5.3.3 Cohérence de l'interrogation : 155
5.4 Impacts des traitements 157
5.4.1 Impacts des traitements de dérivation des données 158
Impacts du traitement « Criticité des données » 158
Impacts des traitements « Suivi des erreurs » et « Remontée des erreurs » 158
Impacts des traitements « Harmonisation » 158
5.4.2 Impacts des traitements de préparation des données . . . 159
Impacts du traitement « Rafraîchissement » 159
Impacts des traitements « Historisation » et « Archivage » 160
Impacts des traitements « Consolidation » et « Calcul » . 161
Impacts du traitement « Disponibilité » 162

5.4.3 Les traitements par concept multidimensionnel 162
5.5 Conception des modules décisionnels 163
5.5.1 Passage du diagramme décisionnel à notre modèle MMDT 164
Règles PMMDT des classes-faits d'analyse 164
Règles PMMDT des classes-dimensions d'analyse 165
Les autres spécificités multidimensionnelles 166
5.5.2 Passage du diagramme décisionnel au modèle E-A 170
Règles PMEA des classes-faits d'analyse 170
Règles PMEA des classes-dimensions d'analyse 171
5.6 Conclusion 173

Notre démarche de développement du Trident décisionnel définit la conception du SID à partir des produits de l'analyse, soit le diagramme décisionnel du SID et les trois graphes de propriétés (un par groupe d'acteurs).

Suivant [Srivastava and Chen 1999], la phase de conception est composée des tâches suivantes « la sélection de l'architecture », « la création du schéma ». Par ailleurs, les schémas de correspondances définissent la construction et l'alimentation du SID [Vassiliadis et al. 2002a]. Ainsi, de par les projets décisionnels menés dans le cadre industriel, nous pensons que la conception du SID doit inclure les schémas de correspondances entre le schéma du SID et le schéma des sources. Des tâches liées à la définition des schémas sont aussi définies dans le processus de développement proposé par [Luján-Mora 2005]. De ce fait, nous proposons une démarche de conception composée de quatre tâches dont les deux premières correspondent à celles définies par [Srivastava and Chen 1999] et les deux autres sont liées aux schémas de correspondances. Ces tâches sont : « Choisir l'architecture », « Concevoir les modules du SID », « Définir les schémas de correspondances » et « Evaluer les schémas de correspondances » (cf. figure 5.1).

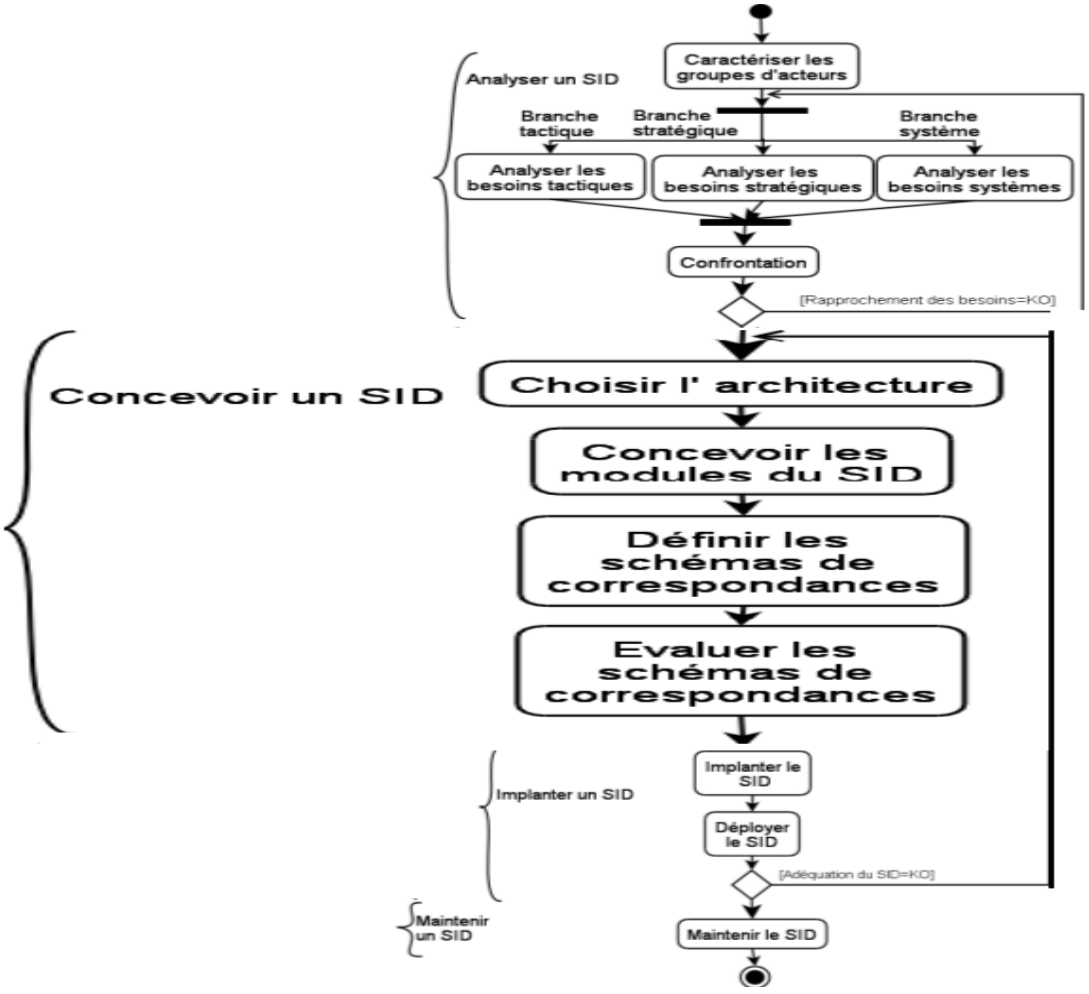


FIG. 5.1 – Zoom sur la phase de conception de notre Trident décisionnel

Dans la section 5.1, nous introduisons la problématique et nos propositions liées à cette phase. Puis, dans la section 5.2, nous décrivons la première tâche qui guide les concepteurs décisionnels dans le choix des modules de l'architecture du SID. Dans la section 5.3, nous proposons notre modèle conceptuel multidimensionnel qui généralise les modèles existants afin de représenter toutes les spécificités du SID dont celles liées à la dynamique. En ce sens, dans la section 5.4, nous présentons les impacts des traitements sur notre modèle multidimensionnel. Au regard de notre dichotomie des espaces de stockage, nous définissons les règles de passage du diagramme décisionnel du SID au modèle multidimensionnel dans la section 5.5.1, respectivement au modèle entité-association dans la section 5.5.2. Enfin, dans la section 5.6, nous dressons le bilan des propositions liées à la phase de conception.

5.1 Introduction

Dans le cadre de nos travaux de recherche, nous nous focalisons sur les deux premières tâches de la conception. Nous ne formulons pas de propositions spécifiques aux schémas de correspondances car pour le moment, lors de la mise en place des projets décisionnels, nous utilisions des tableaux de correspondances et les supports de modélisation proposées par les outils ETL des éditeurs tels que [Sunopsis 2006; Oracle 2006; Microsoft 2006b].

5.1.1 Problématique

Les méthodes de développement existantes ne couvrent même pas les deux tâches définies par [Srivastava and Chen 1999] (sus citées) comme explicitées dans notre état de l'art des méthodes de développement dans la section 2.3. Elles tiennent compte d'un, voire de deux modules décisionnels et elles ne guident pas le choix de l'architecture. [Sen and Sinha 2005] proposent un guidage pour le choix des outils décisionnels du marché mais pas des modules supportés par ces outils.

De plus, elles se focalisent principalement sur la modélisation des données du SID comme l'argumentent [Tsois et al. 2001]. Autrement dit, elles focalisent sur la statique du SID. Cependant, les modèles conceptuels associés ne permettent pas de représenter à l'unisson toutes les spécificités liées à l'aspect statique du SID comme indiqué dans la figure 2.5. Le modèle qui permet de représenter le plus grand nombre de spécificités multidimensionnelles est [Luján-Mora et al. 2006]. Néanmoins certaines spécificités liées à l'aspect statique des SID ne sont pas prises en compte : les rôles multiples, les éléments terminaux multiples, les mesures dérivées, les liens entre les mesures. La pertinence des agrégations est partiellement représentée.

De même, les spécificités liées à la dynamique du SID ne sont prises en compte que partiellement. Ce modèle permet d'exprimer certains traitements des SID et ce

uniquement après la conception du schéma du SID. Tous les traitements de préparation des données liés au contexte décisionnel tels que la fiabilité des données sont occultés alors que l'objectif premier des SID est de faciliter la prise de décision.

Notre problématique est donc : **« comment concevoir un SID fiable, par rapport aux besoins des utilisateurs et aux sources, qui soit basé sur une architecture comprenant des modules ayant un schéma multidimensionnel ou entité-association ? »**

5.1.2 Proposition

Le modèle de [Luján-Mora et al. 2006] est basé sur le paradigme objet, plus précisément sur le modèle du diagramme de classes UML. Ce diagramme présente l'avantage de représenter dans un seul modèle la statique et la dynamique du système, mais il ne permet pas de représenter toutes les spécificités des SID. De plus, il n'exploite pas cet avantage du diagramme de classes UML qui fournit une vue globale et intégrée du SID et par la même facilite la validation. Ainsi, dans la continuité de notre diagramme décisionnel, nous définissons un modèle basé sur le diagramme de classes UML. Dans l'optique de définir un modèle qui généralise les modèles existants et qui répond au besoin d'un modèle conceptuel multidimensionnel standard, nous nous inspirons des travaux de [Luján-Mora et al. 2006] et nous nous positionnons par rapport au méta-modèle standard « Common Warehouse Metamodel » (CWM) [OMG 2003]. Le positionnement par rapport à CWM est réalisé à partir de la spécification du paquetage OLAP.

Le diagramme de classes UML est maîtrisé par les informaticiens, les concepteurs décisionnels sont donc dédouanés de l'apprentissage d'un nouveau modèle. Cependant, n'étant pas familier aux décideurs, il convient donc de faciliter la compréhension de notre modèle généralisé. Pour cela, nous facilitons la compréhension du modèle par les utilisateurs en :

- utilisant des stéréotypes différents pour désigner chaque concept multidimensionnel en fonction de la terminologie utilisée par les décideurs,
- explicitant les liens entre les mesures,
- représentant les éléments terminaux multiples,
- définissant plusieurs rôles pour les dimensions,
- explicitant la pertinence des agrégations,
- définissant les opérations liées à la dérivation et à la préparation des données pour chaque classe associée à un concept multidimensionnel,
- explicitant qu'un attribut fait l'objet d'un traitement,
- indiquant que le traitement ne s'applique pas à toute la classe associée au concept multidimensionnel mais uniquement à un attribut donné.

Ainsi, nous proposons une démarche de conception qui permet de guider le concepteur pour le choix d'une architecture adaptée et pour le passage du modèle de besoins à la modélisation multidimensionnelle ou entité-association. Nous proposons

comme support de la méthode de conception un modèle conceptuel multidimensionnel généralisé utilisant le formalisme UML permettant de représenter les données et les traitements des concepts multidimensionnels.

5.2 Choix de l'architecture décisionnelle

Comme défini dans le chapitre 1, un SID est l'ensemble des moyens humains, matériels et logiciels destinés à la collecte, à la mémorisation, à la restitution des informations issues des sources transactionnelles et destinés à faciliter la prise de décision. Un SID est alimenté par une ou plusieurs sources internes ou externes à l'organisation. Au cours de cette chaîne décisionnelle, les données brutes extraites des sources sont transformées en information élaborée et stockée dans des espaces afin d'être restituées aux décideurs sous une forme qui facilite leur prise de décision. Ces informations sont obtenues par la mise en oeuvre de quatre fonctions qui sont l'extraction, la transformation, le chargement et la présentation des données [Annoni et al. 2005a]. Ces fonctions peuvent être regroupées en deux ensembles suivant les traitements que nous avons définis lors de l'analyse. Les traitements de dérivation regroupent les fonctions d'extraction et de chargement alors que les traitements de préparation regroupent les fonctions de transformation et de présentation. En raison des ces fonctions, l'architecture décisionnelle est une agrégation d'un ou plusieurs modules.

Un module peut assurer une ou plusieurs fonctions. Nous retrouvons classiquement les modules « entrepôts de données » et « magasins de données ». Nous proposons d'harmoniser les définitions des modules en tenant compte de ceux présentés dans la littérature [Jarke et al. 2001] et de ceux rencontrés lors des applications développées au sein de la société I-D6. Nous identifions quatre modules (cf. figure 5.2). Ces modules sont définis comme suit :

- **Passerelle Décisionnelle** (PD) : module d'accès direct aux sources afin de présenter les données détaillées aux décideurs. Ce module permet aux décideurs d'accéder aux données brutes des sources. Il n'y a pas de transformation des données sources. Elle assure la fonction de présentation des données. Un exemple de passerelle décisionnelle est l'outil Microsoft Excel doté d'un lien ODBC vers les bases sources,
- **Système de Collecte des Informations** (SCI) : système intermédiaire qui permet de collecter l'information en amont de la chaîne décisionnelle. Cet espace de stockage temporaire permet le nettoyage, l'harmonisation et la consolidation des données issues de sources réparties, mobiles et hétérogènes. Il offre ainsi une manipulation plus simple et centralisée des sources de données. Cet espace de stockage permet également de réduire la charge du système transactionnel due aux alimentations du SID. Ces systèmes reposent généralement sur des bases de données relationnelles sans gestion de l'historisation des données. Il assure les fonctions d'extraction, de transformation et de chargement

entre les sources et l'entrepôt de données ou le magasin de données. Il n'assure pas la fonction de présentation des données car c'est un espace de stockage temporaire,

- **Entrepôt de données** (ED) : espace de stockage centralisé où les données de l'organisation sont intégrées, résumées, agrégées et historisées pour la prise de décision [Inmon 1996]. Il assure les fonctions d'extraction, de transformation, de chargement et de présentation entre les sources ou le système de collecte de l'information et le magasin de données,
- **Magasin de données** (MD) : ensemble cohérent de données répondant aux besoins spécifiques d'une classe de décideurs. Il assure les fonctions d'extraction, de transformation, de chargement et de présentation à partir des sources, du système de collecte de l'information ou de l'entrepôt de données.

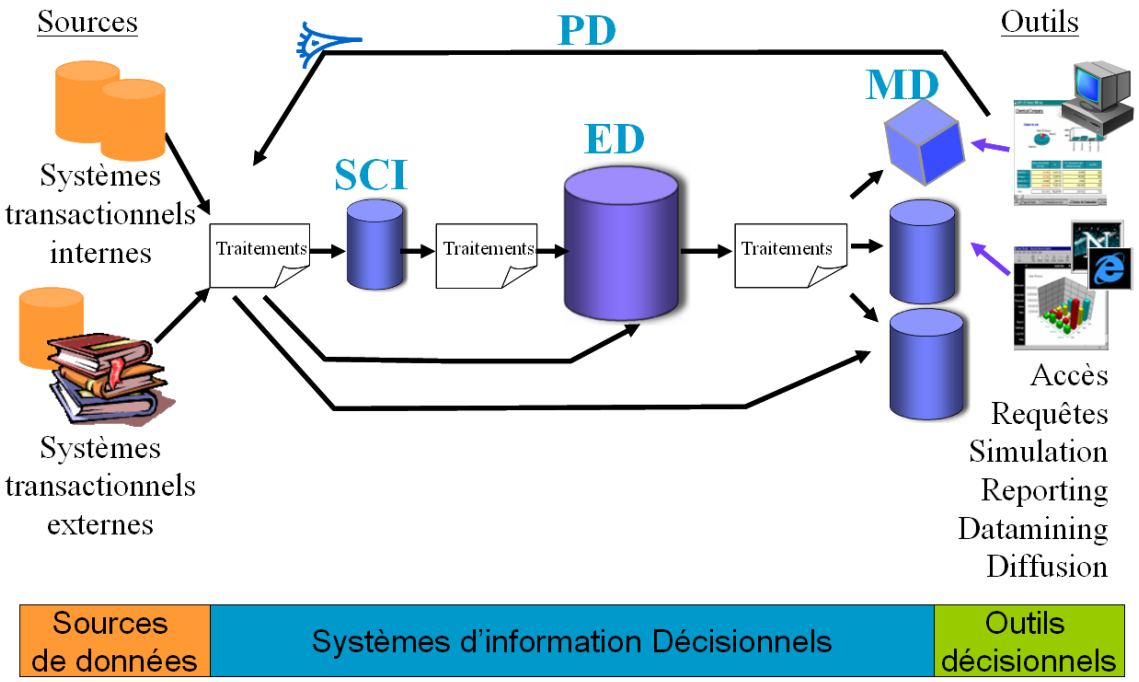


FIG. 5.2 – Les différents modules de l'architecture décisionnelle

Le choix de l'architecture décisionnelle marque le début de la phase de conception. Elle consiste à définir l'architecture adaptée au projet en fonction des besoins des trois groupes d'acteurs. L'un des principes généraux de notre méthode de développement est qu'elle permet de concevoir un SID complet, en l'occurrence basé sur les quatre modules qui composent notre typologie des modules décisionnels.

Ainsi, dans l'optique d'une architecture fiable et adaptée, nous proposons des règles de cohérence pour la définition de celle-ci. Dans la section suivante, nous explicitons ces règles ainsi que la typologie des architectures décisionnelles valides résultantes.

5.2.1 Typologie des architectures décisionnelles valides

Définition 5.1 (Architecture décisionnelle) *Une architecture décisionnelle est une combinaison de modules décisionnels telle les fonctions assurées par ces modules supportent l'hétérogénéité des besoins des acteurs.*

L'architecture est donc définie en fonction des besoins des acteurs. Une architecture peut être composée des 4 types de modules décisionnels sus cités. L'architecture est vide quand l'organisation n'a pas de SID sinon, elle doit respecter les règles de cohérence « CA » suivantes :

- CA1 : tous les modules ainsi que les sources peuvent être interrogés par les utilisateurs afin d'avoir accès aux données à tous les niveaux de détail,
- CA2 : un SCI est un système de stockage temporaire et une PD est un accès direct aux sources donc un SID ne peut être constitué uniquement de {SCI} ou de {PD, SCI}. Ces architectures ne sont donc pas cohérentes,
- CA3 : si l'architecture est composée de plus d'un magasin, alors il existe nécessairement un entrepôt de données pour la centralisation de l'information,
- CA4 : l'architecture peut être constituée de plusieurs modules du même type, mais au plus d'un seul de type entrepôt. Cette règle répond à la caractéristique de l'entrepôt de centraliser toute l'information de l'organisation. De plus, l'information fédérée en un seul espace facilite l'interrogation des données.

Dans le tableau 5.1, nous présentons la typologie des architectures décisionnelles valides, i.e. dont la définition respecte les règles de cohérence « CA ». On pose l'ensemble E = {PD, SCI, ED, MD}. Le nombre de regroupements possibles des quatre modules est égal aux seize combinaisons de un à quatre modules dans l'ensemble E. Nous enlevons les trois architectures non cohérentes :{SCI}, {PD, SCI} et l'architecture vide. Nous obtenons donc treize combinaisons valides.

	Nombre de type de modules			
	1	2	3	4
Architecture	{PD$^{1..N}$}	{PD$^{1..N}$,MD$^{1..N}$}	{PD$^{1..N}$,MD$^{1..N}$, ED}	{PD$^{1..N}$,SCI$^{1..N}$,MD$^{1..N}$,ED}
	{MD$^{1..N}$}	{PD$^{1..N}$,ED}	{PD$^{1..N}$,SCI$^{1..N}$,MD$^{1..N}$}	
	{ED}	{SCI$^{1..N}$,MD$^{1..N}$}	{PD$^{1..N}$,SCI$^{1..N}$,ED}	
		{SCI$^{1..N}$,ED}	{SCI$^{1..N}$,MD$^{1..N}$,ED}	
		{MD$^{1..N}$,ED}		

TAB. 5.1 – Combinaisons valides d'architectures hétérogènes et complètes

Face à l'hétérogénéité de ces architectures décisionnelles, nous proposons de guider le concepteur pour la définition d'une architecture multi-modulaire adaptée. Plus concrètement, nous proposons des critères pour quantifier les besoins des acteurs et un processus automatique pour la définition de l'architecture. Le choix de l'architecture se déroule en deux tâches dont la première est la définition de ces critères et la seconde est l'application du processus automatique pour déterminer les types

de modules de l'architecture. Dans les deux sous-sections suivantes, nous explicitons ces deux tâches et les propositions liées.

5.2.2 Critères de quantification

Nous avons constaté que la définition d'une combinaison de modules adaptée aux besoins d'un projet varie en fonction de six critères mesurables. Nous avons défini les critères suivants [Annoni et al. 2006a] :

- le niveau de couverture des données (NCD) : niveau binaire évaluant la couverture métier des besoins des décideurs de type TCouverture. Les éléments de ce type sont : « vertical » et « transversal ». La couverture du projet est verticale s'il se rapporte à un seul domaine d'activité ou un seul métier ou une seule classe d'utilisateurs sinon, elle est transversale,
- le niveau de traitement des données (NTD) : niveau binaire précisant la qualité des traitements à effectuer sur les sources de données de type TTraitement. Les éléments de ce type sont : « peu travaillé » et « beaucoup travaillé ». Les données sont qualifiées de beaucoup travaillées si les traitements sont des agrégations ou des calculs analytiques,
- le niveau de complexité des sources (NCS) : niveau binaire évaluant les difficultés et les exigences d'interrogation des sources de données de type TComplexité. Les éléments de ce type sont : « peu complexe » et « complexe ». Des sources sont qualifiées de complexes s'il faut les homogénéiser ou les centraliser afin de ne pas pénaliser l'activité quotidienne des sources et de faciliter la gestion des données,
- le niveau d'équipement décisionnel existant (NEE) : niveau multivalué indiquant l'architecture décisionnelle existante au sein de l'organisation, de type TArchitectureExistant. Un élément de ce type a pour valeur l'une des 16 combinaisons comprenant les combinaisons valides et non cohérentes,
- le niveau de pérennité de l'architecture (NPA) : niveau binaire précisant le respect de notre typologie des architectures décisionnelles. Il est de type TPérénnité dont les éléments sont : « pérenne » et « temporaire »,
- le niveau décisionnel souhaité (NDS) : niveau binaire précisant la complétude de l'architecture décisionnelle souhaitée de type TDécisionnel dont les éléments sont : « partiel » et « complet ».

A cette étape de notre démarche du Trident décisionnel, les besoins sont représentés dans le diagramme décisionnel du SID et dans les graphes de propriétés des groupes d'acteurs. Nous utilisons les propriétés de ces graphes pour la quantification de ces critères. Le tableau 5.2 indique les propriétés qui permettent d'évaluer ces six critères. La colonne « Graphes » indique le graphe ou les graphes du groupe d'acteurs dans lequel les propriétés sont définies.

Critère	Propriétés	Graphes
Niveau de couverture des données (NCD)	Périmètre, Couverture	Stratégique, Système
Niveau de traitement des données (NTD)	Hétérogénéité, Calcul	Tactique, Stratégique, Système
Niveau de complexité des sources (NCS)	Complexité	Système
Niveau d'équipement décisionnel existant (NEE)	Niveau d'urbanisation décisionnelle	Tactique, Système
Niveau de pérennité de l'architecture (NPA)	Normalisation du schéma du SID	Système
Niveau décisionnel souhaité (NDS)	Niveau décisionnel souhaité	Stratégique

TAB. 5.2 – Propriétés par critère de quantification

5.2.3 Automatisation

La tâche du choix de l'architecture du SID peut être automatisée car cette architecture est définie suivant six arguments qui prennent des valeurs connues dans des ensembles discrets. De plus, le processus est déterministe suivant la logique procédurale des diagrammes d'activités présentés dans les figures 5.3 et 5.4. Son résultat appartient à l'ensemble des treize architectures valides de notre typologie qui constitue le type TArchitectureValide. Ainsi, nous proposons la fonction Choix_architecture() qui a six arguments correspondant aux six critères et qui renvoie l'architecture adaptée aux besoins. Elle a la signature suivante :

Choix_architecture (NCD :TCouverture, NTD :TTraitement, NCS :TComplexité, NEE :TArchitectureExistant, NPA :TPérennité, NDS :TDécisionnel) : TArchitectureValide

Le nombre de passerelles, de systèmes de collecte de l'information et de magasins n'est pas défini par notre fonction. Cependant, le nombre de magasins de données MD ou encore le nombre de classes distinctes d'utilisateurs C_{sid} se calcule simplement à partir du nombre de magasins de données utilisés par les classes d'utilisateurs du SID existant C_{anc} et du nombre de classes d'utilisateurs distinctes auxquelles s'adresse le futur SID C_{nvx}. Ainsi, les règles de calcul de C_{sid} sont :

$C_{sid} = C_{anc} + C_{nvx}$ si $C_{anc} \cap C_{nvx} = \{\emptyset\}$
$C_{sid} = C_{anc} + C_{nvx}$ - $C_{anc} \cap C_{nvx}$ si $C_{anc} \cap C_{nvx} \neq \{\emptyset\}$

Ce raisonnement est valable pour le calcul du nombre de passerelles décisionnelles. Nous recommandons un seul SCI, mais dans le cas de sources réparties ou mobiles, il importe de les réunir en un SCI au préalable. Puis, de faire le nettoyage et l'harmonisation des données avec les autres sources dans un autre SCI.

La logique procédurale du choix de l'architecture est présentée dans l'annexe A *via* 6 des 32 diagrammes d'activités. Nous présentons ici les deux diagrammes associés au choix de l'architecture dans le cas de la mise en place d'un nouveau SID, i.e. sans existant décisionnel au sein de l'organisation. Pour faciliter la lisibilité des diagrammes d'activités, nous avons réalisé un diagramme par niveau de couverture des données (vertical, tranversal) et par valeur d'architecture existante ($\{\emptyset\}$, $\{PD^{1..N}\}$, $\{SCI^{1..N}\}$, $\{MD^{1..N}\}$, $\{ED\}$, $\{PD^{1..N}, MD^{1..N}\}$, $\{PD^{1..N}, ED\}$, $\{SCI^{1..N}, MD^{1..N}\}$, $\{SCI^{1..N}, ED\}$, $\{MD^{1..N}, ED\}$, $\{PD, SCI\}$, $\{PD^{1..N}, MD^{1..N}, ED\}$, $\{PD^{1..N}, SCI^{1..N}, MD^{1..N}\}$, $\{PD^{1..N}, SCI^{1..N}, ED\}$, $\{SCI^{1..N}, MD^{1..N}, ED\}$, $\{PD^{1..N}, SCI^{1..N}, MD^{1..N}, ED\}$).

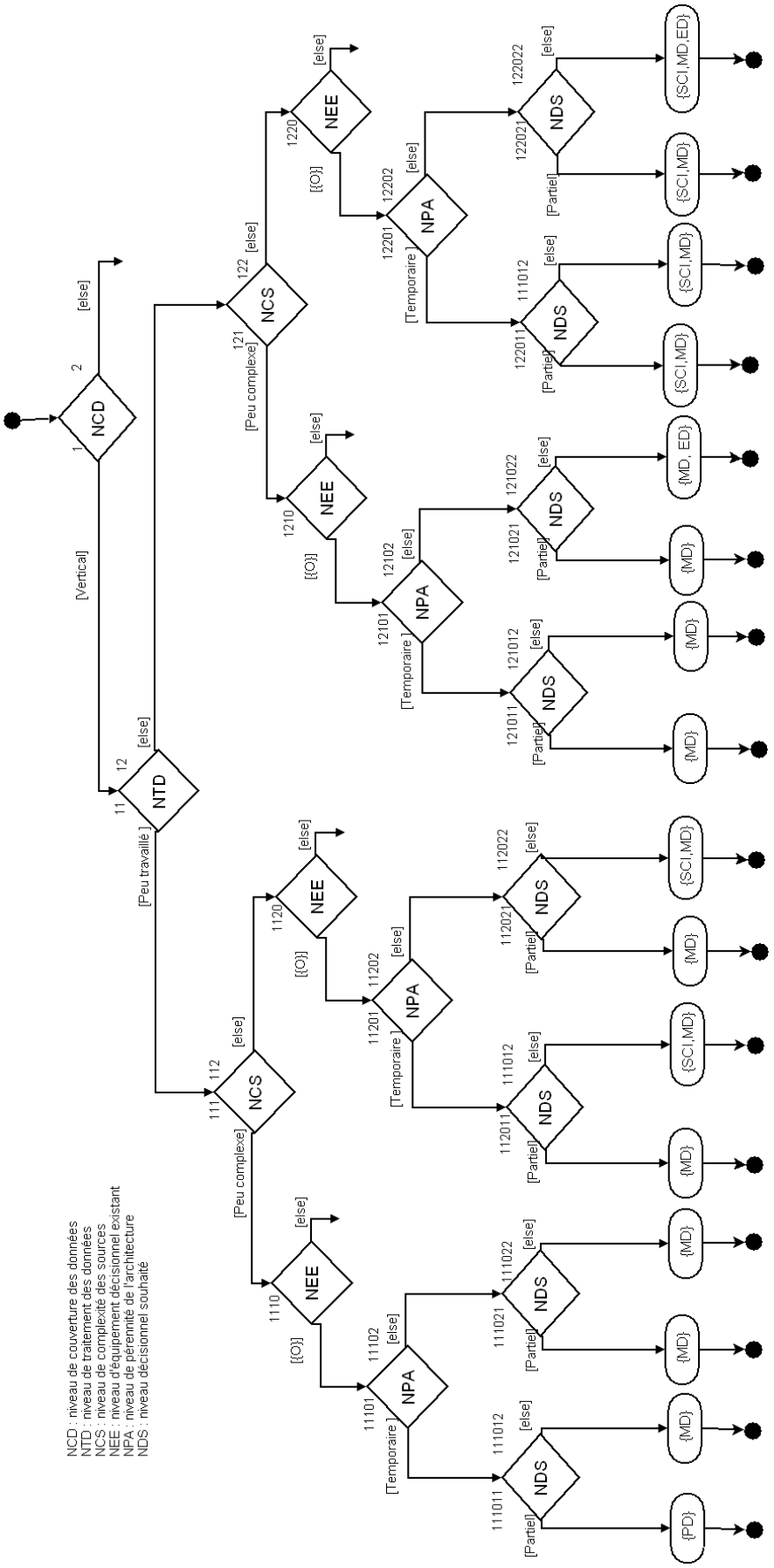


FIG. 5.3 – Diagramme du choix de l'architecture à partir de NCD=Vertical et NEE={∅}

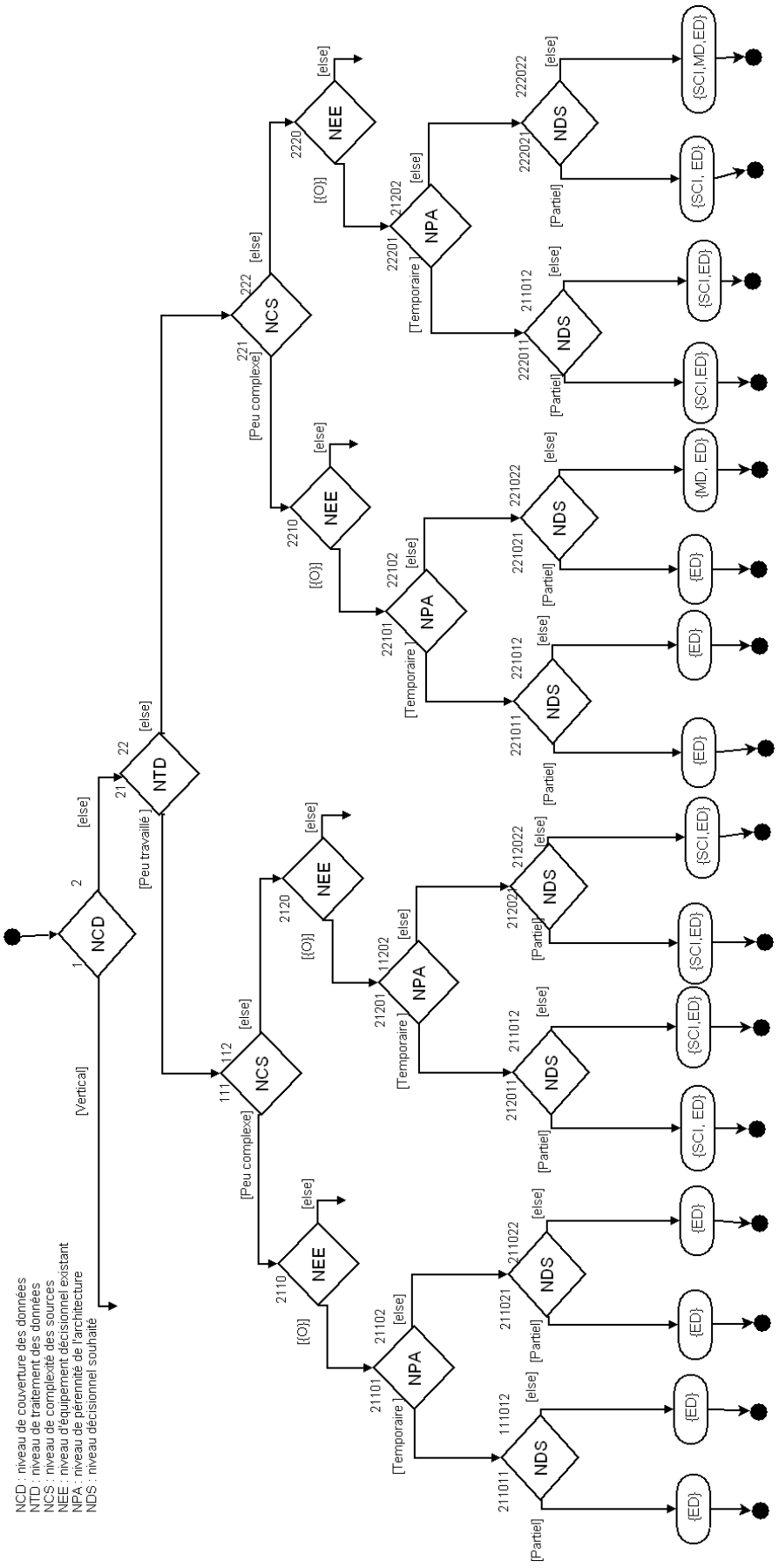


FIG. 5.4 – Diagramme du choix de l'architecture à partir de NCD=Tranversal et NEE={∅}

Nous déroulons trois exemples car le choix de l'architecture décisionnelle est une tâche charnière dans le processus d'ingénierie des SID. Son adéquation par rapport aux besoins des acteurs et aux sources influe sur l'acceptation et la pérennité du SID au sein d'une organisation. Ces exemples correspondent soit à la mise en place d'un nouveau SID, soit à l'extension d'un SID existant. Nous ne présentons pas les graphes de propriétés des exemples 2 et 3, mais nous explicitons l'évaluation des critères en fonction des propriétés annotées de ces derniers.

Exemple : cas n°1. Nous considérons un assureur qui souhaite suivre l'évolution de ses activités. Pilotant de manière empirique son organisation et en raison de la baisse de motivation de certains employés, il souhaite dynamiser son équipe. Son système d'information comprend une base de données associée au progiciel utilisé dans l'organisme et des fichiers de type tableur de présence des employés. Le client ne possède pas de système d'information décisionnel *(NEE = Ø)* et il souhaite établir des rapports peu complexe *(NTD = peu travaillé)* et *(NDS = partiel)* sur l'activité de son équipe *(NCD = vertical)* à partir de sources peu complexes *(NCS = peu complexe)* et qu'il souhaite maintenir *(NPA= pérenne)*. Le 6-uplet a pour valeur **{vertical, peu travaillé, peu complexe, {Ø}, pérenne, partiel}**. Après l'exécution de la fonction « Choix_architecture() », l'architecture du SID à mettre en place est **{PD}**. La logique procédurale est représentée par le chemin qui va du choix 1 au choix 111011 sur la figure 5.3. Cette architecture est monolithique car le SID doit fournir des données peu élaborées concernant une activité à partir de système sources non complexe, autrement dit les traitements de dérivation et de préparation ne sont pas complexes. Les points importants de ce projet sont la présentation des données et la simplicité du SID requis car le client souhaite uniquement analyser plus efficacement l'information dont il dispose afin de proposer des primes aux membres de son équipe.

Exemple : cas n°2. Nous étudions l'architecture du projet d'une grande enseigne nationale fruitière dont l'activité est de collecter et de distribuer des produits frais. En raison de nouveaux marchés signés dans d'autres pays, elle décide d'automatiser la production des tableaux de bord des métiers de la logistique. Elle possède déjà un entrepôt de données et trois magasins de données. Ces derniers se rapportent à la gestion du temps des salariés, à l'activité de dépôt, aux ressources humaines. Les informations relatives aux tableaux de bord de la logistique sont disséminées dans cinq sources de données dont quatre sont délocalisées. Les sources se composent de quatre bases de données et d'un fichier de type tableur. Le contexte d'exploitation est défini par le 6-uplet de valeur **{vertical, beaucoup travaillé, complexe, {ED, 3 MD}, temporaire, complet}**. En effet, l'enseigne possède déjà un système décisionnel composé de plusieurs modules et elle souhaite le compléter pour le pilotage d'un autre domaine d'activité. Les données sont beaucoup travaillées car elles doivent être consolidées en raison des différences de gestion des agences. Les systèmes sources sont complexes car elles sont distantes et délocalisées. Bien que le SID soit défini comme temporaire, son architecture est définie comme complète, la pérennité du SID n'a donc pas d'impact dans ce cas. Ainsi, après l'exécution de la

fonction « Choix_architecture() », l'architecture du SID à mettre en place est **{SCI, ED, MD$^{1..N}$}**. Le choix de cette architecture s'explique de la façon suivante :

- un SCI car certaines sources de données sont délocalisées. Il permet de centraliser et consolider les données,
- quatre MD car le projet concerne l'activité de logistique et auparavant, il existait trois MD. Les classes d'utilisateurs associées aux quatre activités sont distinctes les unes des autres. Les tableaux de bord de la logistique sont actuellement liés à ceux de l'activité de dépôt, mais l'objectif est de les traiter séparément car l'activité s'est accrue lors de la mondialisation de l'enseigne. De plus, les classes d'utilisateurs des autres magasins ne sont pas liées à l'activité de logistique. L'intersection des classes étant nulle, il n'y a qu'un MD par activité. L'architecture est donc composée de quatre MD,
- un ED, en l'occurrence celui qui compose l'architecture décisionnelle existante, afin de centraliser et d'historiser les données car l'architecture du SID comprend plus d'un magasin.

Exemple : cas n°3. Nous définissons l'architecture dans le cadre de notre projet immobilisations à partir des trois graphes de propriétés définis lors de l'analyse des besoins. Ce projet concerne un seul métier suivant la propriété « Couverture » du graphe de propriétés du groupe système donc *(NCD=vertical)*. Ce métier est celui des immobilisations suivant la propriété « Périmètre » du graphe de propriétés du groupe Stratégique. Suivant la propriété « Hétérogénéité » du graphe de propriétés du groupe système et la propriété « Calcul » des graphes de propriétés des trois groupes, les calculs sont de simples calculs arithmétiques donc *(NTD=peu travaillé)*. Suivant la propriété « Complexité » du graphe de propriétés du groupe système, les sources ne sont pas complexes donc (*NCS=peu complexe)*. Suivant la propriété « Niveau d'urbanisation décisionnelle » des graphes de propriétés des groupes tactique et système *(NEE={ED, 1 MD})*. Suivant la propriété « Normalisation du schéma » du graphe des propriétés du groupe système, la normalisation est forte, autrement dit *(NPA=pérenne)*. Enfin, suivant la propriété « Niveau décisionnel souhaité » du graphe de propriétés du groupe stratégique *(NCD=complet)*. Ainsi le contexte d'exploitation est défini par le 6-uplet de valeur **{vertical, peu travaillé, peu complexe, {ED, 1 MD}, pérenne, complet}**. Après l'exécution de la fonction « Choix_architecture() », l'architecture du SID à mettre en place est **{ED, MD$^{1..N}$}**. Le nombre de magasins de données est de 2 car l'existant concerne le service de la finance et cette classe d'utilisateurs est complètement distincte de celle du service des Immobilisations. Nous devons donc concevoir le schéma d'un nouveau magasin de données.

5.3 Modèle multidimensionnel de données et de traitements (MMDT)

Parmi les architectures valides, suivant notre dichotomie des espaces de stockage, certains modules peuvent être basés sur des modèles de représentation classiques (le modèle entité-association ou les modèles orientés objets) ou sur des modèles qui facilitent la prise de décision. La modélisation qui est communément acceptée pour faciliter la prise de décision est la modélisation multidimensionnelle. Elle a fait l'objet de plus d'une dizaine de propositions en une décennie. Cependant, aucun modèle multidimensionnel n'est actuellement reconnu [Ravat et al. 2007b].

Notre modèle de besoins repose sur deux types de classes multidimensionnelles d'analyse. Ces classes sont la classe-fait et la classe-dimension car notre modèle de besoins est destiné à représenter les concepts exprimés par les acteurs et reconnus par les industriels et les chercheurs. Nous avons défini un modèle proche de la vision des données par les décideurs, de compréhension simple et facile car il a pour objectif d'être validé par ces derniers. Cependant, le modèle conceptuel a pour objectif de représenter l'ensemble des concepts (en termes de données et de traitements) et leurs spécificités nécessaires tant aux décideurs qu'aux concepteurs décisionnels.

Pour la définition de notre modèle conceptuel, appelé MDDT, nous étendons le modèle existant [Luján-Mora et al. 2006]. Ce dernier est basé sur trois principaux types de classes multidimensionnelles de conception. Ces classes sont associées aux concepts de « fait », de « dimension » et de « paramètre ou encore niveau » avec les stéréotypes respectifs << Fait >>, << Dimension >>, << Base >>. Elles permettent de représenter un grand nombre de spécificités multidimensionnelles, mais certaines sont partiellement prises en compte voire non traitées. Ces spécificités, classées suivant les propriétés définies dans les listes [Vassiliadis and Sellis 1999; Blaschka et al. 1999b; Abelló et al. 2001a; Rafanelli 2003] et l'aspect dynamique du SID, sont :

- la structure complexe des dimensions et des paramètres :
  - les rôles multiples,
  - les éléments terminaux multiples.
- la structure complexe des faits et des mesures :
  - les mesures dérivées,
  - les liens entre les mesures.
- la cohérence de l'interrogation :
  - la pertinence des agrégations (partiellement prise en compte).
- l'aspect dynamique :
  - les traitements de préparation des données.

La prise en compte de l'aspect dynamique concerne tous les concepts multidimensionnels, nous présentons les spécificités liées à cette propriété lors de la modélisation de chaque concept. Ainsi, dans les trois sections suivantes, nous explicitons :

- comment représenter ces spécificités liées aux trois premières propriétés,

– comment prendre en compte la dynamique du SID pour les concepts multidimensionnels.

Pour une prise en compte continue de l'aspect dynamique des SID au cours du processus d'ingénierie, nous utilisons les concepts d'informativité et les opérations associées aux traitements que nous avons définis dans le chapitre 4 dans le cadre de l'analyse des besoins. Nous rappelons les traitements, les opérations et les concepts d'informativité associés dans le tableau 5.3.

Traitement	**Opération**	**Informativité**
Dérivation		
Harmonisation des données	Harmoniser(c,h)	l
Suivi des erreurs	Trace(l)	t
Remontée des erreurs	Exception(e,m)	e
Permission d'accès	Permission(e,m)	p
Préparation		
Rafraîchissement	Rafraîchir(m,cond)	*
Historisation	Historiser(p,d,cond)	h
Archivage	Archiver(p,d,fct,cond)	a
Consolidation	Consolider(l,{dim_i})	s
Calcul	Calculer({v_i}$^+$)	c
Disponibilité des données	Disponible (d,f,c),	d

TAB. 5.3 – Opération et concept d'informativité par traitement

5.3.1 Structure complexe des dimensions et des paramètres :

La structure complexe des dimensions et des paramètres est caractérisée par les spécificités suivantes : les « hiérarchies simples », les « hiérarchies multiples », les « hiérarchies alternatives », les « hiérarchies non-strictes », les « dimensions dégénérées », les « rôles multiples » et les « éléments terminaux multiples » des dimensions. Les définitions de ces concepts sont présentées dans la section 2.2.1.

Le modèle [Luján-Mora et al. 2006] représente une classe-dimension et ses classes-niveaux organisées en hiérarchies suivant un graphe acyclique direct. Il utilise les rôles « +d » et « +r » qu'il attribue aux classes-niveaux participant à l'association. Cette relation d'association porte le stéréotype << Rolls-upTo >> et elle indique la relation hiérarchique et la navigabilité entre deux classes-niveaux. Le rôle « +d » indique le sens vers lequel se fait l'augmentation du niveau de détail (forage vers le bas) et le rôle « +r » indique le sens vers lequel se fait la réduction du niveau de détail (forage vers le haut). Il utilise la navigabilité des associations UML pour indiquer le forage par défaut (vers le haut ou vers le bas) quand une classe-niveau participe à plusieurs hiérarchies. Il autorise un seul forage vers le haut et un seul forage vers le haut par défaut pour une classe-niveau afin gérer la propagation et la cohérence des

données lors des opérations de forage. Il définit une dimension dégénérée comme un attribut de la classe-fait associée.

Il permet donc de représenter certaines spécificités liées à la structure complexe des dimensions et des paramètres telles que les « hiérarchies simples », les « hiérarchies multiples », les « hiérarchies alternatives », les « hiérarchies non-strictes » et les « dimensions dégénérées ». De ce fait, nous utilisons la même modélisation pour représenter les dimensions et les différents types de hiérarchies.

Cependant, ce modèle ne permet pas de représenter les « rôles multiples » et les « éléments terminaux multiples » des dimensions alors que deux faits d'un schéma en constellation peuvent s'exprimer à deux niveaux de granularité distincts d'une dimension. Il importe de le représenter sur le schéma afin de le traiter correctement au cours du développement du SID.

En ce sens, nous proposons :
- d'ajouter des rôles aux relations multiples entre les classes-faits et les classes-dimensions pour représenter les « rôles multiples » des dimensions,
- d'autoriser des relations d'agrégation entre des classes-faits et les classes associées au concept de niveau pour représenter les « éléments terminaux multiples ».

Nous formulons l'hypothèse selon laquelle les paramètres d'une dimension sont organisés suivant au moins une hiérarchie. Une dimension est donc composée de niveaux organisés en hiérarchie.

Concepts de dimension et de niveau

Une dimension est représentée par une **classe-dimension** indiquée par le stéréotype << Dimension >> et des **classes-niveaux** indiquée par le stéréotype << Niveau >>.

Le compartiment des attributs des classes-niveaux contient les attributs faibles associés au niveau et le paramètre associé au niveau. A chaque attribut sont associés les concepts d'informativité correspondant aux traitements dont il fait l'objet. Le compartiment des opérations des classes-niveaux contient les opérations correspondant aux traitements de dérivation et de préparation des données définies pour toute la classe ou uniquement pour certains de ses attributs.

Le compartiment des attributs des classes-dimensions contient les attributs faibles commun aux niveaux de la dimension et l'attribut associé au paramètre identifiant la dimension. A chaque attribut sont associés les concepts d'informativité correspondant aux traitements dont il fait l'objet. Le compartiment des opérations des classes-dimensions contient les opérations correspondant aux traitements de dérivation et de préparation des données définies pour toute la classe ou uniquement pour certains de ses attributs.

Nous utilisons le stéréotype << Niveau >> afin de manipuler la terminologie des décideurs contrairement à [Luján-Mora et al. 2006] qui utilise le stéréotype << Base >> (cf. figure 5.5). A toute dimension, nous associons un niveau « All » qui est le plus haut niveau de la dimension mais qui n'est pas représenté sur le modèle. La classe-dimension et ce niveau « All » forment la hiérarchie par défaut de toute dimension qui n'a pas de niveau spécifique. Les opérations dont les arguments sont valables pour tous les niveaux et ne diffèrent pas de ceux de la classe-dimension sont définies dans la classe-dimension.

Une hiérarchie est représentée par une note UML, contenant son nom, associée au chemin du graphe représentant la dimension qu'elle organise. Plus précisément, la note est associée à la relation << Rolls-upTo >> qui lie le niveau supérieur avec le niveau précédent de la hiérarchie. Dans le cas de hiérarchies multiples, la note est associée à la relation qui lie le premier niveau de séparation des hiérarchies et le niveau inférieur direct. Le premier niveau de séparation des hiérarchies est défini en partant des feuilles du graphe de la dimension. De plus, afin de préciser les hiérarchies, nous proposons de suffixer le nom du rôle de l'association par les numéros des chemins qui comprennent l'arête associée à cette relation.

Par ailleurs, nous proposons de représenter les dimensions dégénérées de manière explicite. Nous définissons donc une classe-dimension avec le stéréotype << DimensionDegeneree >> contenant uniquement l'attribut dimension dégénérée et les opérations liées à cet attribut. Cette classe est liée à la classe de fait par une relation d'agrégation avec les multiplicités 1-1. De plus, nous ne déclarons pas le concept

d'informativité « s » lié à l'opération « Consolider » pour cet attribut. En effet, les attributs dimensions dégénérées identifient de manière unique le fait. Leurs sommes, leurs moyennes, leurs minima, leurs maxima n'ont donc pas de sens. La seule fonction d'agrégation applicable est la fonction « compte ». Comme la signature de cette opération est la même pour tout attribut dimension dégénérée, afin de ne pas surcharger notre modèle nous ne déclarons pas cette opération pour ces attributs.

Exemple

Nous présentons la dimension « Biens » avec ses deux hiérarchies « h_Catégorie » et « h_Fabricant ». La hiérarchie « h_Catégorie » est formée des classes-niveaux « Sous_catégories » et « Catégories ». La hiérarchie est formée de la classe-niveau « Fabricants ». La classe-dimension est l'élément le plus fin de chaque dimension.

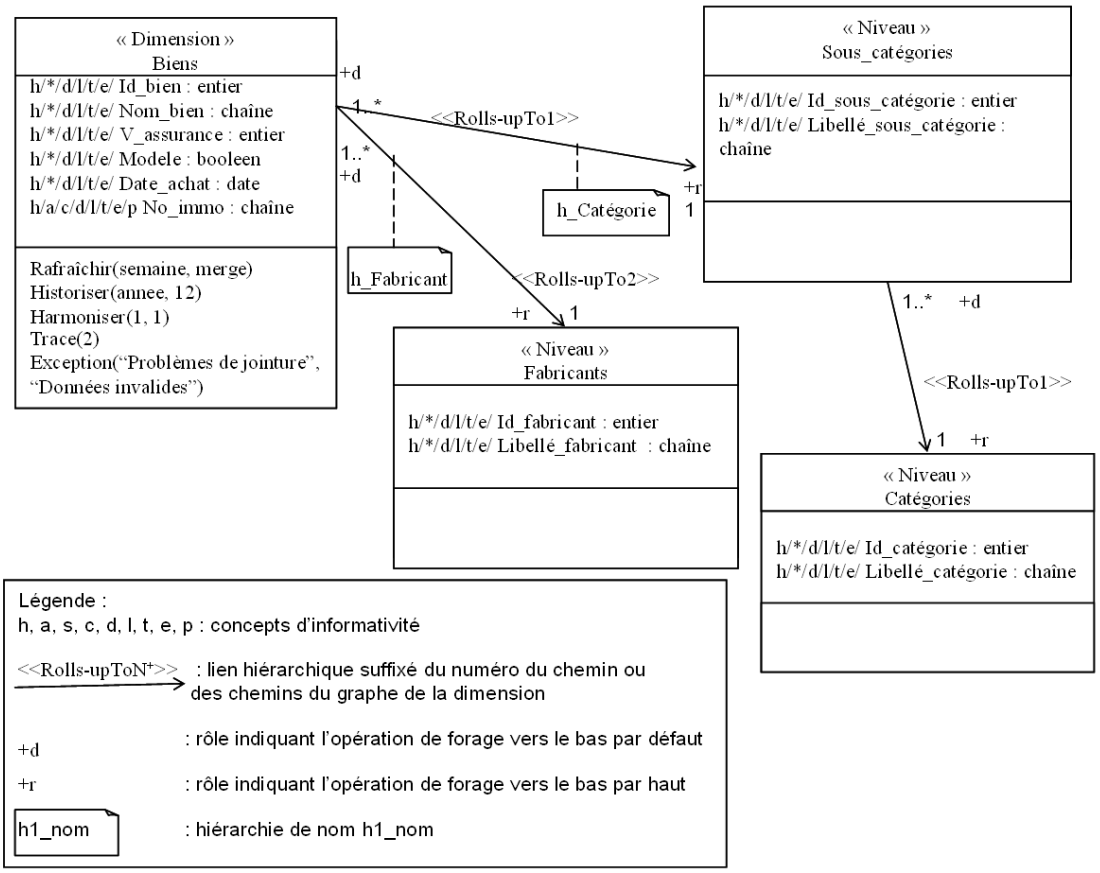


FIG. 5.5 – La dimension « Biens » avec ses deux hiérarchies

Un exemple de représentation des éléments terminaux multiples de la dimension « Biens » est la suivante : la classe-fait « Immobilisations » est connectée à la classe-dimension « Biens » alors que la classe-fait « Achats » est connectée à la classe-niveau

« Sous_catégories », qui est une des classes-niveaux de la dimension « Biens ». Cette représentation permet d'exprimer que les achats sont analysés par sous-catégorie de biens achetés mais, qu'ils ne peuvent pas être analysés par bien.

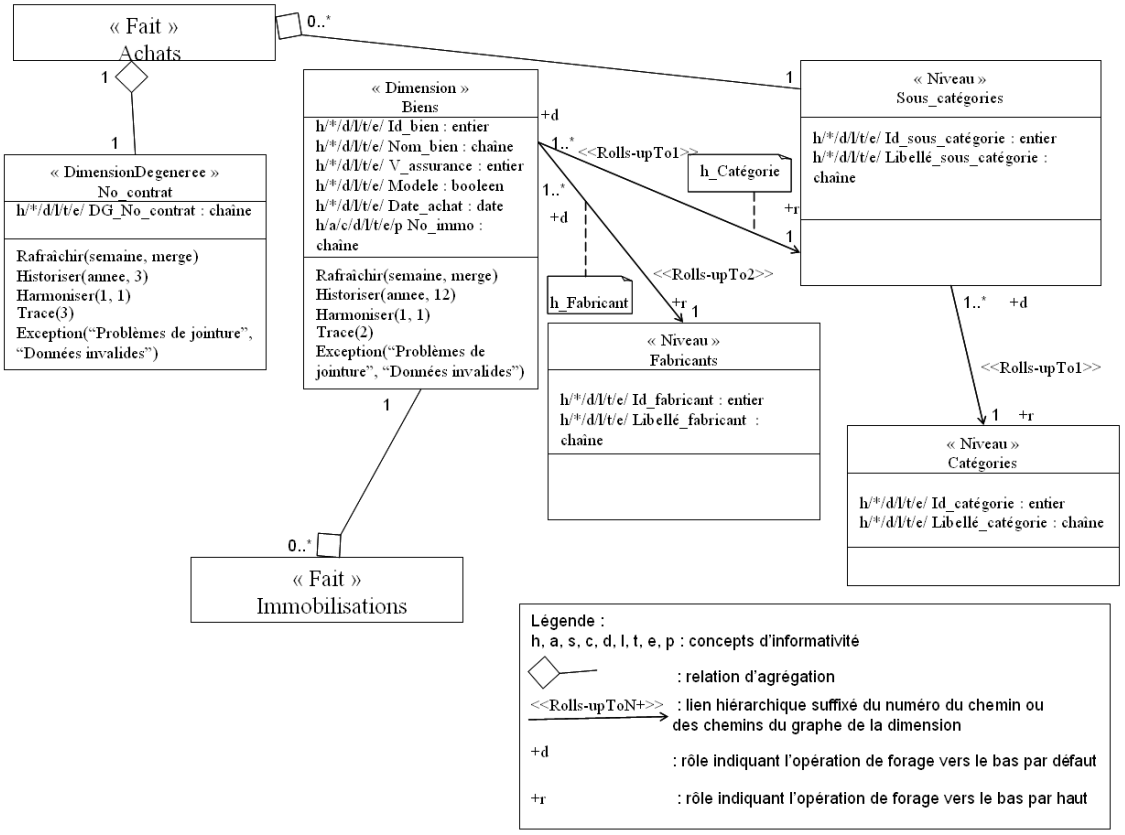


FIG. 5.6 – La dimension « Biens » avec ses deux hiérarchies, ces deux éléments terminaux dans un schéma en constellation et la dimension dégénérée No_contrat représentée explicitement

Nous proposons aussi une représentation simplifiée d'une dimension dégénérée à l'instar de [Luján-Mora et al. 2006]. Elle consiste à définir l'attribut correspondant comme un attribut de la classe-fait auquel la propriété d'informativité « s » n'est pas associée et l'opération « Consolider » n'est pas définie. La représentation simplifiée de la dimension dégénérée « No_contrat » est présentée dans la figure 5.7.

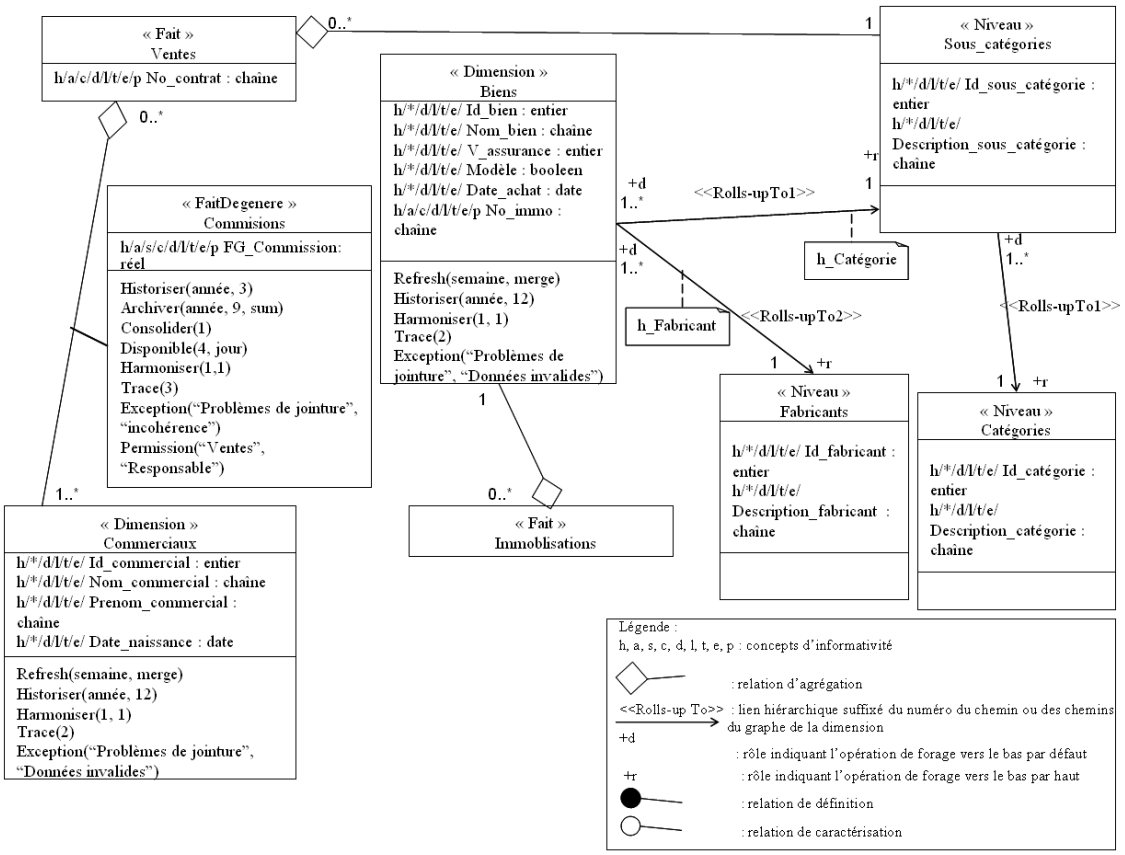


FIG. 5.7 – La représentation implicite de la dimension dégénérée « No_contrat » et la représentation du fait dégénéré « Commissions »

Positionnement par rapport à CWM

Notre classe multidimensionnelle de conception classe-dimension est une instanciation de la méta-classe « Dimension » avec :

- l'attribut booléen « isTime » égal à « True » ou « False » pour respectivement une dimension temporelle ou une dimension autre,
- l'attribut booléen « isMeasure » toujours égal à « False ».

Dans le cas d'un niveau non terminal, notre classe multidimensionnelle de conception classe-niveau est une instanciation de la méta-classe « Level ». Cependant, dans le cas d'un niveau terminal autre que celui par défaut d'une dimension donnée, notre classe multidimensionnelle de conception classe-niveau est une instanciation de la méta-classe « Dimension » avec les mêmes caractéristiques pour les attributs « isTime » et « isMeasure » définies ci-dessus. Le paquetage OLAP du méta-modèle CWM [OMG 2003] spécifie qu'une dimension peut être composée de 0 ou plusieurs hiérarchies, alors que nous considérons au moins une hiérarchie par défaut pour la navigation dans les données du fait connecté à la dimension.

5.3.2 Structure complexe des faits et des mesures :

La structure complexe des faits et des mesures est caractérisée par les spécificités suivantes : les « schémas en constellation », les « mesures multiples » et les « faits dégénérés », les « mesures dérivées » et les « liens entre les mesures ». Les définitions de ces concepts sont présentées dans la section 2.2.1.

Le modèle [Luján-Mora et al. 2006] que nous généralisons représente des classes-faits avec leurs attributs « dimensions dégénérées » et leurs attributs « mesures » sur un même schéma. Un fait dégénéré est un fait « F » défini tel que pour une valeur de ce fait, plusieurs valeurs d'une dimension « D » peuvent être associées. Il est donc représenté par une classe d'association UML avec le stéréotype << FaitDegenere >> lié à la relation (cf. figure 5.7). Il permet donc de représenter des spécificités de la structure complexe des faits et des mesures telles que les « schémas en constellation », les « mesures multiples » et les « faits dégénérés ». De par ces caractéristiques, nous utilisons la même modélisation pour représenter ces spécificités.

Cependant, le modèle [Luján-Mora et al. 2006] ne représente pas les « mesures dérivées » et les « liens entre les mesures » alors que l'analyse de ces mesures permettent de préciser les tendances et les corrélations. Il ne propose pas une représentation qui distingue les mesures dérivées des autres mesures et il ne représente pas sur le schéma les règles de calcul des mesures. La méthode « Calculer() » de la valeur_vénale définie dans le diagramme décisionnel du SID (cf. figure 4.17) en est un exemple. Cette dernière possède les arguments « Valeur_achat » et « Dépréciation ». « Dépréciation » est une mesure du fait.

En ce sens, nous proposons :
- de distinguer le concept de mesure de celui de fait et de considérer les mesures comme des classes à part entière pour représenter les « mesures dérivées »,
- de définir les liens existants entre les classes multidimensionnelles associées aux mesures et aux faits. Nous définissons aussi les liens existants entre les classes multidimensionnelles associées aux mesures elles-mêmes pour représenter les « liens entre les mesures » exprimés dans les règles de calcul à l'origine des mesures dérivées.

Concepts de fait et de mesure

Un fait est représenté par une classe multidimensionnelle appelée **classe-fait** indiquée par le stéréotype << Fait >>. Le compartiment des attributs des classes-faits contient des attributs spécifiques aux faits. Il peut être vide. A chaque attribut sont associés les concepts d'informativité correspondant aux traitements dont il fait l'objet. Le compartiment des opérations des classes-faits contient les opérations correspondant aux traitements de dérivation et de préparation des données définies pour toute la classe ou uniquement pour certains de ses attributs.

Une mesure est représentée par une classe multidimensionnelle appelée **classe-mesure** indiquée par le stéréotype << Mesure >>. Le compartiment des attributs des classes-mesures contient des attributs qui décrivent la mesure. A chaque attribut sont associés les concepts d'informativité correspondant aux traitements dont il fait l'objet. Le compartiment des opérations des classes-mesures contient les opérations correspondant aux traitements de dérivation et de préparation des données définies pour toute la classe ou uniquement pour certains de ses attributs.

Considérer les mesures comme des classes à part entières implique de définir des relations entre ces deux types de classes pour associer une mesure dérivée à un fait. Une mesure est une caractéristique du fait ; elle n'est pas un élément constitutif du fait au même titre qu'un niveau est un élément constitutif d'une hiérarchie d'une dimension donnée. De plus, certaines mesures sont nécessairement définies pour un fait car elles ne peuvent pas être nulles dans le sens où elles portent la sémantique du fait. Ainsi, pour indiquer les classes mesures qui caractérisent ou qui définissent une classe-fait, nous proposons deux nouvelles relations appelées **caractérisation** et **définition**. Nous définissons ces relations en s'inspirant respectivement des relations d'agrégation et de composition UML car il existe un couplage qui peut être faible ou fort entre une mesure et un fait. Cependant, la propagation des valeurs d'attributs n'est pas pertinente entre des classes-faits et des classes-mesures.

Définition 5.2 (La relation de caractérisation) *La relation de caractérisation entre une classe-fait et une classe-mesure spécifie que la classe-mesure peut ne pas être définie pour cette classe-fait. Une relation de caractérisation existe uniquement entre une classe-fait et une de ses classes-mesures.*

Définition 5.3 (La relation de définition) *La relation de définition entre une classe-fait et une classe-mesure spécifie que la classe-mesure est nécessairement définie pour cette classe-fait. Une relation de définition existe uniquement entre une classe-fait et une de ses classes-mesures.*

Ces relations expriment les liens entre les classes-faits et les classes-mesures. La relation de caractérisation est représentée par un trait terminé par un rond vide,

alors que la relation de définition qui est un lien fort est représentée par un trait terminé par un rond noirci.

Nous pouvons maintenant représenter les liens entre les classes-mesures participant à des mesures dérivées. Nous avons constaté que les liens entre les mesures d'un fait, qui sont des mesures dérivées ou qui participent aux calculs de ces dernières, correspondent à deux types de dépendances. Ces dépendances sont la « dépendance utilise » et la « dépendance hiérarchique ».

Définition 5.4 (La dépendance utilise) *La dépendance utilise spécifie qu'une mesure d'un fait est calculée à partir d'autres mesures de ce fait et d'autres données non présentes dans le SID. Autrement dit, le calcul des données d'une classe-mesure M_a utilise les données de classes-mesures $\{M_b\}^+$ (soit une ou plusieurs classes-mesures M_b) et des données OLTP non présentes dans le SID.*

Définition 5.5 (La dépendance hiérarchique) *La dépendance hiérarchique indique qu'une mesure d'un fait est calculée uniquement à partir d'autres mesures de ce même fait. Autrement dit, le calcul des données d'une classe-mesure M_k utilise uniquement les données des classes-mesures $\{M_c\}^+$ (soit une ou plusieurs classes-mesures M_c).*

Afin d'éviter un modèle complexe, nous réutilisons la relation de dépendance du diagramme de classes UML avec les stéréotypes << utilise >> et << hiérarchique >> pour représenter respectivement la relation de « dépendance utilise » et la relation de « dépendance hiérarchique ». Nous avons constaté au cours des projets que la dépendance utilise est le lien le plus commun entre deux classes-mesures d'un fait.

Exemple

Nous présentons la modélisation du fait « Immobilisations » avec les mesures « Valeur_vénale », « Dépréciation » et « Amortissement ». Ces mesures sont définies comme des classes à part entière. La mesure « Dépréciation » n'est pas connue au début d'une immobilisation, la classe-mesure « Dépréciation » est donc liée à la classe-fait Immobilisations par une relation de caractérisation. La « Valeur_vénale » est au moins égale à la « Valeur_achat » du bien car la « Valeur_achat » d'un bien initialise et définit l'immobilisation liée. La classe-mesure « Valeur_vénale » est donc liée à la classe-fait par une relation définition. Il existe une relation de dépendance « utilise » entre la mesure « Valeur_vénale » et la mesure « Dépréciation » car elle est calculée à partir de la mesure « Dépréciation » et de la donnée des sources « Valeur_achat ».

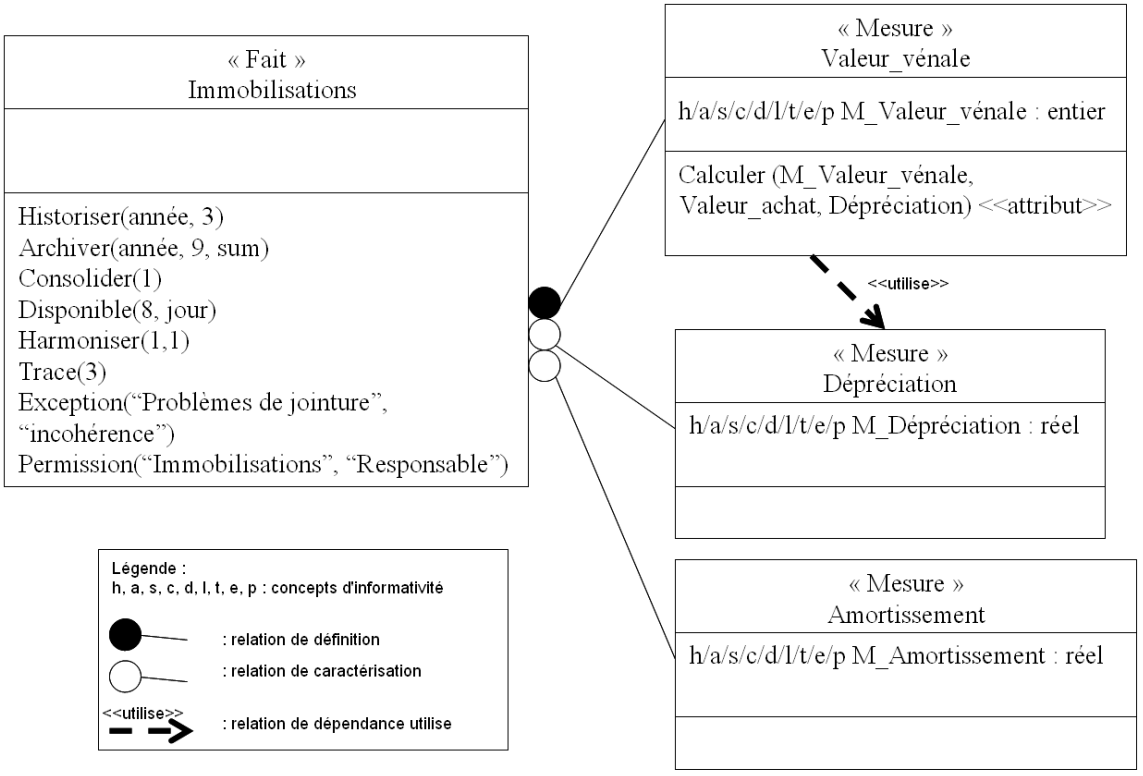


FIG. 5.8 – Modélisation conceptuelle du fait « Immobilisations » et des liens entre ses mesures

Dans le domaine de la comptabilité européenne, il existe des relations de « dépendance hiérarchique » car la balance commerciale est composée de la mesure « Services » et de la mesure « Biens ». Cette dernière qui est composée à son tour des mesures « Import » et « Export » comme indiqué dans la figure 5.9.

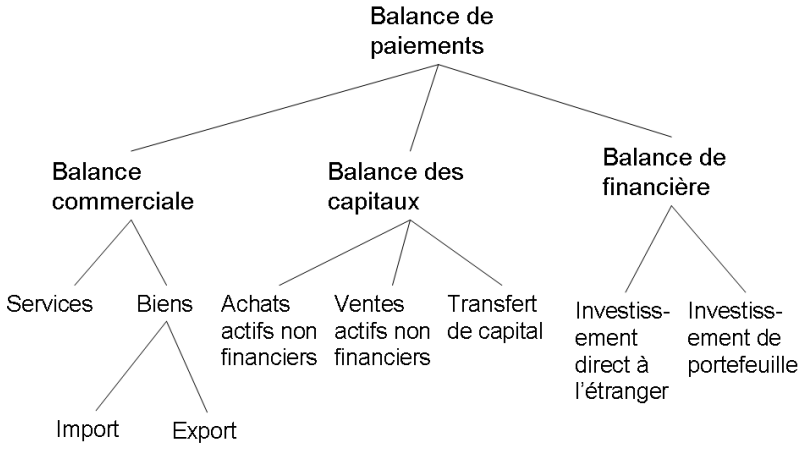


FIG. 5.9 – Hiérarchie des mesures de la comptabilité européenne (cf. INSEE [INSEE 2007])

Nous proposons aussi une représentation simplifiée des mesures indépendantes qui ne participent pas à la règle de calcul de mesures dérivées. Cette représentation consiste à définir les mesures indépendantes comme des attributs des classes-faits (cf. figure 5.10). Ainsi, dans le cas d'une représentation simplifiée de notre modèle, le compartiment des attributs des classes-faits contient des attributs qui sont des dimensions dégénérées et des attributs qui sont des mesures indépendantes. Nous privilégions la représentation implicite dans le reste du mémoire afin d'obtenir un schéma avec une lisibilité plus importante.

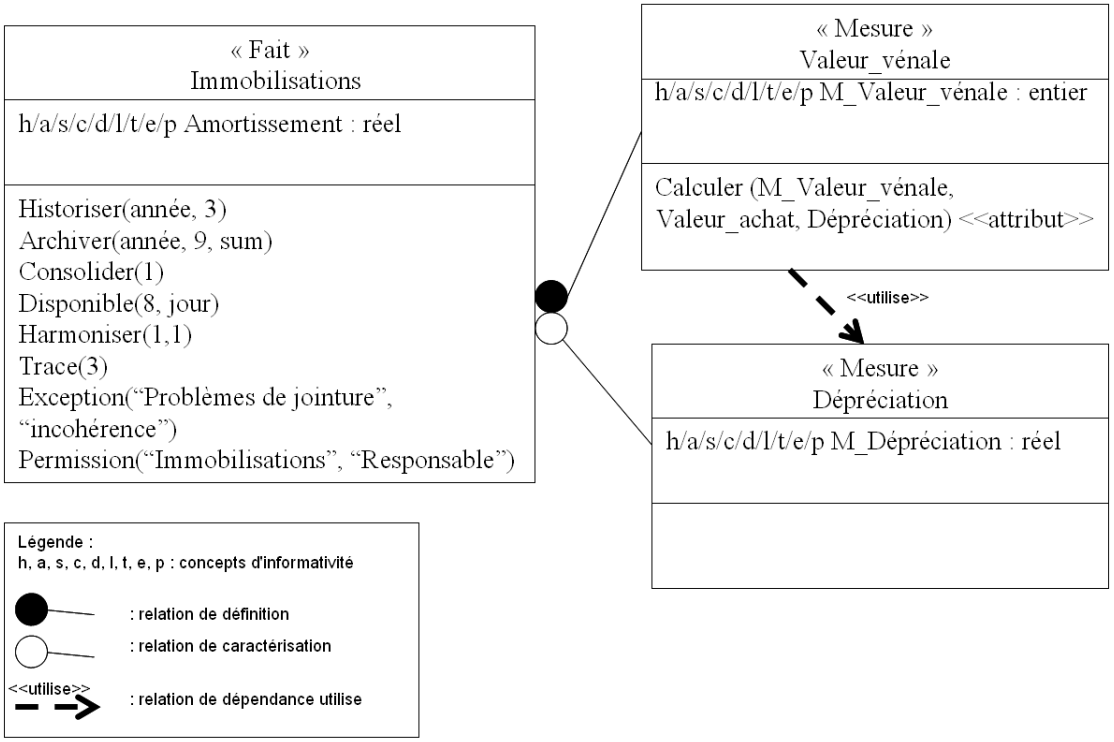


FIG. 5.10 – La représentation implicite du fait « Immobilisations » et des liens entre ses mesures

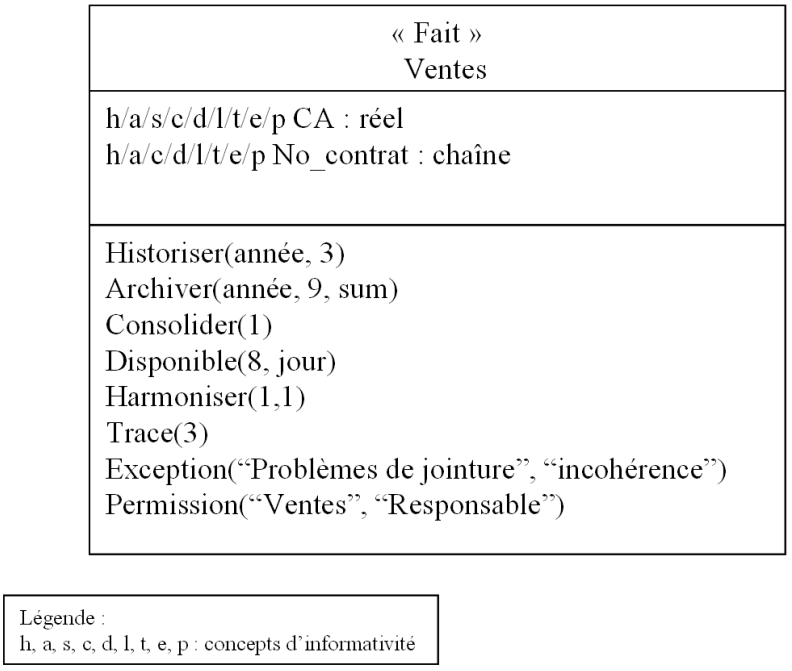


FIG. 5.11 – La représentation implicite du fait « Ventes » avec l'attribut mesure indépendant « CA » et l'attribut dimension dégénérée « No_contrat »

Un avantage de la définition de ces liens entre les mesures est celui de faciliter le contrôle des données et ainsi de garantir leur fiabilité. De plus, la prise en compte de leurs besoins de mesures dérivées permet d'une part, de répondre à leurs besoins et d'autre part, d'améliorer les performances du SID car ces données sont disponibles dans le SID ; elles ne sont pas re-calculées dynamiquement.

Positionnement par rapport à CWM

Notre classe multidimensionnelle classe-fait est une instanciation de la méta-classe « Cube » avec l'attribut booléen « isVirtual » toujours égal à « False ».

Cependant, notre classe multidimensionnelle classe-mesure ne correspond pas à l'instanciation d'une classe du paquetage OLAP de CWM [OMG 2003] car ce méta-modèle ne définit pas une mesure comme une classe à part entière. Notre classe-mesure constitue donc une originalité par rapport au méta-modèle CWM pour la représentation des liens entre les mesures.

5.3.3 Cohérence de l'interrogation :

La cohérence de l'interrogation est basée principalement sur la « pertinence des agrégations ». La définition de cette spécificité est présentée dans la section 2.2.1.

Le modèle [Luján-Mora et al. 2006] représente partiellement la pertinence des agrégations. Il spécifie les fonctions d'agrégation qui ne sont pas applicables pour une mesure donnée du fait dans une note UML associée à la relation d'agrégation entre la classe-fait et la classe-dimension. Les agrégations des autres mesures, toutes fonctions d'agrégation confondues, peuvent donc être jugées pertinentes. Cependant, comme l'argumente [Husemann et al. 2000], la pertinence des agrégations dépend des faits et des mesures ainsi que des fonctions d'agrégation.

La fonction d'agrégation par défaut du modèle [Luján-Mora et al. 2006] est la somme. Cependant, les agrégations suivant la fonction « somme » de certains faits et de certaines mesures n'ont pas de sens. Par exemple, la fréquence d'absence d'un salarié au cours d'un mois à un sens, mais la somme des fréquences d'absence de ce salarié au cours des douze mois d'une année civile n'a pas de sens.

Ainsi, nous définissons la fonction « compte » comme la fonction d'agrégation par défaut de notre modèle, soit la fonction associée au plus faible niveau d'agrégation (pour lequel une fonction existe) défini par [Pedersen and Jensen 1999]. De plus, nous représentons explicitement les agrégations pertinentes et les agrégations non pertinentes des mesures et des faits en fonction des dimensions via la définition de l'opération « Consolider ». Cette opération est liée au traitement de préparation des données appelé « Consolidation ». Elle a deux arguments :

- l : niveau de consolidation l parmi les niveaux définis par [Pedersen and Jensen 1999] (cf. tableau 4.7),
- $\{dim_i\}^+$: 0 à n dimensions suivant lesquelles les agrégations ne sont pas pertinentes. Par défaut, les agrégations sont pertinentes suivant toutes les dimensions connectées au fait.

Avec la possibilité de définir des opérations au niveau des attributs que fournit notre modèle, les concepteurs décisionnels peuvent explicitement représenter les agrégations pertinentes et les agrégations non pertinentes des faits et des mesures dépendantes mais, aussi celles des mesures indépendantes qui sont des attributs des classes-faits.

Ainsi, notre modèle conceptuel MMDT repose sur quatre types de classes multidimensionnelles de conception qui sont les classes-faits, les classes-dimensions, les classes-mesures et les classes-niveaux. De plus, il permet de représenter les données et les traitements relatifs à ces concepts dans un seul schéma car nous définissons les opérations associées aux traitements dans le compartiment des opérations des classes multidimensionnelles de conception. Nous avons donc modélisé les concepts de faits, de mesures, de dimensions, de niveaux, d'attributs faibles, de hiérarchies. A travers les exemples, nous avons modélisé les spécificités de l'aspect statique des SID et de l'aspect dynamique des SID qui sont partiellement prises en compte ou qui ne

sont pas prises en compte par notre modèle de référence [Luján-Mora et al. 2006] (les éléments terminaux multiples, les rôles multiples, les mesures dérivées, les liens entre les mesures, la pertinence des agrégations et les traitements de préparation des données). La figure 5.12 présente notre modèle MMDT.

Afin de garantir la validité des schémas conceptuels multidimensionnels de données et de traitements définis, nous définissons les règles de cohérence des relations entre les classes-dimensions, les classes-niveaux, les classes-faits et les classes-mesures suivantes :

- CCM1 : une classe-dimension peut être reliée à une classe-fait par une relation d'agrégation et à une classe-niveau par une relation d'association,
- CCM2 : une classe-niveau peut être reliée à une classe-dimension et à une autre classe-niveau par des relations d'association,
- CCM3 : une classe-dimension peut être reliée plusieurs fois à une même classe-fait avec des rôles différents. Ces rôles sont spécifiés par des stéréotypes ajoutés sur la relation d'agrégation entre la classe-fait et la classe-dimension,
- CCM4 : une classe-niveau peut être reliée par une relation d'agrégation à une classe-fait afin de modéliser le niveau en fonction duquel le fait est exprimé,
- CCM5 : une classe-fait peut être reliée à une classe-mesure par une relation de caractérisation ou de définition,
- CCM6 : deux classes-mesures peuvent être reliées entre elles par une relation de dépendance utilise ou de dépendance hiérarchique.

La prise en compte des traitements dans la conception du SID a des impacts sur sa modélisation. Nous proposons d'évaluer ces impacts et leur représentation dans la section suivante.

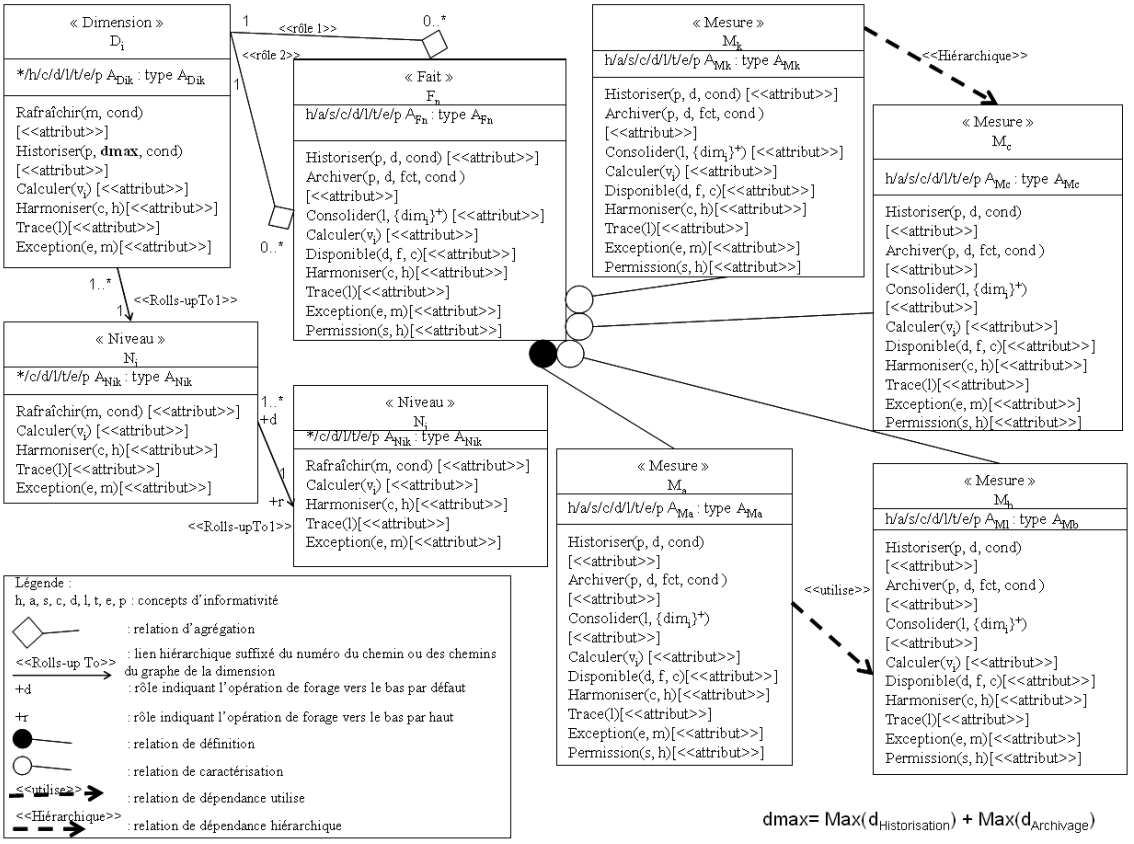


FIG. 5.12 – Modèle multidimensionnel de données et de traitements (MMDT)

5.4 Impacts des traitements

La définition des traitements permet de spécifier la dynamique des SID. Un autre intérêt est de garantir la définition de toutes les données nécessaires pour assurer la dérivation du SID à partir des systèmes sources et la préparation des données pour faciliter la prise de décision. Cependant, la mise en oeuvre de certains traitements d'un concept décisionnel implique des modifications des données des concepts qui lui sont liés. Nous proposons de spécifier les impacts des traitements sur le modèle afin d'indiquer les concepts qui sont concernés en cas de modification des autres concepts.

Nous avons constaté que certains traitements ne sont pertinents que pour certains concepts. De même, la spécification de certaines opérations suffit pour à la compréhension des impacts liés. En ce sens, dans cette section, nous passons en revue les traitements afin d'analyser leurs impacts et la représentation de ces derniers sur le schéma multidimensionnel du SID, puis nous récapitulons les traitements valables par classe multidimensionnelle.

5.4.1 Impacts des traitements de dérivation des données

Impacts du traitement « Criticité des données »

Les décideurs exploitent les données en naviguant *via* des opérations de forage afin d'avoir plus de détails. Les opérations de forage s'effectuent sur le fait suivant les hiérarchies de cette dimension car les données d'un fait étant calculées en fonction des valeurs de dimensions. Définir des droits d'accès aux données du SID correspond à définir les droits d'accès aux faits du SID. La « criticité » du fait correspond donc à celle du SID. Le traitement associé est déclaré que pour les classes-faits et les classes-mesures car la criticité du fait implique celle des dimensions connectées. Cet impact étant directement lié à la définition de l'opération « Permission() » dans les classes-faits et les classes-mesures, nous n'ajoutons pas une autre représentation. La « criticité » d'une classe-fait en relation de caractérisation ou de définition avec ses classes-mesures sont contraints par celle de la classe-mesure ayant les arguments les plus limitants.

Dans le cas d'un schéma en constellation, il n'y pas de problème relatif à la criticité de l'accès aux données car l'accès est donné pour un fait et l'accès à une dimension ne donne pas l'accès à tous les faits qui lui sont connectés.

Impacts des traitements « Suivi des erreurs » et « Remontée des erreurs »

Les traitements « Suivi des erreurs » et « Remontée des erreurs » permettent de surveiller la cohérence des données restituées aux décideurs. Le niveau de suivi ne peut prendre la valeur trois, soit l=3, que pour les classes-faits ou les classes-dimensions. Pour les classes-niveaux et classes-mesures, cet argument peut pendre les valeurs 1 et 2. Dans le cas d'un niveau de suivi l=3, le suivi et la gestion des exceptions des classes-dimensions (respectivement classes-faits) impliquent le suivi et la gestion des exceptions de leurs classes-niveaux (respectivement classes-mesures). Ainsi, pour l=3, l'opération « Trace() » et l'opération « Exception() » ne sont pas déclarées pour les classes-niveaux ou les classes-mesures. Cet impact n'est pas associé à un symbole afin de ne pas surcharger le schéma et de faciliter sa compréhension par les décideurs.

Impacts des traitements « Harmonisation »

Le traitement « Harmonisation » est défini pour toutes classes multidimensionnelles car ces dernières dérivent d'éléments des sources qui ont des caractéristiques d'extraction et de chargement propres.

5.4.2 Impacts des traitements de préparation des données

Impacts du traitement « Rafraîchissement »

Le traitement « Rafraîchissement » est déclaré pour les classes-dimensions ou les classes-niveaux. En effet, pour assurer la cohérence des données du SID la mise à jour des ces classes-dimensions déclenche la mise à jour des classes-faits connectées. Cependant, la mise à jour de certaines classes-dimensions ou classes-niveaux n'impacte pas le fait. Par exemple, la mise à jour de la classe-dimension Temps par l'ajout de dates, de mois et d'années n'a aucun incident sur les classes-faits connectées. Ainsi, pour spécifier que le rafraîchissement d'une classe-dimension implique le rafraîchissement de la classe-fait, nous ajoutons sur la relation d'agrégation entre le fait et la dimension deux chevrons précédant le losange de l'agrégation (cf. figure 5.13).

Dans le cas de classes-dimensions connectées à une classe-fait suivant deux rôles, le rafraîchissement est défini pour un rôle donné. Dans le cadre de la modélisation conceptuelle, nous n'abordons pas les problèmes de l'implantation, mais suivant l'implantation de la classe-dimension basée sur une ou deux structures distinctes, la prise en compte de cet impact est différente.

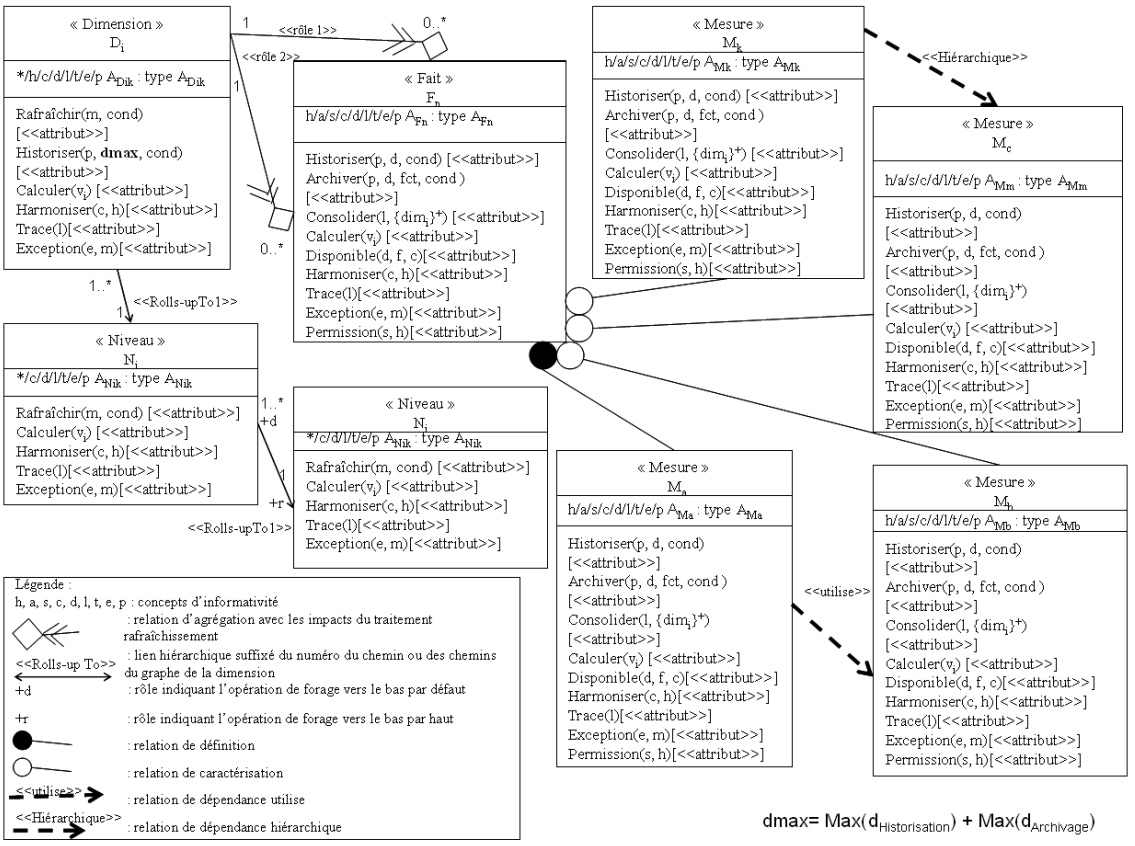


FIG. 5.13 – Modèle multidimensionnel de données et de traitements du domaine Immobilisations avec les impacts de traitements

Impacts des traitements « Historisation » et « Archivage »

Les traitements « Historisation » et « Archivage » sont principalement liés aux classes-faits et classes-mesures mais sachant que les décideurs réalisent des requêtes sur les anciennes données, il faut historiser les données des classes-dimensions et des classes-niveaux. Afin, que le SID soit cohérent durant tout son cycle de vie, nous définissons la règle selon laquelle la période d'historisation d'une classe-dimension est égale à la somme de la plus grande période d'historisation et la plus grande période d'archivage des classes-faits connectées. Ainsi, afin que cet impact soit validé aussi bien par les concepteurs que les décideurs, l'opération d'historisation est déclarée pour les classes-dimensions avec une période respectant cette règle. L'opération « Archiver() » n'est pas définie pour les classes-dimensions et les classes-niveaux.

Pour des classes-mesures n'ayant pas les mêmes périodes d'historisation et d'archivage en raison des besoins des décideurs, les périodes d'historisation et d'archivage les plus importantes doivent être prises comme référence pour les classes-faits qu'elles caractérisent ou définissent.

La définition complète du traitement Archivage requiert une table d'archivage indiquant pour chaque classe-fait et classe-mesure, par classe-dimension, la classe-niveau ou l'attribut de la classe-dimension « Temps » où l'archivage est défini. Les cellules de la table d'archivage sont renseignées à partir de l'argument « fct » des opérations « Archiver() » du diagramme décisionnel du SID. Cet argument est défini à partir de la propriété « Consolidation » du graphe de propriétés. Les niveaux d'archivage sont définis par les utilisateurs.

Exemple : la table d'archivage de notre projet immobilisations est définie dans le tableau 5.4.

Niveau d'archivage		classe-fait Immobilisations		
		classe-mesure Valeur_vénale	classe-mesure Dépréciation	classe-mesure Amortissement
classe-dimension Biens	classe-niveau Biens			
	classe-niveau Sous_catégories			somme
	classe-niveau Catégories	somme	somme	
	classe-niveau Fabricants	somme	somme	somme
classe-dimension Etats	classe-dimension Etats			
classe-dimension Temps	Id_temps			
	Mois			
	Trimestre	somme	somme	somme
	Semestre			
	Année			

TAB. 5.4 – Table d'archivage du projet immobilisations

Impacts des traitements « Consolidation » et « Calcul »

Le traitement « Consolidation » est défini que pour les classes-faits et les classes-mesures car cette opération n'est pertinente que pour les concepts de faits et de mesures [Husemann et al. 2000]. Ce traitement est généralement spécifié par des opérations attributs car tous les attributs d'une classe multidimensionnelle de conception ne sont pas nécessairement consolidés suivant la même fonction d'agrégation. Quand un attribut d'une classe-fait ou d'une classe-mesure ne peut pas être consolidé suivant une ou plusieurs dimensions, ces classes-dimensions sont passées en paramètre de l'opération Consolider() après le niveau de consolidation.

Le traitement « Calcul » est défini pour toutes classes multidimensionnelles car tout attribut d'une classe multidimensionnelle peut faire l'objet d'une règle de calcul.

Exemple : la règle de calcul de l'attribut quadigramme de la classe-dimension « Employés » du SID de la société I-D6 est la concaténation de la première lettre du prénom et des trois premières lettres du nom d'un employé.

Impacts du traitement « Disponibilité »

L'analyse des données correspond à l'analyse des faits et des mesures le long des dimensions. La « disponibilité » du fait correspond donc à celle du SID. Le traitement associé est déclaré que pour les classes-faits et les classes-mesures car la disponibilité du fait implique celle des dimensions connectées. Cet impact étant directement lié à la définition de l'opération « Disponible() » dans les classes-faits et les classes-mesures, nous n'ajoutons pas une autre représentation. La « disponibilité » d'une classe-fait en relation de caractérisation ou de définition avec ses classes-mesures sont contraints par celles de la classe-mesure ayant les arguments les plus limitants.

Dans le cas d'un schéma en constellation, la disponibilité des dimensions communes à plusieurs classes-faits est contrainte par la classe-fait ayant la plus faible disponibilité.

5.4.3 Les traitements par concept multidimensionnel

Comme tous les traitements ne peuvent pas être appliqués à toutes les classes multidimensionnelles, il importe de définir pour chacune uniquement les traitements qui lui sont applicables comme présenté dans la figure 5.5.

Classe	Traitements applicables
Classe-fait	Historisation, Archivage, Consolidation, Calcul, Disponibilité des données, Harmonisation, Suivi des erreurs, Remontée des erreurs, Permission d'accès
Classe-mesure	Historisation, Archivage, Consolidation, Calcul, Disponibilité des données, Harmonisation, Suivi des erreurs (sauf si niveau de suivi l=3), Remontée des erreurs (sauf si niveau de suivi l=3), Permission d'accès
Classe-dimension	Rafraîchissement, Historisation (avec une durée d=Max($d_{F_{Archivage}}$) + Max($d_{F_{Historisation}}$), Calcul, Harmonisation, Suivi des erreurs, Remontée des erreurs
Classe-niveau	Rafraîchissement, Calcul, Harmonisation, Suivi des erreurs (sauf si niveau de suivi l=3), Remontée des erreurs (sauf si niveau de suivi l=3)

TAB. 5.5 – Traitements par classe multidimensionnelle

Ainsi, six opérations sont définies pour les classes-dimensions au lieu de neuf pour les classes-faits et les classes-mesures, et cinq pour les classes-niveaux. Les opérations qui ont les mêmes arguments pour toutes les classes-niveaux sont déclarées une seule fois et elles le sont dans la classe-dimension. De même, les opérations qui ont les mêmes arguments pour toutes les classes-mesures sont déclarées une seule fois et dans la classe-fait.

La spécification des impacts implicites et explicites des traitements associés aux classes multidimensionnelles sur le schéma conceptuel du SID présente un avantage pour les concepteurs décisionnels car elle indique les concepts qui sont concernés en cas de modification d'un concepts. L'analyse des impacts des modifications sur un système existant est une tâche qui représente 80% du coût total de ce système [Barros 1995; Ecudié 1997; Barros 1997]. Notre modélisation conceptuelle contribue à faciliter les tâches liées à l'analyse d'impacts au cours de l'étape de maintenance du système.

5.5 Conception des modules décisionnels

La conception des nouveaux modules décisionnels composant l'architecture du SID est réalisée à partir de son diagramme décisionnel défini lors de l'analyse. Suivant le module décisionnel, il convient de le transformer en un schéma multidimensionnel *via* notre proposition de modèle conceptuel multidimensionnel de données et de traitements ou de le transformer en un schéma entité-association. Dans les deux sections suivantes, nous développons la transformation via notre modèle conceptuel MMDT, puis nous présentons la transformation via le modèle entité-association.

5.5.1 Passage du diagramme décisionnel à notre modèle MMDT

Le diagramme décisionnel contient les deux types de classes d'analyse qui sont les classes-faits et les classes-dimensions. A partir de ces classes, nous proposons de dériver les quatre types de classes de conception qui sont les classes-faits, les classes-mesures, les classes-dimensions et les classes-niveaux suivant un ensemble de règles appelées PMMDT (Passage au Modèle Multidimensionnel de Données et de Traitements). Nous commençons par transformer les classes-faits d'analyse, puis nous transformons les classes-dimensions d'analyse.

Règles PMMDT des classes-faits d'analyse

L'objectif de la transformation des classes-faits d'analyse est de représenter la structure complexe des faits et des mesures, la cohérence des interrogations et la dynamique du SID associée au concept de fait et de mesure exprimées implicitement dans le diagramme décisionnel. Ainsi, pour chaque classe-fait d'analyse, nous appliquons les règles suivantes :

- **PMMDT1** : toute classe-fait d'analyse F_A est transformée en une classe-fait de conception F_C,
- **PMMDT2** : tout attribut de la classe-fait F_A qui est un iième (i>1) argument d'une opération d'attribut « Calculer() » est défini comme une classe-mesure. Cette classe-mesure est à la source d'un traitement, nous la désignons donc par M_s. Cette classe-mesure contient l'attribut dont le nom est préfixé par « M_ » qui correspond à la mesure ainsi que les opérations associées à ce dernier,
- **PMMDT3** : tout attribut de la classe-fait F_A qui est le premier argument d'une opération d'attribut « Calculer() », dont au moins un des autres arguments est un attribut de la classe-fait, est défini comme une classe-mesure. Cette classe-mesure est la cible d'un traitement, nous la désignons donc par M_c. Cette classe-mesure contient l'attribut dont le nom est préfixé par « M_ » qui correspond à la mesure, l'opération « Calculer() » ainsi que les autres opérations associées à ce dernier. Cette règle permet de définir les mesures dérivées,
- **PMMDT4** : tout attribut de la classe-fait F_A qui ne vérifie pas les règles PMMDT2 et PMMDT3 est un attribut de la classe-fait de conception F_C associée. Cette règle permet de définir les mesures indépendantes,
- **PMMDT5** : une classe-mesure M_s ou M_c est liée à une classe-fait F_C par une relation de définition si cette classe-mesure ne peut pas être vide et si elle est indissociable de cette classe-fait,
- **PMMDT6** : une classe-mesure M_s ou M_c est liée à la classe-fait F_C par une relation de caractérisation si cette classe-mesure peut être vide ou si elle n'est pas nécessairement connu pour cette classe-fait,
- **PMMDT7** : deux classes-mesures M_c et M_s sont reliées par une relation de « dépendance hiérarchique » si M_s est un argument de l'opération « Calculer() » de M_c et si les autres arguments sont exclusivement des classes-mesures,

- **PMMDT8** : deux classes-mesures M_c et M_s sont reliées par une relation de « dépendance utilise » si M_s est un argument de l'opération « Calculer() » de M_c et si les autres arguments ne sont pas exclusivement des classes-mesures, autrement dit les arguments sont des classes-mesures et des données OLTP non présentes dans le SID ou des données OLTP uniquement,
- **PMMDT9** : toute opération de F_A est définie uniquement pour la classe-fait, sauf les opérations spécifiques aux attributs à l'origine des classes-mesures. Cependant, les traitements définis pour une classe-fait ou une classe-mesure doivent respecter les impacts des traitements et doivent appartenir à l'ensemble des traitements valables pour une classe-fait respectivement une classe-mesure comme défini dans le tableau 5.5.

Règles PMMDT des classes-dimensions d'analyse

L'objectif de la transformation des classes-dimensions d'analyse est de représenter la structure complexe des dimensions et la dynamique du SID associée aux concepts de dimension et de niveau. Ainsi, pour chaque classe-dimension, nous appliquons les règles suivantes :

- **PMMDT10** : toute classe-dimension d'analyse D_A connectée à une classe-fait d'analyse F_A et non cible d'une dépendance fonctionnelle est transformée en une classe-dimension de conception D_C,
- **PMMDT11** : toute classe-dimension d'analyse D_A cible d'une dépendance fonctionnelle est transformée en une classe-niveau de conception N_C,
- **PMMDT12** : tout attribut et toute opération d'une classe-dimension d'analyse D_A correspondant à une D_C est un attribut de la classe-dimension D_C associée,
- **PMMDT13** : tout attribut et toute opération d'une classe-dimension d'analyse D_A correspondant à une N_C est un attribut de la classe-niveau N_C associée,
- **PMMDT14** : toute relation « En Fonction » entre une classe-fait F_A et une classe-dimension D_A est transformée en une relation d'agrégation vers la classe-fait F_C 0..* - 1 D_C si D_A n'est pas cible d'une dépendance fonctionnelle. Si une classe-fait est reliée deux fois à une classe-dimension alors, il faut donner des rôles différents aux relations d'agrégation. Cette règle définit les éléments terminaux multiples et les rôles multiples. Elle évite qu'une classe-fait soit reliée à deux éléments terminaux distincts d'une dimension,
- **PMMDT15** : toute relation de dépendance fonctionnelle entre deux classes-dimensions (transformées en D_C et N_C) est transformée en une relation d'association,
- **PMMDT16** : toute relation de dépendance fonctionnelle entre deux classes-dimensions (transformées en deux classes-niveaux source N_{C_s} et cible N_{C_c}) est transformée en une relation d'association ayant le stéréotype << RollsupTo >> avec les multiplicités N_{C_c} 1 - 1..* N_{C_s} ou N_{C_c} 1 - 1 N_{C_s} suivant le projet. La navigabilité de l'association par défaut est de N_{C_s} vers N_{C_c}. Le rôle « +d » est associée à la classe-niveau N_{C_s} et le rôle « +r » est associé à la classe-niveau N_{C_c}. Cette règle permet de déterminer les types de hiérarchies,

– **PMMDT17** : toute opération de D_C est définie pour la classe-dimension ou la classe-niveau associée. Cependant, les traitements définis doivent respecter les impacts des traitements et doivent appartenir à l'ensemble des traitements valables pour une classe-dimension respectivement une classe-niveau comme défini dans le tableau 5.5.

Les autres spécificités multidimensionnelles

La structuration complexe des faits et mesures ainsi que des dimensions inclut la définition des faits dégénérés et des dimensions dégénérées. Les règles suivantes ne permettent pas de prendre en compte ces spécificités. Nous proposons donc les suivantes :

– **PMMDT18** : tout attribut d'une classe-dimension D qui constitue l'origine d'une dépendance fonctionnelle directe, par rapport à l'ensemble des dimensions connectées au fait, dont la cible est l'ensemble des attributs d'un fait F est une dimension dégénérée. Cet attribut est déplacé de la dimension D vers le fait F,
– **PMMDT19** : tout attribut d'une classe-fait F tel que pour une valeur de la classe-fait, il est associé à plusieurs valeurs d'une classe-dimension D. Cette mesure devient une classe-fait dégénérée. Elle est définit entre la classe-fait F dont elle est extraite et la classe-dimension D avec laquelle elle définit une relation N-M.

Exemple : l'application des règles de passage à partir du diagramme décisionnel de notre projet immobilisations (présenté dans le chapitre 4 que nous rappelons dans la figure 5.14) est présenté ci-après.

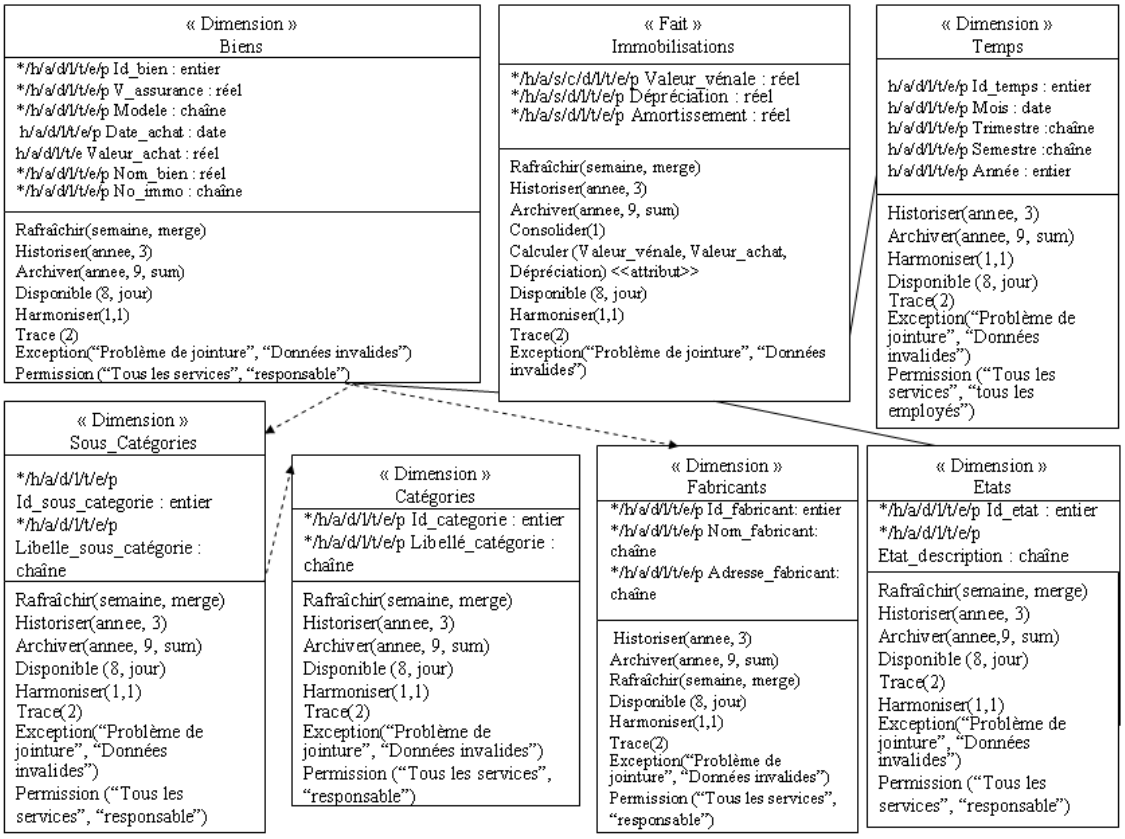


FIG. 5.14 – Diagramme décisionnel du SID du projet immobilisations (rappel)

Nous déroulons les règles PMMDT pour les classes-faits d'analyse :

- PMMDT1 : la classe-fait d'analyse « Immobilisations » est transformée en une classe-fait de conception appelée « Immobilisations »,
- PMMDT2 : l'attribut « Dépréciation » est transformée en une classe-mesure avec l'attribut M_Dépréciation. La classe-mesure « Dépréciation » n'a pas d'opération car il n'y a pas d'opération d'attribut définie pour cette mesure dans la classe-fait d'analyse « Immobilisations »,
- PMMDT3 : l'attribut « Valeur_vénale » est transformée en une classe-mesure avec l'attribut M_Valeur_vénale et l'opération Calculer (M_Valeur_vénale, Valeur_achat, Dépréciation) << attribut >> uniquement car il n'y a pas d'opération d'attribut définie pour cette mesure dans la classe-fait d'analyse « Immobilisations »,
- PMMDT4 : l'attribut « Amortissement » est l'attribut de la classe-fait « Immobilisations »,
- PMMDT5 : la classe-mesure « Valeur_vénale » est reliée à la classe-fait par une relation de définition car elle ne peut pas être vide et elle définit la sémantique de la classe-fait « Immobilisations »,
- PMMDT6 : la classe-mesure « Dépréciation » est reliée à la classe-fait par une relation de caractérisation car elle peut être vide lors de la déclaration du bien,

- PMMDT7 : l'opération « Calculer() » de la classe-fait d'analyse n'a pas uniquement des classes-mesures comme argument alors il n'y a pas de relation de dépendance hiérarchique dans notre schéma,
- PMMDT8 : la classe-mesure « Valeur_vénale » et la classe-mesure « Dépréciation » sont reliées par une relation de « dépendance utilise » car les arguments de l'opération Calculer() de la classe-mesure « Valeur_vénale » sont « Dépréciation » et la donnée OLTP « Valeur_achat »,
- PMMDT9 : l'opération « Rafraîchir() » n'est pas une opération valable pour les classes-faits donc elle est supprimée. Nous obtenons donc les opérations suivantes dont les arguments respectent les impacts des traitements « Historiser(année, 3) », « Archiver(année, 9, sum) », « Consolider(1) », « Disponible(8, jour) », « Harmoniser(1,1) », « Trace(2) », « Exception(« Problème de lien », « incohérence ») », « Permission(« Immobilisations », « Responsable ») ». Les traitements de Trace() et Exception sont définis au niveau des classes-mesures car le niveau de suivi de la classe-fait est l=2.

Nous déroulons les règles PMMDT pour les classes-dimensions d'analyse :

- PMMDT10 : les classes-dimensions d'analyse « Biens », « Etats » et « Temps » sont transformées en des classes-dimensions de conception car elles sont connectées à la classe-fait d'analyse « Immobilisations » et ne sont pas cibles d'une dépendance fonctionnelle,
- PMMDT11 : les classes-dimensions d'analyse « Sous_catégories », « Catégories » et « Fabricants » sont transformées en des classes-niveaux de conception car elles sont les cibles de relations de dépendance fonctionnelle,
- PMMDT12 : les attributs et les opérations des classes-dimensions de conception « Biens », « Etats » et « Temps » sont ceux de la classe-dimension d'analyse associée,
- PMMDT13 : les attributs et les opérations des classes-dimensions de conception « Sous_catégories », « Catégories » et « Fabricants » sont ceux respectivement des classes-dimensions d'analyse « Sous_catégories », « Catégories » et « Fabricants »,
- PMMDT14 : les relations « En Fonction » entre la classe-fait « Immobilisations » et les classes-dimensions « Biens », « Etats » et « Temps » sont transformées en des relations d'agrégation,
- PMMDT15 : la relation de dépendance fonctionnelle entre « Biens » et « Sous_catégories » et celle entre « Biens » et « Fabricants » sont transformées en des relations d'association,
- PMMDT16 : la relation de dépendance fonctionnelle entre « Sous_catégories » et « Catégories » est transformée en une relation d'association ayant le stéréotype << Rolls-upTo >> dont la navigabilité va de « Sous_catégories » vers « Catégories » avec les multiplicités « Catégories » 1 - 1..* « Sous_catégories » car une sous-catégorie est associée à une et une seule catégorie mais une catégorie peut regrouper plusieurs sous-catégories. Le rôle « +d » est associée à la classe-niveau « Sous_catégories » et « +r » est associé à la classe-niveau « Catégories »,

– PMMDT17 : pour les classes-dimensions « Biens », « Etats » et « Temps », les opérations « Archiver(), Disponible(), Permission() » ne sont pas valables pour les classes-dimensions donc elles sont supprimées. Nous obtenons donc les opérations suivantes dont les arguments respectent les impacts des traitements : « Rafraîchir(semaine, merge) », « Historiser(annee, 12) », « Harmoniser(1, 1) », « Trace(2) », » Exception(« Problèmes de jointure », « Données invalides ») ». L'argument durée de l'opération Historiser() est égale à 12, soit la somme de la durée d'historisation et de la durée d'archivage de la classe-fait « Immobilisations ». Pour les classes-niveaux, les opérations « Historiser(), Archiver(), Disponible(), Permission() » ne sont pas valables. Les valeurs des arguments des autres opérations sont identiques pour tous les niveaux et elles ne diffèrent pas de celles de la classe-dimension « Biens ». Les opérations ne sont donc pas déclarées dans les classes-niveaux.

Nous déroulons les règles pour la prise en compte des dimensions dégénérées et des faits dégénérés :

– PMMDT18 : il existe la dépendance fonctionnelle {No_immo -> Dépréciation , Valeur_vénale, Amortissement}, mais elle n'est pas valide lorsque que l'on tient compte de l'ensemble des dimensions qui sont connectées au fait « Immobilisations ». Autrement dit, le « No_immo » ne permet pas de déterminer de manière unique une immobilisation par bien, par état et au cours du temps car le « No_immo » ne change au cours du temps et suivant l'état. « No_immo » n'est donc pas une dimension dégénérée,
– PMMDT19 : Aucun attribut de la classe-fait « Immobilisations » ne définit une relation N-M avec une des classes-dimensions. Il n'y a donc pas de fait dégénéré dans notre exemple.

Le schéma conceptuel multidimensionnel du nouveau magasin du projet immobilisations obtenu est présenté dans la figure 5.15.

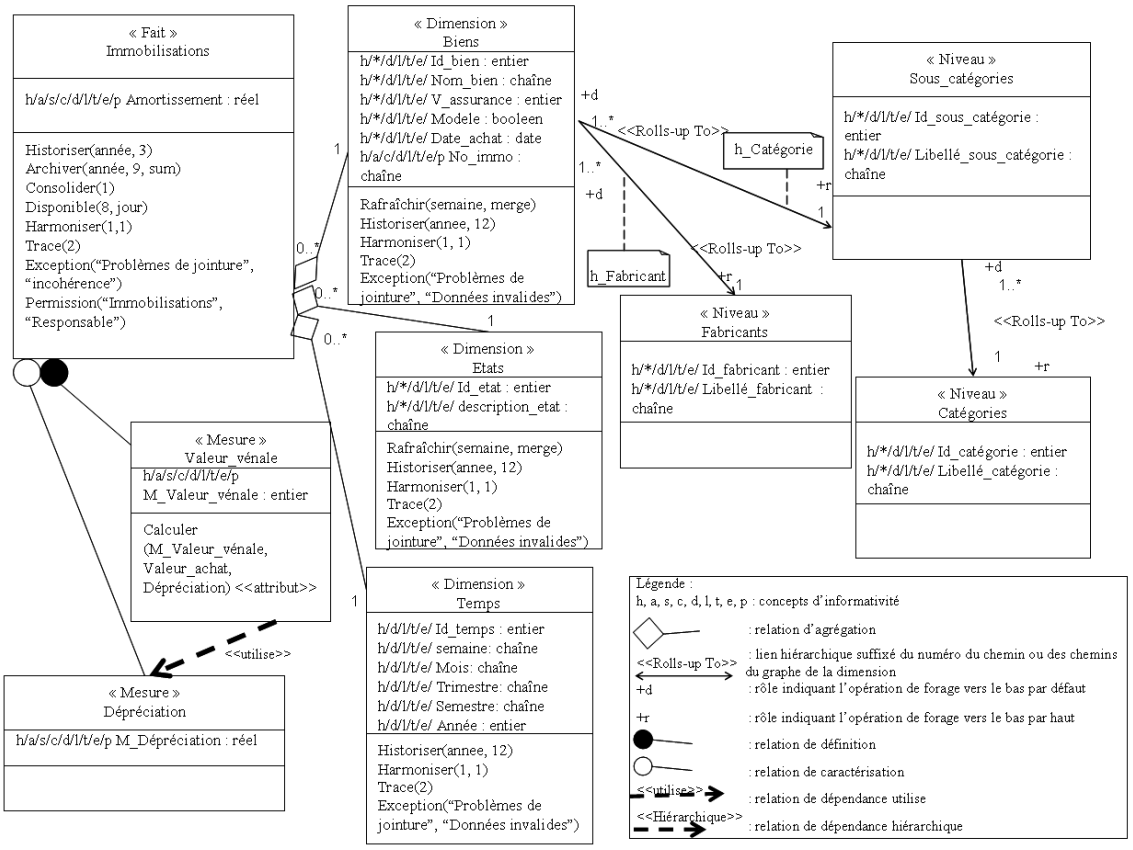


FIG. 5.15 – Schéma conceptuel multidimensionnel du magasin de données du projet immobilisations

5.5.2 Passage du diagramme décisionnel au modèle E-A

Pour définir le schéma des modules décisionnels basé sur le modèle entité-association à partir des classes-faits et des classes-dimensions d'analyse, nous proposons de dériver les entités et les associations du modèle entité-association suivant un ensemble de règles appelées PMEA (Passage au Modèle Entité-Association). Nous commençons par transformer les classes-faits d'analyse, puis nous transformons les classes-dimensions d'analyse.

Règles PMEA des classes-faits d'analyse

La transformation du diagramme décisionnel en un schéma entité-association est une opération moins intuitive car le modèle E-A ne respecte pas la représentation multidimensionnelle des données. Cependant, ce dernier est dédié à la manipulation des données. Par ailleurs, l'objectif d'expliciter la structure complexe des faits et des

mesures demeure. Ainsi, pour chaque classe-fait du diagramme décisionnel du SID, nous appliquons les règles suivantes :

- **PMEA1** : toute classe-fait d'analyse F_A est transformée en une entité E_F. La clé primaire de E_F est une clé de substitution Id_NomE_F,
- **PMEA2** : tout attribut de la classe-fait F_A qui est un iième (i>1) argument d'une opération d'attribut « Calculer() » est défini comme une entité E_M. Cette entité est à la source d'un traitement, nous la désignons par E_{M_s}. Cette entité contient un attribut dont le nom est préfixé par « M_ » qui correspond à la mesure. La clé primaire de E_s est une clé de substitution Id_NomE_{M_s},
- **PMEA3** : tout attribut de la classe-fait F_A qui est le premier argument d'une opération d'attribut « Calculer() », dont au moins un des autres argument est un attribut de la classe-fait, est défini comme une entité. Cette entité est la cible d'un traitement, nous la désignons par E_{M_c}. Cette entité contient un attribut dont le nom est préfixé par « M_ » qui correspond à la mesure. La clé primaire de E_M est une clé de substitution Id_NomE_{M_c},
- **PMEA4** : tout attribut de la classe-fait F_A qui ne vérifie pas les règles PMEA2 et PMEA3 forme les attributs de la classe-fait multidimensionnelle de conception F_C associée,
- **PMEA5** : une entité définissant une mesure E_M forme une association binaire, appelée « Définir_NomE_M », avec l'entité E_F définissant un fait F_C ayant les cardinalités E_M(1,N)-(1,1)E_F si cette entité E_M ne peut pas être nulle et si elle est indissociable de cette classe-fait,
- **PMEA6** : une entité définissant une mesure E_M forme une association binaire, appelée « Caractériser_NomE_M », avec l'entité E_F définissant un fait F_C ayant les cardinalités E_M(0,N)-(1,1)E_F si cette entité E_M peut être nulle ou si elle n'est pas nécessairement connu pour cette classe-fait,
- **PMEA7** : deux entités définissant des mesures E_{M_s} et E_{M_c} sont liées par une association binaire « ComposerE_{M_s}_E_{M_c} » de cardinalités E_{M_s}(0,1) (respectivement (1,1)) - (0,N)E_{M_c} si E_{M_c} est un argument de l'opération d'attribut « Calculer() » et les autres arguments sont exclusivement des classes-mesures,
- **PMEA8** : deux entités définissant des mesures E_{M_s} et E_{M_c} forme une association binaire « utiliseE_{M_s}_E_{M_c} » de cardinalités E_{M_s} (1,1) - (0,N)E_{M_c} si E_{M_c} est un argument de l'opération d'attribut « Calculer() » et les autres arguments ne sont pas exclusivement des classes-mesures. Autrement dit, l'association binaire est définie si les arguments sont des classes-mesures et des données OLTP non stockées dans le SID ou encore uniquement des données OLTP.

Règles PMEA des classes-dimensions d'analyse

Afin de représenter la structure complexe des dimensions, la transformation des classes-dimensions repose sur les règles suivantes :

- **PMEA10** : toute classe-dimension d'analyse D_A connectée à une classe-fait F_A et non cible d'une dépendance fonctionnelle est transformée en une entité E_D. La clé primaire de E_D est l'identifiant de la classe-dimension D_A,
- **PMEA11** : toute classe-dimension d'analyse D_A cible d'une dépendance fonctionnelle est transformée en une entité E_N. La clé primaire de E_N est l'identifiant de la classe-dimension D_A,
- **PMEA12** : tout attribut d'une classe-dimension d'analyse D_A transformée en une E_D est un attribut de cette entité,
- **PMEA13** : tout attribut d'une classe-dimension d'analyse D_A transformée en une E_N est un attribut de cette entité,
- **PMEA14** : toute relation « En Fonction » entre une classe-fait F_A et une classe-dimension D_A qui n'est pas cible d'une dépendance fonctionnelle est transformée en une association « Evaluer_NomE_D_NomE_F » de cardinalités E_D (0,N) - (0,1) E_F ou E_D (0,N) - (1,1) E_F suivant les caractéristiques du lien entre les deux entités,
- **PMEA15** : toute relation de dépendance fonctionnelle entre deux classes-dimensions D_A transformées en une E_D et E_{N1} est transformée en une association « Composer_NomE_N_NomE_D » de cardinalités E_D (1,1) - (0,N) E_N ou E_D (1,1) - (1,N) E_N suivant le projet,
- **PMEA16** : toute relation de dépendance fonctionnelle entre deux classes-dimensions D_A transformées en une E_N et E_{N1} est transformée en une association « Composer_NomE_{N1}_NomE_N » de cardinalités E_{N1} (*,1) - (*,N) E_N ou E_{N1} (*,N) - (*,N) E_N suivant le projet. L'astérisque « * » signifie 0 ou 1. Cette règle permet de définir les différents types de hiérarchies.

Ainsi, pour illustrer les règles de passage au modèle entité-association, nous considérons la conception d'un nouvel entrepôt de données au lieu d'un magasin de données à partir du diagramme décisionnel du SID (cf. figure 5.14). Suite à l'application de ces règles de passage, nous obtenons le schéma entité-association du nouvel entrepôt de données de l'organisation.

Exemple : le schéma conceptuel multidimensionnel du projet immobilisations dans le cas d'un nouveau module entrepôt de données est présenté dans la figure 5.16.

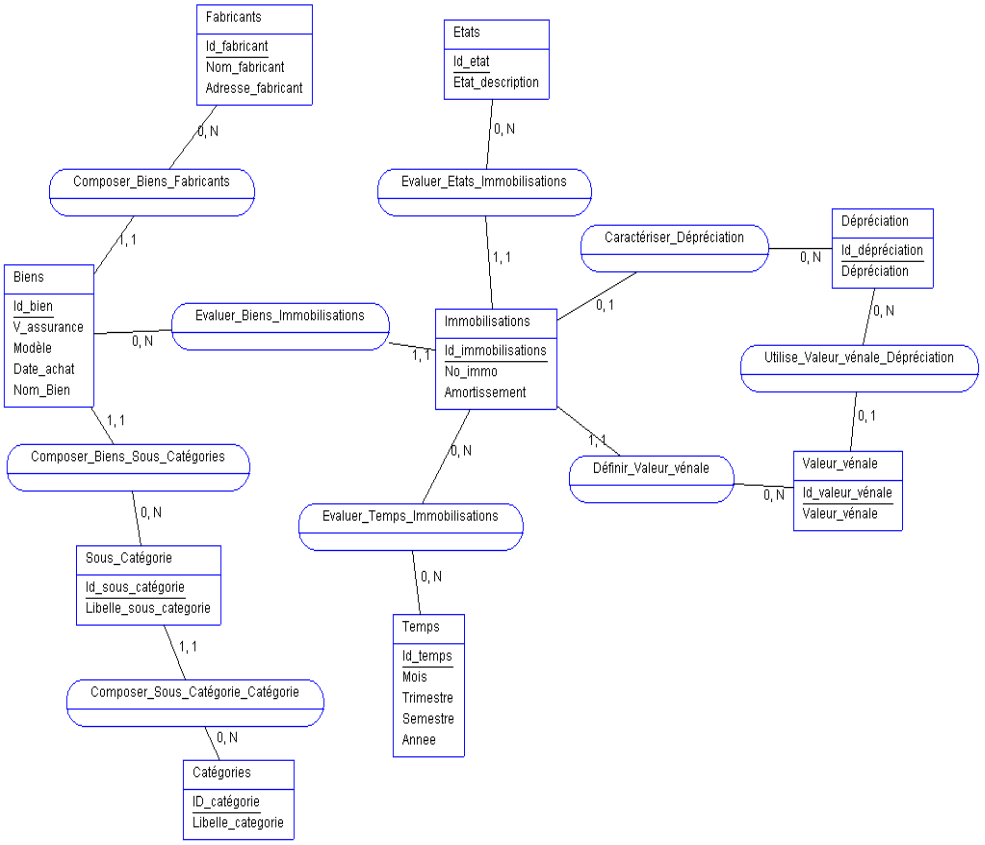


FIG. 5.16 – Schéma conceptuel entité-association de l'entrepôt de données du domaine des immobilisations

Le passage au modèle entité-association présente l'inconvénient de ne pas représenter la pertinence des agrégations et la dynamique du SID. Les traitements de dérivation et de préparation de données ne peuvent pas être définis dans le schéma entité-association du SID.

5.6 Conclusion

Les méthodes de conception existantes ne couvrent pas toutes les étapes de conception au regard du processus de développement proposé par [Srivastava and Chen 1999] car elles définissent un voire deux modules décisionnelles et elles ne guident pas le choix de l'architecture du SID. De plus, elles sont basées sur des modèles conceptuels multidimensionnels qui ne représentent que les données. Ces modèles ne permettent pas de représenter toutes les spécificités liées à la structure complexe des dimensions, des faits et des mesures ainsi que celles liées aux agrégations pertinentes et à la dynamique du SID.

Au cours de notre phase de conception du Trident décisionnel, à partir du graphe de propriétés des acteurs et du diagramme décisionnel du SID, nous guidons le choix de la combinaison de modules décisionnels et la conception de ces derniers. Nous proposons six critères pour qualifier les besoins et un processus automatique basé sur l'évaluation de ces six critères pour le choix de l'architecture adaptée au projet. Ces critères sont définis à partir des annotations de neuf propriétés des graphes de propriétés des groupes d'acteurs, tous groupes d'acteurs confondus.

La conception de ces modules consiste à définir le schéma entité-association ou le schéma multidimensionnel au regard de notre dichotomie des espaces des architectures décisionnelles. Nous proposons donc des règles de passage de notre diagramme décisionnel (modèle des besoins) :

- à notre modèle conceptuel multidimensionnel des données et des traitements (MMDT),
- au modèle entité-association.

Nous proposons de concevoir les modules qui reposent la modélisation multidimensionnelle avec notre modèle multidimensionnel conceptuel de données et de traitements (MMDT). Ce dernier est une extension du profil UML pour la modélisation multidimensionnelle proposée par [Luján-Mora et al. 2006] afin de bénéficier de la forte expressivité de ce profil et de la notoriété de la modélisation UML. Il est aussi une généralisation des spécificités représentées par les modèles existants afin de représenter toutes les propriétés définies dans les listes caractérisant les SID [Vassiliadis and Sellis 1999; Blaschka et al. 1999b; Abelló et al. 2001a; Rafanelli 2003].

Il représente les données au travers d'une constellation de classes-faits et de classes-dimensions. Les classes-faits sont caractérisées ou définies par des classes-mesures reliées. Les liens définis dans les règles de calcul à l'origine des mesures dérivées sont explicitement représentés entre les classes-mesures. Les classes-dimensions sont composées de classes-niveaux organisées suivant des hiérarchies simples, multiples, alternatives et non-strictes. Elles peuvent avoir différents rôles pour une même classe-fait et plusieurs éléments terminaux. Les classes-faits peuvent contenir des attributs correspondant à des dimensions dégénérées ou encore elles peuvent être définies comme dégénérées.

De plus, notre modèle MMDT représente les données et les traitements de ces classes de conception dans un seul schéma multidimensionnel car il utilise la partie opération des classes UML pour définir les traitements de dérivation et de préparation des données du SID. Ainsi, les décideurs et les concepteurs décisionnels ont une vue globale des concepts et des spécificités des SID qui leurs sont nécessaires, proche de la manière dont les décideurs se représentent les données. La représentation des impacts implicites et explicites sur le schéma conceptuel du SID contribue à faciliter les tâches d'analyse d'impacts qui ont lieu lors de la maintenance du système sont facilitées.

Ainsi, notre typologie des architectures valides, les deux ensembles de règles de passage ainsi que notre modèle conceptuel MMDT permettent de concevoir un SID fiable et adapté aux besoins des acteurs du projet décisionnel. De plus, notre méthode de développement étant basée sur un modèle de besoins et un modèle conceptuel multidimensionnel orienté-objet, elle favorise la capitalisation et la réutilisation de la connaissance au cours des projets. Ces propositions de réutilisation sont nécessaires mais pas suffisantes pour améliorer de manière significative les temps de développements. Nous proposons donc, dans le chapitre suivant des composants réutilisables.

Chapitre 6

Capitalisation dans un catalogue de patrons

Table des matières

6.1 Introduction 177
6.1.1 Problématique 178
6.1.2 Proposition 178
6.2 Présentation du catalogue 179
6.2.1 Découpage du catalogue 179
6.2.2 Formalisme P-SIGMA étendu 181
Formalisme P-SIGMA initial 181
Extension pour la gestion de la documentation 183
Extension pour la formalisation du contexte 185
6.3 Ingénierie pour la réutilisation 187
6.3.1 Patrons d'analyse 189
Patrons processus 190
Patrons produits 208
6.3.2 Patron de conception 212
Patrons processus 213
Patrons produits 220
6.4 Ingénierie par la réutilisation : mise en application de notre catalogue 221
6.4.1 Réutilisation des patrons au cours de phase d'analyse . . 223
6.4.2 Réutilisation des patrons au cours de phase de conception 238
6.5 Conclusion 241

Les besoins de méthodes permettant le développement rapide et fiable de SID (Systèmes d'Information Décisionnels) se font nettement ressentir depuis les années 1998 car plus d'une quinzaine de méthodes ont été proposées. La démocratisation de l'utilisation de tels systèmes au sein des organisations induit une profusion de contextes différents et variés à prendre en compte par les sociétés de services informatiques spécialisées et les départements décisionnels. Ce besoin est principalement motivé par un gain de compétitivité et de productivité. Une proposition de solution est de capitaliser la connaissance (en termes de démarche et de produit) et de faciliter la réutilisation de celle-ci afin de ne pas partir de zéro à chaque projet.

Dans le cadre de notre collaboration avec la société [I-D6 2007], nous avons constaté de nombreux points communs dans les processus de développement de projets décisionnels. Ce constat est corroboré par [Sen and Sinha 2005]. A chaque début de projet, il n'est pas rare que des tâches du processus d'ingénierie soient réutilisées de façon empirique par les concepteurs décisionnels. Cette réutilisation est opportuniste car elle dépend de l'expérience, de la capacité de corrélation et de la politique de réutilisation des concepteurs affectés à ce projet. De plus, ce type de réutilisation est généralement non documentée et non disponible pour tous. Cette re-définition quasi systématique du processus représente « une perte de temps » ainsi qu'un coût pour les organisations.

Ainsi, face à ces nombreux points communs dans les processus de développement des SID d'architectures variées, nous proposons de capitaliser la connaissance et l'expertise dans l'ingénierie des SID dans un catalogue de composants réutilisables pour l'analyse et la conception de SID et de faciliter leur réutilisation. Notre catalogue BIPAD (Business Intelligence Patterns for Analysis and Design) facilite la réutilisation des processus et des produits de notre méthode par la formalisation du contexte des composants et la gestion de la documentation intégrée aux composants.

Dans la section 6.1, nous introduisons la problématique de nos travaux et nos propositions liées à la réutilisation. Puis, dans la section 6.2, nous explicitons notre catalogue de patrons au travers de son découpage et du formalisme utilisé pour la représentation de nos composants. Dans la section 6.3, nous présentons les patrons processus d'analyse, les patrons produits d'analyse, les patrons processus de conception et les patrons produit de conception . Dans la section 6.4, nous déroulons une mise en application de notre catalogue de patrons. Enfin dans la section 6.5, nos concluons sur nos propositions liées à la réutilisation dans le domaine de l'ingénierie des SID.

6.1 Introduction

Nos objectifs par rapport à cette thématique sont les suivants :
– systématiser la réutilisation au sein des projets décisionnels,

- faciliter la réutilisation aussi bien par des concepteurs décisionnels débutants que des concepteurs décisionnels experts,
- améliorer la communication au cours des projets,
- gérer la traçabilité de la documentation entre les acteurs du projet,
- corriger les vocabulaires hétérogènes des concepteurs décisionnels,
- faire face aux changements d'équipe inhérents et croissants au sein des sociétés de services spécialisées dans le décisionnel.

6.1.1 Problématique

Les travaux relatifs à la capitalisation et à la réutilisation dans le domaine des SID sont à leurs débuts. Seules deux équipes ont fait des propositions relatives à la réutilisation dans le domaine des SID. Ces deux propositions fournissent des patrons produits d'analyse ou de conception, mais ces patrons produits concernent que certains concepts tels que les dimensions ou encore ils concernent qu'une partie d'une phase du processus d'ingénierie. De plus, elles ne facilitent pas et elle ne systématise pas la réutilisation car les concepteurs ne sont pas guider pour la mise en place du SID *via* des patrons processus. Ces patrons sont des blocs indépendants où les relations entre les patrons ne sont pas définies. Les auteurs ne proposent pas d'outil pour concevoir un SID par réutilisation.

Comme mis en avant dans notre état de l'art, des composants produits qui couvrent toute l'analyse et la conception ainsi que des composants processus pour guider et pour faciliter la mise en oeuvre de ces phases sont nécessaires.

Ainsi, notre problématique est **« comment capitaliser la connaissance en terme de produit et de processus afin de systématiser la réutilisation et la traçabilité au sein des projets ? Comment définir cette capitalisation afin que tout concepteur dispose des supports pour la mise en place rapide de SID fiables et pertinents ? »**

6.1.2 Proposition

La thématique de la réutilisation dans l'ingénierie des systèmes d'information classiques a atteint une maturité confirmée. De ce fait, nous nous inspirons des propositions relatives à cette dernière afin de répondre à notre problématique et atteindre nos objectifs liés à cette thématique. Les patrons ont fait l'objet de nombreux travaux. Ce sont des composants dont le contenu est disponible et adaptable par les concepteurs décisionnels, soit des boîtes blanches, avec un degré de variabilité qui favorise une adaptation simple. Nous définissons donc un catalogue de patrons.

Les patrons d'analyse et les patrons de conception de notre catalogue sont définis suivant l'approche orientée problème des composants par des triplets <Problème, So-

lution, Contexte> [Alexander. 1977; Gamma et al. 1995]. Cette approche distingue la spécification du problème de sa réalisation. Le problème est un élément de démarche à réaliser ou un produit à définir. La solution est une démarche ou un modèle représenté respectivement par des algorithmes, des diagrammes d'activités, des modèles de documents ou des modèles de données. Enfin, le contexte définit la situation pour laquelle la solution décrite dans le composant est utilisable.

Comme le soulignent les travaux de [Barbier et al. 2004], la réutilisation de composants permet de réduire les coûts et les délais de conception, d'implantation et de maintenance si elle s'allie à la traçabilité. Nous utilisons le formalisme P-SIGMA [Conte et al. 2001] pour représenter nos patrons. Cependant, ce dernier ne formalise pas l'expression du contexte du patron et il ne gère pas la documentation projet. Ainsi, nous proposons d'étendre ce formalisme par la formalisation de l'expression du contexte des patrons et l'intégration de la gestion de la documentation.

6.2 Présentation du catalogue

Définition 6.1 (Un catalogue de patrons) *Un catalogue de patrons est un ensemble de patrons qui permet de résoudre des problèmes liés à une même problématique dans un contexte donné [Gzara 2000].*

Notre catalogue permet de résoudre l'ensemble des problèmes liés à l'analyse et à la conception des SID. Il est composé d'une trentaine de patrons. Il capitalise les produits et les processus de notre méthode de développement des SID. Il est composé de quatre catégories de patrons définies à partir des concepts de processus et de produit, soit les patrons processus d'analyse, les patrons produits d'analyse, les patrons processus de conception et les patrons produits de conception . Il guide le concepteur décisionnel *via* les relations inter-patrons existantes [Rieu 1999] durant les deux premières phases de l'ingénierie des SID, ce qui facilite et contribue à systématiser la réutilisation des patrons.

Nous avons défini nos patrons à partir des points communs et de la variabilité d'un échantillon varié d'une quinzaine de projets qui ont fait preuve de fiabilité et de pérennité lors de mise en oeuvre chez les organisations clientes de la société I-D6. Ces projets ont porté sur de nombreux domaines et métiers qui requièrent la mise en place de SID.

6.2.1 Découpage du catalogue

A partir de la documentation de ces projets, tels que les cahiers des charges, les documents de spécifications détaillées et les documents de cadrage, nous avons

mis en exergue les points communs des différents développements. Nous avons non seulement identifié les cinq tâches de [Sen and Sinha 2005], mais nous avons trouvé d'autres processus et produits communs d'un projet SID à l'autre. Ces éléments de développement communs constituent le noyau de notre catalogue. Afin de faciliter l'utilisation des patrons, nous avons défini des patrons de granularité basée sur les quatre niveaux d'éléments processus de notre démarche du Trident décisionnel et le niveau produit définis comme suit :

- les quatre niveaux de patrons processus :
  - l'étape : un projet SID se déroule en deux étapes qui sont « mettre en place le SID » et « maintenir le SID »,
  - la phase : un élément de granularité plus fine que l'étape,
  - la tâche : un élément qui compose une phase,
  - la sous-tâche : un ensemble de sous-tâches constitue une tâche.
- le niveau de patron produit :
  - le modèle : les produits de notre démarche sont des modèles de documents, des modèles de tableaux, des modèles de graphes, des modèles conceptuels de données et des modèles conceptuels de données et de traitements.

La figure 6.1 présente le diagramme de collaboration de patrons de l'étape « Mettre en place un SID ». Ce diagramme définit les relations entre les principaux patrons des phases d'analyse et de conception de granularité étape, phase, tâche, sous-tâche et produit. Ce diagramme utilise l'occurrence de collaboration que suggère [Fowler 2004] pour la représentation des patrons processus. Nous utilisons une représentation propriétaire pour distinguer rapidement les patrons produits.

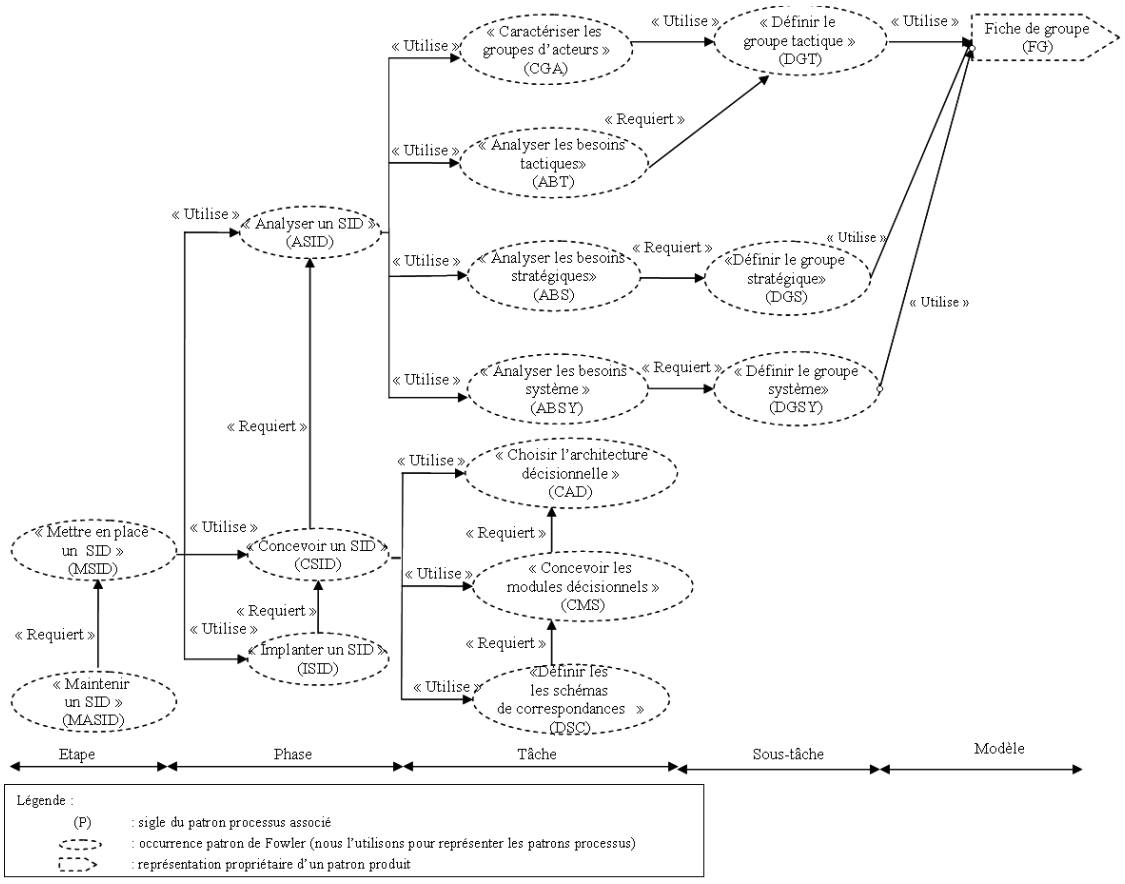


FIG. 6.1 – Diagramme de collaboration partiel de notre catalogue

Exemple : pour analyser le SID, le concepteur décisionnel doit d'abord caractériser les groupes d'acteurs (patron CGA) avant d'analyser en parallèle les besoins tactiques (patron ABT), les besoins stratégiques (patron ABS) et les besoins systèmes (patron ABSY). Les patrons ABT, ABS et ABSY requièrent des patrons utilisés par le patron CGA tel que le patron « Définir le groupe tactique » (DGT).

6.2.2 Formalisme P-SIGMA étendu

Formalisme P-SIGMA initial

Comme pour les systèmes d'information classique [Guzelian 2004], tous les besoins des acteurs du SID ne peuvent pas être définis dès le début du processus d'ingénierie. De ce fait, suivant l'itération de développement du SID, les patrons mis en oeuvre diffèrent. Il importe de prendre en compte la variabilité des contextes pour lesquels un patron peut être utilisé. Ce contexte est spécifié dans la représentation du patron.

La représentation d'un patron nécessite l'utilisation d'un formalisme. Selon [Duong 2000], il existe deux types de formalismes : narratif et structuré. Le premier repose sur l'expression du patron dans un texte en langage naturel tandis que le second est organisé par rubrique au sein d'un formulaire. Nous privilégions le formalisme structuré en vue de systématiser la réutilisation au cours de l'ingénierie des SID.

Le formalisme P-SIGMA [Conte et al. 2001] est une tentative d'unification des formalismes structurés qui ont été proposés [Coad 1992; Gamma et al. 1995; Buschmann et al. 1996]. Nous l'avons choisi comme la base de notre représentation car il intègre l'expression des aspects produits et processus et un grand nombre de relations inter-patrons. De plus, il facilite la sélection, la réutilisation et l'organisation des composants. Cet avantage est basé sur le regroupement de l'information en trois parties. Chaque partie regroupe des rubriques, obligatoires ou non, comprenant un ou plusieurs champs. Ainsi, le nombre de rubriques par partie du formalisme varie d'un patron à l'autre suivant les informations associées. Les parties sont les suivantes :

- interface : elle permet la sélection du patron,
- réalisation : elle exprime la solution proposée par le patron,
- relation : elle permet d'organiser les relations entre patrons.

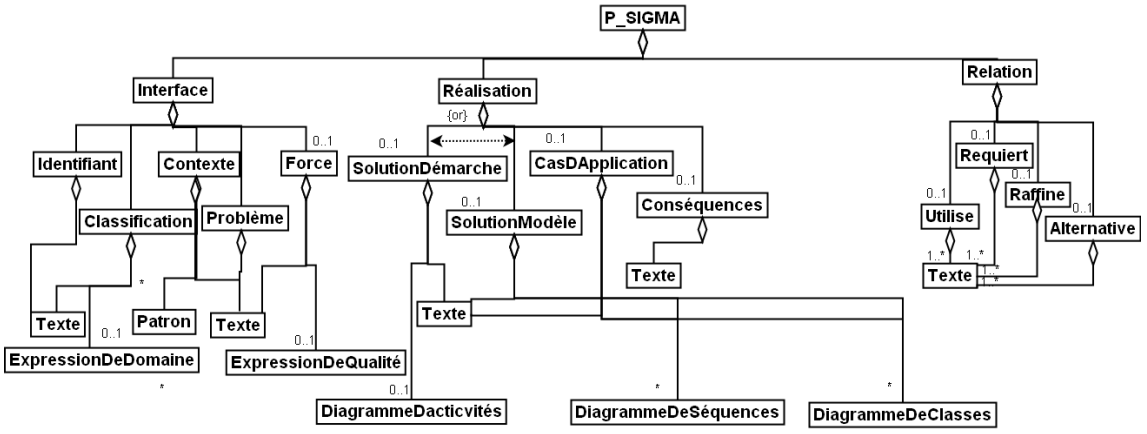


FIG. 6.2 – Diagramme de classes UML du formalisme P-SIGMA

Cependant, ce formalisme ne gère pas la documentation projet. La définition du contexte qu'il offre ne favorise pas une réutilisation systématique des patrons. En ce sens, dans les deux sections suivantes, nous présentons la proposition pour la gestion de la documentation et celle pour la formalisation du contexte. Nous utilisons le sigle du patron qui permet de l'identifier aussi. Nous les ajoutons donc dans le formalisme P-SIGMA.

Extension pour la gestion de la documentation

Le formalisme P-SIGMA permet de définir de nombreuses informations liées à la réalisation de la solution-démarche ou de la solution-modèle du patron. La documentation projet est indispensable dans un projet décisionnel car de nombreux acteurs de types et de domaines d'activités différents interviennent (cf. le diagramme de classe des acteurs présenté dans la figure 3.2. L'intégration de la gestion des documents dans la spécification des patrons favorise une meilleure coordination entre les différents acteurs et un vocabulaire commun. Elle implique aussi la présence de supports approuvés pour une future maintenance.

Nous proposons d'étendre le formalisme P-SIGMA pour assurer la traçabilité de la documentation par l'ajout de deux rubriques « Documents sources » et « Documents cibles ». Ces rubriques précisent la documentation liée au problème traité par le patron (cf. figure 6.2) comme suit :

- Documents sources : la documentation qui sera utilisée au cours la solution-démarche afin d'obtenir la solution-modèle,
- Documents cibles : la documentation résultant de l'application de la solution-démarche.

Tout d'abord, il est possible d'envisager la spécification des documents sources au niveau de la rubrique contexte de la partie interface (cf. figure 6.2). Mais cela reviendrait à rompre les regroupements qui font la force du formalisme initial P-SIGMA. De plus, la gestion de la documentation est une tâche parallèle au processus de développement qui ne peut le contraindre en raison des impératifs de productivité des organisations. Elle est liée à la résolution du problème posé par le patron. Ainsi, les rubriques « Documents sources » et « Documents cibles » se rapportent à la partie « Réalisation » du patron. La figure 6.3 présente le formalisme P-SIGMA étendu sous forme tabulaire que nous utilisons pour la définition de notre catalogue.

Parties	***Rubriques***	***Champs***
INTERFACE	Sigle	Le sigle du patron
	Nom (ou encore identifiant)	le problème résolu par ce patron
	Classification	le type et la portée du patron
	Contexte	les situations dans lesquelles le problème est résolu et/ou le patron est appliqué
	Problème	la description du problème à résoudre
	Motivation (ou classification)	un scénario d'application décrit de manière textuelle ou graphique
	Forces	les raisons de proposer le patron
REALISATION	Solution-modèle	le modèle de solution proposé (diagrammes UML, schéma E-A et/ou modèles multidimensionnels, …)
	Solution-démarche	la démarche de solution proposée pour résoudre le problème. (diagramme UML, …)
	Cas d'applications	des exemples d'applications de la solution modèle
	Conséquences d'application	les limites et les bénéfices de l'application de la solution
	Documents sources	la documentation qui sera utilisée au cours de la solution démarche afin d'obtenir la solution modèle
	Documents cibles	la documentation résultant de l'application de la solution démarche
RELATION	Utilise	la liste des patrons «utilisés» par le patron considéré. Un patron A utilise un patron B, si une partie des problèmes posés par A peuvent être résolus en partie ou complètement par B
	Requiert	la liste des patrons «requis» par défaut par le patron considéré. Un patron A requiert un patron B, si l'application de B est un pré requis à l'application de A
	Raffine	la liste des patrons «raffinés» par le patron considéré. Un patron A raffine un patron B, si le problème posé par A est une spécialisation de celui posé par B
	Alternative	la liste des patrons qui constituent une alternative au patron considéré. Un patron A est une alternative d'un patron B, si A a le même problème que B mais propose une solution différente.

FIG. 6.3 – Formalisme P-SIGMA étendu

Extension pour la formalisation du contexte

Dans le formalisme P-SIGMA, le contexte est une rubrique définie par une agrégation de patrons et de textes exprimés en langage naturel. Cette définition du contexte rend difficile la recherche du composant lors de la réutilisation [Guzelian 2004] ; ce qui contraint la réutilisation systématique des patrons.

La rubrique « Contexte » du formalisme P-SIGMA indique les situations dans lesquelles le problème est résolu ou encore celles dans lesquelles le patron peut être est appliqué ainsi que les différents patrons requis par ce patron. Les informations apportées par cette rubrique sont donc :
- les conditions d'utilisation de ce patron,
- les patrons requis pour l'utilisation de ce patron.

[Guzelian 2004] utilisent des modèles de buts pour formaliser le contexte, mais cette proposition de solution n'est pas adéquate pour notre catalogue car la granularité des patrons étant basée sur celle de notre démarche, le but de développement est défini par l'identifiant du patron même. De plus, dans notre catalogue de patrons, les conditions d'utilisation d'un patron dépendent de :
- x : l'itération en cours du processus de développement,
- y : le nombre d'utilisations de ce patron au cours du processus de développement complet du SID.

Les patrons p_r requis par un patron p_s dépendent aussi des paramètres sus cités. Les conditions d'utilisation d'un patron p_s et les conditions dans lesquelles ce patron p_s requiert un patron p_r ne sont pas nécessairement identiques.

Exemple : nous avons défini un cycle de vie basé sur deux itérations de la phase d'analyse au cours du cadrage et du prototypage pour rapprocher les besoins des trois groupes d'acteurs. Puis, seules les phases de conception et d'implantation du SID sont réitérées pour modifier le schéma du SID et les schémas de correspondances entre les sources et le SID. Autrement dit, au cours de la deuxième itération en termes de patrons, le patron CSID « Concevoir le SID » est utilisable et il requiert le patron ASID « Analyser le du SID ». Cependant, au cours des itérations ultérieures, le patron CSID est toujours applicable mais il ne requiert pas ASID.

A partir de cet exemple, nous pouvons aussi noter que la relation « requiert » entre un patron p_s et un p_r peut être optionnelle. De même, la relation « requiert » entre un patron p_s et un ensemble de patrons p_{r_i} peut être exclusive. Un patron p_s qui requiert par défaut nb patrons (avec nb>1) suivant la rubrique « Requiert » du formalisme P-SIGMA étendu peut requérir exclusivement p_{r_i} (avec $1 \leq i < nb$) suivant l'itération du processus de développement et du nombre d'utilisations.

Afin de formaliser dans quelles situations un patron p_s peut être utilisé et dans quelles conditions il requiert un patron p_r, nous proposons deux fonctions $\mathbf{u}_s$ et $\mathbf{r}_s$.

Soit $\mathbf{u}_s$ la fonction qui indique si les conditions d'utilisation d'un patron p_s sont vérifiées. Cette fonction est définie comme suit :

$$\begin{array}{ll} u_s : \mathbb{N}^2 \rightarrow \{0,1\} & \\ (x,y) \mapsto & u_s(x,y)=0 \\ & \text{ou} \\ & u_s(x,y)=1 \end{array}$$

Dans le cas où $u_s(x,y)=1$, p_s est utilisable sinon, p_s n'est pas utilisable.

Exemple : le contexte du patron MSID « Mettre en place un SID » qui est utilisable indépendamment de l'itération en cours et du nombre de fois qu'il a été utilisé au cours du processus de développement. Son contexte est défini comme suit :

$$\forall (x,y) \in \mathbb{N}^2_+,\ u_s(x,y)=1$$

Soit $\mathbf{r}_s$ la fonction qui indique si un patron p_s requiert un autre patron p_r. Elle indique dans quelles situations un patron p_r est requis par un patron p_s. Cette fonction est définie comme suit.

Soit $\mathbb{P}$: l'ensemble des patrons de notre catalogue de patrons eBIPAD. MSID est un élément de $\mathbb{P}$.

$$\begin{array}{ll} r_s : \mathbb{P} \text{ x } \mathbb{N} \text{ x } \mathbb{N} \rightarrow \{0,1\} & \\ (p,x,y) \mapsto & r_s(p_r,x,y)=0 \\ & \text{ou} \\ & r_s(p_r,x,y)=1 \\ \text{avec} & p_r : \text{patron requis par défaut par } p_s. \end{array}$$

Dans le cas où $r_s(p_r,x,y)=1$, le patron p_s requiert p_r sinon, p_s ne requiert pas p_r.

Exemple : le contexte du patron ISID « Implanter le SID » qui est utilisable dans tous les cas et qui requiert toujours le patron CSID « Concevoir le SID » est défini comme suit :

$$\exists \text{ CSID} \in \mathbb{P} \backslash \forall x \in \mathbb{N}, \forall y \in \mathbb{N},\ u_s(x,y)=1 \text{ et } r_s(\text{CSID},x,y)=1$$

Ainsi, le contexte C_s d'un patron p_s est défini au cours du cycle de vie du SID par une agrégation de fonctions u_s et r_s comme présenté dans la figure 6.4. Dans

notre catalogue, pour l'expression du contexte des patrons, nous nous intéressons uniquement aux cas où les fonctions u_s et r_s sont égales à 1, en l'occurrence les cas où le patron est utilisable et où ce dernier requiert un patron p_r.

La formalisme P-SIGMA étendu résultant de ces extensions est présenté dans la figure 6.4.

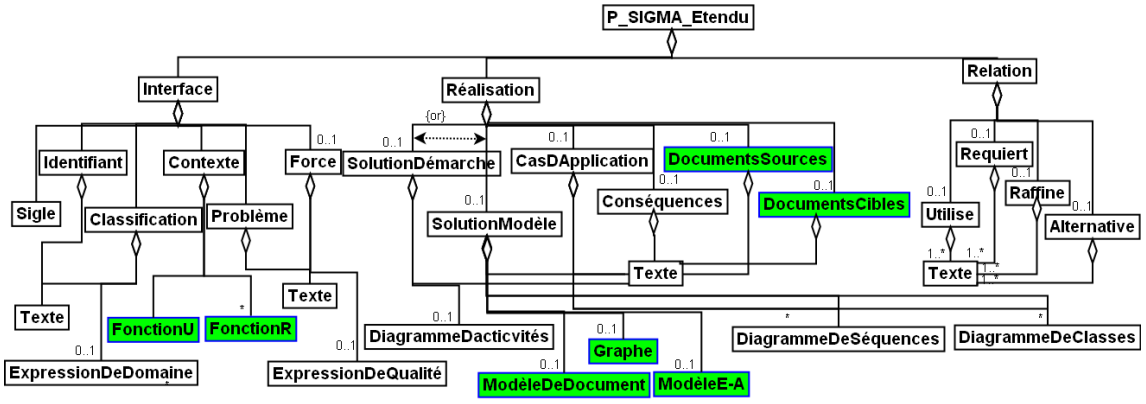


FIG. 6.4 – Diagramme de classes UML du formalisme P-SIGMA étendu

6.3 Ingénierie pour la réutilisation

Le catalogue est introduit par le patron MSID « Mettre en place un SID ». Ce patron guide les tâches de développement du SID liées à l'étape initiale de l'ingénierie. La formalisation du patron introducteur MSID est donnée dans la figure 6.5.

Notre méthode est basée sur un cycle de vie itératif incrémental comprenant une itération pour le cadrage du projet. Le cadrage peut se dérouler sur plusieurs itérations de la phase d'analyse. Comme, nous l'avons précisé au chapitre 3, le cadrage est généralement réalisé en une itération de la phase d'analyse. Afin que notre méthode soit générique, nous considérons un nombre k d'itérations de la phase d'analyse au cours pour du cadrage.

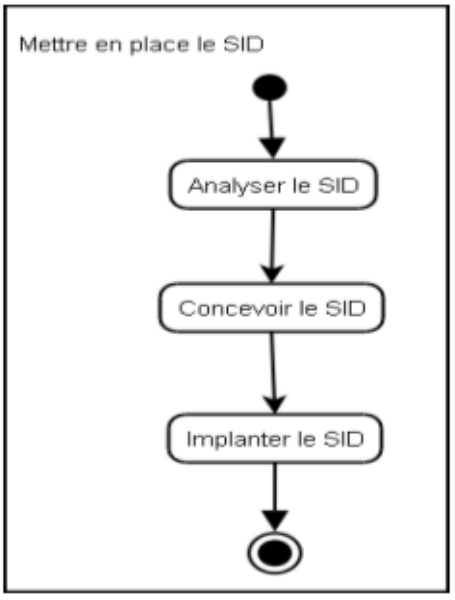
Parties	***Rubriques***	***Champs***
INTERFACE	Sigle	MSID
	Nom	Mettre en place un SID
	Classification	analyse $\wedge$ conception $\wedge$ implantation $\wedge$ processus $\wedge$ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, u_s(x, y) = 1$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet u_s : la fonction qui indique les conditions dans lesquelles ce patron est applicable
	Problème	Comment mettre en place un système d'information décisionnel ? La problématique du stratégique est un facteur de pérennité, de rentabilité et de prospérité d'une organisation. Un système décisionnel est un système d'information qui permet de mesurer les indicateurs clés pour connaître les événements et ainsi prévoir les tendances. Face à cette problématique, il convient de mettre en place un SID
	Motivation	Une méthode de développement d'un SID s'articule autour de trois phases, soit : - l'analyse du SID, - la conception du SID, - l'implantation du SID.
	Forces	Ce patron guide le concepteur dans la mise en place d'un système d'information décisionnel.
REALISATION	Solution-démarche	La solution est la mise en oeuvre du diagramme d'activités UML ci-dessous. A une activité est associée un patron. 
	Solution-modèle	
	Cas d'applications	Tout domaine critique où la prise de décision doit être rapide et fiable
	Conséquences d'application	
	Documents sources	Le cahier des charges
	Documents cibles	
RELATION	Utilise	« Analyser un SID », « Concevoir un SID » et « Implanter un SID »
	Requiert	
	Raffine	
	Alternative	

FIG. 6.5 – Patron « Mettre en place un SID » (MSID)

Ce patron MSID utilise directement des patrons processus d'analyse et des processus patrons de conception. A partir de ces derniers, la mise en place d'un SID requiert des patrons produits d'analyse et des patrons produits de conception. Ainsi, dans les deux sections suivantes, nous présentons ces quatre types de patrons.

6.3.1 Patrons d'analyse

Les patrons d'analyse permettent de capitaliser les fragments de démarche et de produits liés à la phase d'analyse de notre Trident décisionnel. Outre les points communs notés d'un projet décisionnels à un autre, nous avons constaté des points communs entre les tâches de notre démarche dus à la nécessité de comparaison et de rapprochement des besoins des trois groupes d'acteurs. En ce sens, nous définissons des patrons qui sont communs à différentes tâches. Le diagramme de collaboration de patrons de la phase « Analyser un SID » suivant notre démarche du Trident décisionnel est présenté dans la figure 6.6.

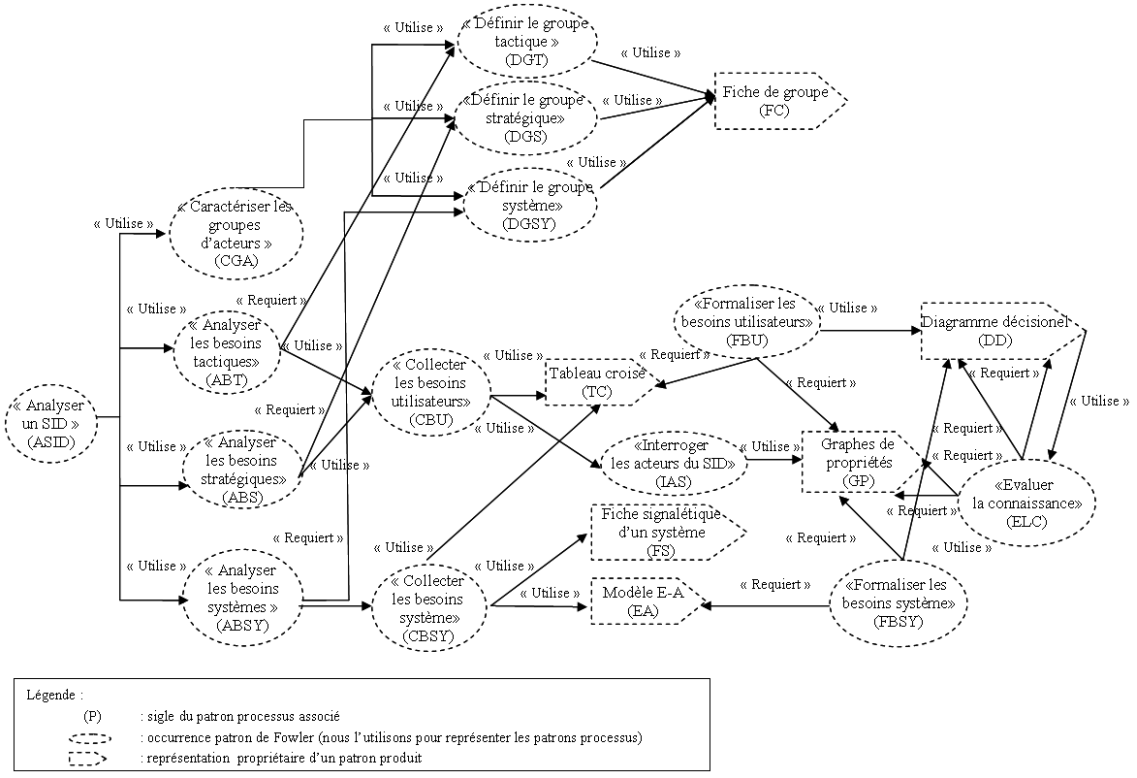


FIG. 6.6 – Diagramme de collaboration de patrons d'analyse

Patrons processus

Un patron processus d'analyse permet de représenter les processus liés à l'analyse d'un SID. Il met en avant les activités liées au problème étudié. Nous présentons successivement les patrons suivant le diagramme de collaboration des patrons d'analyse dans les figures ci-dessous :

Parties	***Rubriques***	***Champs***
INTERFACE	Sigle	ASID
	Identifiant	Analyser un SID
	Classification	analyse $\wedge$ processus $\wedge$ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, (x \leq k, y \leq k \Rightarrow (u_s(x, y) = 1, r_s(MSID, x, y) = 1))$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable r_s : la fonction qui indique les conditions dans lesquelles ce patron requiert le patron MSID
	Problème	Comment analyser un système d'information décisionnel ? L'analyse du SID correspond à l'étude de la mise en place du SID. Elle est réalisée afin d'évaluer la faisabilité du projet, en l'occurrence le rapprochement entre les besoins
	Motivation	L'analyse du SID consiste à caractériser les groupes d'acteurs et analyser leurs besoins en vue de les rapprocher.
	Forces	Ce patron guide le concepteur pour l'analyse d'un système d'information décisionnel. Il définit le contexte d'exploitation du SID
REALISATION	Solution-démarche	La solution consiste à la mise en oeuvre de ce diagramme d'activités. Analyser le SID Caractériser les groupes d'acteurs Analyser les besoins tactiques Analyser les besoins stratégiques Analyser les besoins système Evaluer la connaisance utilisateurs Evaluer la connaisance de l'environnement
	Solution-modèle	
	Cas d'applications	Au début d'un projet décisionnel ou quand les besoins n'ont pas été rapprochés ou lorsque que les besoins ont été modifiés
	Conséquences d'application	Définition des acteurs et de leurs besoins
	Documents sources	Le cahier des charges
	Documents cibles	
RELATION	Utilise	« Caractériser les groupes acteurs », « Analyser les besoins tactiques », «Analyser les besoins stratégiques », « Analyser les besoins systèmes », « Evaluer la connaissance »
	Requiert	«Mettre en place un SID »
	Raffine	
	Alternative	

FIG. 6.7 – Patron « Analyser un SID » (ASID)

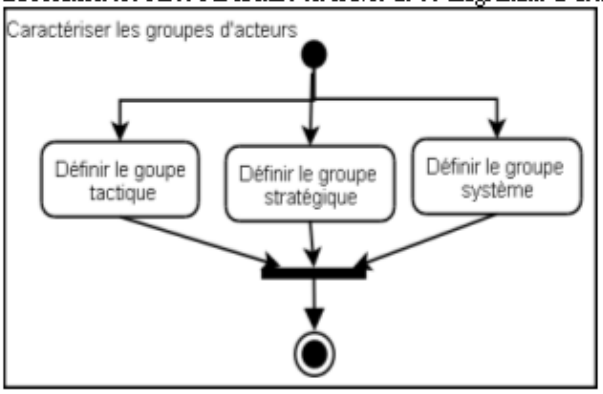
Parties	Rubriques	Champs
INTERFACE	Sigle	CGA
	Nom	Caractériser les groupes d'acteurs
	Classification	analyse ∧ processus ∧ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \leq k, y = 0 \Rightarrow u_s(x, y) = 1$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable
	Problème	Quels sont les rôles des différents utilisateurs qui interviennent sur le projet? Comment déterminer ces rôles ?
	Motivation	Suivant le niveau hiérarchique et le rôle d'un utilisateur dans le développement et l'exploitation du SID, les caractéristiques et l'impact de ses besoins varient.
	Forces	Ce patron aide le concepteur à caractériser les acteurs du SID.
REALISATION	Solution-démarche	La solution est basée sur la mise en œuvre de ce diagramme d'activités : 
	Solution-modèle	
	Cas d'applications	Définir les membres de chaque groupe
	Conséquences d'application	Les acteurs ayant les doubles casquettes qui feront la passerelle entre les groupes sont déterminés
	Documents sources	Le cahier des charges
	Documents cibles	
RELATION	Utilise	« Définir le groupe tactique », « Définir le groupe stratégique », « Définir le groupe système»
	Requiert	
	Raffine	
	Alternative	

FIG. 6.8 – Patron « Caractériser las groupes d'acteurs » (CGA)

Parties	Rubriques	Champs
INTERFACE	Sigle	DGT
	Nom	Définir le groupe tactique
	Classification	analyse $\wedge$ processus $\wedge$ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \leq k, y = 0 \Rightarrow u_s(x,y) = 1$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable
	Problème	Comment définir les membres du groupe tactique ?
	Motivation	Le groupe tactique comprend des acteurs qui sont des utilisateurs directs du SID, en l'occurrence les analystes et les décideurs liés spécifiquement à un métier relatif au projet
	Forces	Ce patron aide le concepteur pour la définition des besoins du groupe tactique.
REALISATION	Solution-démarche	A partir du document de cadrage, il faut identifier les collaborateurs métiers qui ont une connaissance fine des métiers relatifs au projet
	Solution-modèle	
	Cas d'applications	Pour la définition des acteurs liés à la gestion de systèmes OLTP existants
	Conséquences d'application	
	Documents sources	Le cahier des charges
	Documents cibles	La fiche du groupe système à inclure dans le dossier de cadrage
RELATION	Utilise	« Fiche de groupe »
	Requiert	
	Raffine	
	Alternative	

FIG. 6.9 – Patron « Définir le groupe tactique » (DGT)

Parties	Rubriques	Champs
INTERFACE	Sigle	DGS
	Nom	Définir le groupe stratégique
	Classification	analyse ∧ processus ∧ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \leq k, y = 0 \Rightarrow u_s(x, y) = 1$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable
	Problème	Comment définir les membres du groupe pilote ?
	Motivation	La conduite d'un projet décisionnel implique la participation de nombreux acteurs. Ils peuvent assurer plusieurs rôles suivant la taille de l'organisation. Il convient de connaître l'interlocuteur auquel il faut s'adresser pour un problème donné.
	Forces	Ce patron aide le concepteur pour la définition des besoins du groupe de stratégique dans les grands groupes et les PME car les projets décisionnels concernent principalement les grands groupes
REALISATION	Solution-démarche	A partir du cahier des charges, il faut identifier un ou plusieurs représentants de la direction, qui définissent dans le choix des priorités de l'organisation, puis. si c'est une PME alors - identifier un ou des représentants métiers concernés qui ont une connaissance globale des métiers, - identifier un ou des représentants du service informatique qui ont une connaissance globale des systèmes sources, - identifier un ou des représentants du service informatique décisionnel ou de la société de service spécialisée dans le décisionnel. sinon - identifier un ou des représentants métiers concernés qui ont une connaissance globale des métiers, - identifier un ou des responsables système qui s'occuperont des problèmes techniques inhérents au développement logiciel et de la future maintenance du SID, - identifier un responsable réseaux, s'il est distinct du responsable système, qui va s'occuper de l'échange des données, - identifier un ou des responsables base de données ayant une connaissance globale des systèmes sources, - identifier un ou des représentants du service informatique spécialisé dans décisionnel ou de la société de service spécialisée dans le décisionnel. finsi Remplir la fiche de groupe pour le groupe stratégique
	Solution-modèle	
	Cas d'applications	Définition du groupe stratégique dans les grands groupes ou les PME.
	Conséquences d'application	
	Documents sources	Le cahier des charges
	Documents cibles	La fiche du groupe stratégique à inclure dans le dossier de cadrage
RELATION	Utilise	« Fiche de groupe »
	Requiert	
	Raffine	
	Alternative	

FIG. 6.10 – Patron « Définir le groupe stratégique » (DGS)

Parties	Rubriques	Champs
INTERFACE	Sigle	DGSY
	Nom	Définir le groupe système (DGSY)
	Classification	analyse $\wedge$ processus $\wedge$ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \leq k, y = 0 \Rightarrow u_s(x,y) = 1$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable
	Problème	Comment définir les membres du groupe système ?
	Motivation	Les responsables des sources de données sont aussi bien les membres de l'équipe de maître d'œuvre que la composante sources de données de l'équipe de maître d'ouvrage.
	Forces	Ce patron aide le concepteur pour la définition des besoins du groupe qui gère les systèmes sources et les modules décisionnels existants du SID.
REALISATION	Solution-démarche	A partir du cahier des charges, il faut : - identifier les collaborateurs de la maîtrise d'œuvre et de la maîtrise d'ouvrage qui gèrent les systèmes OLTP de l'organisation - identifier des collaborateurs du service informatique qui ont une connaissance fine des systèmes sources, - identifier des collaborateurs du service informatique décisionnel ou de la société de service spécialisée dans le décisionnel.
	Solution-modèle	
	Cas d'applications	Pour la définition des acteurs liés à la gestion de systèmes OLTP et du SID existant
	Conséquences d'application	
	Documents sources	Le cahier des charges
	Documents cibles	La fiche du groupe système à inclure dans le dossier de cadrage
RELATION	Utilise	« Fiche de groupe »
	Requiert	
	Raffine	
	Alternative	

FIG. 6.11 – Patron « Définir le groupe système » (DGSY)

L'analyse des besoins du groupe tactique se réalise par l'instanciation des patrons suivants :

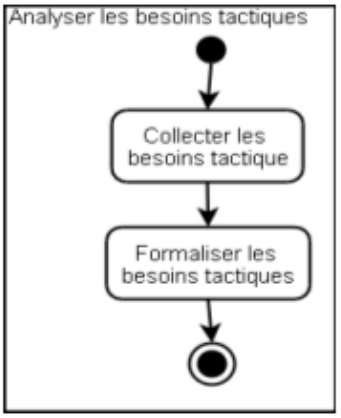
Parties	Rubriques	Champs
INTERFACE	Sigle	ABT
	Nom	Analyser les besoins tactiques
	Classification	analyse $\wedge$ processus $\wedge$ SID,
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, (x \leq k \Rightarrow u_5(x,y) = 1), (y = 0 \Rightarrow r_5(DGT, x, y) = 1)$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_5 : la fonction qui indique les conditions dans lesquelles ce patron est utilisable r_5 : la fonction qui indique les conditions dans lesquelles ce patron requiert le patron DGT
	Problème	Comment analyser les besoins tactiques d'un projet décisionnel ?
	Motivation	Il permet de déterminer les données et la complexité des traitements sur les données
	Forces	Ce patron guide le concepteur dans l'analyse des besoins tactiques.
REALISATION	Solution-démarche	La solution est basée sur la mise en œuvre du diagramme d'activités suivant :  Il convient d'utiliser le patron « Collecter les besoins utilisateurs » respectivement, « Formaliser les besoins utilisateurs » car la réalisation de ces tâches sont très proches pour les groupes d'acteurs tactiques et stratégiques qui sont les utilisateurs du SID.
	Solution-modèle	
	Cas d'applications	
	Conséquences d'application	
	Documents sources	Le document d'expression des besoins des décideurs tactiques
	Documents cibles	
RELATION	Utilise	« Collecter les besoins utilisateurs», « Formaliser les besoins utilisateurs»
	Requiert	« Définir le groupe tactique»
	Raffine	
	Alternative	

FIG. 6.12 – Patron « Analyser les besoins tactiques » (ABT)

Parties	Rubriques	Champs
INTERFACE	Sigle	CBU
	Nom	Collecter les besoins utilisateurs
	Classification	analyse $\wedge$ processus $\wedge$ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \leq k \Rightarrow u_s(x,y) = 1$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable
	Problème	Quels sont les besoins des décideurs liés directement au métier relatif à la statique et à la dynamique du SID ?
	Motivation	Le taux d'appropriation d'un SID par les décideurs liés directement au métier est proportionnel au taux de satisfaction de ce dernier. Ces besoins permettent de déterminer le niveau de complexité des traitements sur les données. Ils contribuent aussi à définir les techniques et les technologies qui seront mises en œuvre dans le projet SID.
	Forces	Ce patron guide le concepteur dans la spécification des besoins utilisateurs.
REALISATION	Solution-démarche	1. Sélectionner les tableaux à 1 ou 2 dimensions définies dans le document d'expression des besoins du groupe afin de déterminer les tableaux croisés synthétiques ou les tableaux croisés, simples. 2. Déterminer les tableaux croisés qui ne se recouvrent pas complètement tels que – soit TM l'ensemble des tableaux croisés TM_i d'un groupe d'acteurs (tactique ou stratégique), – soit TM_A l'ensemble des tableaux croisés de l'échantillon des besoins d'un groupe d'acteurs, – soit F_{TM_i} l'ensemble des faits analysés fi du tableau croisé TM_i, – soit D_{TM_i} l'ensemble des dimensions di du tableau croisé TM_i. $\forall TM_i \in TM, \forall TM_j \in TM, ((F_{TM_i} = F_{TM_j}) \wedge (D_{TM_i} = D_{TM_j}))$ $\Rightarrow TM_i \notin TM_A, TM_j \notin TM_A$ 3. Interroger les acteurs du SID
	Solution-modèle	
	Cas d'applications	Analyse des besoins exprimés sous forme de tableaux croisés ou tableaux croisés synthétiques
	Conséquences d'application	
	Documents sources	Le document d'expression des besoins des décideurs (tactiques ou stratégiques)
	Documents cibles	Les tableaux croisés du groupe tactique ou stratégique
RELATION	Utilise	« Tableau croisé», « Interroger les acteurs du SID»
	Requiert	
	Raffine	
	Alternative	

FIG. 6.13 – Patron « Collecter les besoins utilisateurs » (CBU)

Parties	Rubriques	Champs
INTERFACE	Sigle	FBU
	Nom	Formaliser les besoins utilisateurs
	Classification	analyse $\wedge$ processus $\wedge$ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \leq k \Rightarrow (c(x,y)=1, r_s(TC,x,y)=1, r_s(GP,x,y)=1)$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable r_s : la fonction qui indique les conditions dans lesquelles ce patron requiert le patron TC, respectivement le patron GP
	Problème	Comment formaliser les besoins utilisateurs ?
	Motivation	Les besoins liés à la statique du SID exprimés sous forme de tableaux croisés et les besoins liés à la dynamique du SID exprimés via un graphe sont transformés en un diagramme proche de la vision des utilisateurs afin qu'ils puissent valider l'interprétation de leurs
	Forces	Ce patron aide le concepteur à formaliser les besoins des utilisateurs
REALISATION	Solution-démarche	La formalisation des besoins en termes de traitements requiert le tableau indiquant les propriétés qui permettent de définir les traitements par groupe d'acteurs, défini ci-dessous :

Traitement	Propriété	Informativité	Groupe
	Dérivation		
Harmonisation des données	Complexité, Hétérogénéité	l	Système
Suivi des erreurs	Suivi	t	Tactique, Système
Remontée des erreurs	Remontée	e	Tactique, Système
Permission d'accès	Ouverture, Criticité	p	Stratégique
	Préparation		
Rafraîchissement	Rafraîchissement	*	Tactique, Stratégique, Système
Historisation	Historisation	h	Tactique, Stratégique, Système
Archivage	Archivage	a	Tactique, Stratégique, Système
Consolidation	Consolidation	s	Tactique, Stratégique, Système
Calcul	Calcul	c	Tactique, Stratégique, Système
Disponibilité des données	Disponibilité, Réactivité	d	Tactique, Système

La formalisation est basée sur l'application des règles de transformation, syntaxiques et fusion :
Les règles de transformation :
– Règles de transformation de l'environnement du projet Ex
– Information EI :
– EI1 : un diagramme décisionnel est associé à tout tableau croisé type,
– EI2 : tout tableau croisé spécifiant les besoins utilisateurs doit avoir une structure proche de celle du tableau croisé type. Dans le cas contraire, il est transformé en un tableau à deux dimensions qui sont la dimension « Temps » et une des dimensions principales du domaine d'application lié. Cette dernière est positionnée à la valeur « A1 ».
– Règles de transformation du fait Sx :
– Information SI :
– SI1 : une classe-fait avec le stéréotype << Fait >> est associée à tout fait,
– SI2 : un attribut est associé à toute mesure.
– Traitements SP :
– SP1 : une opération est définie pour chaque traitement (applicable à ce groupe en fonction du tableau) dont les propriétés associées sont annotées dans le graphe de propriétés,
– SP2 : une opération d'attribut est associée à toute mesure qui possède des arguments différents de ceux de la classe pour un traitement donné,
– SP3 : les concepts d'informativité sont associés aux attributs de la classe-fait en fonction des traitements déclarés.

FIG. 6.14 – Patron « Formaliser les besoins utilisateurs »- début (FBU)

REALISATION		– Règles de transformation des dimensions Ai : – Information AI : – AI1 : une classe-dimension avec le stéréotype << Dimension >> est associée à toute dimension, – AI2 : un attribut «Id» est associé à chaque classe-dimension, – AI3 : un attribut est associé à tout paramètre, – Traitements AP : – AP1 : une opération est définie pour chaque traitement (applicable à ce groupe en fonction du tableau des traitements) dont les propriétés associées sont annotées dans le graphe de propriétés, – AP2 : une opération d'attribut est associée à tout paramètre qui possède des arguments différents pour un traitement donné. La classe-dimension « Temps » est généralement pas rafraîchie, – AP3 : les concepts d'informativité sont associés aux attributs de la classe-dimension en fonction des traitements déclarés, Les règles syntaxiques permettent de vérifier la cohérence des diagrammes décisionnels. – Information SDI : – SDI1 : une classe-dimension ne peut pas être reliée une autre classe-dimension, – SDI2 : une classe-fait ne peut pas être reliée à une autre classe-fait, – SDI3 : à toute mesure est associé le concept d'informativité et le traitement d'historisation sur l'exercice précédent pour l'analyse des tendances, – SDI4 : à tout paramètre est associé le concept d'informativité d'historisation et le traitement sur l'exercice précédent pour l'analyse des tendances. – Traitements SDP : – SDP1 : si le concept d'informativité porte sur tous les attributs de la classe et avec les mêmes paramètres alors l'opération liée est spécifiée au niveau de la classe, – SDP2 : si une des mesures du fait de l'analyse possède les concepts d'informativité liée à l'historisation «h» ou à l'archivage «a» alors toutes les dimensions liées doivent posséder aussi cette propriété. De plus, la période et la condition de l'opération associée doivent être au moins égales à celles de la mesure, – SDP3 : l'opération « Rafraîchir() » ne doit pas être appliquée à la dimension temps car elle est initialisée pour assurer tout le cycle de vie du SID. Les règles de fusion : – FUS : regrouper les diagrammes décisionnels ayant la même classe-fait et des classes-dimensions en commun, – FUS1 : fusionner les classes-dimensions partagées par ajout des attributs et des opérations, – FUS2 : fusionner les classes-faits par ajout des attributs et des opérations, – FUS3 : ajouter les classes-dimensions propres à chaque diagramme, – FDS : regrouper les diagrammes décisionnels ayant des classes-faits différents et des classes-dimensions en commun, – FDS1 : fusionner les classes-dimensions par ajout des attributs et des opérations, – FDR : définir les rôles multiples des classes-dimensions qui ont des attributs en commun, – FDR1 pour les classes-dimensions de noms différents ayant peu d'attributs en commun, définir les relations de dépendance fonctionnelle directe entre les attributs des classes-dimensions et les attributs communs.. Puis, il faut définir les relations de dépendance fonctionnelle entre les classes-dimensions qui ont un faible nombre d'attributs en commun. Enfin, il faut supprimer les attributs communs dans la classe-dimension dont les attributs communs et les attributs de la classe ne définissent pas une dépendance fonctionnelle directe,. – FDR2 : fusionner les classes-dimensions de noms différents mais, dont les attributs sont identiques à l'identifiant près, – FRD3 : définir des n relations « En Fonction » entre la classe-fait et la classe-dimension résultant de la fusion de n classes-dimensions, – FRD4 : définir n rôles différents exprimant la sémantique du lien entre la classe-fait et cette classe-dimension. – FRC : fusion des liens d'association, – FRC1 : pour la cohérence des données une classe-dimension ne peut pas être cible d'une dépendance fonctionnelle et participé à un lien d'association entre une classe-fait,
	Solution-modèle	
	Cas d'applications	
	Conséquences d'application	
	Documents sources	Les tableaux croisés du groupe d'acteurs et le graphe de propriété du groupe d'acteurs
	Documents cibles	Le diagramme décisionnel du groupe d'acteurs à inclure dans le document de spécifications détaillées de ce groupe d'acteurs
RELATION	Utilise	« Diagramme décisionnel»
	Requiert	« Tableau croisé », « Graphe de propriétés »
	Raffine	
	Alternative	

Fig. 6.15 – Patron « Formaliser les besoins utilisateurs »- fin (FBU)

Parties	Rubriques	Champs
INTERFACE	Sigle	IAS
	Nom	Interroger les acteurs du SID
	Classification	analyse $\wedge$ processus $\wedge$ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \leq k \Rightarrow u_s(x,y) = 1$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable
	Problème	Quels sont les besoins des acteurs liés à la dynamique du SID ?
	Motivation	Les besoins liés à la dynamique du SID sont exprimés durant l'interview des acteurs. La dynamique d'un SID est évaluée suivant différentes propriétés.
	Forces	Ce patron aide le concepteur dans la spécification des besoins liés à la dynamique du SID.
REALISATION	Solution-démarche	1. Interroger le groupe d'acteurs pour les propriétés qui lui sont associées :

Catégorie	Propriété	Groupe
Profil informatique	Habitudes informatiques	Tactique, Stratégique
	Aptitudes informatiques	Tactique, Stratégique
Gestion des erreurs	Suivi	Tactique, Système
	Remontée	Tactique, Système
Sécurité	Criticité	Stratégique
	Ouverture	Système
Extraction - Chargement	Disponibilité	Système
	Hétérogénéité	Système
	Couverture	Système
	Complexité	Système
Niveau décisionnel existant	Niveau d'informatisation décisionnelle	Tactique, Système
	Niveau d'urbanisation décisionnelle	Tactique, Système
Exploitation	Accès	Tactique
	Interrogation	Tactique
	Reporting	Tactique, Stratégique
	Analyse	Tactique, Stratégique
Restitution	Modalités	Tactique, Stratégique
	Fragmentation	Tactique, Stratégique
	Réactivité	Tactique
Validité	Rafraîchissement	Tactique, Stratégique, Système
	Historisation	Tactique, Stratégique, Système
	Archivage	Tactique, Stratégique, Système
Transformation	Consolidation	Tactique, Stratégique, Système
	Calcul	Tactique, Stratégique, Système
Niveau décisionnel	Mode de pilotage	Stratégique
	Périmètre	Stratégique
Pérennité	Dichotomie des espaces de stockage	Système
	Normalisation du schéma du SID	Système

		2. Annoter les feuilles du graphe de propriétés à partir des réponses aux questions concernant les propriétés et annoter les nœuds de rang 2 en posant des questions sur le poids des catégories de propriétés associées
	Solution-modèle	
	Cas d'applications	Interview des acteurs d'un projet décisionnel
	Conséquences d'application	
	Documents sources	Le document d'expression des besoins du groupe d'acteurs
	Documents cibles	Le graphe de propriétés à inclure dans le document de spécifications détaillées du groupe d'acteurs
RELATION	Utilise	« Graphe des propriétés»
	Requiert	
	Raffine	
	Alternative	

FIG. 6.16 – Patron « Interroger les acteurs du SID » (IAS)

L'analyse des besoins du groupe stratégique se réalise par l'instanciation des patrons suivants :

Parties	Rubriques	Champs
INTERFACE	Sigle	ABS
	Nom	Analyser les besoins stratégiques
	Classification	analyse $\wedge$ processus $\wedge$ SID,
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, (x \leq k \Rightarrow u_5(x,y) = 1), (y = 0 \Rightarrow r_5(DGS, x, y) = 1)$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_5 : la fonction qui indique les conditions dans lesquelles ce patron est utilisable r_5 : la fonction qui indique les conditions dans lesquelles ce patron requiert le patron DGS
	Problème	Comment analyser les besoins stratégiques d'un projet décisionnel ?
	Motivation	Il permet de déterminer les orientations de l'organisation ainsi que le niveau décisionnel et la portée souhaitée du projet.
	Forces	Ce patron guide le concepteur dans l'analyse des besoins stratégiques.
REALISATION	Solution-démarche	La solution est basée sur la mise en œuvre du diagramme d'activités suivant : Analyser les besoins stratégiques Collecter les besoins stratégiques Formaliser les besoins stratégiques Il convient d'utiliser le patron « Collecter les besoins utilisateurs » respectivement, « Formaliser les besoins utilisateurs » car la réalisation de ces tâches sont très proches pour les groupes d'acteurs tactiques et stratégiques qui sont les utilisateurs du SID.
	Solution-modèle	
	Cas d'applications	
	Conséquences d'application	
	Documents sources	Les documents d'expression des besoins stratégiques
	Documents cibles	
RELATION	Utilise	« Collecter les besoins utilisateurs», « Formaliser les besoins utilisateurs»
	Requiert	« Définir le groupe stratégique»
	Raffine	
	Alternative	

FIG. 6.17 – Patron « Analyser les besoins stratégiques » (ABS)

L'analyse des besoins du groupe système se réalise par l'instanciation des patrons suivants :

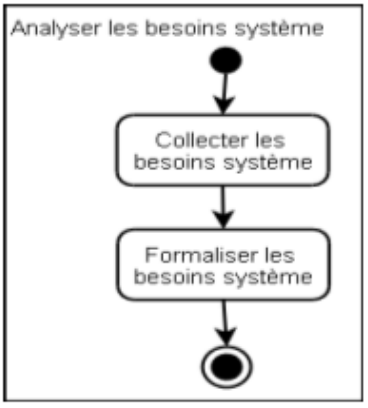
Parties	Rubriques	Champs
INTERFACE	Sigle	ABSY
	Nom	Analyser les besoins systèmes
	Classification	analyse $\wedge$ processus $\wedge$ SID,
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, (x \leq k \Rightarrow u_5(x,y) = 1), (y = 0 \Rightarrow r_5(DGSY, x, y) = 1)$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_5 : la fonction qui indique les conditions dans lesquelles ce patron est utilisable r_5 : la fonction qui indique les conditions dans lesquelles ce patron requiert le patron DGSY
	Problème	Comment analyser les besoins systèmes d'un projet décisionnel ?
	Motivation	Il permet de déterminer les caractéristiques des sources et des modules décisionnels existants. Le but d'un SID est de valoriser le capital informationnel de l'entreprise donc les données opérationnelles. De plus, la phase d'alimentation des données sur la réalisation d'un projet SID représente environ 80% de charge suivant les moyens et les outils mis en œuvre. Afin de réduire ce pourcentage et passer plus de temps à la restitution des données sous un format adapté aux décideurs, il faut une analyse complète et pertinente des systèmes existants. Cette dernière permet de déterminer le niveau d'équipement décisionnel existant et le niveau de complexité de l'ensemble des systèmes existants.
	Forces	Ce patron guide le concepteur dans l'analyse des besoins stratégiques.
REALISATION	Solution-démarche	La solution est basée sur la mise en œuvre du diagramme d'activités suivant : 
	Solution-modèle	
	Cas d'applications	
	Conséquences d'application	
	Documents sources	Les documents d'expression des besoins systèmes
	Documents cibles	
RELATION	Utilise	« Collecter les besoins systèmes», « Formaliser les besoins systèmes»
	Requiert	« Définir le groupe système»
	Raffine	
	Alternative	

FIG. 6.18 – Patron « Analyser les besoins systèmes » (ABSY)

Parties	Rubriques	Champs
INTERFACE	Sigle	CBSY
	Nom	Collecter les besoins systèmes
	Classification	analyse $\wedge$ processus $\wedge$ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \leq k \Rightarrow u_s(x,y) = 1$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable
	Problème	Comment caractériser les sources de données existantes et en extraire l'information décisionnelle ?
	Motivation	La détermination de l'ensemble des systèmes existants aussi bien systèmes d'information classiques que des modules décisionnels permet de définir le capital informationnel de liés au métier du projet et de déterminer le niveau d'équipement décisionnel existant utile pour le choix l'architecture adaptée.
	Forces	Ce patron guide le concepteur dans la spécification des besoins systèmes.
REALISATION	Solution-démarche	1. Sélectionner les sources de données pertinentes dont celles des applications transactionnelles des domaines d'activités liés au projet ainsi que les modules décisionnelles existants. Récupérer les modèles de données des sources avec le patron « Modèle Entité-Association » 2. Réaliser la fiche signalétique de chaque système existant avec le patron « Fiche signalétique d'un système » 3. Interroger les acteurs du SID avec le patron « Interroger les acteurs du SID »
	Solution-modèle	
	Cas d'applications	Analyse des besoins exprimés sous forme de tableaux croisés ou tableaux croisés synthétiques
	Conséquences d'application	
	Documents sources	Le document d'expression des besoins du groupe système
	Documents cibles	Les schémas des sources de données et la fiche signalétique de chaque source à inclure dans le document de spécifications détaillées techniques
RELATION	Utilise	« Modèle Entité-Association», « Fiche signalétique», « Interroger les acteurs du SID»
	Requiert	
	Raffine	
	Alternative	

FIG. 6.19 – Patron « Collecter les besoins systèmes » (CBSY)

<table>
<tr><th>Parties</th><th>Rubriques</th><th>Champs</th></tr>
<tr><td rowspan="6">INTERFACE</td><td>Sigle</td><td>FBSY</td></tr>
<tr><td>Nom</td><td>Formaliser les besoins systèmes</td></tr>
<tr><td>Classification</td><td>analyse ∧ processus ∧ SID</td></tr>
<tr><td>Contexte</td><td>$\forall x \in N_+^*, \forall y \in N_+^*, x \leq k \Rightarrow (c(x,y)=1, r_s(EA,x,y)=1, r_s GP,x,y)=1)$ avec
x : itération en cours du processus de développement du SID,
y : nombre d'utilisations de ce patron au cours du processus complet
k : nombre maximal d'itérations de la phase d'analyse
u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable
r_s : la fonction qui indique les conditions dans lesquelles ce patron requiert le patron EA, respectivement le patron GP</td></tr>
<tr><td>Problème</td><td>Comment formaliser les besoins systèmes ?</td></tr>
<tr><td>Motivation</td><td>La formalisation des besoins utilisateur permet de déterminer l'information décisionnelle contenue dans les sources et préciser les hiérarchies des axes d'analyse.</td></tr>
<tr><td>Forces</td><td>Ce patron aide le concepteur à formaliser les besoins systèmes</td></tr>
<tr><td>REALISATION</td><td>Solution-démarche</td><td>La formalisation des besoins en termes de traitements requiert le tableau indiquant les propriétés qui permettent de définir les traitements par groupe d'acteurs, défini ci-dessous :
<table>
<tr><th>Traitement</th><th>Propriété</th><th>Informativité</th><th>Groupe</th></tr>
<tr><td colspan="4">Dérivation</td></tr>
<tr><td>Harmonisation des données</td><td>Complexité, Hétérogénéité</td><td>l</td><td>Système</td></tr>
<tr><td>Suivi des erreurs</td><td>Suivi</td><td>t</td><td>Tactique, Système</td></tr>
<tr><td>Remontée des erreurs</td><td>Remontée</td><td>e</td><td>Tactique, Système</td></tr>
<tr><td>Permission d'accès</td><td>Ouverture, Criticité</td><td>p</td><td>Stratégique</td></tr>
<tr><td colspan="4">Préparation</td></tr>
<tr><td>Rafraîchissement</td><td>Rafraîchissement</td><td>*</td><td>Tactique, Stratégique, Système</td></tr>
<tr><td>Historisation</td><td>Historisation</td><td>h</td><td>Tactique, Stratégique, Système</td></tr>
<tr><td>Archivage</td><td>Archivage</td><td>a</td><td>Tactique, Stratégique, Système</td></tr>
<tr><td>Consolidation</td><td>Consolidation</td><td>s</td><td>Tactique, Stratégique, Système</td></tr>
<tr><td>Calcul</td><td>Calcul</td><td>c</td><td>Tactique, Stratégique, Système</td></tr>
<tr><td>Disponibilité des données</td><td>Disponibilité, Réactivité</td><td>d</td><td>Tactique, Système</td></tr>
</table>
Soit la matrice des faits candidats qui liste les entités en fonction du nombre de liens des entités et des associations et du nombre d'attributs numériques qui ne sont pas clé primaire. Le nombre de lignes (Nb_att) de la matrice est égal au nombre maximal d'attributs numériques qui ne participent pas à une clé primaire et qui ne sont pas uniques (max(Nb_att)) que possède une entité ou une association du schéma des sources. Les entêtes des lignes prennent leur valeur dans l'intervalle [1 ; max(Nb_att)]. Le nombre de colonnes (Nb_liens) est égal au plus grand nombre de liens que possède une entité ou une association du schéma des sources. Les entêtes des colonnes prennent leur valeur dans l'intervalle [1 ;max(Nb_liens)]. La cellule(n, p) contient les noms des entités et/ou associations qui ont n attributs numériques qui ne participent pas à une clé primaire, non-uniques et qui ont une arité égale à p.

Soit la fonction ArityR : CF → CF qui indique quelles sont les entités ou les associations qui sont liées à une association respectivement à une entité qui est un fait candidat CFk.
Soit CFk le fait candidat caractérisé par la mesure candidate CMCFk . Il est connecté à la dimension candidate CDCFk .
Soit fk le fait valide caractérisé par la mesure mfk . Il est connecté à la dimension valide dfk .</td></tr>
</table>

FIG. 6.20 – Patron « Formaliser les besoins systèmes »- début (FBSY)

Règles de sélection des faits (SF) et des mesures (SM) :
– SF1 : toute entité ou association contenue dans la matrice est un fait candidat CFk,
– SF2 : si CFi ∈ ArityR(CFk) alors CFk est un fait f∈ F et CFi ∉ F. Dans le cas où CFi ∈ ArityR(CFk) et CFk ∈ ArityR(CFi), les faits candidats qui sont des associations sont définis comme faits valides et les faits candidats qui sont des entités ne sont pas définis comme valides,
– SF3 : si CFk– > CFi (dépendance fonctionnelle) alors CFk est un fait ∈ F, CFi ∉ F,
– SM1 : tous les attributs numériques des CFk qui ne participent pas à une clé primaire sont des mesures candidates CMCFk ,
– SM2 : tous les attributs numériques des faits valides fk qui ne participent pas à une clé primaire sont des mesures,
– SM3 : si CFi ∈ ArityR(fk) alors CMfk = {CMCFi}∪({CMCFk} \{CMCFi}).

Règles de sélection des dimensions (SD) :
– SD1 : toute entité en relation directe avec un fait candidat CFk qui n'est pas un fait candidat ou un fait est une dimension candidate CDCFk ,
– SD2 : toute entité ∈ ArityR (fk) qui n'est pas un fait est une dimension dfk ∈ D,
– SD3 : pour fk, les attributs ∉ Mfk qui ne sont pas de type date forment une nouvelle dimension dfk ∈ D,
– SD4 : toute entité en relation directe avec une dimension dk qui n'est pas un fait fk est une dimension valide ddk ∈ D. Cette règle s'applique de manière récursive a fin de définir les niveaux des hiérarchies,
– SD5 : les attributs de type date de fk sont extrait de ce fait afin de créer la dimension Temps. Le fait est relié à la dimension n fois avec n=nombre de d'attributs de type date. Dans le cas o`u n>1, alors il faut ajouter un rôle à l'association,
– SD6 : dans le cas o`u la dimension « Temps » n'a pas été créée auparavant,
elle est rajoutée parmi les concepts multidimensionnels valides,
– SD7 : les liens avec une notion de temps entre un fait fk et une entité ou une association ∈ ArityR(CFk) sont transformés en plusieurs relations « En fonction » avec des rôles différents,
– SD8 : vérifier la non-existence de dépendances fonctionnelles entre les attributs d'une classe-fait et ceux d'une classe-dimension. Si une dépendance fonctionnelle existe, alors les attributs de la classe-fait qui dépendent fonctionnellement d'un ou de plusieurs attributs de la classe-dimension sont transférés dans cette dernière,
– SD9 : vérifier la non-existence de dépendances fonctionnelles entre les attributs des classes-dimensions. Si une dépendance fonctionnelle existe, elle est représentée par un trait discontinu entre les classes-dimensions.

Règles de sélection des paramètres (SP) et des attributs faibles (SW) :
– SP1 :tout attribut numérique, dont les attributs qui participent à la clé primaire, d'une dimension dfk est transformé à un paramètre pk ∈ P,
– SW1 : tout attribut non-numérique dune dimension dfk est transformé en attribut faible ak ∈ A.

Règles de structuration liées au groupe système : elles permettent de représenter via le diagramme décisionnel les concepts multidimensionnels définis à partir des règles de sélection.
Règles de structuration des données :
– SRD1 : tout fait fk est transformé en une classe-fait avec des attributs correspondant à ses mesures mCFk ,
– SRD2 : toute dimension dfk est transformée en une classe-dimension avec des attributs correspondant à ses paramètres pk et ses attributs faibles ak,
– SRD3 : tout classe-fait fk est associée à chacune de ses classes-dimensions dfk par une association représentée un trait continu qui signifie « fk analysé EN FONCTION de dfk ». Ce lien est précisé avec les rôles définis à partir des règles de sélection,
– SRD4 : toute classe-dimension est reliée à ses classes-dimensions par des traits discontinus représentant un lien hiérarchique,
– SRD5 : toute règle de calcul définie dans le graphe de propriétés via la propriété « Calcul » de la catégorie « Transformation » est spécifiée par une opération attribut dans la classe-fait ou la classe-dimension liée.
Règles de structuration des processus :
– SRP1 : les opérations Disponible(d, f, c),Harmoniser(c, h), Trace(l), Exception(e,m), Calculer({vi}+), Historiser(p, d, cond), Rafraîchir(f, m), Archiver(p, d, fct, cond) sont définies en fonction des annotations du graphe de propriétés du groupe système,
– SRP2 : une opération d'attribut est associée à tout paramètre qui possède des arguments différents pour un traitement donné.
– SRP3 : les concepts d'informativité sont associés aux attributs des classes multidimensionnelles en fonction des traitements déclarés,

FIG. 6.21 – Patron « Formaliser les besoins systèmes »- suite (FBSY)

		Règles syntaxiques liées au groupe système : elles permettent de contrôler la consistance des SDD dérivés à partir des sources. – WR1 : une classe-fait peut être reliée à une autre classe-fait. – WR2 : une classe-dimension peut être reliée à une autre classe classe-dimension si et seulement s'il existe une dépendance fonctionnelle entre les deux entités. – WR3 : une classe-dimension peut être reliée à une classe-fait par un trait continu et à une autre classe-dimension par un trait discontinu. Les règles de fusion : – FUS : regrouper les diagrammes décisionnels ayant la même classe-fait et des classes-dimensions en commun, – FUS1 : fusionner les classes-dimensions partagées par ajout des attributs et des opérations, – FUS2 : fusionner les classes-faits par ajout des attributs et des opérations, – FUS3 : ajouter les classes-dimensions propres à chaque diagramme, – FDS : regrouper les diagrammes décisionnels ayant des classes-faits différents et des classes-dimensions en commun, – FDS1 : fusionner les classes-dimensions par ajout des attributs et des opérations, – FDR : définir les rôles multiples des classes-dimensions qui ont des attributs en commun, – FDR1 pour les classes-dimensions de noms différents ayant peu d'attributs en commun, définir les relations de dépendance fonctionnelle directe entre les attributs des classes-dimensions et les attributs communs.. Puis, il faut définir les relations de dépendance fonctionnelle entre les classes-dimensions qui ont un faible nombre d'attributs en commun. Enfin, il faut supprimer les attributs communs dans la classe-dimension dont les attributs communs et les attributs de la classe ne définissent pas une dépendance fonctionnelle directe,, – FDR2 : fusionner les classes-dimensions de noms différents mais, dont les attributs sont identiques à l'identifiant près, – FRD3 : définir des n relations « En Fonction » entre la classe-fait et la classe-dimension résultant de la fusion de n classes-dimensions, – FRD4 : définir n rôles différents exprimant la sémantique du lien entre la classe-fait et cette classe-dimension. – FRC : fusion des liens d'association, – FRC1 : pour la cohérence des données une classe-dimension ne peut pas être cible d'une dépendance fonctionnelle et participé à un lien d'association entre une classe-fait,
	Solution-modèle	
	Cas d'applications	
	Conséquences d'application	
	Documents sources	Les modèles de données Entité-Association des sources et le graphe de propriété du groupe d'acteurs système
	Documents cibles	Le diagramme décisionnel du groupe système à inclure dans le document de spécifications techniques détaillées
RELATION	Utilise	« Diagramme décisionnel»
	Requiert	« Modèle E-A », « Graphe de propriétés »
	Raffine	
	Alternative	

FIG. 6.22 – Patron « Formaliser les besoins systèmes »- fin (FBSY)

Parties	Rubriques	Champs
INTERFACE	Sigle	ELC
	Nom	Evaluer la connaissance
	Classification	analyse $\wedge$ processus $\wedge$ SID,
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \leq k \Rightarrow (c(x,y)=1, r_s(DD1,x,y)=1, r_s(DD2,x,y)=1,$ $r_s(GP1,x,y)=1, r_s(GP2,x,y)=1)$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable r_s : la fonction qui indique les conditions dans lesquelles ce patron requiert le patron DD1, respectivement le patron DD2. Les patrons DD1 et DD2 sont des patrons DD que nous numérotons pour les distinguer. De même, GP1 et GP2 sont des patrons GP. GP1 est associé à DD1 du groupe d'acteurs 1 et GP2 est associé à DD2 du groupe d'acteurs 2.
	Problème	Comment évaluer la connaissance dans le cadre de la confrontation des besoins ?
	Motivation	Il permet de confronter les besoins du groupe 1 et du groupe 2.
	Forces	Ce patron guide le concepteur dans la confrontation des besoins exprimés sous forme de deux diagrammes décisionnels en fonction des poids des catégories de propriétés définis dans les graphes de propriétés de chaque groupe.
REALISATION	Solution-démarche	Si DD1 et DD2 sont un diagramme décisionnel du groupe tactique et un diagramme décisionnel du groupe stratégique alors : Parcours des graphes des propriétés annotées suivant une recherche en profondeur en comparant les propriétés exprimées au travers de leurs besoins à partir des règles définies ci-après. Nous commençons par le sous-graphe « Décision » puis nous finissons par le sous-graphe « Technique » et nous nous focalisons sur les catégories nécessaires pour la définition des traitements communs aux deux graphes : – UCONF1 : appliquer les règles de fusion pour les classes multidimensionnelles de même nom, – UCONF2 : pour la confrontation des attributs de même nom appartenant à deux classes multidimensionnelles différentes, – UCONF2A : si les attributs sont partagés par des classes-faits distinctes alors il convient de préciser fonctionnellement l'environnement afin de déterminer si ces données représentent un ou plusieurs concepts, – UCONF2B : si les attributs sont partagés par des classes-dimensions distinctes alors il faut vérifier les dépendances fonctionnelles entre ces classes, – UCONF3 : pour la confrontation des opérations de même nom, ayant des paramètres différents, appartenant à deux classes multidimensionnelles de même nom, nous évaluons les propriétés associées à ces traitements : – UCONF3A : si les contraintes de valeurs des propriétés sont proches alors nous considérons la moins restrictive, – UCONF3B : si les coefficients sont égaux et les contraintes de valeurs sont contraires alors nous considérons les contraintes de valeurs du graphe de propriétés du groupe tactique car ces acteurs maîtrisent le domaine d'activité, – UCONF3C : si les contraintes de poids ne sont pas égales, les contraintes de valeurs sont contraires et la contrainte de poids la plus importante est celle du groupe stratégique alors nous considérons la contrainte de valeurs définie par ce groupe, – UCONF3D : si les contraintes de poids ne sont pas égales, les contraintes de valeurs sont contraires et la contrainte de poids la plus importante est celle du groupe tactique alors nous considérons que l'itération de l'analyse en cours est clôturée et qu'une nouvelle itération est définie. Sinon – SCONF1 : les classes multidimensionnelles définies dans le diagramme utilisateur (UDD) qui ne sont pas définies dans le diagramme décisionnel des sources (SDD) doivent être évaluées et budgétisées, – SCONF2 : les classes multidimensionnelles définies dans le diagramme système (SDD) qui ne sont pas définies dans le diagramme décisionnel utilisateur (UDD) sont supprimées dans le cas où ces données sont des données supplémentaires non nécessaires à la cohérence des données,

FIG. 6.23 – Patron « Evaluer la connaissance »- début (ELC)

		– SCONF3 : appliquer les règles de fusion pour les classes multidimensionnelles de même nom, – SCONF4 : pour la confrontation des attributs de même nom appartenant à deux classes multidimensionnelles différentes, – SCONF4A : si les attributs sont partagés par des classes-faits distinctes alors il convient de préciser fonctionnellement l'environnement afin de déterminer si ces données représentent un ou plusieurs concepts, – SCONF 4B : si les attributs sont partagés par des classes-dimensions distinctes alors il faut vérifier les dépendances fonctionnelles entre ces classes, – SCONF5 : pour la confrontation des opérations de même nom, ayant des paramètres différents, appartenant à deux classes multidimensionnelles de même nom, nous évaluons les propriétés associées à ces traitements : – SCONF5A : si la contrainte de valeurs retenue pour la définition du diagramme décisionnel utilisateur est proche de celle du groupe système sont proches alors nous considérons celle retenue pour la définition du diagramme décisionnel utilisateur, – SCONF5B : si la contrainte de poids retenue pour la définition au diagramme décisionnel utilisateur est égale à celle du groupe système et si la contrainte de valeurs retenue pour la définition du diagramme décisionnel utilisateur est contraire à celle du groupe système alors nous considérons la contrainte de valeurs retenue pour la définition du diagramme décisionnel utilisateur, – SCONF5C : si la contrainte de poids retenue pour la définition du diagramme décisionnel utilisateur n'est pas égale à celle du groupe système et si la contrainte de valeurs retenue pour la définition du diagramme décisionnel utilisateur est contraire à celle du groupe système alors nous considérons la contrainte de valeurs associée à la plus grande contrainte de poids.
	Solution-modèle	
	Cas d'applications	
	Conséquences d'application	
	Documents sources	
	Documents cibles	Le diagramme décisionnel de la confrontation
RELATION	Utilise	« Diagramme décisionnel»
	Requiert	« Diagramme décisionnel», « Graphe de propriétés»
	Raffine	
	Alternative	

FIG. 6.24 – Patron « Evaluer la connaissance »- fin (ELC)

Patrons produits

Un patron produit d'analyse permet de représenter le résultat de la suite d'action dont il découle. Nous utilisons le formalisme P-SIGMA étendu pour représenter les patrons produits tels que la fiche de catégorisation des groupes d'acteurs et la fiche descriptive des sources. Les autres patrons-produits sont les produits type de notre démarche, en l'occurrence le tableau croisé type (cf. tableau 6.1), le graphe de propriétés général (cf. figure 6.27), le diagramme décisionnel type (cf. figure 6.28). Nous les rappelons dans ce chapitre. Dans cette section, nous présentons donc les patrons produits représentés avec le formalisme P-SIGMA étendu.

Parties	Rubriques	Champs
INTERFACE	Sigle	FG
	Nom	Fiche de groupe
	Classification	analyse ∧ produit ∧ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \leq k \Rightarrow u_s(x,y) = 1$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable
	Problème	Comment spécifier les membres d'un groupe ?
	Motivation	Afin de pouvoir comparer les acteurs du SID, ils doivent être spécifiés dans la même structure.
	Forces	Ce patron aide le concepteur à spécifier les membres d'un groupe.
REALISATION	Solution-démarche	
	Solution-modèle	Les informations qui caractérisent un membre d'un groupe sont définies dans le tableau :

Informations	Groupe du projet (Tactique, Stratégique, Système)	Direction administrative et financière	Direction générale	Commercial	Gestion financière	Gestions des achats	Gestion des ventes	Logistique	Ressources humaines	Autre domaine :	Informatique	Informatique décisionnelle
Nom												
Prénom												
Fonction												
Coordonnées site 1												
Coordonnées site 2												
Types de problème à charge												
Disponibilités												

REALISATION	Cas d'applications	Pour la définition des acteurs liés à la gestion de systèmes OLTP existants
	Conséquences d'application	
	Documents sources	
	Documents cibles	La fiche du groupe système à inclure dans le dossier de cadrage
RELATION	Utilise	
	Requiert	
	Raffine	
	Alternative	

FIG. 6.25 – Patron « Fiche de groupe » (FG)

Parties	Rubriques	Champs
INTERFACE	Sigle	FS
	Nom	Fiche signalétique
	Classification	analyse ∧ produit ∧ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \leq n \Rightarrow u_s(x,y) = 1$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable
	Problème	Comment spécifier les systèmes existants ?
	Motivation	Les systèmes qui vont fournir les données au système d'information décisionnel ont généralement des caractéristiques différentes qui influent sur la faisabilité technique et organisationnelle du système à construire. Cette fiche doit mettre en avant les aspects technique, volumétriques et les possibilités de restitution des sources. De même, elle permet de connaître l'ouverture des systèmes sources vers des outils de la chaîne décisionnelle.
	Forces	Ce patron guide le concepteur dans la spécification des systèmes existants. Il permet un comparatif des systèmes sources et facilite le choix du type d'outils requis par la chaîne décisionnelle.
REALISATION	Solution-démarche	
	Solution-modèle	Les informations qui caractérisent un membre d'un groupe sont définies dans le tableau :

Caractéristiques	Source S_n
Numéro du système source	
Nom du système source	
Description	
Editeur	
Fonction	
Responsable	
Fenêtre horaire de disponibilité	
Sécurité des données	
Informations fonctionnelles	
Cohérence de la source	
Tables concernées	
Informations techniques	
Configuration de la machine	
SGBD	
Langage	
Volume de données	
Historique	
Ouverture du système vers d'autres outils	
Problèmes rencontrés lors de l'utilisation	

	Cas d'applications	Pour la définition des acteurs liés à la gestion de systèmes OLTP et des modules décisionnels existants
	Conséquences d'application	
	Documents sources	
	Documents cibles	La fiche du groupe système à inclure dans le dossier de cadrage
RELATION	Utilise	
	Requiert	
	Raffine	
	Alternative	

FIG. 6.26 – Patron « Fiche signalétique » (FS)

$S.C_r$	$A_j.E_l$		
	$Val1_{E_{l_{A_j}}}$	...	$Valm_{E_{l_{A_j}}}$
$A_i.E_k$			
$Val1_{E_{k_{A_i}}}$	$\{ValC_{r_S}(Val1_{E_{k_{A_i}}}, Val1_{E_{l_{A_j}}})\}$	...	$\{ValC_{r_S}(Val1_{E_{k_{A_i}}}, Valm_{E_{l_{A_j}}})\}$
$Val2_{E_{k_{A_i}}}$	$\{ValC_{r_S}(Val2_{E_{k_{A_i}}}, Val1_{E_{l_{A_j}}})\}$	...	...
$Valn_{E_{k_{A_i}}}$	$\{ValC_{r_S}(Valn_{E_{k_{A_i}}}, Val1_{E_{l_{A_j}}})\}$	...	$\{ValC_{r_S}(Valn_{E_{k_{A_i}}}, Valm_{E_{l_{A_j}}})\}$

TAB. 6.1 – Patron « Tableau croisé » (TC)

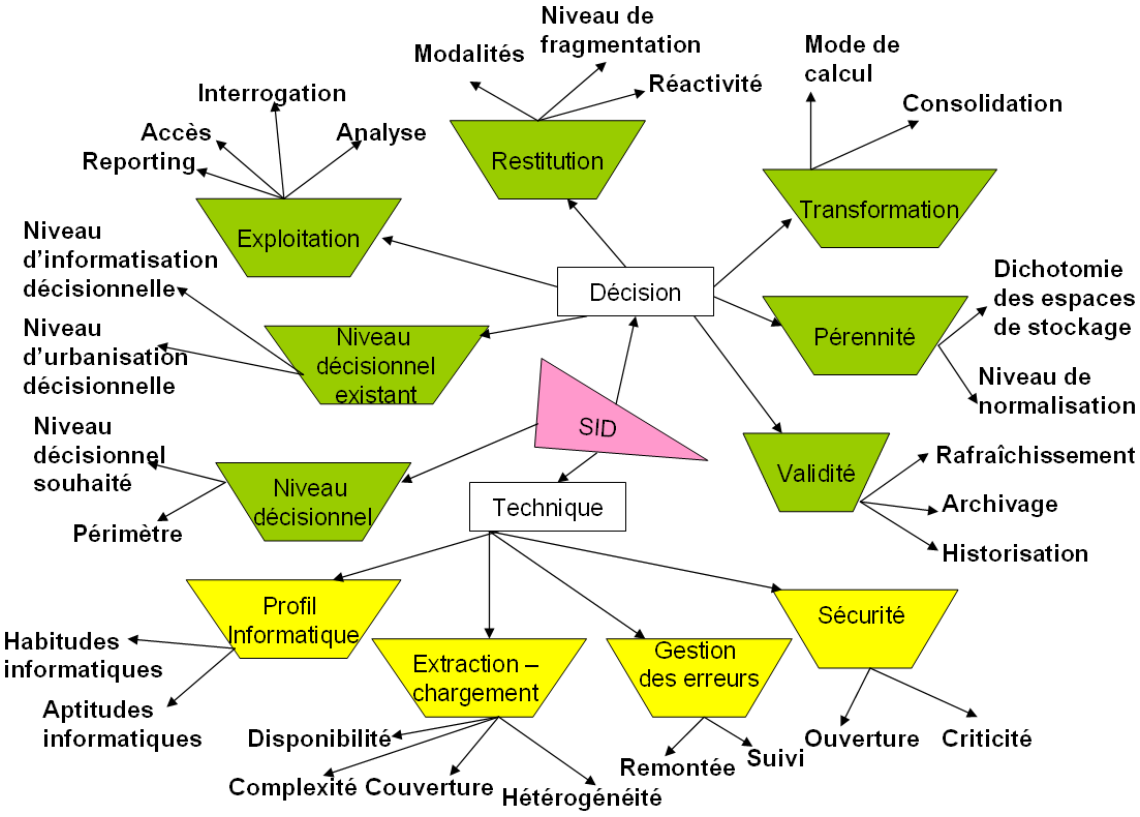


FIG. 6.27 – Patron « Graphe de propriétés » (GP)

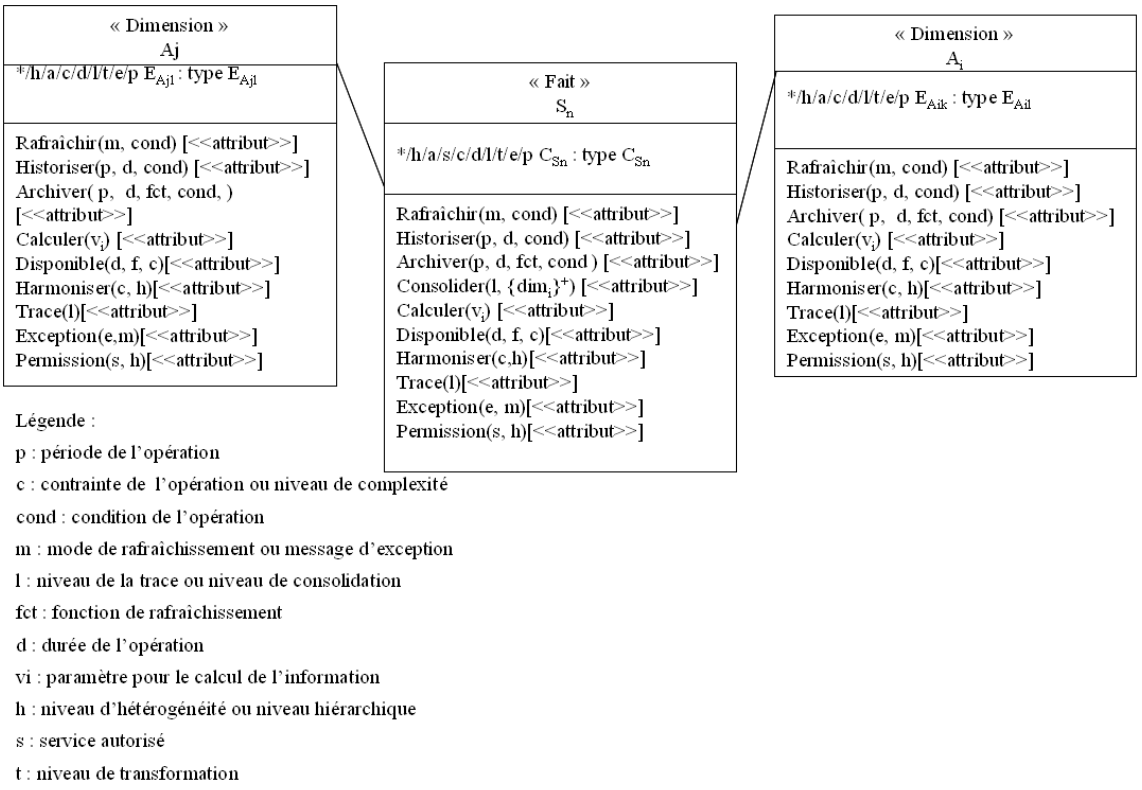


FIG. 6.28 – Patron « Diagramme décisionnel » (DD)

6.3.2 Patron de conception

Les patrons de conception permettent de capitaliser les fragments de démarche et de produits liés à la phase de conception de notre Trident décisionnel. Ces patrons permettent de choisir l'architecture adaptée du SID et de concevoir les schémas de données des modules du SID. Le diagramme de collaboration de patrons présenté dans la figure 6.29 indique les relations entre les patrons de conception de notre catalogue.

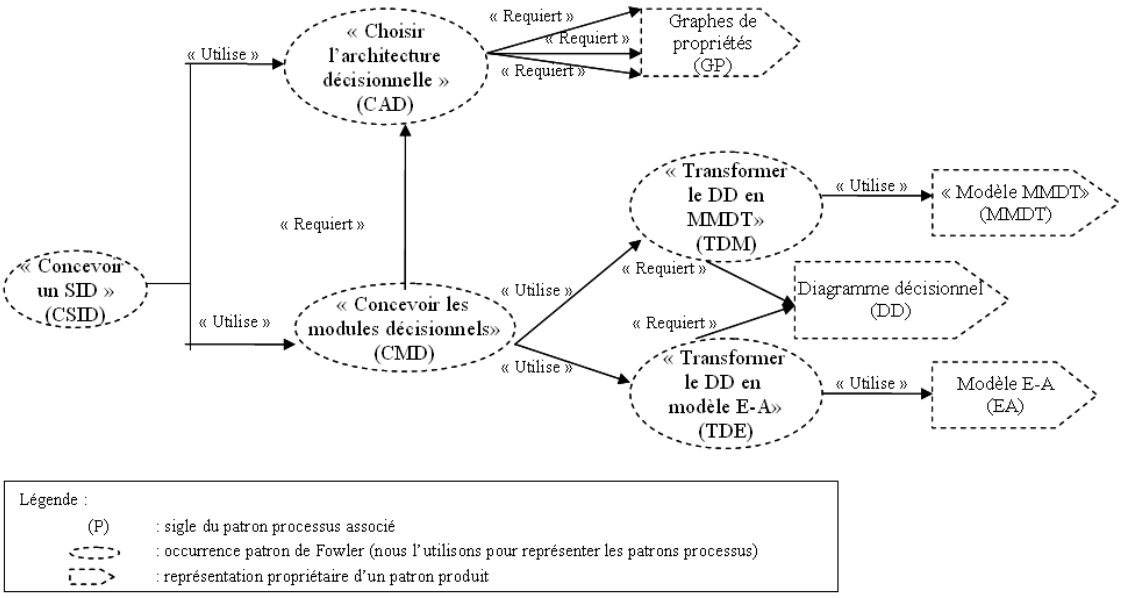


FIG. 6.29 – Diagramme de collaboration de patrons de conception

Patrons processus

Un patron processus de conception permet de représenter les processus liés à la conception du SID suivant l'architecture décisionnelle. Il décrit comment modéliser les données et les traitements afin de définir un système complet et fiable pour la prise de décision.

Le patron qui amorce la conception du SID est le patron processus « Concevoir le SID ».

Parties	*Rubriques*	*Champs*
INTERFACE	Sigle	CSID
	Nom	Concevoir le SID
	Classification	conception$\wedge$ processus$\wedge$ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, (x \succ k \Rightarrow u_s(x,y)=1), (x = k \Rightarrow u_s(x,y)=1, r_s(ASID,x,y)=1)$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable r_s : la fonction qui indique les conditions dans lesquelles ce patron requiert le patron ASID
	Problème	Comment concevoir un système d'information décisionnel ? La conception du SID » correspond à la conception des différents modules décisionnels en suivant l'architecture du SID et à la définition des correspondances entre les sources et les nouveaux modules.
	Motivation	La conception du SID s'articule autour de trois tâches afin de définir le schéma conceptuel de données et de traitements des différents modules décisionnels et les schémas de correspondance.
	Forces	Ce patron guide le concepteur pour la conception d'un système d'information décisionnel. Il permet de : - définir la combinaison de modules décisionnels qui composent l'architecture du SID, - modéliser les nouveaux modules décisionnels, - modéliser les schémas de correspondances,
REALISATION	Solution-démarche	La solution est basée est sur la mise en oeuvre du diagramme d'activités suivants : Concevoir le SID Choisir l'architecture décisionnelle Concevoir les modules décisionnels Définir les schémas correspondances Evaluer les correspondances
	Solution-modèle	
	Cas d'applications	Pour concevoir tout système dédié à la prise de décision
	Conséquences d'application	
	Documents sources	Le dossier de cadrage, le document de spécifications détaillées de chaque groupe
	Documents cibles	
RELATION	Utilise	« Caractériser l'architecture décisionnelle », « Concevoir les modules du SID »
	Requiert	« Analyser un SID »
	Raffine	
	Alternative	

Fig. 6.30 – Patron « Concevoir un SID » (CSID)

Parties	Rubriques	Champs
INTERFACE	Sigle	CAD
	Nom	Choisir l'architecture décisionnelle
	Classification	conception∧ processus∧ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \geq k \Rightarrow u_s(x,y) = 1, r_s(GP1,x,y) = 1,$ avec $r_s(GP2,x,y) = 1, r_s(GP3,x,y) = 1)$ x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable r_s : la fonction qui indique les conditions dans lesquelles ce patron requiert le patron GP1, GP2 respectivement le patron GP3. Les patrons GP1, GP2 et GP3 sont des patrons GP que nous numérotons pour les distinguer. Ils sont les GP des groupes tactique, stratégique et système.
	Problème	Comment choisir l'architecture décisionnelle adaptée ?
	Motivation	Face à l'hétérogénéité des modules décisionnels et au contexte nombreux et variés des projets décisionnels, le choix d'une architecture adaptée qui répond aux besoins des utilisateurs est indispensable pour assurer l'utilisation et la pérennisation du SID au sein de l'organisme.
	Forces	Ce patron aide le concepteur pour le choix de l'architecture du SID
REALISATION	Solution-démarche	La solution est basée l'application de la fonction Choix_architecture() qui prend six paramètres qui définis à partir des propriétés des graphes des propriétés comme indiqués dans le tableau suivant : Critère / Propriétés / Graphe Niveau de couverture des données (NCD) / Périmètre, Couverture / Stratégique, Système Niveau de traitement des données (NTD) / Hétérogénéité, Calcul / Tactique, Stratégique, Système Niveau de complexité des sources (NCS) / Complexité / Système Niveau d'équipement décisionnel existant (NEE) / Niveau d'urbanisation décisionnelle / Tactique, Système Niveau de pérennité de l'architecture (NPA) / Normalisation du schéma du SID / Système Niveau décisionnel souhaité (NDS) / Niveau décisionnel souhaité / Tactique, Système 1. Evaluer les 6 niveaux, 2. Appliquer la fonction Choix_architecture qui repose sur un diagramme d'activité que nous avons décomposé en 32 pour des raisons de lisibilité. Ces diagrammes sont représentées dans l'annexe A de cette thèse.
	Solution-modèle	
	Cas d'applications	
	Conséquences d'application	
	Documents sources	Le dossier de cadrage, le document de spécifications détaillées de chaque groupe
	Documents cibles	La combinaison des modules de l'architecture décisionnelle à inclure dans le document de spécifications techniques détaillées
RELATION	Utilise	
	Requiert	
	Raffine	
	Alternative	

FIG. 6.31 – Patron « Choisir l'architecture » (CAD)

Parties	Rubriques	Champs
INTERFACE	Sigle	CMD
	Nom	Concevoir les modules décisionnels
	Classification	conception ∧ processus ∧ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \geq k \Rightarrow u_s(x,y) = 1, r_s(CAD, x, y) = 1)$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable r_s : la fonction qui indique les conditions dans lesquelles ce patron requiert le patron CAD
	Problème	Comment concevoir les modules du SID ?
	Motivation	L'hétérogénéité des modules décisionnels implique que certains soit basé sur une modélisation Entité-Association et d'autres soient basés sur une modélisation multidimensionnelle
	Forces	Ce patron aide le concepteur à définir le schéma des modules décisionnels.
REALISATION	Solution-démarche	Pour chaque nouveau module décisionnel de l'architecture décisionnelle définie dans le document de spécifications détaillées technique, si le module est basé sur une modélisation multidimensionnelle (un magasin de données) alors il faut utiliser le patron « Transformer le DD en MMDT » sinon, il faut utiliser le patron « Transformer le DD en un modèle E-A »
	Solution-modèle	
	Cas d'applications	
	Conséquences d'application	
	Documents sources	Le document de spécifications détaillées de chaque groupe
	Documents cibles	
RELATION	Utilise	
	Requiert	
	Raffine	
	Alternative	

FIG. 6.32 – Patron « Concevoir les modules décisionnels » (CMD)

Parties	Rubriques	Champs
INTERFACE	Sigle	TDM
	Nom	Transformer le DD en MMDT
	Classification	conception∧ processus∧ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \geq k \Rightarrow u_s(x,y) = 1, r_s(DD,x,y) = 1)$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable r_s : la fonction qui indique les conditions dans lesquelles ce patron requiert le patron DD qui est celui du SID
	Problème	Comment concevoir le diagramme décisionnel en notre modèle multidimensionnel de données et de traitements ?
	Motivation	La transformation du modèle de besoins en un modèle conceptuel multidimensionnel permet de représenter les concepts multidimensionnels et leurs spécificités nécessaires tant aux décideurs qu'aux acteurs systèmes
	Forces	Ce patron aide le concepteur à définir le schéma multidimensionnel d'un module décisionnel
REALISATION	Solution-démarche	Règles PMMDT des classes-faits d'analyse : – **PMMDT1** : toute classe-fait d'analyse FA est transformée en une classe-fait de conception FC, – **PMMDT2** : tout attribut de la classe-fait FA qui est un iième (i>1) paramètre d'une opération d'attribut « Calculer() » est défini comme une classe-mesure. Cette classe-mesure est à la source d'un traitement, nous la désignons donc par Ms. Cette classe-mesure contient l'attribut dont le nom est préfixé par « M » qui correspond à la mesure ainsi que les opérations associées à ce dernier, – **PMMDT3** : tout attribut de la classe-fait FA qui est le premier paramètre d'une opération d'attribut « Calculer() », dont au moins un des autres paramètres est un attribut de la classe-fait, est défini comme une classe-mesure. Cette classe-mesure est la cible d'un traitement, nous la désignons donc par Mc. Cette classe-mesure contient l'attribut dont le nom est préfixé par «M » qui correspond à la mesure, l'opération « Calculer()» ainsi que les autres opérations associées à ce dernier. Cette règle permet de définir les mesures dérivées, – **PMMDT4** : tout attribut de la classe-fait FA qui ne vérifie pas les règles PMMDT2 et PMMDT3 est un attribut de la classe-fait de conception FC associée. Cette règle permet de définir les mesures indépendantes, – **PMMDT5** : une classe-mesure Ms ou Mc est liée à une classe-fait FC par une relation de définition si cette classe-mesure ne peut pas être nulle et si elle est indissociable de cette classe-fait, – **PMMDT6** : une classe-mesure Ms ou Mc est liée à la classe-fait FC par une relation de caractérisation si cette classe-mesure peut être nulle ou si elle n'est pas nécessairement connu pour cette classe-fait, – **PMMDT7** : deux classes-mesures Mc et Ms sont reliées par une relation de « dépendance hiérarchique » si Ms est un paramètre de l'opération « Calculer() » de Mc et si les autres paramètres sont exclusivement des classes-mesures, – **PMMDT8** : deux classes-mesures Mc et Ms sont reliées par une relation de « dépendance utilise » si Ms est un paramètre de l'opération « Calculer() » de Mc et si les autres paramètres ne sont pas exclusivement des classes-mesures, autrement dit les paramètres sont des classes-mesures et des données OLTP non présentes dans le SID ou des données OLTP uniquement, – **PMMDT9** : toute opération de FA est définie uniquement pour la classe-fait, sauf les opérations spécifiques aux attributs à l'origine des classes-mesures. Cependant, les traitements définis pour une classe-fait ou une classe-mesure doivent respecter les impacts des traitements et doivent appartenir à l'ensemble des traitements valables pour une classe-fait respectivement une classe-mesure comme défini dans le tableau des traitements par classe multidimensionnelle de conception.

FIG. 6.33 – Patron « Transformer le DD en MMDT »- début (TDM)

<table>
<tr><td></td><td></td><td>Règles PMMDT des classes-dimension d'analyse :
– PMMDT10 : toute classe-dimension d'analyse DA connectée à une classe-fait d'analyse FA et non cible d'une dépendance fonctionnelle est transformée en une classe-dimension de conception DC,
– PMMDT11 : toute classe-dimension d'analyse DA cible d'une dépendance fonctionnelle est transformée en une classe-niveau de conception NC,
– PMMDT12 : tout attribut et toute opération d'une classe-dimension d'analyse DA correspondant à une DC est un attribut de la classe-dimension DC associée,
– PMMDT13 : tout attribut et toute opération d'une classe-dimension d'analyse DA correspondant à une NC est un attribut de la classe-niveau NC associée,
– PMMDT14 : toute relation « En Fonction » entre une classe-fait FA et une classe-dimension DA est transformée en une relation d'agrégation vers la classe-fait FC 0..* - 1 DC si DA n'est pas cible d'une dépendance fonctionnelle. Si une classe-fait est reliée deux fois à une classe-dimension alors, il faut donner des rôles différents aux relations d'agrégation. Cette règle définit les éléments terminaux multiples et les rôles multiples. Elle évite qu'une classe-fait soit reliée à deux éléments terminaux distincts d'une dimension,
– PMMDT15 : toute relation de dépendance fonctionnelle entre deux classes-dimension (transformées en DC et NC) est transformée en une relation d'association,
– PMMDT16 : toute relation de dépendance fonctionnelle entre deux classes-dimension (transformées en deux classes-niveaux source NCs et cible NCc) est transformée en une relation d'association ayant le stéréotype << RollupTo >> avec les multiplicités NCc 1 - 1..* NCs ou NCc 1 - 1 NCs suivant le projet. La navigabilité de l'association par défaut est de NCs vers NCc. Le rôle « +d » est associée à la classe-niveau NCs et le rôle « +r » est associé à la classe-niveau NCc . Cette règle permet de déterminer les types de hiérarchies,
– PMMDT17 : toute opération de DC est définie pour la classe-dimension ou la classe-niveau associée. Cependant, les traitements définis doivent respecter les impacts des traitements et doivent appartenir à l'ensemble des traitements valables pour une classe-dimension respectivement une classe-niveau comme défini dans le tableau des traitements par classe multidimensionnelle de conception.

Le tableau qui définit les traitements par classe multidimensionnelle de conception est présenté ci-dessus :

<table>
<tr><th>Classe</th><th>Traitements valables</th></tr>
<tr><td>Classe-Fait</td><td>Historisation, Archivage, Consolidation, Calcul, Disponibilité des données, Harmonisation, Suivi des erreurs, Remontée des erreurs, Permission d'accès</td></tr>
<tr><td>Classe-mesure</td><td>Historisation, Archivage, Consolidation, Calcul, Disponibilité des données, Harmonisation, Suivi des erreurs (sauf si niveau de suivi l=3), Remontée des erreurs (sauf si niveau de suivi l=3), Permission d'accès</td></tr>
<tr><td>Classe-dimension</td><td>Rafraîchissement, Historisation (avec une durée d=Max($d_{FArchivage}$) + Max($d_{FHistorisation}$), Calcul, Harmonisation, Suivi des erreurs, Remontée des erreurs</td></tr>
<tr><td>Classe niveau</td><td>Rafraîchissement, Calcul, Harmonisation, Suivi des erreurs (sauf si niveau de suivi l=3), Remontée des erreurs (sauf si niveau de suivi l=3)</td></tr>
</table>
</td></tr>
<tr><td></td><td>Solution-modèle</td><td></td></tr>
<tr><td></td><td>Cas d'applications</td><td></td></tr>
<tr><td></td><td>Conséquences d'application</td><td></td></tr>
<tr><td></td><td>Documents sources</td><td>Le document de spécifications détaillées de chaque groupe</td></tr>
<tr><td></td><td>Documents cibles</td><td>Le modèle conceptuel multidimensionnel de données et de traitements du SID</td></tr>
<tr><td rowspan="4">RELATION</td><td>Utilise</td><td></td></tr>
<tr><td>Requiert</td><td></td></tr>
<tr><td>Raffine</td><td></td></tr>
<tr><td>Alternative</td><td></td></tr>
</table>

FIG. 6.34 – Patron « Transformer le DD en MMDT »- fin (TDM)

Parties	Rubriques	Champs
INTERFACE	Sigle	TDE
	Nom	Transformer le DD en un modèle E-A
	Classification	conception∧ processus∧ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \geq k \Rightarrow u_s(x,y) = 1, r_s(DD,x,y) = 1)$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable r_s : la fonction qui indique les conditions dans lesquelles ce patron requiert le patron DD qui est celui du SID
	Problème	Comment concevoir le diagramme décisionnel en notre modèle Entité-Association de données ?
	Motivation	La transformation du modèle de besoins en un modèle conceptuel entité-association permet de représenter de définir des systèmes avec une gestion plus efficace pour la manipulation des données
	Forces	Ce patron aide le concepteur à définir le schéma Entité-Association d'un module décisionnel
REALISATION	Solution-démarche	Règles PMEA des classes-faits d'analyse : – **PMEA1** : toute classe-fait d'analyse FA est transformée en une entité EF. La clé primaire de EF est une clé de substitution Id NomEF, – **PMEA2** : tout attribut de la classe-fait FA qui est un iième (i>1) paramètre d'une opération d'attribut « Calculer() » est défini comme une entité EM. Cette entité est à la source d'un traitement, nous la désignons par EMs . Cette entité contient un attribut dont le nom est préfixé par « M » qui correspond à la mesure. La clé primaire de Es est une clé de substitution Id NomEMs, – **PMEA3** : tout attribut de la classe-fait FA qui est le premier paramètre d'une opération d'attribut « Calculer() », dont au moins un des autres paramètre est un attribut de la classe-fait, est défini comme une entité. Cette entité est la cible d'un traitement, nous la désignons par EMc. Cette entité contient un attribut dont le nom est préfixé par « M » qui correspond à la mesure. La clé primaire de EM est une clé de substitution Id NomEMc, – **PMEA4** : tout attribut de la classe-fait FA qui ne vérifie pas les règles PMEA2 et PMEA3 forme les attributs de la classe-fait multidimensionnelle de conception FC associée, – PMEA5 : une entité définissant une mesure EM forme une association binaire, appelée «Définir NomEM», avec l'entité EF définissant un fait FC ayant les cardinalités EM(1,N)-(1,1)EF si cette entité EM ne peut pas être nulle et si elle est indissociable de cette classe-fait, – **PMEA6** : une entité définissant une mesure EM forme une association binaire, appelée «Caractériser NomEM», avec l'entité EF définissant un fait FC ayant les cardinalités EM(0,N)- (1,1)EF si cette entité EM peut être nulle ou si elle n'est pas nécessairement connu pour cette classe-fait, – **PMEA7** : deux entités définissant des mesures EMs et EMc sont liées par une association binaire « ComposerEMs EMc » de cardinalités EMs(0,1) (respectivement (1,1)) - (0,N)EMc si EMc est un paramètre de l' opération d'attribut « Calculer() » et les autres paramètres sont exclusivement des classes-mesures, – **PMEA8** : deux entités définissant des mesures EMs et EMc forme une association binaire «utiliseEMs EMc» de cardinalités EMs (1,1) - (0,N)Emc si EMc est un paramètre de l' opération d'attribut « Calculer() » et les autres paramètres ne sont pas exclusivement des classes-mesures, autrement dit les paramètres sont des classes-mesures et des données OLTP non stockés dans le SID ou encore que des données OLTP .

FIG. 6.35 – Patron « Transformer le DD en modèle E-A »- début (TDE)

Parties	Rubriques	Champs
INTERFACE	Sigle	TDE
	Nom	Transformer le DD en un modèle E-A
	Classification	conception∧ processus∧ SID
	Contexte	$\forall x \in N_+^*, \forall y \in N_+^*, x \geq k \Rightarrow u_s(x,y) = 1, r_s(DD,x,y) = 1)$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable r_s : la fonction qui indique les conditions dans lesquelles ce patron requiert le patron DD qui est celui du SID
	Problème	Comment concevoir le diagramme décisionnel en notre modèle Entité-Association de données ?
	Motivation	La transformation du modèle de besoins en un modèle conceptuel entité-association permet de représenter de définir des systèmes avec une gestion plus efficace pour la manipulation des données
	Forces	Ce patron aide le concepteur à définir le schéma Entité-Association d'un module décisionnel
REALISATION	Solution-démarche	Règles PMEA des classes-faits d'analyse : – **PMEA1** : toute classe-fait d'analyse FA est transformée en une entité EF. La clé primaire de EF est une clé de substitution Id NomEF, – **PMEA2** : tout attribut de la classe-fait FA qui est un iième (i>1) paramètre d'une opération d'attribut « Calculer() » est défini comme une entité EM. Cette entité est à la source d'un traitement, nous la désignons par EMs . Cette entité contient un attribut dont le nom est préfixé par « M » qui correspond à la mesure. La clé primaire de Es est une clé de substitution Id NomEMs, – **PMEA3** : tout attribut de la classe-fait FA qui est le premier paramètre d'une opération d'attribut « Calculer() », dont au moins un des autres paramètre est un attribut de la classe-fait, est défini comme une entité. Cette entité est la cible d'un traitement, nous la désignons par EMc. Cette entité contient un attribut dont le nom est préfixé par « M » qui correspond à la mesure. La clé primaire de EM est une clé de substitution Id NomEMc, – **PMEA4** : tout attribut de la classe-fait FA qui ne vérifie pas les règles PMEA2 et PMEA3 forme les attributs de la classe-fait multidimensionnelle de conception FC associée, – PMEA5 : une entité définissant une mesure EM forme une association binaire, appelée «Définir NomEM», avec l'entité EF définissant un fait FC ayant les cardinalités EM(1,N)-(1,1)EF si cette entité EM ne peut pas être nulle et si elle est indissociable de cette classe-fait, – **PMEA6** : une entité définissant une mesure EM forme une association binaire, appelée «Caractériser NomEM», avec l'entité EF définissant un fait FC ayant les cardinalités EM(0,N)- (1,1)EF si cette entité EM peut être nulle ou si elle n'est pas nécessairement connu pour cette classe-fait, – **PMEA7** : deux entités définissant des mesures EMs et EMc sont liées par une association binaire « ComposerEMs EMc » de cardinalités EMs(0,1) (respectivement (1,1)) - (0,N)EMc si EMc est un paramètre de l' opération d'attribut « Calculer() » et les autres paramètres sont exclusivement des classes-mesures, – **PMEA8** : deux entités définissant des mesures EMs et EMc forme une association binaire «utiliseEMs EMc» de cardinalités EMs (1,1) - (0,N)Emc si EMc est un paramètre de l' opération d'attribut « Calculer() » et les autres paramètres ne sont pas exclusivement des classes-mesures, autrement dit les paramètres sont des classes-mesures et des données OLTP non stockés dans le SID ou encore que des données OLTP .

FIG. 6.36 – Patron « Transformer le DD en modèle E-A »- fin (TDE)

Patrons produits

Un patron produit de conception décrit les objets en sortie des différentes tâches de conception tels que le modèle conceptuel de données et le modèle conceptuel de données et de traitements. Les patrons produits spécifiques à la phase de conception

sont le modèle de données conceptuel entité-association et notre modèle conceptuel multidimensionnel de données et de traitements. Nous rappelons dans ce chapitre notre modèle conceptuel multidimensionnel de données et de traitements (cf. figure 6.37).

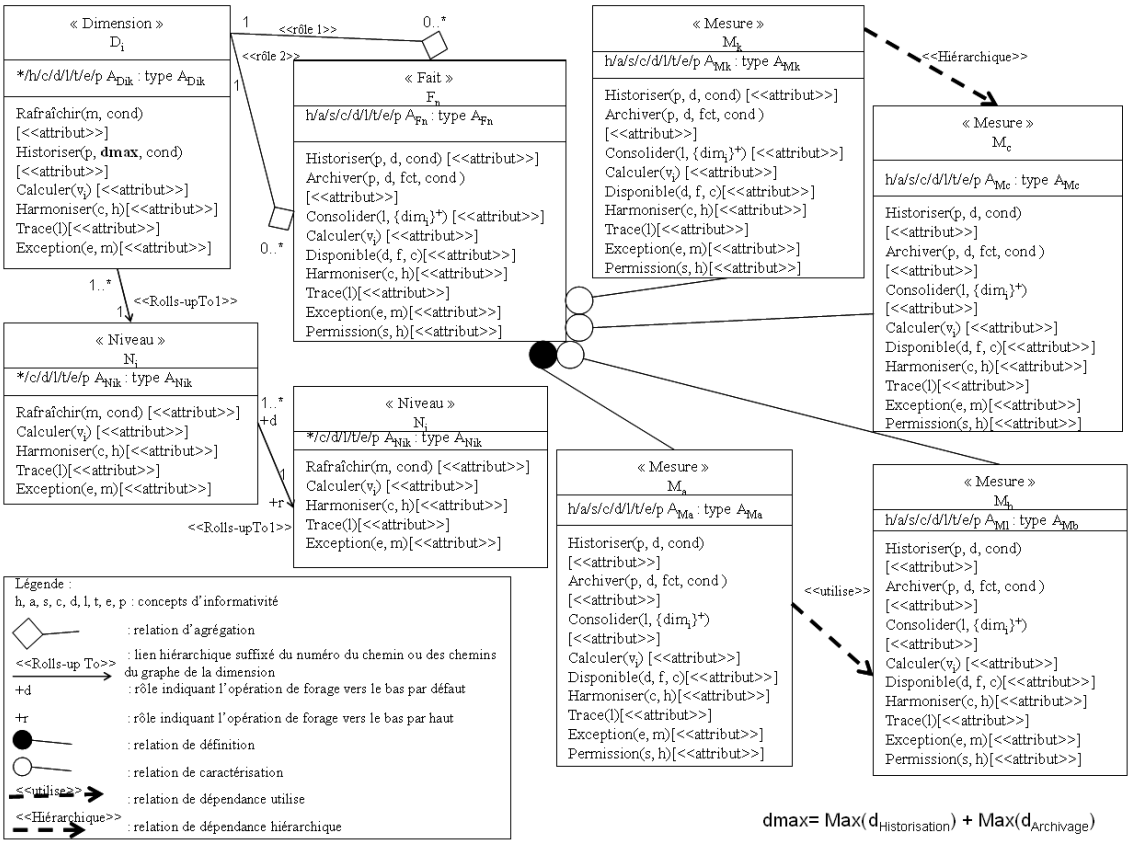


FIG. 6.37 – Modèle multidimensionnel de données et de traitements généralisé MMDT

6.4 Ingénierie par la réutilisation : mise en application de notre catalogue

Tout au long de cette thèse, nous avons déroulé l'exemple de la mise en place d'un SID pour le domaine des immobilisations. Dans un souci pédagogique, nous avons commencé par un exemple simpliste car le domaine des immobilisations n'est pas l'activité principale de l'organisation que nous citons. Afin de montrer le bénéfice de notre catalogue de patrons de manière générale et le bénéfice qu'il procure à la mise en oeuvre de notre méthode de développement, nous présentons dans cette section l'application de notre catalogue de patrons pour la mise du SID d'un des principaux domaines d'activités de cette organisation, soit celui de la redevance force motrice (RFM). Nous nous focalisons sur les résultats de la mise en oeuvre.

Des exemples d'application très développés ont été présentés dans les chapitres 4 et 5. Nous déroulons le cadrage et le prototypage de ce projet. Le délai de mise à disposition du SID étant très court ; le cadrage et le prototypage ont été fusionnés. Nous présentons des versions simplifiées des schémas sources car ces informations sont critiques pour l'organisation.

La redevance force motrice est l'activité qui calcule le coût du traitement de l'eau pour la production de l'électricité depuis des stations de captage telles que des cours d'eau, des lacs.

Pour ce projet, en raison des délais imposés par les acteurs stratégiques et tactiques, les concepteurs décisionnels réalisent le cadrage et le prototypage au cours d'une seule itération définie suivant le cycle de notre processus de développement présenté dans le chapitre 3. Le nombre maximal d'itérations de la phase d'analyse est donc égal à 1 (soit k=1). Par rapport au projet immobilisations, le domaine d'activité a changé donc les acteurs tactiques et les acteurs système aussi. Les acteurs tactiques sont des décideurs liés au domaine de la redevance force motrice et les acteurs systèmes s'occupent des systèmes sources liés au domaine de la redevance force motrice et des modules décisionnels existants. Les acteurs stratégiques n'ont pas changé car c'est une organisation de taille moyenne avec une soixantaine d'employés dont un responsable des projets décisionnels et deux responsables stratégiques.

Pour la mise en place de ce SID, nous utilisons le patron associé à cette étape de la démarche, soit le patron « Mettre en place un SID » (MSID) présenté dans la figure 6.5. La granularité des patrons fondée sur les éléments processus et produits de notre démarche du Trident décisionnel permet de mettre en oeuvre le patron dédié à l'élément processus que l'on souhaite réaliser ou au produit que l'on souhaite définir. Le patron MSID est utilisable indépendamment de tout contexte. Suivant la solution-démarche de ce patron, il faut analyser le SID puis, il faut concevoir le SID et enfin il faut implanter le SID. Nous utilisons dans l'ordre les patrons suivants : « Analyser un SID », « Concevoir un SID » et « Implanter le SID ». La réalisation de ce patron utilise le document source qui est le cahier des charges du projet. Nous nous focalisons sur les tâches d'analyse et de conception. Ainsi dans les deux sections suivantes, nous explicitons la réutilisation des patrons pour l'analyse et la conception du projet redevance force motrice.

Tous les patrons d'analyse que nous instancions au cours de cette itération de la phase d'analyse (x=1) sont utilisables car le nombre maximal k d'itérations de la phase d'analyse est égal à 1 (x=k) et le nombre d'utilisations de chaque patron est égal à 0 (y=0). Nous n'explicitons donc pas le contexte de chaque patron au cours de la phase d'analyse. Cependant, au cours de la phase de conception, nous explicitons individuellement l'évaluation du contexte de chaque patron car tous les patrons ne sont pas utilisables et ils ne requièrent pas tous les patrons qu'ils requièrent par défaut suivant leur rubrique « Requiert ».

6.4.1 Réutilisation des patrons au cours de phase d'analyse

Début de l'utilisation du patron « Analyser un SID » ASID

Pour analyser le SID, nous utilisons le patron « Analyser un SID » (ASID) présenté dans la figure 6.7. Nous sommes au cours de la première itération du processus de développement donc ce patron requiert MSID. Nous avons déjà utilisé ce patron requis au cours de cette itération, ASID peut donc être instancié. Suivant sa solution-démarche, il faut caractériser les groupes d'acteurs. Puis, il faut analyser en parallèle les besoins tactiques, les besoins stratégiques et les besoins systèmes. A partir de l'analyse des besoins tactiques et stratégiques, il faut évaluer la connaissance des utilisateurs. Enfin, il faut évaluer la connaissance de l'environnement en ayant au préalable analysé les besoins systèmes. La réalisation de ce patron utilise le document source qui est le cahier des charges du projet.

Nous commençons donc par caractériser les groupes d'acteurs avec le patron du même nom et de sigle (CGA), présenté dans la figure 6.8. Suivant sa solution-démarche, il faut définir en parallèle le groupe tactique, le groupe stratégique et le groupe système à partir du document source qui est le cahier des charges du projet.

Nous réutilisons la fiche de groupe du groupe stratégique définie au cours du projet immobilisations car les acteurs de ce groupe n'ont pas changé. Nous définissons le groupe tactique avec le patron « Définir le groupe tactique » (DGT) (cf. figure 6.9) et le groupe système avec le patron « Définir le groupe système » (DGT) (cf. figure 6.11). Ils utilisent le patron « Fiche de groupe » pour la définition des acteurs de chaque groupe. En raison de la confidentialité de ces informations, nous ne pouvons pas présenter ces fiches dans cette thèse. Ces fiches permettent aux concepteurs décisionnels de s'adresser au bon interlocuteur pour obtenir les informations pertinentes en cas de besoins et de diffuser la documentation aux acteurs concernés.

Début de l'utilisation du patron « Analyser les besoins tactiques »

Maintenant, nous procédons à l'analyse parallèle des besoins tactiques, stratégiques et systèmes. Nous commençons par l'analyse des besoins tactiques avec le patron « Analyser les besoins tactiques » (ABT). Ce patron requiert le patron DGT car il n'a pas encore été utilisé. Nous avons déjà utilisé le patron DGT donc nous pouvons instancier le patron ABT. Suivant sa solution-démarche, il faut collecter les besoins tactiques avec le patron « Collecter les besoins utilisateurs » et il faut formaliser les besoins stratégiques avec le patron « Formaliser les besoins utilisateurs ».

Début de l'utilisation du patron « Collecter les besoins utilisateurs »

L'utilisation du patron « Collecter les besoins utilisateurs » ne requiert pas d'autre patron car il n'a pas de fonction r_s défini dans son contexte. Suivant sa solution-démarche, à partir du document d'expression des besoins tactiques, nous sélectionnons les tableaux croisés et nous déterminons ceux qui ne se recouvrent

pas. Puis, nous interrogeons les acteurs tactiques avec le patron « Interroger les acteurs du SID ».

Les tableaux croisés contenus dans le document de spécification des besoins du groupe tactique sont définis dans les tableaux 6.2, 6.3 et 6.4 suivants :

Force Motrice. Volume turbiné		Temps.Semaine			
		S1	S2	S3	S4
Ouvrage physique. Nom ouvrage	Ouvrage physique. Puissance				
Plan d'Arem	40	152	99	98	127
Barrage Carbonne	26	145	150	123	130
Argentat	48	340	310	297	308
Barrage de L'aigle	360	550	546	519	420

TAB. 6.2 – Tableau croisé du volume turbiné par ouvrage physique au cours de quatre dernières semaines pour un exploitant

Force Motrice. Energie produite ; Hauteur chute ; Volume turbiné			Exploitant.Raison sociale	
			Syndicat E	Syndicat D
Nom Département	Nom Bassin	Ouvrage Physique		
Haute-Garonne	Garonne	Plan d'Arem	5 230,77,14 ; 255 ; 4521	4 533,33 ; 15 ; 221
		Barrage Carbonne	6 562,14 ; 19 ; 152	11 052,02,14 ; 17 ; 256
Dordogne	Dordogne	Argentat	32 324,56 ; 31 ; 725	23 630,36 ; 25 ; 530
Corrèze	Limousin	Barrage de l'aigle	251 523,23 ; 84 ; 974	273 989,88 ; 77 ; 1061

TAB. 6.3 – Tableau croisé des ouvrages physiques au cours du dernier mois

Force Motrice.			Payeur.Raison sociale		
Montant redevance			Payeur1	Payeur2	Payeur3
Nom Département	Nom Bassin	Ouvrage fonctionnel			
Haute-Garonne	Garonne	Usine d'Arem	7 523,14		9 409.21
		Usine Carbonne		14 424.25	6 409.21
Dordogne	Dordogne	Usine Argentat	21 715,45	35 698,14	
Corrèze	Limousin	Usine de l'aigle	58 245,85	47 019,24	36 985,23

TAB. 6.4 – Tableau croisé du montant de la redevance par ouvrage fonctionnel hydroélectrique et par payeur pour le dernier mois

Ces tableaux croisés partagent des dimensions et des faits mais ils ne se recouvrent pas totalement, les trois sont donc évalués pour l'analyse.

L'interrogation des décideurs tactiques suivant le patron « Interroger les acteurs du SID » consiste à leurs poser des questions liées aux propriétés suivantes :
- les habitudes informatiques,
- les aptitudes informatiques,
- le suivi des erreurs,
- la remontée des erreurs,
- le niveau d'informatisation décisionnelle,
- le niveau d'urbanisation décisionnelle,
- l'accès aux données,
- l'interrogation des données,
- le reporting des données,
- l'analyse des données,
- les modalités de restitution,
- la fragmentation de la restitution,
- la réactivité de la restitution,
- le rafraîchissement des données,
- l'historisation des données,
- l'archivage des données,
- la consolidation des données,
- le calcul des données.

Les réponses à ces questions ainsi que celles relatives aux poids des catégories de propriétés permettent d'annoter le graphe de propriétés défini suivant le patron « Graphe de propriétés ». Nous obtenons le graphe de propriétés du groupe tactique présenté dans la figure 6.38.

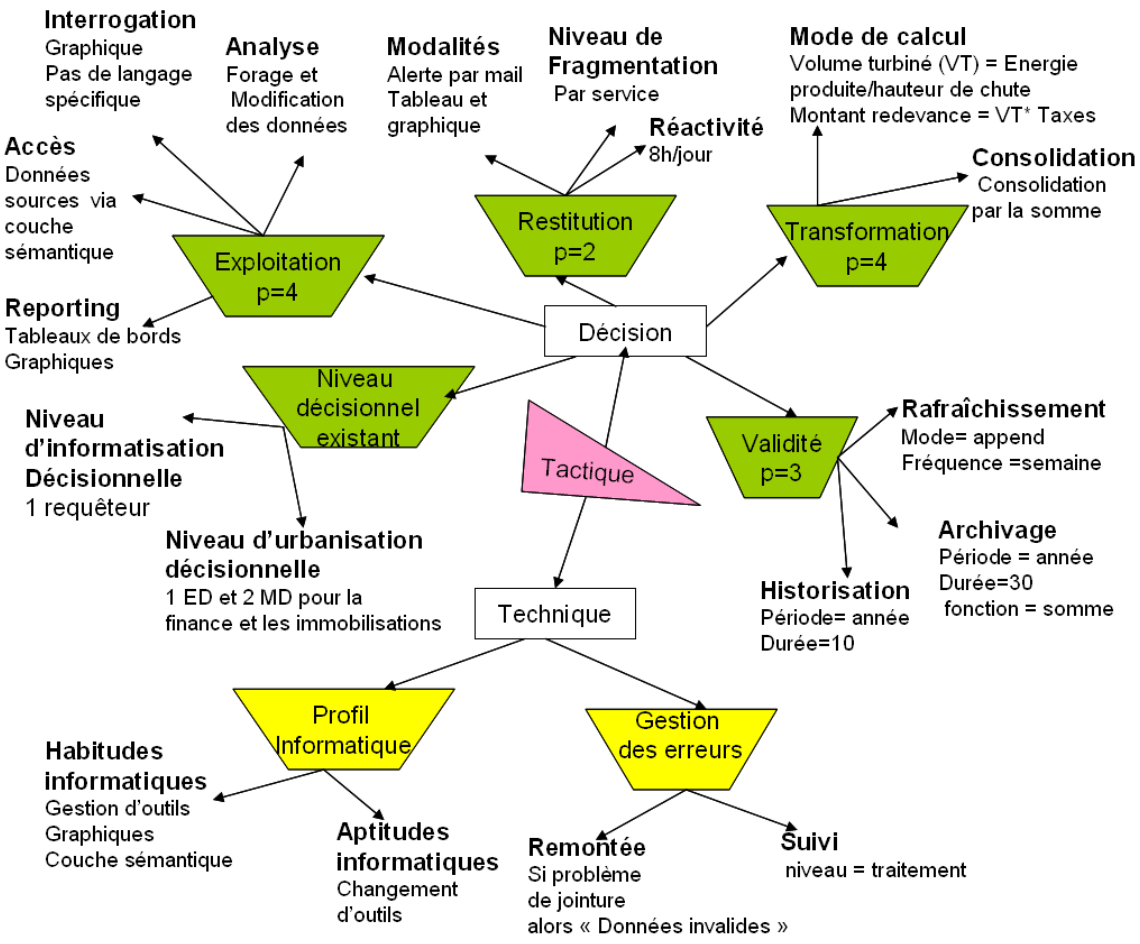


FIG. 6.38 – Graphe de propriétés CSD du groupe tactique du projet RFM

Fin de l'utilisation du patron « Collecter les besoins utilisateurs »

Début de l'utilisation du patron « Formaliser les besoins utilisateurs »

Suivant la solution-démarche du patron ABT, après la collecte des besoins, il faut procéder à la formalisation des besoins tactiques en utilisant le patron « Formaliser les besoins utilisateurs ». La mise en oeuvre de la solution démarche basée sur les règles de transformation et les règles syntaxiques suivie de l'utilisation du patron produit « Diagramme décisionnel » génère les diagrammes 6.39, 6.40 et 6.41 associés respectivement aux tableaux croisés 6.2, 6.3 et 6.4.

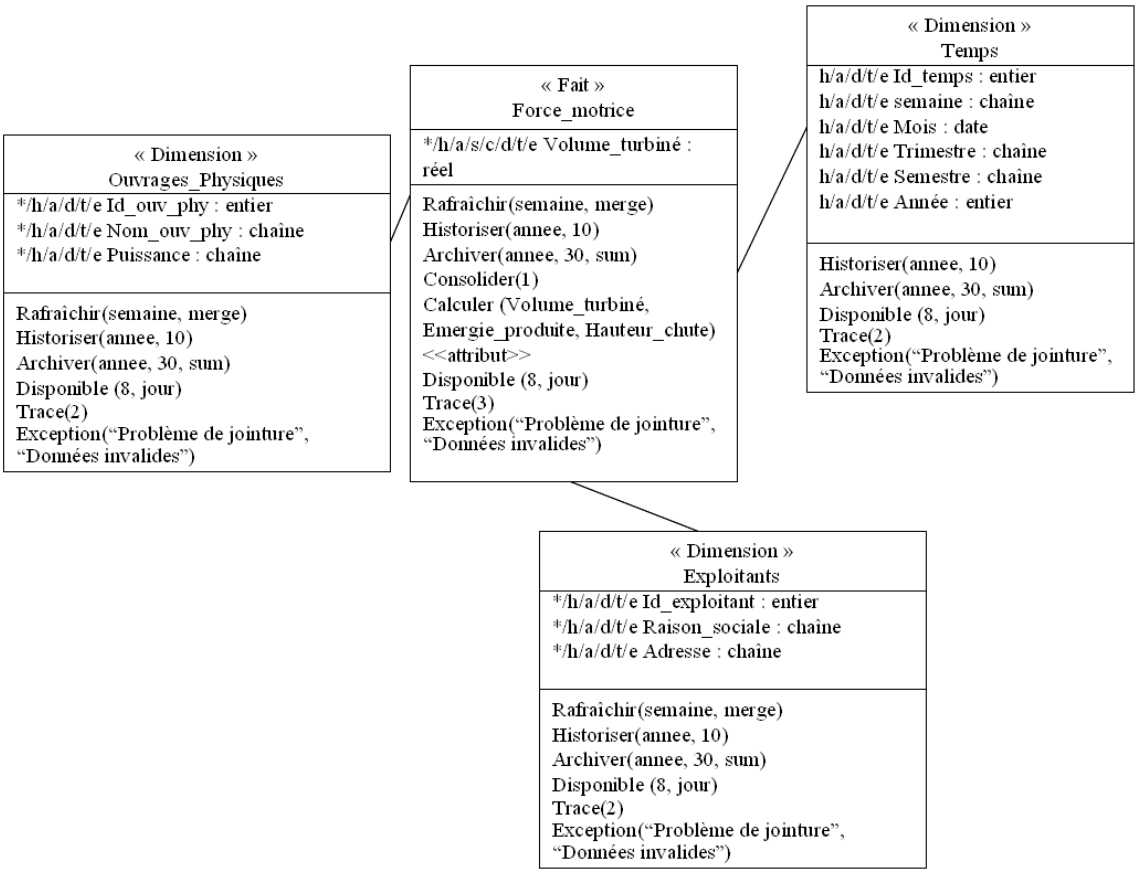


FIG. 6.39 – Diagramme décisionnel associé au tableau 6.2

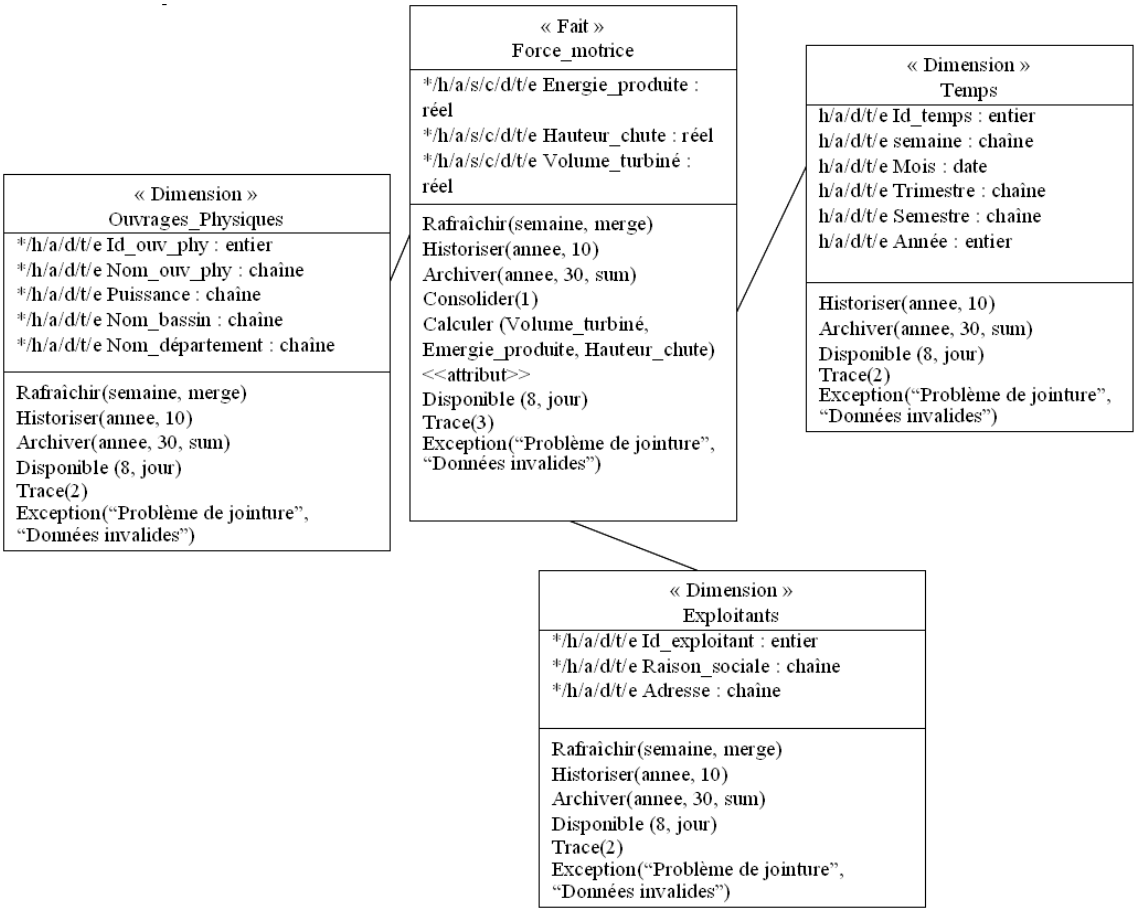


FIG. 6.40 – Diagramme décisionnel associé au tableau 6.3

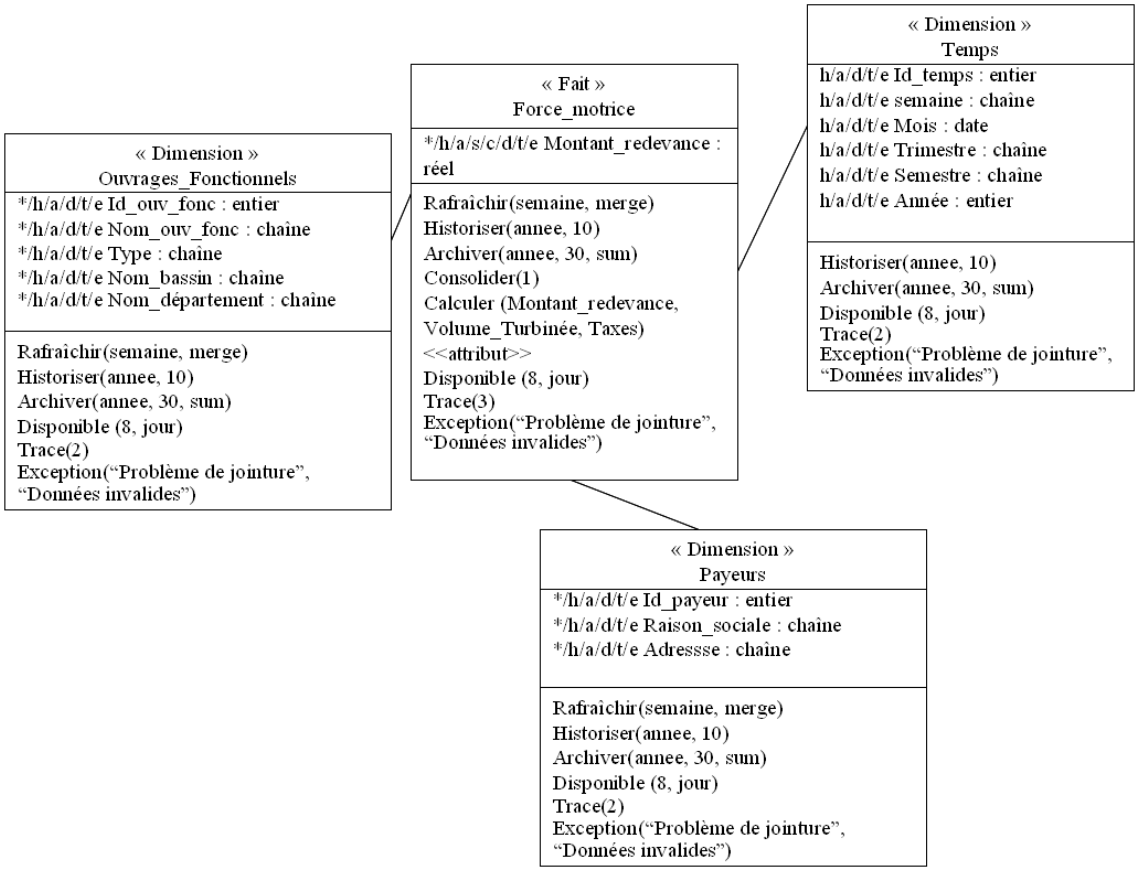


FIG. 6.41 – Diagramme décisionnel associé au tableau 6.4

L'application des règles de fusion de la solution-démarche pour les trois diagrammes décisionnels génère un diagramme décisionnel composé d'une classe-fait et de quatre classes-dimensions. La fusion des diagrammes décisionnels se fait deux à deux. Le diagramme décisionnel résultant est présenté dans la figure 6.42.

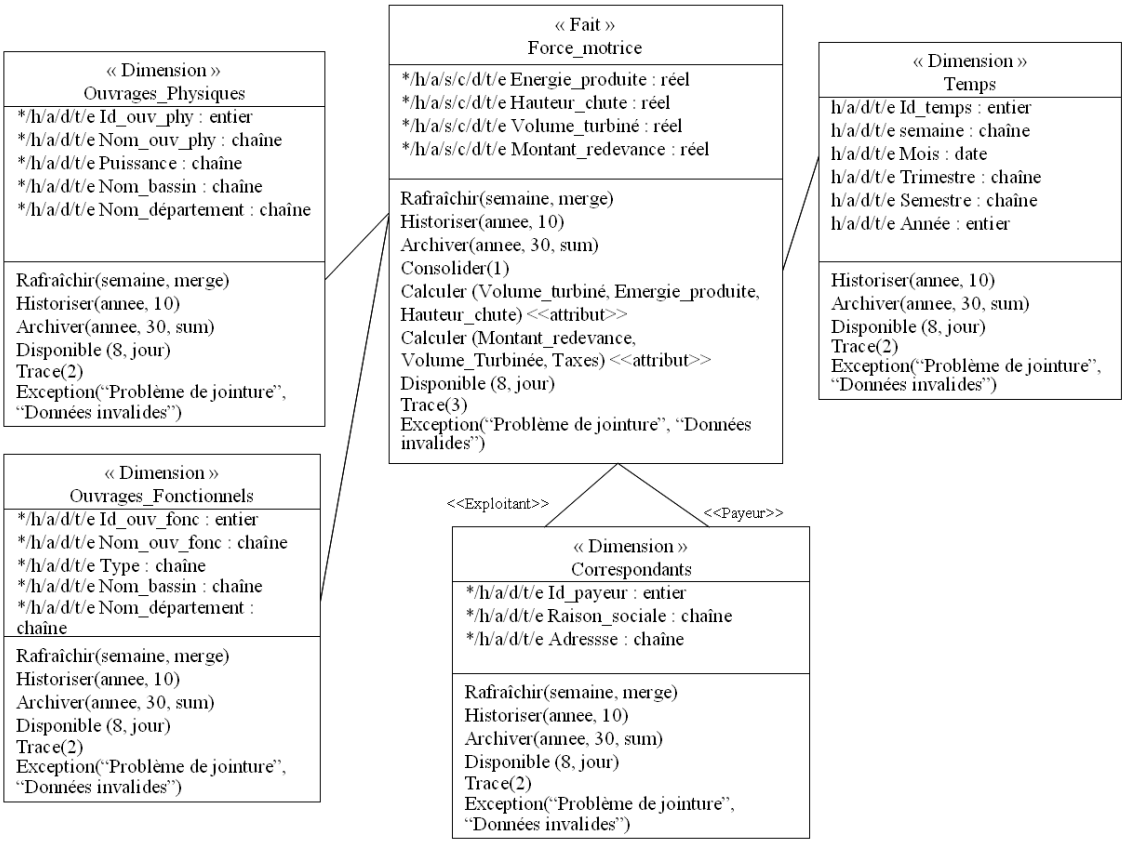


FIG. 6.42 – Diagramme décisionnel du groupe tactique du projet RFM

Fin de l'utilisation du patron « Formaliser les besoins utilisateurs »

Fin de l'utilisation du patron « Analyser les besoins tactiques »

Début de l'utilisation du patron « Analyser les besoins stratégiques »

Suivant la solution-démarche du patron « Analyser les besoins stratégiques », nous procédons à l'analyse des besoins stratégiques *via* l'utilisation du patron « Collecter les besoins utilisateurs » (CBU).

Fin de l'utilisation du patron « Collecter les besoins utilisateurs »

La solution-démarche du patron CBU préconise de sélectionner les tableaux croisés du groupe d'acteurs qui ne se recouvrent pas. Le tableau croisé extrait du document d'expression des besoins du groupe stratégique est présenté dans le tableau 6.5.

Force Motrice	Temps.Mois		
	Janvier	Février	Mars
Energie produite	40 568 823,85	35 632 24,30	45 263 224,54
Volume turbiné	950 758,19	1 095 058,19	1 116 726,79
Montant redevance	9 757 215,25	9 895 456,38	10 823 340,82

TAB. 6.5 – Tableau croisé synthétique de la RFM sur le premier trimestre 2007

Le troisième point de la solution-démarche préconise d'interroger ces acteurs. Nous utilisons le patron « Interroger les acteurs du SID ». Nous les interrogeons pour les propriétés suivantes :

- les habitudes informatiques,
- les aptitudes informatiques,
- la criticité des données,
- le reporting des données,
- l'analyse des données,
- les modalités de restitution,
- la fragmentation de restitution,
- le rafraîchissement des données,
- l'historisation des données,
- l'archivage,
- la consolidation des données,
- le calcul des données,
- le mode de pilotage de l'organisation,
- le périmètre du projet.

A partir des réponses aux questions relatives à ces propriétés et au poids de ces propriétés. Nous annotons le graphe de propriétés défini par le patron produit « Graphe de propriétés ».

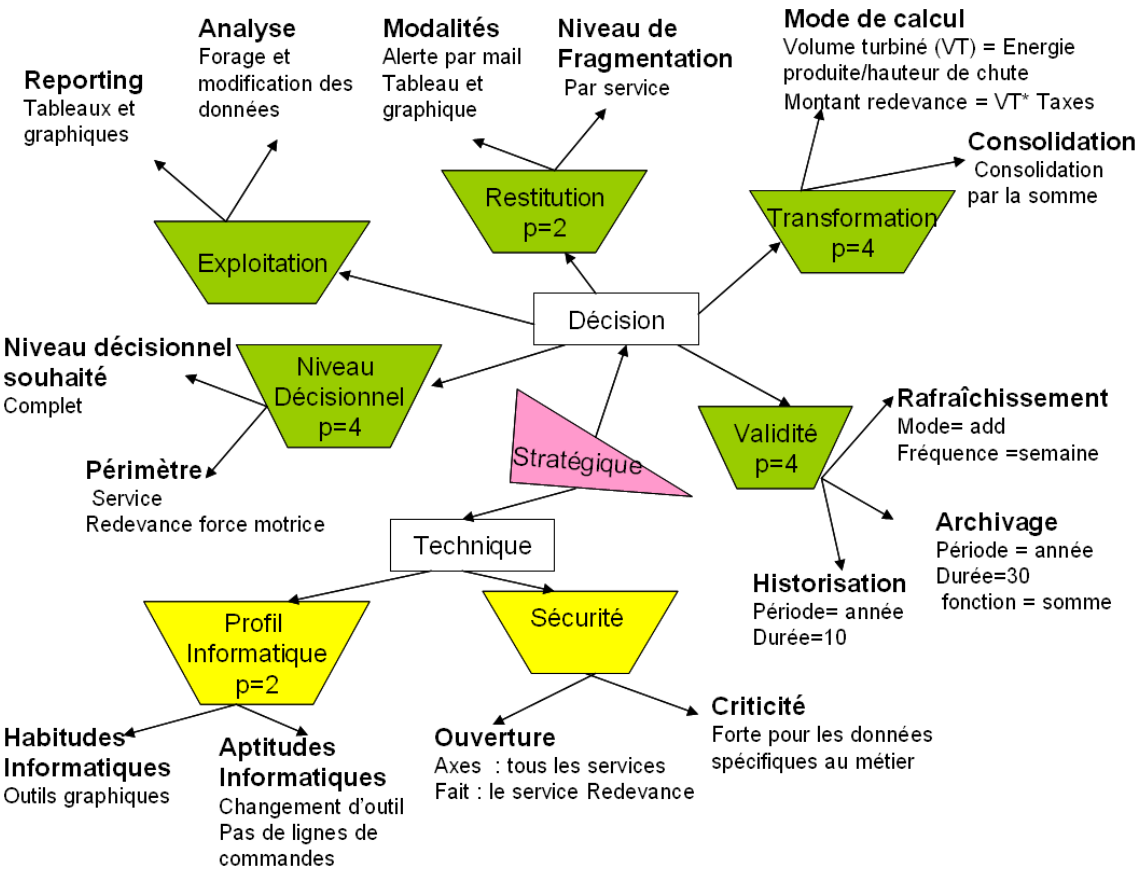


FIG. 6.43 – Graphe de propriétés CSD du groupe stratégique du projet RFM

Fin de l'utilisation du patron « Collecter les besoins utilisateurs »

Début de l'utilisation du patron « Formaliser les besoins utilisateurs »

La formalisation des besoins stratégiques est réalisée à partir de l'utilisation du patron « Formaliser les besoins utilisateurs ». La mise en oeuvre des règles de transformation et des règles syntaxiques suivie de l'utilisation du patron produit « Diagramme décisionnel » génère le diagramme décisionnel du groupe stratégique présenté dans la figure 6.44

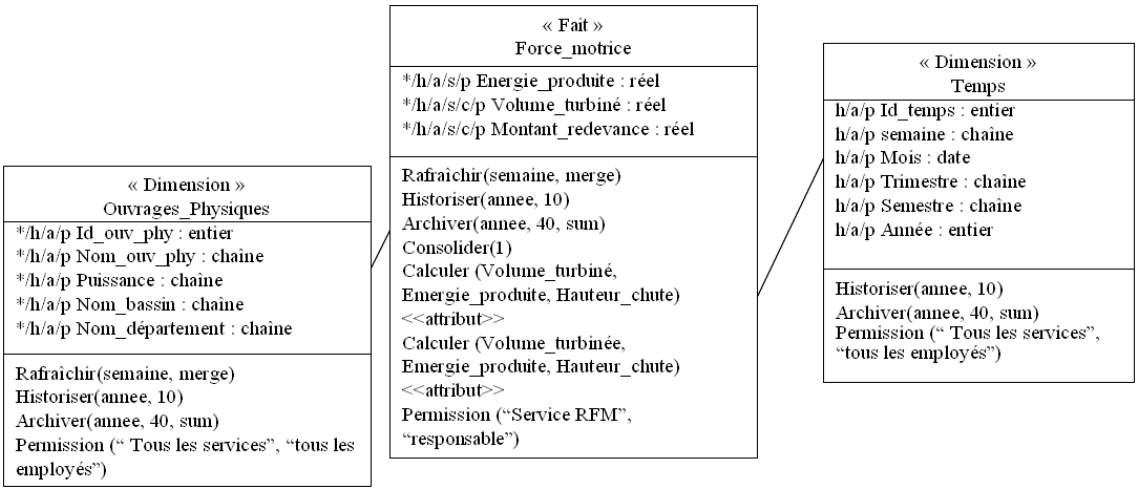


FIG. 6.44 – Diagramme décisionnel associé au tableau 6.5

Fin de l'utilisation du patron « Formaliser les besoins utilisateurs »

Fin de l'utilisation du patron « Analyser les besoins stratégiques »

Début de l'utilisation du patron « Analyser les besoins systèmes »

L'analyse des besoins systèmes suivant la solution démarche du patron processus « Analyser les besoins systèmes » utilise le patron « Collecter les besoins systèmes » (CBSY) pour la collecte de ces besoins. La solution démarche du patron CBSY consiste à sélectionner les systèmes existants aussi bien les sources de données que les modules décisionnels. Nous utilisons le patron « Modèle entité-association » pour évaluer le schéma de données des sources de données. Le schéma entité-association récupéré par rétro-ingénierie est présenté dans la figure 6.45. Puis, nous utilisons le patron produit « Fiche signalétique » pour la définition de cette source de données. Pour des raisons de confidentialité à l'égard de l'organisation cliente de la société I-D6, nous ne présentons pas la fiche signalétique de la source de données.

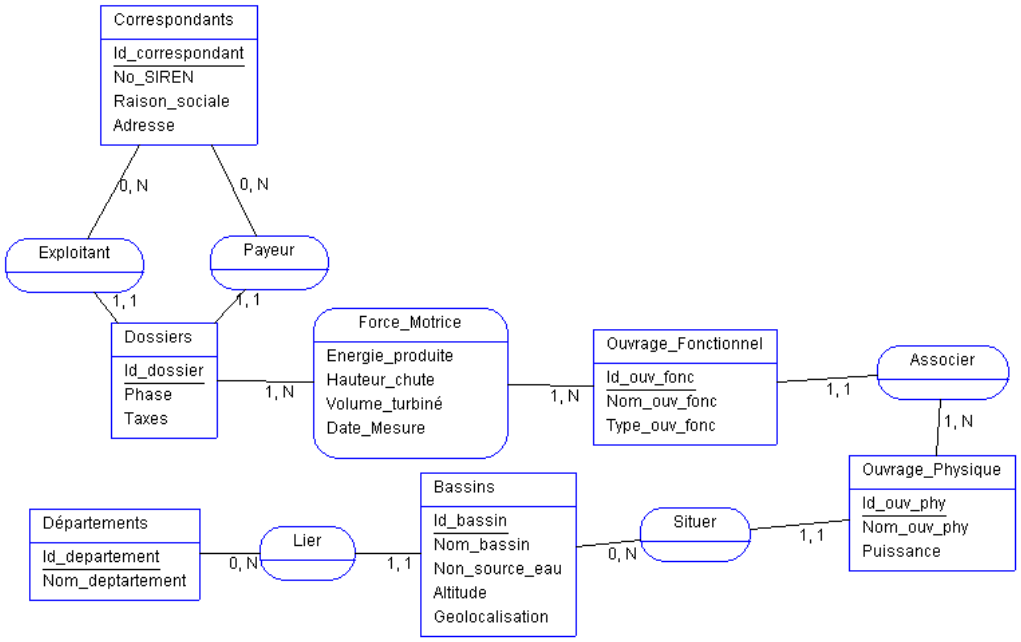


FIG. 6.45 – Modèle entité-association de l'application RFM

Nous utilisons le patron « Interroger les acteurs du SID » pour l'interview des acteurs systèmes. Nous posons les questions liées aux propriétés suivantes ainsi que celles liées aux poids des catégories de propriétés pour ces acteurs systèmes. Les propriétés pour lesquelles nous les interrogeons sont :

- le suivi des erreurs,
- la remontée des erreurs,
- l'ouverture des sources,
- la disponibilité des sources,
- l'hétérogénéité des sources,
- la couverture des sources,
- la complexité des sources,
- le rafraîchissement des données,
- l'historisation des données,
- l'archivage des données,
- la consolidation des données,
- le calcul des données,
- la dichotomie des espaces de stockage,
- la normalisation du schéma du SID.

A partir des réponses, nous définissons le graphe de propriétés (cf. figure 6.46) du groupe système suivant l'utilisation du patron produit « Graphe de propriétés ».

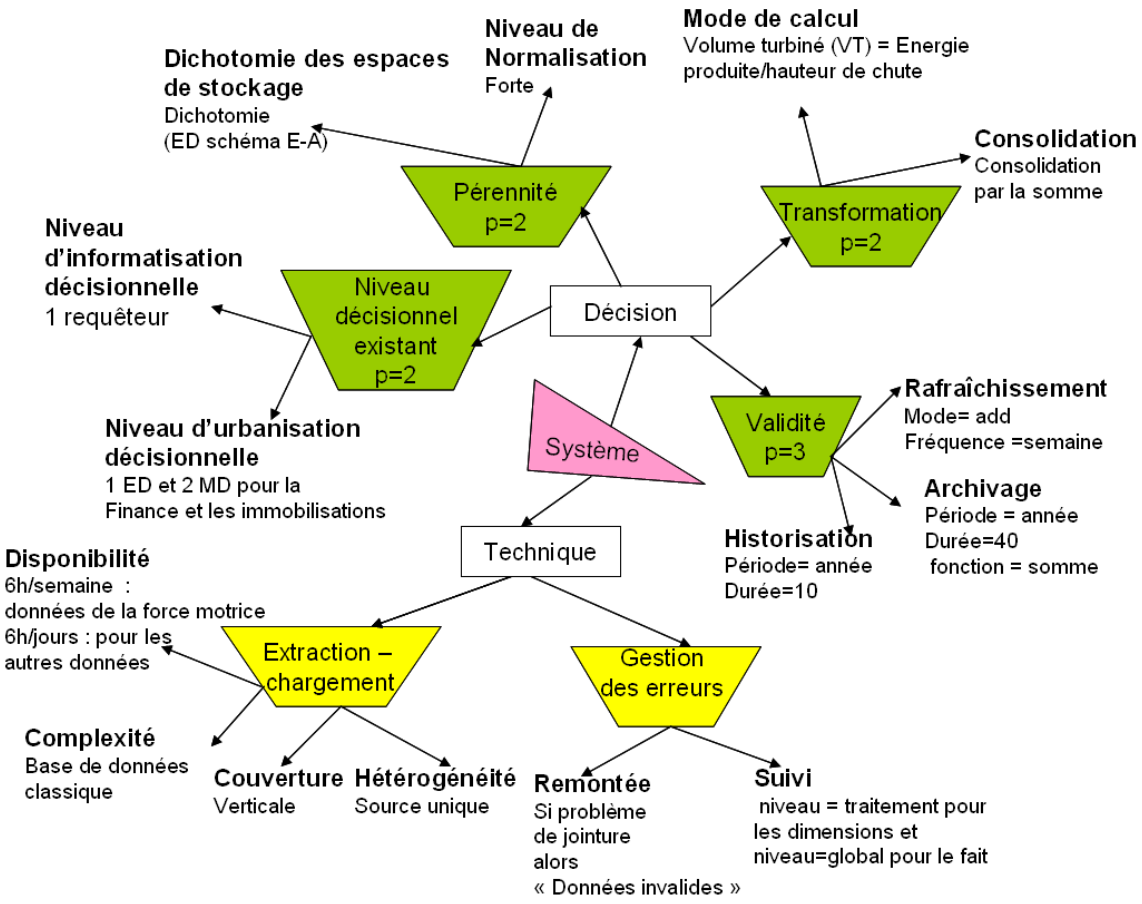


FIG. 6.46 – Graphe de propriétés CSD du groupe système du projet RFM

Fin de l'utilisation du patron « Collecter les besoins systèmes »

Début de l'utilisation du patron « Formaliser les besoins systèmes »

L'utilisation du patron « Formaliser les besoins systèmes » permet la formalisation des besoins systèmes. Nous appliquons uniquement les règles de sélection, de structuration et syntaxiques car il y a qu'une source et donc qu'un diagramme décisionnel. Nous obtenons la matrice des faits candidats présentée dans le tableau 6.6

		N_liens		
		3	2	1
N_att	3		Force_Motrice	
	2			
	1	Dossier, Bassins Correspondants, Ouvrage_Physiques		

TAB. 6.6 – Matrice des faits candidats du projet RFM

Puis, suivant le patron produit « Diagramme décisionnel », nous définissons le diagramme décisionnel du groupe système présenté dans la figure 6.47.

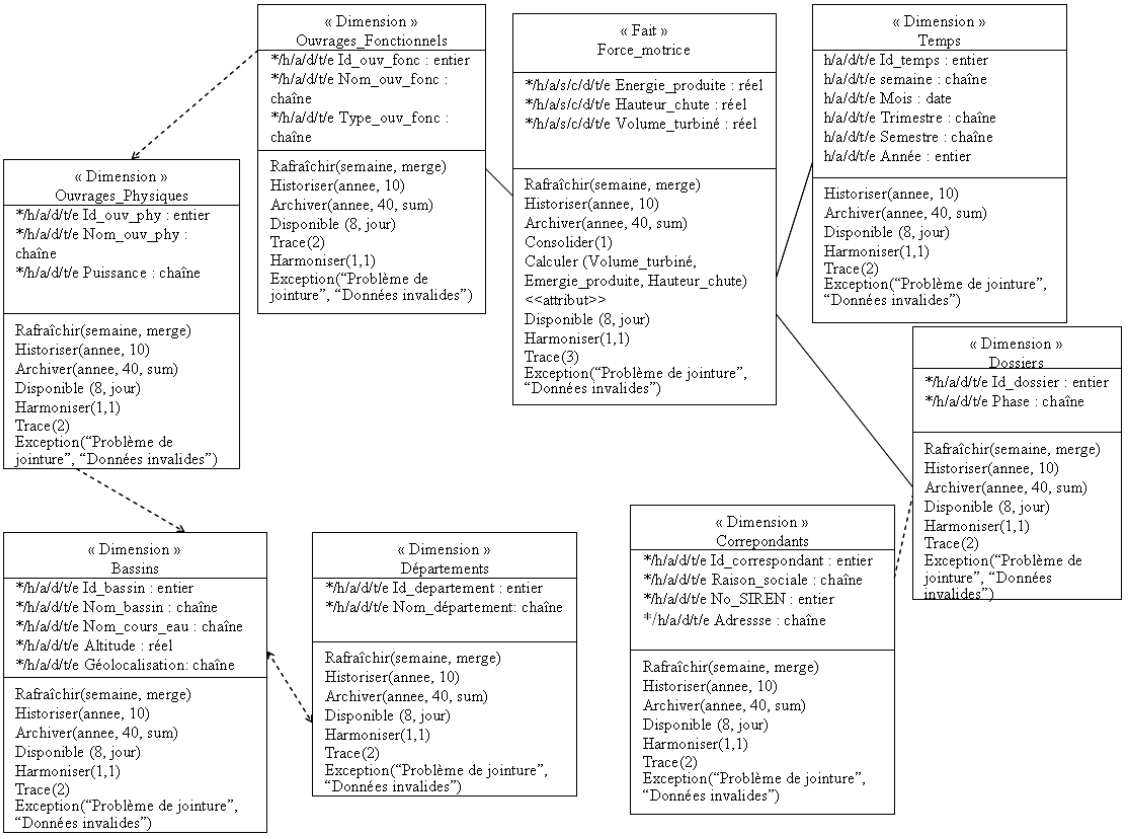


FIG. 6.47 – Diagramme décisionnel du groupe système du projet RFM

Fin de l'utilisation du patron « Formaliser les besoins systèmes »

Début du patron « Evaluer la connaissance » pour l'évaluation de la connaissance utilisateurs

Puis, suivant la solution démarche du patron « Analyser un SID », l'évaluation de la connaissance utilisateurs se réalise avec le patron « Evaluer la connaissance ». Ce patron requiert deux diagrammes décisionnels. Pour l'évaluation de la connaissance utilisateurs, les diagrammes sont le diagramme décisionnel du groupe tactique et le diagramme décisionnel du groupe stratégique. L'application des règles de confrontation utilisateur génère le diagramme décisionnel présenté dans la figure 6.48.

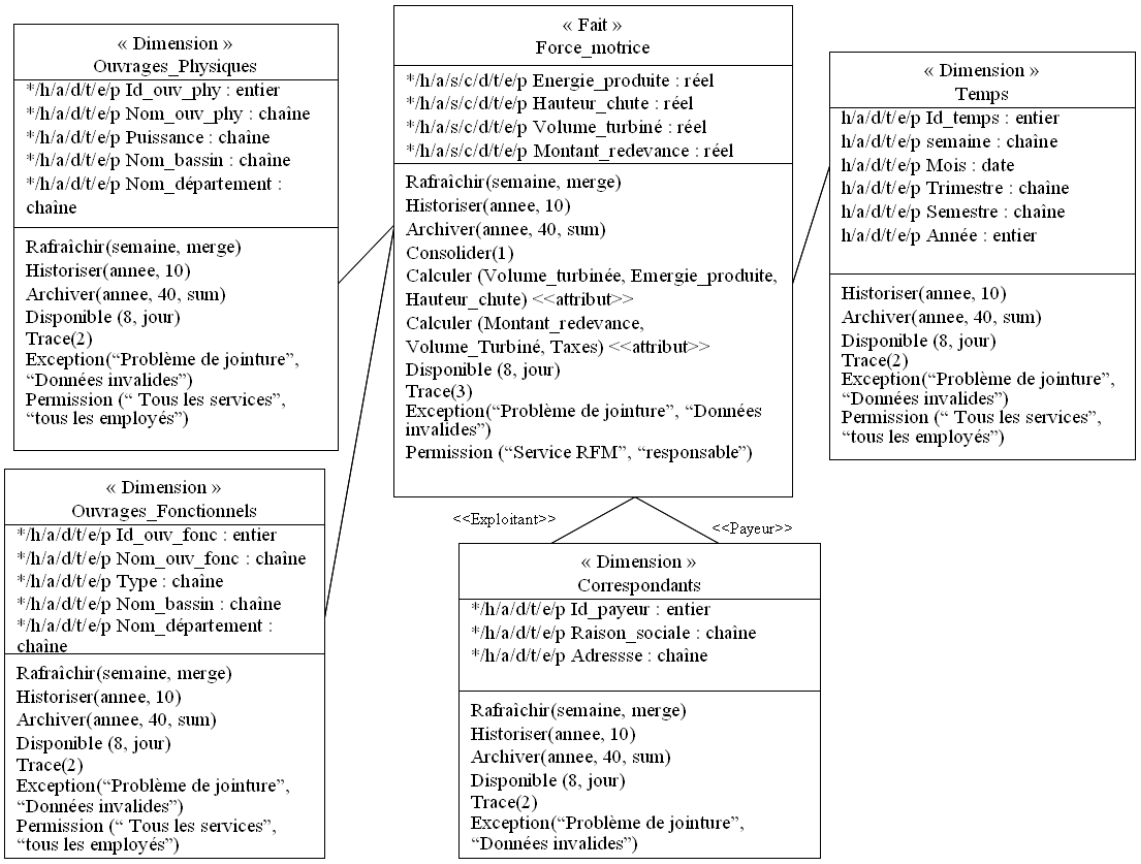


FIG. 6.48 – Diagramme décisionnel utilisateur du projet RFM

Fin de l'utilisation du patron « Evaluer la connaissance » pour l'évaluation de la connaissance utilisateurs

Début de l'utilisation du patron « Evaluer la connaissance » pour l'évaluation de la connaissance de l'environnement

Enfin, suivant la solution-démarche du patron « Analyser un SID », l'évaluation de la connaissance de l'environnement se réalise avec le patron « Evaluer la connaissance ». Pour l'évaluation de la connaissance de l'environnement, les deux diagrammes requis par ce patron sont le diagramme décisionnel utilisateur et le diagramme décisionnel du groupe système.

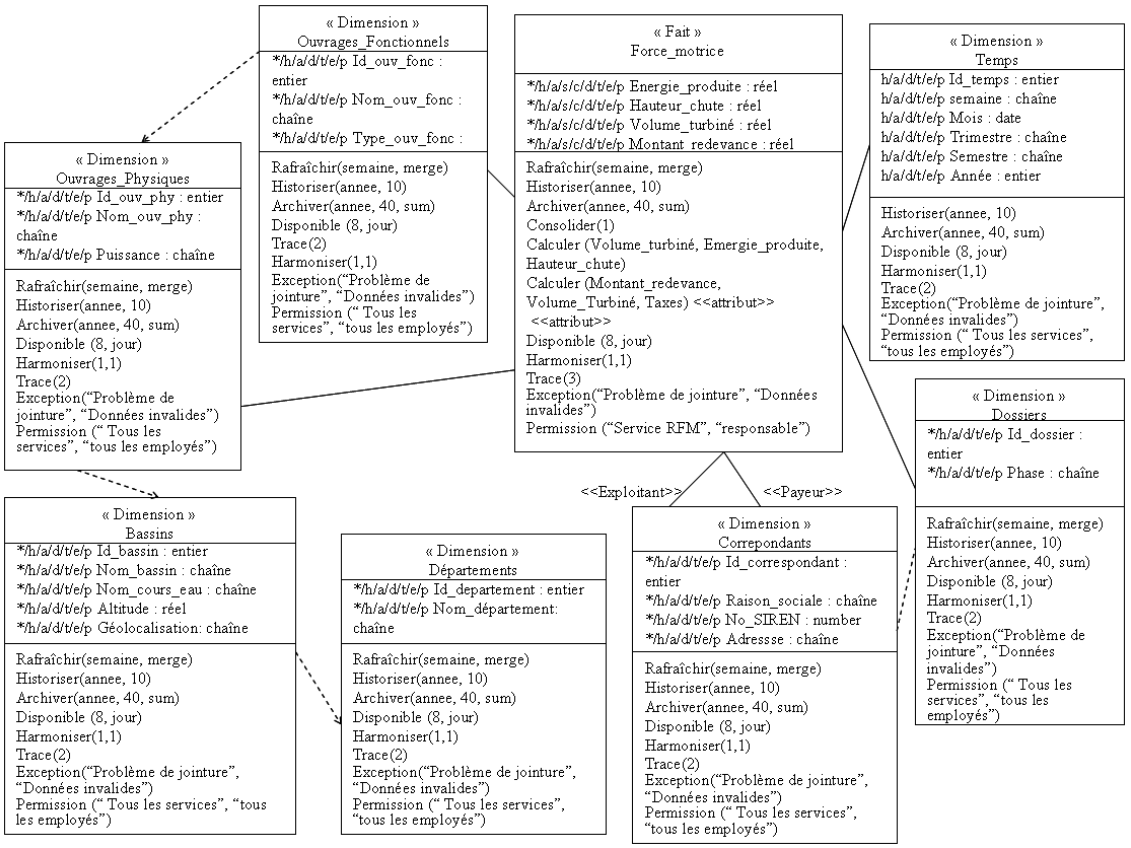


FIG. 6.49 – Diagramme décisionnel du SID du projet RFM

Fin du patron « Evaluer la connaissance » pour l'évaluation de la connaissance de l'environnement

La source de données fournit les données nécessaires pour répondre aux besoins des utilisateurs, la phase d'analyse est donc close. Nous passons donc à la phase de conception.

6.4.2 Réutilisation des patrons au cours de phase de conception

Suivant la solution-démarche du patron « Mettre en place un SID », la conception du SID est réalisée par l'utilisation du patron « Concevoir le SID ». Ce patron requiert l'application du patron ASID car l'itération en cours est égale au nombre d'itération maximal, x=k=1. Nous avons déjà appliqué ce patron donc, le patron CSID peut être instancié. Suivant, sa solution-démarche, nous utilisons séquentiellement les patrons « Choisir l'architecture décisionnelle » et « Concevoir les modules du SID ». Nous n'appliquons les patrons « Définir les schémas de correspondances »

et « Evaluer les schémas de correspondances » car nous n'avons pas fait de propositions spécifiques concernant cette thématique.

Début du patron « Choisir l'architecture décisionnelle »

Le patron « Choisir l'architecture décisionnelle » requiert trois patrons « Graphe de propriétés » afin de disposer des graphes de propriétés des trois groupes d'acteurs. Ces trois graphes de propriétés ont été définis au cours de l'analyse. Nous pouvons donc instancier le patron CAD. Suivant, sa solution-démarche, nous définissons les critères comme indiqué dans le tableau 6.7.

Critère	**Propriétés**	**Valeurs des propriétés**	**Valeurs du critère**
Niveau de couverture des données (NCD)	Périmètre, Couverture	Service RFM, Verticale	vertical
Niveau de traitement des données (NTD)	Hétérogénéité, Calcul	Source unique, Volume turbiné (VT) = Energie produite/hauteur de chute et Montant redevance = VT* Taxes	peu travaillé
Niveau de complexité des sources (NCS)	Complexité	Base de données classiques	peu complexe
Niveau d'équipement décisionnel existant (NEE)	Niveau d'urbanisation décisionnelle	1 ED et 2 MD (Finance et Immobilisations)	{ED, 2 MD}
Niveau de pérennité de l'architecture (NPA)	Normalisation du schéma du SID	Forte	Pérenne
Niveau décisionnel souhaité (NDS)	Niveau décisionnel souhaité	Complet	Complet

TAB. 6.7 – Propriétés par critère de quantification

A partir de ces valeurs de niveaux nous appliquons la fonction « Choix_architecture() » et nous obtenons l'architecture composée d'un entrepôt de donnée et de trois magasins de données. Un nouveau magasin a été crée car les trois classes d'utilisateurs qui sont liées respectivement à la finance, aux immobilisations et à la redevance force motrice sont distinctes. La réutilisation du patron CAD couplée à cette automatisation guident et accélèrent le choix de l'architecture adaptée au projet décisionnel.

Fin du patron « Choisir l'architecture décisionnelle »

Nous utilisons maintenant le patron « Concevoir les modules décisionnels » pour la conception des nouveaux modules décisionnels. Ce patron requiert le patron CAD que nous avons déjà utilisé au cours de cette itération, nous pouvons donc l'instan-

cier. Le nouveau module à concevoir est un magasin de données, nous utilisons donc le patron « Transformer le DD en MMDT » (TDM).

L'utilisation du patron TDM requiert le patron DD pour la définition du schéma conceptuel multidimensionnel de données à partir du diagramme décisionnel du SID. Suivant la solution-démarche, à partir du DD du SID, nous appliquons les règles de passage pour les classes-faits et les classes-dimensions et nous obtenons le schéma conceptuel multidimensionnel de données et de traitements de notre projet décisionnel RFM présenté dans la figure 6.50 (dont la légende est présentée dans la figure 6.51).

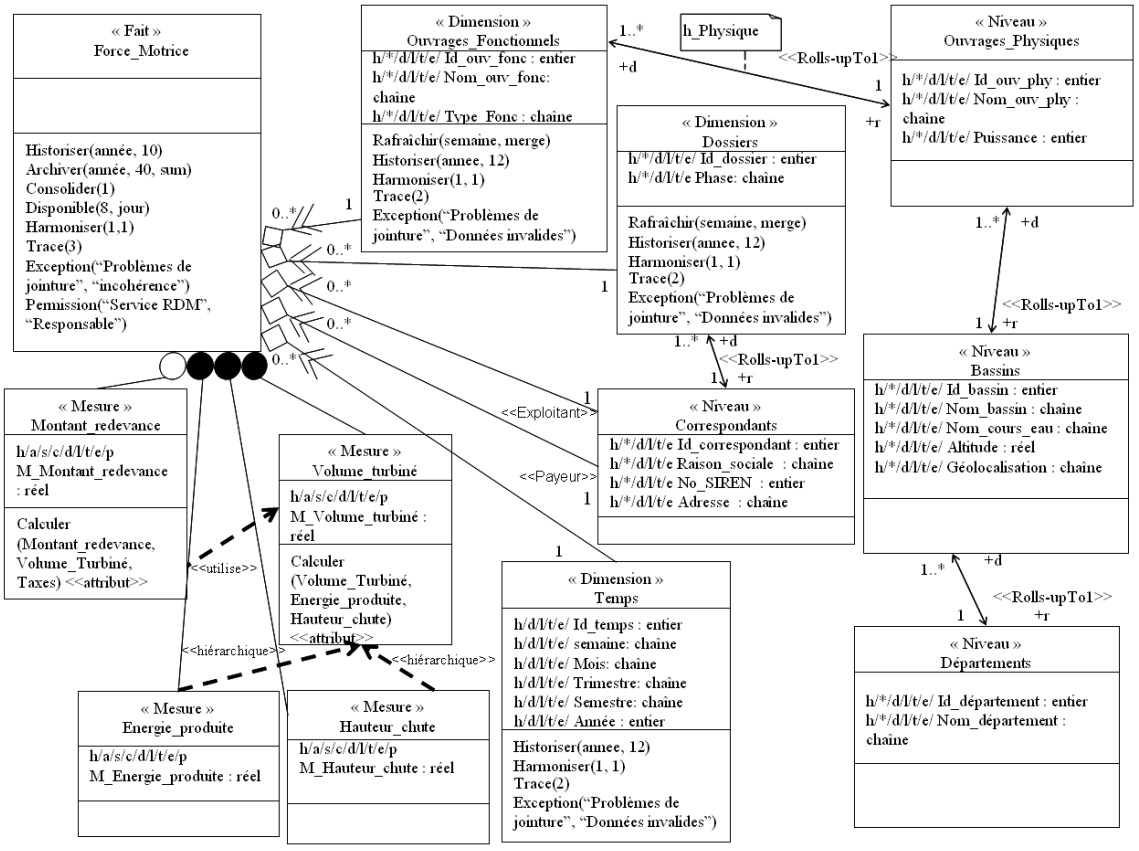


FIG. 6.50 – Schéma conceptuel multidimensionnel du magasin de données du projet RFM

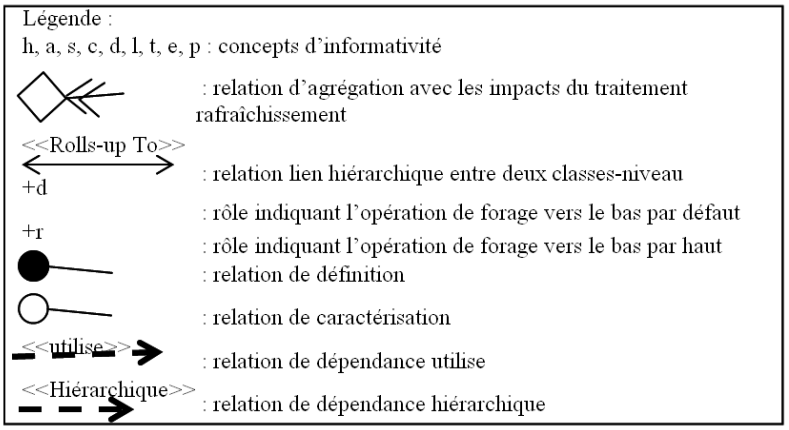


FIG. 6.51 – Légende du schéma conceptuel multidimensionnel du magasin de données du projet RFM

6.5 Conclusion

Nous proposons un ensemble de patrons dont la granularité est liée à la décomposition de notre processus de développement de SID. Ces derniers assurent la traçabilité de la documentation SID et favorisent la réutilisation systématique des patrons dans les différents contextes liés au décisionnel *via* la formalisation du contexte et la gestion de la documentation.

Pour cela, nous avons étendu le formalisme P-SIGMA par l'ajout dans la partie « Réalisation » des patrons de deux rubriques supplémentaires « Documents sources » et « Documents cibles » et par la définition de deux fonctions u_s et r_s indiquant si les conditions d'utilisation d'un patron p_s sont vérifiées respectivement, si les conditions dans lesquelles un patron p_s requiert un patron p_r sont vérifiées.

L'intérêt de notre catalogue est de favoriser la réutilisation systématique lors de l'analyse et de la conception des SID. De plus, il procure comme tout catalogue de patrons les avantages suivants à notre méthode de développement des SID :

- faciliter l'interchangeabilité des ressources humaines sur un projet. En effet, les débutants disposent de l'expertise de l'ingénierie des SID dans le catalogue de patrons,
- instaurer un vocabulaire commun et univoque. Toutes les tâches sont associées à des patrons dont le problème et la solution proposée dans un contexte donné y sont définis explicitement,
- favoriser un gain de temps et une augmentation de productivité. La méthode étant définie, il faut uniquement la réutiliser en suivant l'enchaînement des patrons donné par les relations inter-patrons. « Les concepteurs écriront moins pour mieux produire » [Meyer 2000],

- favoriser une communication régulière et approuvée entre les différents acteurs et de faciliter la future maintenance car le formalisme des patrons P-SIGMA étendu, que nous avons aussi proposé, intègre la gestion des documents.

Par ailleurs, il permet spécifiquement :
- une réutilisation plus efficace via la formalisation du contexte et la définition de patrons dont la granularité est basée sur celle des éléments processus et produit de l'ingénierie des SID,
- un processus de développement guidé tout au long du cycle de vie car l'identifiant de tout patron de notre catalogue est un objectif de développement,
- une traçabilité des développements qui permet aux concepteurs décisionnels de tirer profit, en l'occurrence de réutiliser, des processus réalisés et produits obtenus lors de la réutilisation des patrons.

Troisième partie

Bilan industriel et développements

Chapitre 7

Applications et développements

Table des matières

7.1 Introduction **245**
7.2 Bilan des projets décisionnels dans l'industrie **245**
7.2.1 Montée en compétence sur l'analyse des besoins 246
7.2.2 Montée en compétence sur la conception du SID 247
7.3 Démarche qualité suivant la Roue de Deming **248**
7.3.1 Présentation de la Roue de Deming 248
7.3.2 Mise en application de la Roue de Deming 249
7.4 Développement de l'outil eBIPAD **251**
7.4.1 Généralités de l'outil 251
7.4.2 Fonctionnalités générales 253
7.4.3 Fonctionnalités pour l'ingénieur de patrons 255
Cas d'utilisation « **Ajouter un patron** » 256
Cas d'utilisation « **Ajouter un champ** » 259
7.4.4 Fonctionnalités pour l'ingénieur d'applications 262
Cas d'utilisation « **Accéder à la solution-démarche** » 263
Cas d'utilisation « **Rechercher un patron** » 267
7.4.5 Fonctionnalités pour l'administrateur système 269
7.4.6 Base de patrons de eBIPAD 269
Spécifications de l'outil 270
Schémas de la base de patrons 272
7.5 Evaluation de nos propositions **275**
7.5.1 Synopsis de nos propositions 275
7.5.2 Avantages de nos propositions 277
7.5.3 Inconvénients rencontrés de nos propositions 279
7.6 Conclusion **280**

7.1 Introduction

Dans le cadre de la collaboration industrielle [I-D6 2007], nous avons mené à bien une dizaine de projets décisionnels pour la définition de notre méthode de développement et de notre catalogue de patrons. Au cours de ceux-ci, nous avons rencontré des problèmes liés principalement à l'analyse des besoins et à la conception des SID. Les propositions associées ont été définies suivant le principe de la démarche qualité de la Roue de Deming afin de mettre à profit le cadre industriel d'infogérance. Les itérations de cette démarche qualité ont mis en avant des problèmes concernant la méthode, que nous avons résolus et présentés dans les chapitre 3, 4 et 5, et des problèmes concernant l'utilisation du catalogue de patrons, que nous présentons dans ce chapitre.

Ainsi dans la section 7.2, nous présentons le bilan des projets décisionnels à partir desquels nous avons défini nos propositions. Puis dans la section 7.3, nous présentons la mise en application de la démarche qualité dans le cadre de la définition de ces propositions. Dans la section 7.4, nous décrivons l'outil eBIPAD développé pour répondre aux problèmes posés par l'utilisation du catalogue de patrons. Dans la section 7.5, nous présentons l'évaluation et l'expérimentation de nos propositions. Enfin, dans la section 7.6, nous concluons par la synthèse de nos applications, de nos développements et de l'évaluation de nos propositions.

7.2 Bilan des projets décisionnels dans l'industrie

Les projets menés dans le cadre de la collaboration avec la société I-D6 ont concerné le domaine de la finance, de la grande distribution, de l'environnement, du médical, de la pharmaceutique. Ils avaient pour objectif de mettre en place des SID liés à des métiers tels que la finance, la comptabilité, la redevance, les immobilisations. Le domaine et le métier ainsi que les organisations clientes changeaient régulièrement. Ces changements nous ont permis de constater qu'au-delà des différences de terminologies des métiers, le processus d'ingénierie des SID présentait des tâches récurrentes d'un métier à un autre, d'un domaine à un autre et d'un projet à un autre.

Cependant, la récurrence de ces tâches ne font pas d'elles des tâches maîtrisées par les concepteurs décisionnels car il n'y a pas de méthode qui guide ces tâches spécifiques aux SID tels que l'analyse des besoins du SID et le choix de l'architecture décisionnelle. Nous ne présentons pas un projet particulier mais, nous dressons une synthèse des problèmes et des procédures de définition des solutions qui ont jalonnés nos travaux de recherche.

7.2.1 Montée en compétence sur l'analyse des besoins

L'analyse des besoins au cours d'un projet décisionnel était souvent occultée dans le sens où, après la collecte des besoins au cours de nombreuses réunions, les concepteurs décisionnels passaient directement à la modélisation conceptuelle des données du SID. De plus, le dictionnaire des données des sources n'était pas toujours disponible et la sémantique des données contenues dans les sources n'était pas nécessairement connue par un membre de l'organisation cliente.

Lors de l'implantation des SID, les développements rencontraient de nombreuses difficultés car il manquait des informations importantes pour la mise en place du SID. Cela nécessitait des demandes de compléments d'information aux utilisateurs. Face à ce problème, le contexte technique des données a été précisé afin de charger dans le SID les bonnes données issues des bonnes sources. Ces résultats ont été acceptables.

Suite à des changements d'exercices comptables, des problèmes concernant la qualité des données pour la prise de décision ont été détectés. Ces derniers ont mis en lumière l'importance de la prise en compte du contexte décisionnel des données, au plus tôt dans le processus d'ingénierie pour assurer la fiabilité des données. Les règles de calcul ont été précisées afin d'évaluer les liens entre les données. Cela a permis un contrôle progressif, organisé et plus rapide des données chargées dans le SID, bien que les propositions formulées n'étaient pas suffisantes. En effet, les critères de conservation et de mise à jour des données que nous avions interprétés ne correspondaient pas aux besoins réels de l'organisation cliente en question. Afin de détecter ces problèmes avant la conception du SID, nous avons procédé en deux étapes.

Dans un premier temps, afin d'éviter une expression informelle dont l'interprétation serait biaisée par la connaissance technique des concepteurs décisionnels, nous avons demandé les tableaux de bord que l'organisation cliente utilisait ou souhaitait manipuler. Ces tableaux ont mis en avant des périodes d'historisation et d'archivage que les utilisateurs n'avaient pas mentionnées au cours des réunions. Nous avons donc décidé d'ajouter ces deux colonnes dans une version étendue du traditionnel dictionnaire de données. Cette structure était simple sur des projets de petite taille mais, pour des projets de grosse taille, ce qui est plus souvent le cas, elle était difficilement « gérable ». Elle pouvait donc difficilement faire l'objet d'une maintenance. Nous avons souhaité structurer les réunions d'interviews des utilisateurs car de nombreux points n'étaient pas traités suivant l'expérience des concepteurs décisionnels ou d'autres facteurs. Nous avons donc défini des propriétés importantes des SID et nous les avons regroupés par catégorie. Ces propriétés ont été représentées sous forme d'un graphe orienté afin de guider le concepteur et de bénéficier de la possibilité d'annoter les noeuds. De ce fait, au cours des projets suivants, nous disposions des informations nécessaires pour l'implantation. De plus, l'ordre et la hauteur de ce graphe sont toujours les mêmes indépendamment de la taille du projet.

Cependant, un problème majeur persistait : comment exploiter de manière pertinente et univoque ces propriétés ? Ainsi, à l'instar des traitements ETL définis après la conception du schéma du SID, nous avons défini pour la phase d'analyse des traitements liés à la dérivation et à la préparation des données. Ces traitements montrent leur pertinence car les demandes de compléments d'information diminuaient et les implantations *via* les outils ETL se sont révélées plus rapides.

Dans un second temps, la confrontation entre les sources de données et le SID était trop tardive car elle se faisait entre le schéma conceptuel du SID et les schémas de données des sources. De plus, ces schémas sont difficilement comparables. De ce fait, nous avons réfléchi à un modèle qui vise à représenter aussi bien les concepts nécessaires aux décideurs que ceux nécessaires aux concepteurs décisionnels et qui soit compréhensible par tous les acteurs du projet. Notre modèle multidimensionnel de besoins en est le résultat.

7.2.2 Montée en compétence sur la conception du SID

En parallèle de cette montée en compétence sur l'analyse des besoins, nous avons rencontré des problèmes concernant la conception.

Les modules décisionnels présents dans les organisations ne sont pas nécessairement adaptés aux besoins qui étaient à leur origine comme le confirme le chiffre présenté par [Schiefer et al. 2002], soit 40% des projets décisionnels qui échouent. Nous avons constaté que le choix des modules est souvent imposé par le marché des outils décisionnels. Cependant, les propriétés du SID exprimées au travers des besoins des acteurs concernent ce choix. Nous avons évalué neuf propriétés qui orientent le choix de l'architecture. A partir de ces propriétés, nous avons défini des critères de quantification afin de déterminer l'architecture la mieux adaptée à un projet.

La modélisation qui est la plus utilisée pour la définition des schémas de ces modules décisionnels, soit la modélisation multidimensionnelle, n'intègre pas actuellement la représentation des traitements au cours de la conception du SID. Afin de prendre en compte de manière continue les traitements au cours du processus d'ingénierie, nous avons étendu un modèle conceptuel de données existant.

De plus, nous avions constaté que la précision des règles de calcul et des liens entre les données augmentait la fiabilité des données pour les décideurs et améliorait le contrôle des données par les concepteurs décisionnels. Nous avons donc représenté la pertinence de ces liens entre les données sur notre modèle de données et de traitements.

7.3 Démarche qualité suivant la Roue de Deming

Nous avons défini notre méthode de développement de SID suivant une démarche qualité [Annoni et al. 2006d].

7.3.1 Présentation de la Roue de Deming

Les principales méthodes d'amélioration de la qualité proposées par Phillip Crosby [Crosby 2006], Joseph Juran [Juran 2006] sont liées aux travaux de William Edwards Deming. La **Roue de Deming**, présentée dans la figure 7.1, proposée dans ses travaux [Deming 1953] est celle qui a montré les performances les plus importantes aussi bien dans le domaine économique avec la révolution qualité du Japon et le prix Deming de la qualité que dans d'autres domaines tel que le développement de logiciel [Paulk et al. 1991]. L'idée principale de cette méthode est de commencer par améliorer ce que nous savons faire, mais pas encore assez bien, ensuite nous innoverons, mais pas l'inverse. Ce qui concorde avec l'objectif de nos travaux de recherche : **améliorer ce que la société I-D6 fait au quotidien et proposer des outils et des techniques afin de répondre à ses besoins de méthode efficace, fiable et de réutilisation des composants mis en oeuvre au cours des projets antérieures**.

Cette démarche qualité repose sur un cycle de vie itératif incrémental. Une itération est décrite en quatre temps qui sont les secteurs de cette roue que nous avons traduit par : Cadrage, Développement, Évaluation et Capitalisation-Planification. La définition de ces secteurs présentés dans la figure 7.1 est :

- Cadrage : correspond au secteur "Plan". C'est l'étape au cours de laquelle nous définissons les objectifs et la démarche pour les atteindre,
- Développement : correspond au secteur "Do". C'est l'étape durant laquelle, nous développons les solutions pour répondre aux objectifs définis précédemment,
- Évaluation : correspond au secteur "Check". C'est l'étape durant laquelle, nous appliquons les propositions de solutions dans une optique de validation,
- Capitalisation-Planification : correspond au secteur "Act". C'est l'étape durant laquelle nous définissons et capitalisons ce qui est stable dans notre solution. Puis, nous déterminons les écarts entre notre proposition de solution et nos attentes afin de définir les orientations de la prochaine étape de cadrage.

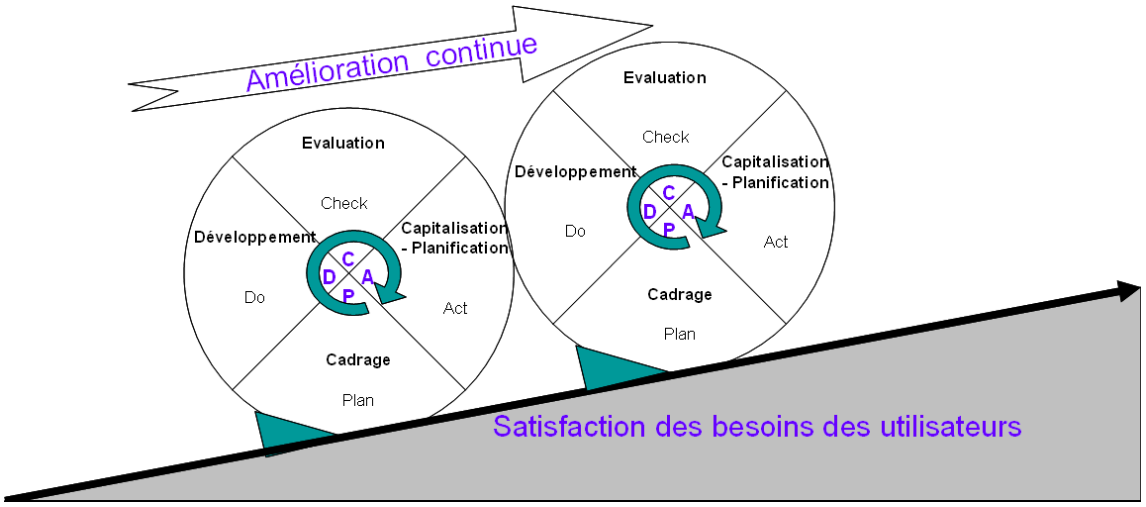


FIG. 7.1 – La démarche qualité Roue de Deming

7.3.2 Mise en application de la Roue de Deming

Pour répondre aux besoins de la société I-D6, nous avons itéré quatre fois la Roue de Deming. Nous avons étudié la documentation existante des projets réalisés au sein de la société afin de dresser un état de l'existant des modèles, des méthodes, des techniques, des outils utilisés par la société I-D6. Cet état a mis en avant un besoin d'homogénéisation et d'harmonisation des terminologies et des méthodes utilisées au sein de la société. Nous avons donc regroupé les concepteurs décisionnel en trois groupes suivant l'expertise dans l'ingénierie des SID (expert, intermédiaire, débutant). Un concepteur expert est un collaborateur ayant une expérience et une maîtrise des projets SID ; il fait preuve d'une bonne expertise de toutes les étapes de l'ingénierie des SID. Au contraire, un concepteur débutant est celui qui débute dans le développement de projets SID. Entre les deux, il y a les collaborateurs qui ont de l'expérience mais pas suffisamment pour diriger un projet SID.

Nous avons travaillé d'itération en itération avec un nombre croissant de ces groupes de collaborateurs afin de satisfaire les besoins des différents groupes. Plus précisément, nous intégrons progressivement, itération par itération, les besoins des concepteurs décisionnels, des plus expérimentés aux débutants. Ainsi, au fil des itérations, la méthode se généralise à l'ensemble des exigences de ces derniers et se raffine grâce aux tâches plus ou moins simples qui auraient pu être occultées par les experts.

A chaque étape, nous posons nos hypothèses, nos bases et nos objectifs afin de mettre en avant des éléments de solution. Nous avons validé partiellement nos éléments de solution en implantant la solution qui répondait aux objectifs définis ensemble. Ces travaux nous ont permis de mettre en lumière des configurations

d'erreurs et des points incontournables à vérifier pour le développement d'un SID satisfaisant les besoins des tous les acteurs.

Au début du processus de définition de la méthode, nous avons recueilli auprès des collaborateurs experts les informations suivantes :
- la démarche mise en oeuvre à chaque début de projet,
- les modèles les plus utilisés,
- les tâches récurrentes d'un projet à l'autre,
- les tâches récurrentes d'un domaine à l'autre,
- les pièges récurrents lors de l'analyse et de la conception,
- les solutions alternatives utilisées lors de la conception et l'analyse.

Nous avons défini la démarche, les modèles, la méthode et le catalogue de patrons réutilisables à partir de trois mécanismes qui sont la récurrence des modèles, des outils, des techniques et des tâches mis en oeuvre, les retours de qualité des organisations clientes et l'évaluation qualitative des temps de mise en oeuvre de chaque étape au regard de la taille du projet.

L'adéquation et le passage d'une itération à l'autre sont fonction de la satisfaction par la méthode des besoins des différents concepteurs décisionnels de la société I-D6. La méthode résultante de chaque itération a été testée progressivement par 1 à 3 groupes de collaborateurs afin d'évaluer :
- l'application de la démarche,
- l'adéquation de nos modèles,
- l'application du catalogue de composants réutilisables,
- la pertinence de l'enchaînement des tâches,
- la validité de la solution proposée.

Chaque itération a été clôturée par la remise de fiches d'évaluation. Toutes les itérations n'ont pas eu la même durée comme présenté dans la figure 7.2. Les deux itérations les plus longues sont la première et la quatrième. La première a été très longue car nous avons pris soin de bien poser les hypothèses de travail et de recueillir toute la documentation industrielle et de recherche. La quatrième itération a duré car nous avons développé le prototype pour une gestion plus efficace de notre catalogue de patrons.

Ainsi, suivant la démarche qualité de la Roue de Deming, la méthode d'analyse et de conception des SID que nous proposons résulte, dans un premier temps, de l'harmonisation de l'expérience et l'expertise des concepteurs décisionnels d'I-D6. Dans un second temps, nous avons innové pour répondre aux besoins de la société et à certaines limites des propositions existantes dans le domaine des SID.

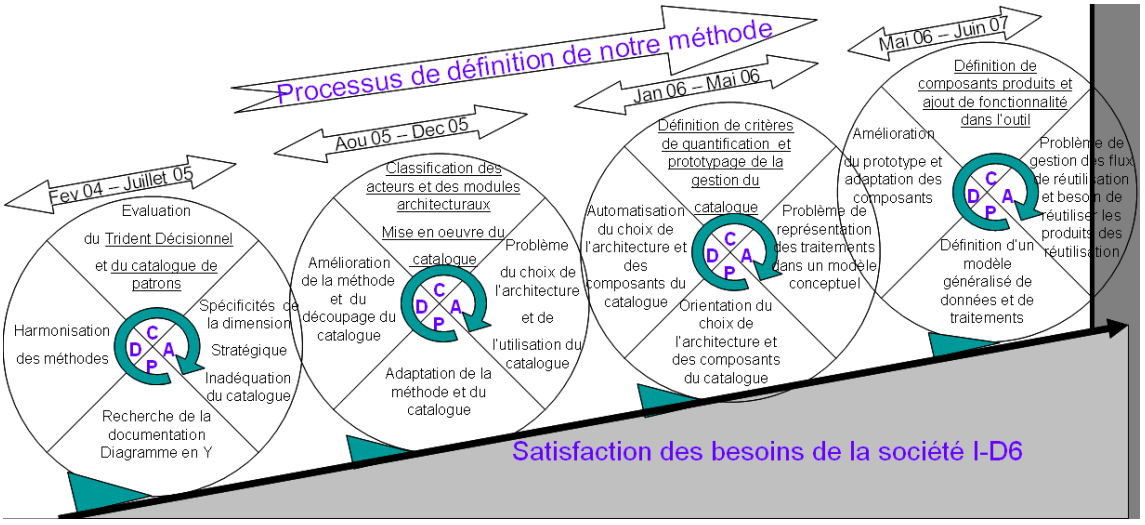


FIG. 7.2 – Notre mise en oeuvre de la démarche qualité Roue de Deming

7.4 Développement de l'outil eBIPAD

Les concepteurs décisionnels ont rencontré des difficultés lors de la mise en oeuvre du catalogue de patrons présenté sous forme de listing. Ils ont exprimé le besoin d'une utilisation plus intuitive et efficace qui prenne en compte l'aspect itinérant de leur fonction. Afin de favoriser l'utilisation systématique du catalogue et donc de notre méthode, nous avons développé l'outil eBIPAD (electronic Business Intelligence Pattern Design). Il constitue un guide méthodologique fondé sur une base constituée des patrons de notre catalogue BIPAD. Le développement de SID par réutilisation consiste à utiliser les patrons dans un contexte qui est celui de tout projet décisionnel, en l'occurrence de mise en place d'un SID. Cet outil permet :

- la gestion des patrons par l'administrateur des patrons, c'est à dire la création, la modification et la suppression des patrons du catalogue,
- la consultation des patrons par le concepteur décisionnel pour les utiliser au cours d'un projet décisionnel.

Cet outil répond aux spécifications d'une bibliothèque en ligne de patrons avec des fonctions d'organisation des patrons [Ralyté 2001]. Ainsi, nous présentons dans les sections suivantes les spécifications fonctionnelles et les schémas de données du système de gestion du catalogue.

7.4.1 Généralités de l'outil

L'outil eBIPAD supporte les deux processus d'ingénierie, soit l'ingénierie pour la réutilisation et l'ingénierie par la réutilisation [Rieu 1999]. Ces processus sont utilisés par deux des trois acteurs de eBIPAD qui sont :

- l'ingénieur de patrons : il est aussi appelé administrateur des patrons. Il est un acteur principal et il est responsable de la gestion du catalogue. Il crée, modifie, supprime et édite les patrons. Les produits générés par l'ingénieur de patrons, en l'occurrence les patrons, les diagrammes UML des patrons processus et les documents types des patrons produits, sont stockés sur le serveur de l'outil eBIPAD dans la base ou dans des répertoires spécifiques.
- l'ingénieur d'applications : il est un concepteur décisionnel. C'est un acteur principal aussi. Il est responsable du développement du SID par réutilisation des patrons du catalogue. Il applique la démarche du Trident décisionnel capitalisée dans les patrons. Il peut consulter uniquement en lecture seule un document stocké sur le serveur. Tous les produits générés par l'ingénieur d'applications ne sont pas stockés sur le serveur. Leurs sauvegardes et leurs emplacements sont de la responsabilité de l'ingénieur d'applications. Ils ne sont définis et possibles que sur le poste informatique de l'ingénieur d'applications,
- l'administrateur du système : il est un acteur secondaire et il est responsable de la gestion des acteurs qui accèdent à l'outil.

L'ingénieur de patrons et l'ingénieur d'applications utilisent des logiciels ad-hoc pour définir ou visualiser des documents. Ces logiciels permettent à l'ingénieur de patrons d'éditer un document de modélisation, un document tableau ou un document texte, de le stocker sur le serveur et d'insérer l'adresse du document dans le contenu du patron.

Le diagramme statique de système eBIPAD est présenté dans la figure 7.3. Il permet de répertorier les acteurs et de spécifier le nombre d'instances d'acteurs connectés à un moment donné.

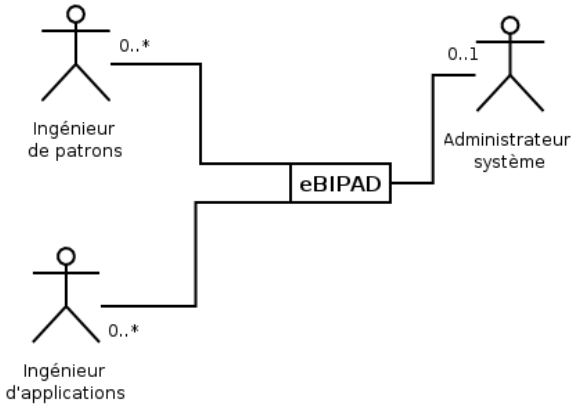


FIG. 7.3 – Diagramme statique de l'outil eBIPAD suivant [Roques 2004]

7.4.2 Fonctionnalités générales

Lors de la connexion d'un acteur à eBIPAD, la fenêtre de connexion est celle présentée dans la figure 7.4. Puis, un état du catalogue ou des utilisateurs est présenté à l'utilisateur[1] suivant le type d'utilisateurs. Outre les tâches liées spécifiquement à la gestion et à l'utilisation de la base de patrons, l'utilisateur peut contacter les ingénieurs de patrons afin de notifier une erreur ou un autre constat fait lors de l'utilisation de l'outil. Comme toute application web, les utilisateurs peuvent transmettre leurs remarques et leurs appréciations.

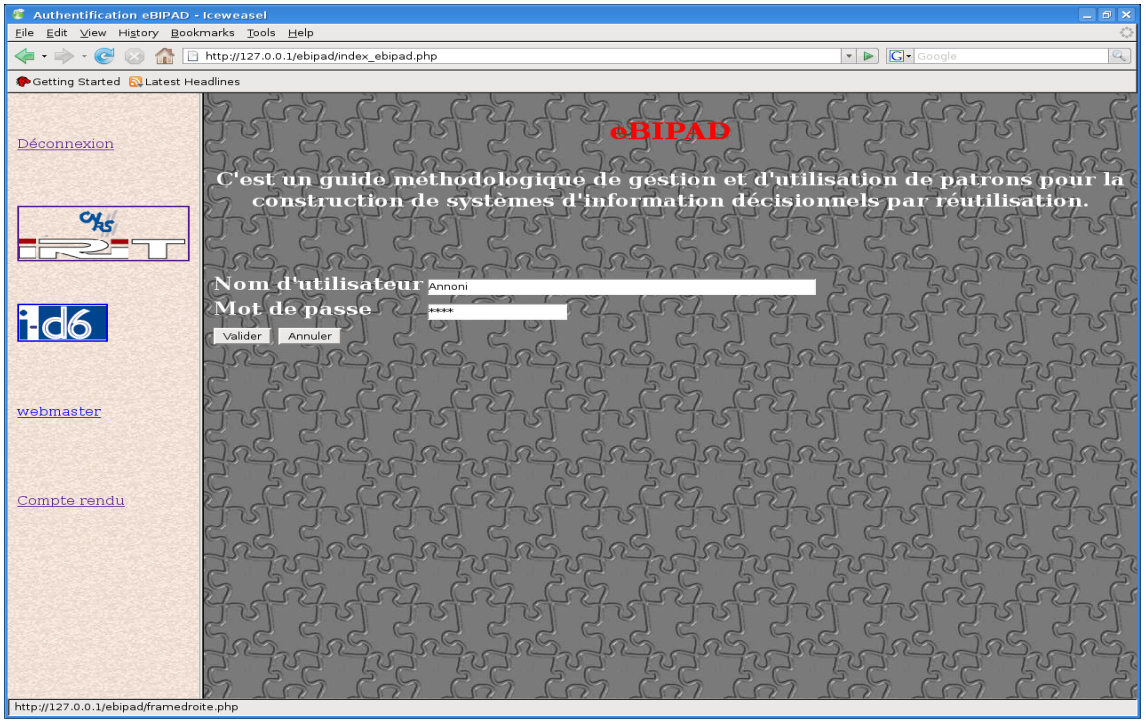


FIG. 7.4 – Fenêtre de connexion à l'outil eBIPAD

Les cas d'utilisations des différents utilisateurs de l'outil eBIPAD sont présentés dans la figure 7.5.

[1]Pour le système de gestion de patrons, nous utilisons le terme « utilisateurs » pour désigner aussi bien les utilisateurs que les acteurs.

Fig. 7.5 – Diagramme de cas d'utilisation de l'outil eBIPAD

Les cas d'utilisation « S'authentifier » et « Afficher un patron » ne sont pas représentés mais ils sont inclus dans tous les autres cas d'utilisation. Nous proposons de structurer les opérations liées aux cas d'utilisation dans des paquetages par acteur comme indiqué dans le diagramme de paquetages présenté dans la figure 7.6.

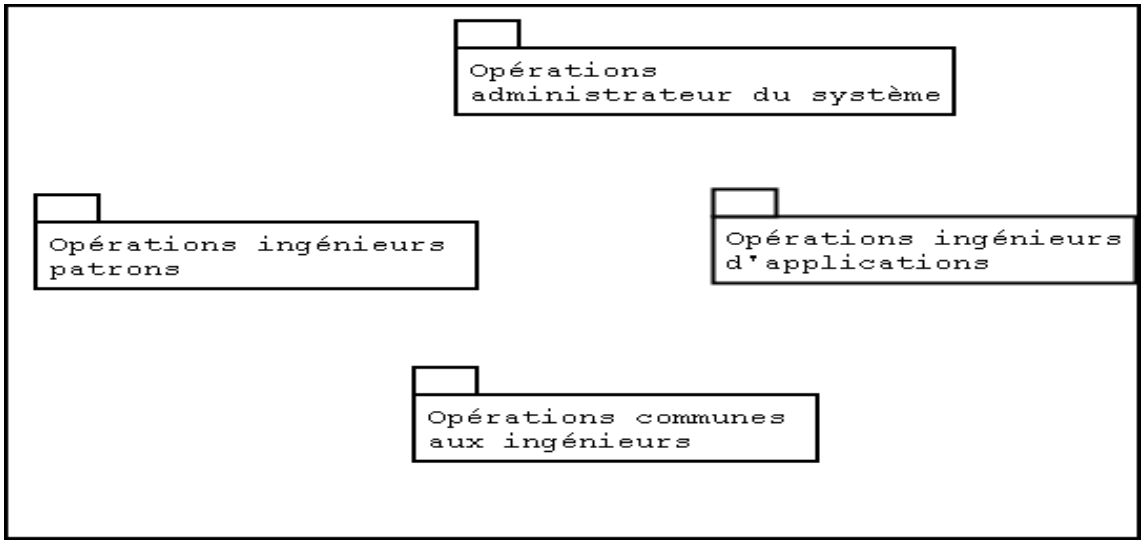


FIG. 7.6 – Diagramme de paquetages de l'outil eBIPAD

7.4.3 Fonctionnalités pour l'ingénieur de patrons

L'ingénieur de patrons est celui qui peut réaliser le plus grand nombre d'opérations. Les cas d'utilisation qui lui sont associés peuvent être regroupés en trois groupes, soit :

- la gestion du contenu :
  - ajouter un patron,
  - modifier un patron,
  - supprimer un patron,
  - ajouter un champ,
  - modifier un champ,
  - supprimer un champ.
- la gestion de la structure :
  - ajouter une partie,
  - modifier une partie,
  - supprimer une partie,
  - ajouter une rubrique,
  - modifier une rubrique,
  - supprimer une rubrique,
  - afficher la structure des patrons.
- l'exploitation du catalogue :
  - lister les patrons,
  - rechercher des patrons,
  - imprimer un patron.

Nous présentons dans la figure 7.7 l'interface d'accueil de notre outil pour un ingénieur de patrons.

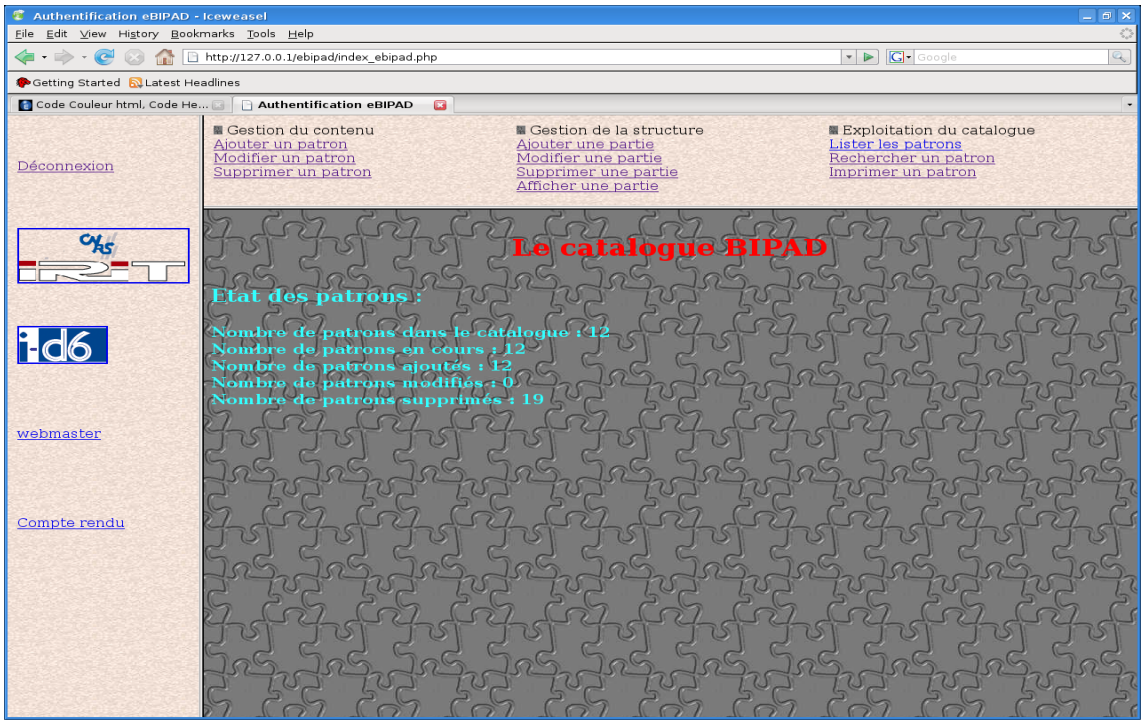


FIG. 7.7 – Interface d'accueil de l'outil eBIPAD d'un ingénieur de patrons

Exemple : nous développons deux des cas d'utilisations de l'ingénieur de patrons, soit les cas d'utilisation « **Ajouter un patron** » et « **Ajouter un champ** ».

Cas d'utilisation « Ajouter un patron »

1. Identification
Titre : Ajouter un patron
Résumé : ce cas d'utilisation permet à l'ingénieur de patrons d'ajouter un patron à la base de patrons
Acteurs : Ingénieurs de patrons
Date de création : 15/05/05
Date de mise à jour : 16/05/05
Version : 0.1 Responsable : Estella Annoni

2. Description des scénarios
2.1. Pré-conditions :
2.2. Scénario nominal :

1. l'utilisateur saisit le nom du patron,

2. eBIPAD vérifie que ce patron n'existe pas dans la base,
3. eBIPAD affiche la représentation de base d'un patron avec toutes les rubriques et les champs,
4. l'utilisateur ajoute les champs par rubrique de la représentation de base (cf. cas d'utilisation " Ajouter un champ "),
5. l'utilisateur enregistre son patron,
6. eBIPAD vérifie si les rubriques obligatoires possèdent au moins un champ non vide,
7. eBIPAD insère le patron dans la base.

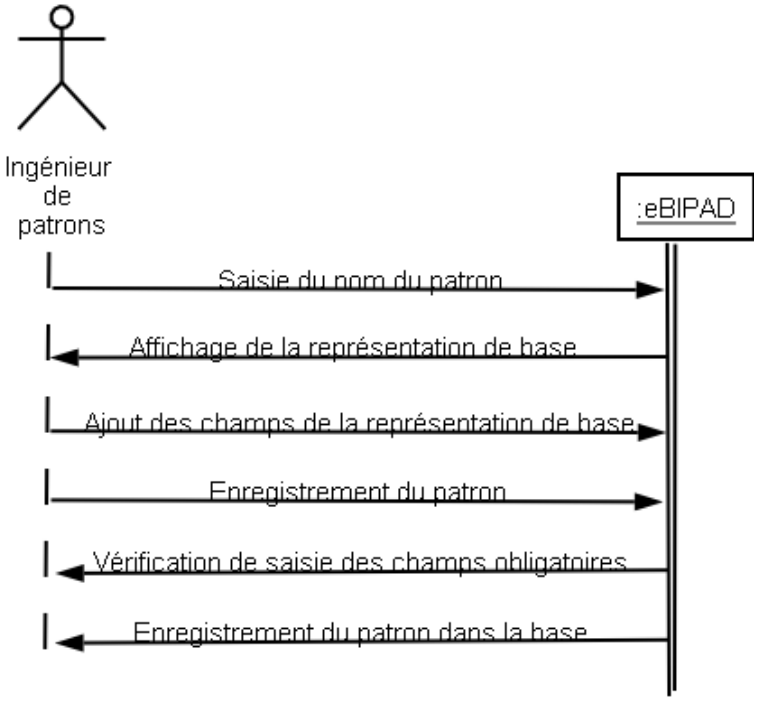


Fig. 7.8 – Diagramme de séquence système du scénario nominal d' " Ajouter un patron "

2.3. Enchaînements alternatifs :
A1 : Un patron de ce nom existe déjà
L'enchaînement A1 démarre à l'étape 3 du scénario nominal ; les étapes 1 et 2 sont identiques.
3. eBIPAD indique à l'ingénieur de patrons que ce patron existe
Le scénario reprend à l'étape 1.

A2 : Au moins une rubrique obligatoire ne possède pas de champs, ou possède un champ de valeur nulle.
L'enchaînement A2 démarre à l'étape 7 du scénario nominal.
7. eBIPAD indique à l'ingénieur de patrons que la contrainte sur les rubriques obligatoires n'est pas vérifiée.

Le scénario reprend à l'étape 4.

A3 : Les données du patron à insérer ne vérifient pas les spécifications de la base.
L'enchaînement A2 démarre à l'étape 7 du scénario nominal.
7. eBIPAD indique à l'ingénieur de patrons que les spécifications sur les données de la base ne sont pas vérifiées.
Le scénario reprend à l'étape 4.

2.4. Enchaînements d'erreurs :
Pas d'erreur

2.5. Post-conditions :
Etape 8 du scénario nominal : la base de patrons contient un patron de plus.
Etape 9 du scénario nominal : le nouveau patron contient au moins des champs non vides pour les rubriques obligatoires.

3. Exigences non fonctionnelles
Intégrité : les contraintes d'intégrité de la base doivent être vérifiées.

4. Besoins d'IHM
Il faut que l'utilisateur puisse lancer des logiciels ad-hoc depuis eBIPAD :
- lancer un éditeur de textes,
- lancer un tableur,
- lancer un outil de modélisation.

La fenêtre de saisie du nom du patron associée à l'étape 2 du scénario nominal du cas d'utilisation « Ajouter un patron » est présentée dans la figure 7.9.

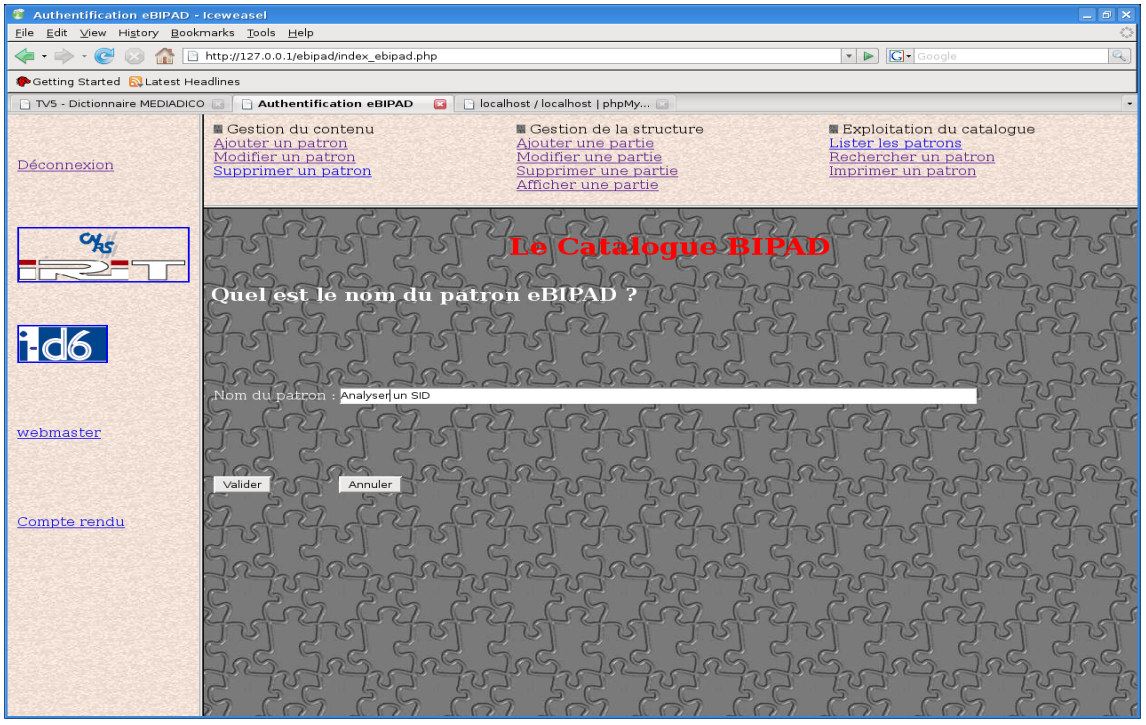


FIG. 7.9 – Fenêtre de saisie du nom du patron

Cas d'utilisation « Ajouter un champ »

1. Identification
Titre : Ajouter un champ
Résumé : cas d'utilisation permet d'ajouter un champ lors de l'ajout ou la modification d'un patron
Acteurs : Ingénieurs de patrons
Date de création : 15/05/05
Date de mise à jour : 16/05/05
Version : 0.1
Responsable : Estella Annoni

2. Description des scénarios
2.1. Pré-conditions :
2.2. Scénario nominal :

1. eBIPAD affiche la fenêtre d'ajout d'un nouveau champ,
2. l'utilisateur saisit le nom du champ,
3. l'utilisateur sélectionne dans la liste déroulante le type du champ,
4. l'utilisateur choisit le type texte,

5. l'utilisateur saisit directement les champs textuels dans la représentation de base,
6. l'utilisateur valide l'ajout du champ,
7. eBIPAD contrôle les données du champ,
8. eBIPAD enregistre le champ dans la base.

2.3. Enchaînements alternatifs :
A1 : Un champ de ce nom pour ce patron existe déjà.
L'enchaînement A1 démarre à l'étape 3 du scénario nominal.
3. eBIPAD indique à l'ingénieur de patrons que ce champ existe pour ce patron et cette rubrique.
Le scénario reprend à l'étape 1.

A2 : Les données ne vérifient pas les contraintes relatives au champ.
L'enchaînement A5 démarre à l'étape 8 du scénario nominal.
8. eBIPAD indique à l'ingénieur de patrons que les contraintes ne sont pas vérifiées.
Le scénario reprend à l'étape 4.

2.4. Enchaînements d'erreurs :

2.5. Postconditions :
La base de patrons contient un champ de plus.
Si le nouveau champ est associé à une rubrique obligatoire alors sa valeur doit être non nulle.

Les fenêtres de l'outil eBIPAD lié à l'ajout d'un champ pour le patron « Analyser un SID » sont présentées dans les figures 7.10, 7.11 et 7.12.

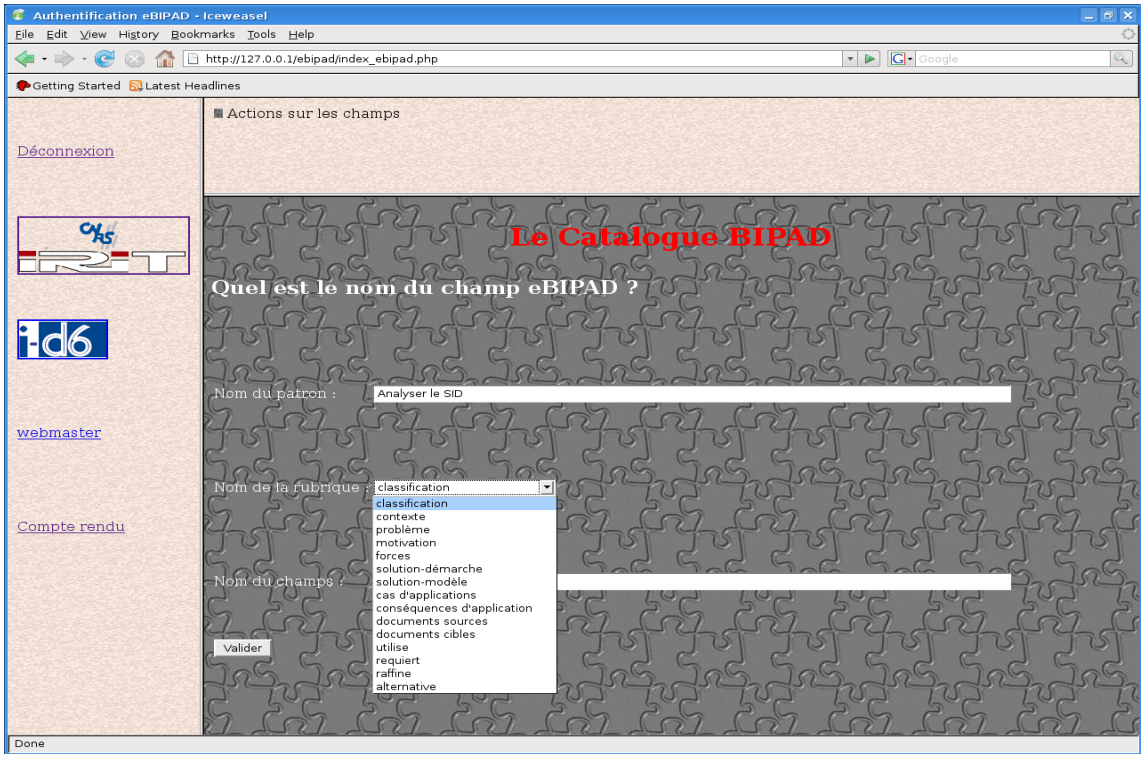


FIG. 7.10 – Saisie du nom du nouveau champ

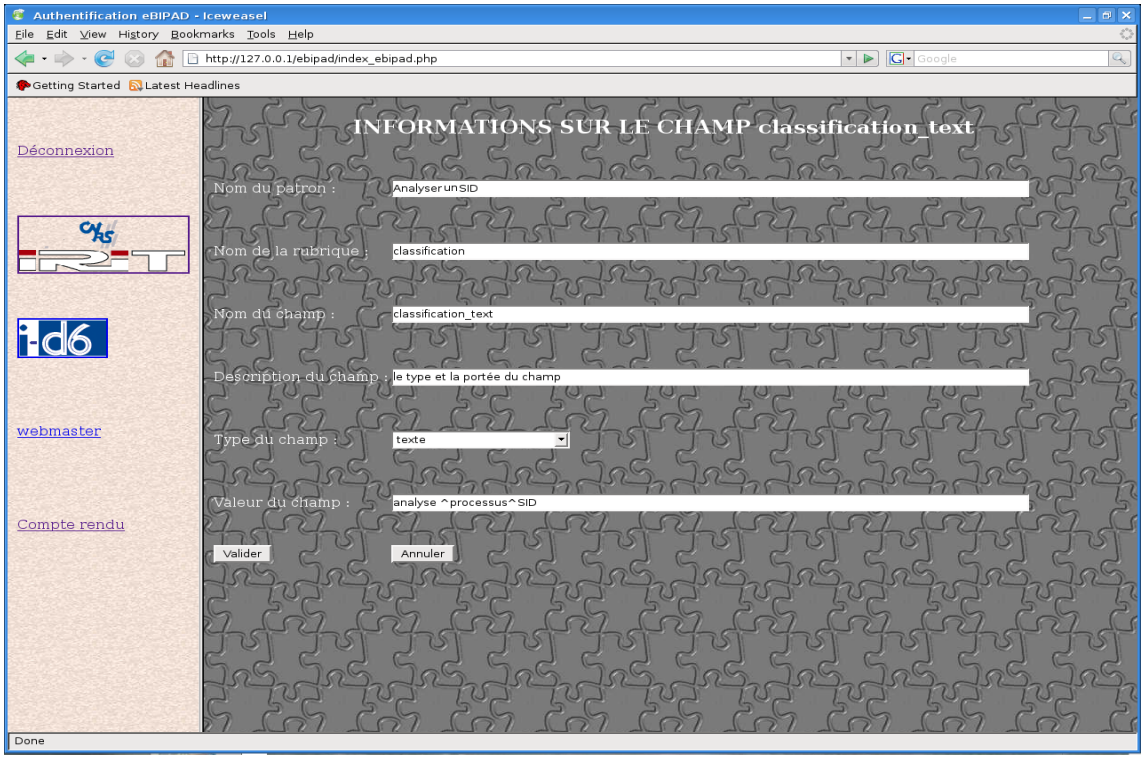


FIG. 7.11 – Définition du type du champ et de la valeur du champ

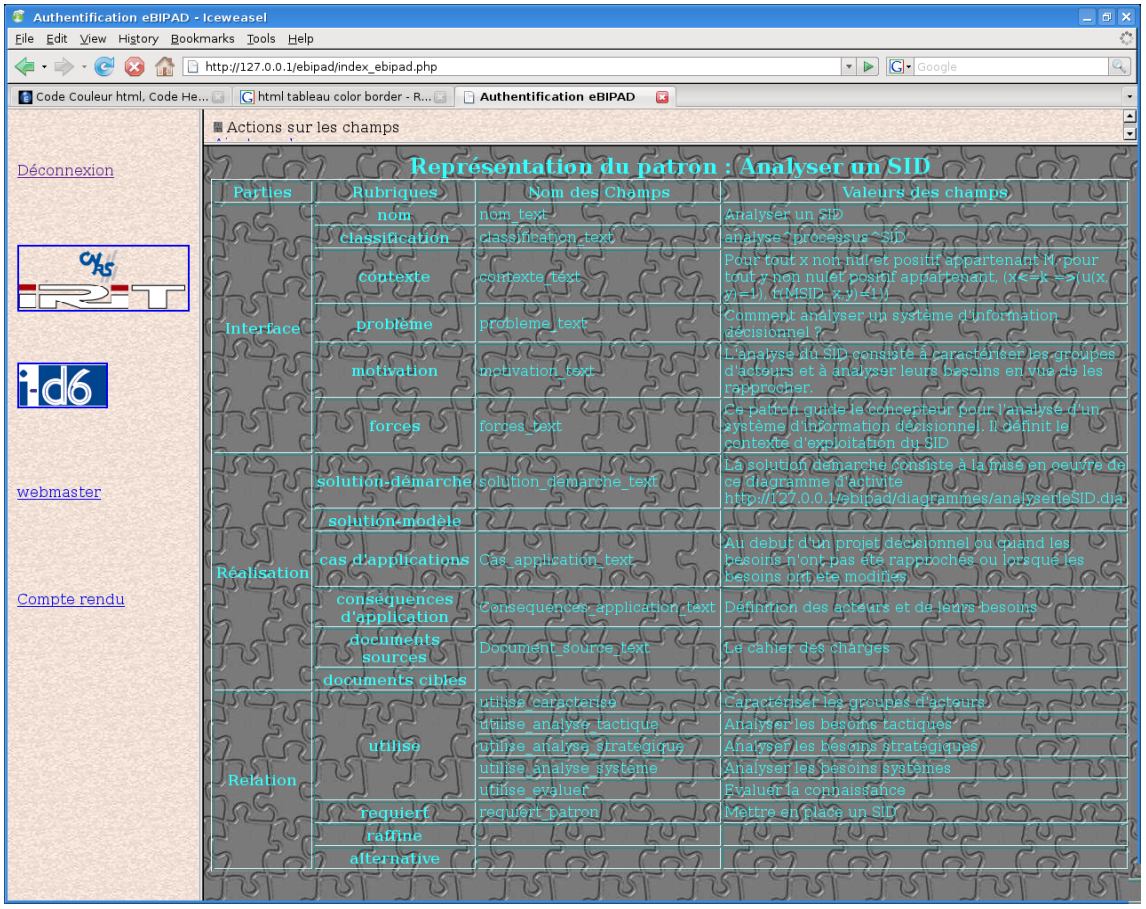


FIG. 7.12 – Affichage du patron après l'ajout de tous les champs

7.4.4 Fonctionnalités pour l'ingénieur d'applications

L'ingénieur d'applications réutilise les patrons suivant les éléments processus et produits de la démarche du Trident décisionnel. L'ingénieur d'applications réalise les tâches suivantes. Ces tâches peuvent être regroupées en deux groupes :

- la réutilisation des patrons :
  - évaluer le contexte du patron,
  - Accéder à la solution-démarche,
  - Accéder à la solution-modèle.
- l'exploitation du catalogue :
  - lister les patrons,
  - rechercher un patron,
  - imprimer un patron.

L'interface d'accueil d'un ingénieur d'applications est présenté à la figure 7.13.

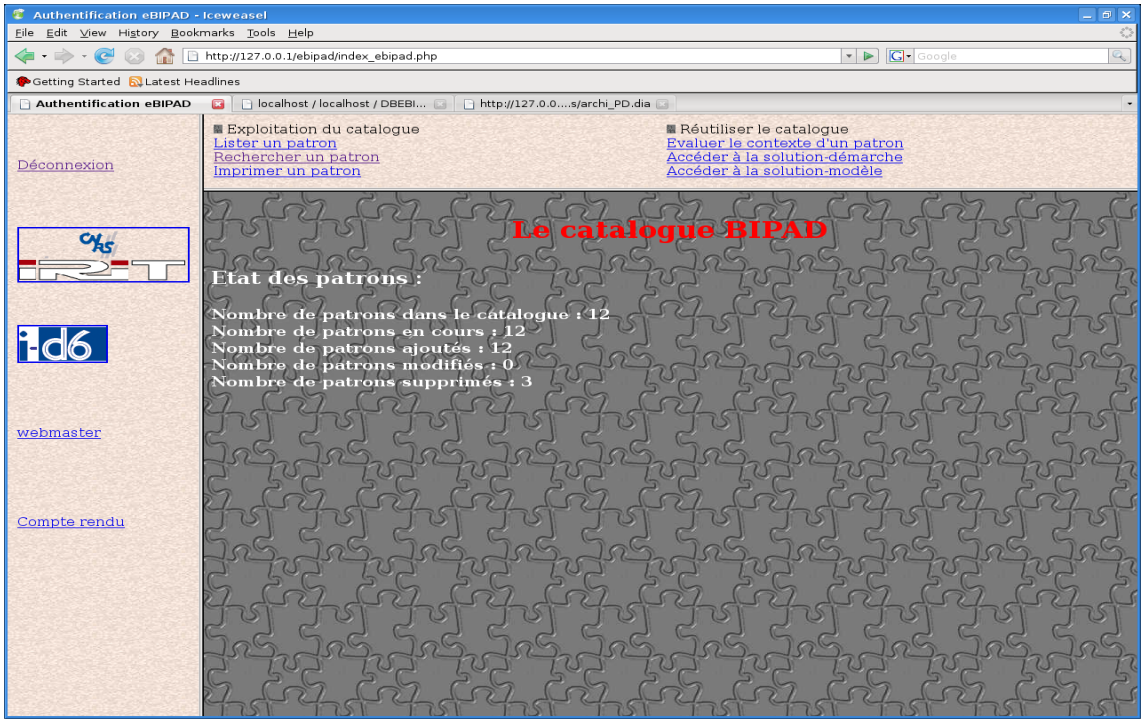


FIG. 7.13 – Interface d'accueil d'un ingénieur d'applications

Exemple : nous développons deux des cas d'utilisations de l'ingénieur d'applications, soit les cas d'utilisation « **Accéder à la solution-démarche** » et « **Rechercher un patron** ».

Cas d'utilisation « Accéder à la solution-démarche »

1. Identification
Titre : Accéder à la solution-démarche
Résumé : ce cas d'utilisation permet à l'ingénieur d'applications d'accéder à la solution -démarche d'un patron donné
Acteurs : Ingénieurs d'applications
Date de création : 15/05/05
Date de mise à jour : 16/05/05
Version : 0.4 Responsable : Estella Annoni

2. Description des scénarios
2.1. Pré-conditions :
2.2. Scénario nominal :

1. l'utilisateur choisit le nom du patron parmi la liste déroulante des noms de patrons,

2. eBIPAD affiche la représentation de base d'un patron en activant le lien associé à l'adresse du diagramme UML de la solution démarche,
3. l'utilisateur affiche le menu contextuel de ce lien,
4. l'utilisateur sauvegarde le diagramme en local sur son poste.

2.3. Enchaînements alternatifs :

2.4. Enchaînements d'erreurs :
Pas d'erreur

3. Exigences non fonctionnelles
Concurrence : plusieurs utilisateurs peuvent accéder à la même solution-démarche.

4. Besoins d'IHM
Lancer un outil de modélisation.

Dans notre exemple, l'utilisation d'applications doit pouvoir ouvrir le diagramme d'activités UML de la solution-démarche.

Les fenêtres de l'outil eBIPAD lié à l'accès à la solution-démarche du patron « Analyser un SID » sont présentées dans les figures 7.14, 7.15 et 7.16.

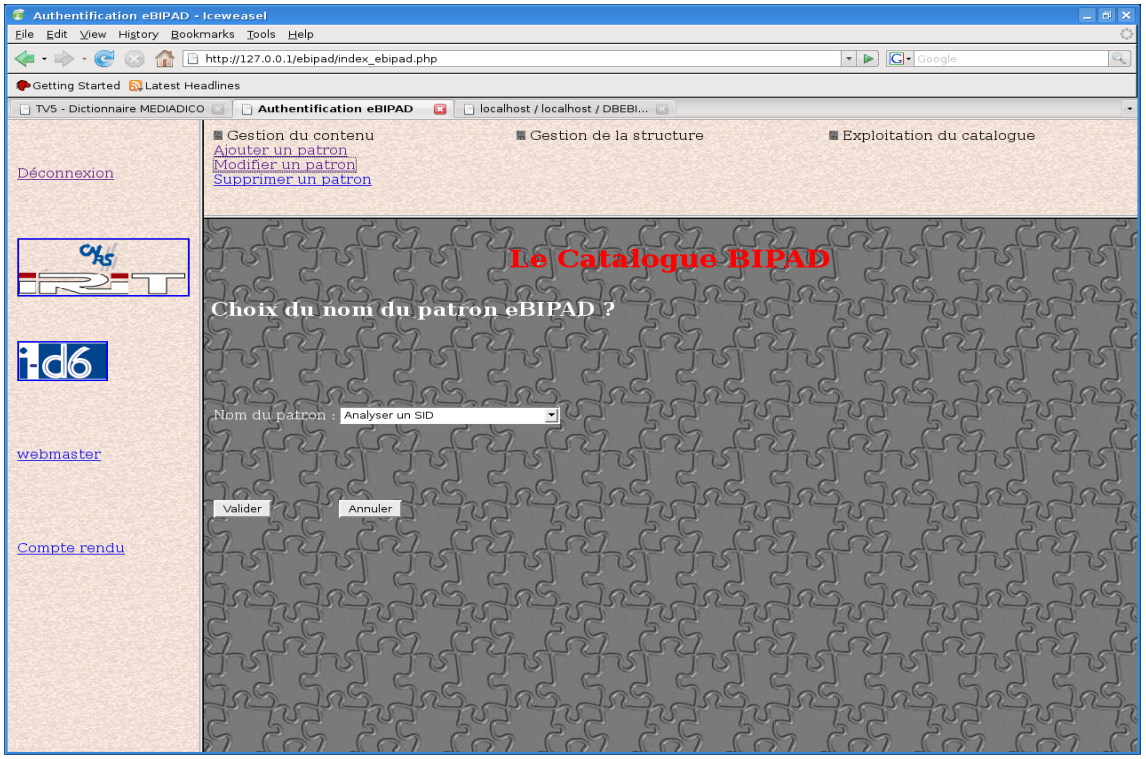


FIG. 7.14 – Choix du nom patron

FIG. 7.15 – Représentation du patron avec le lien de l'adresse du diagramme UML actif

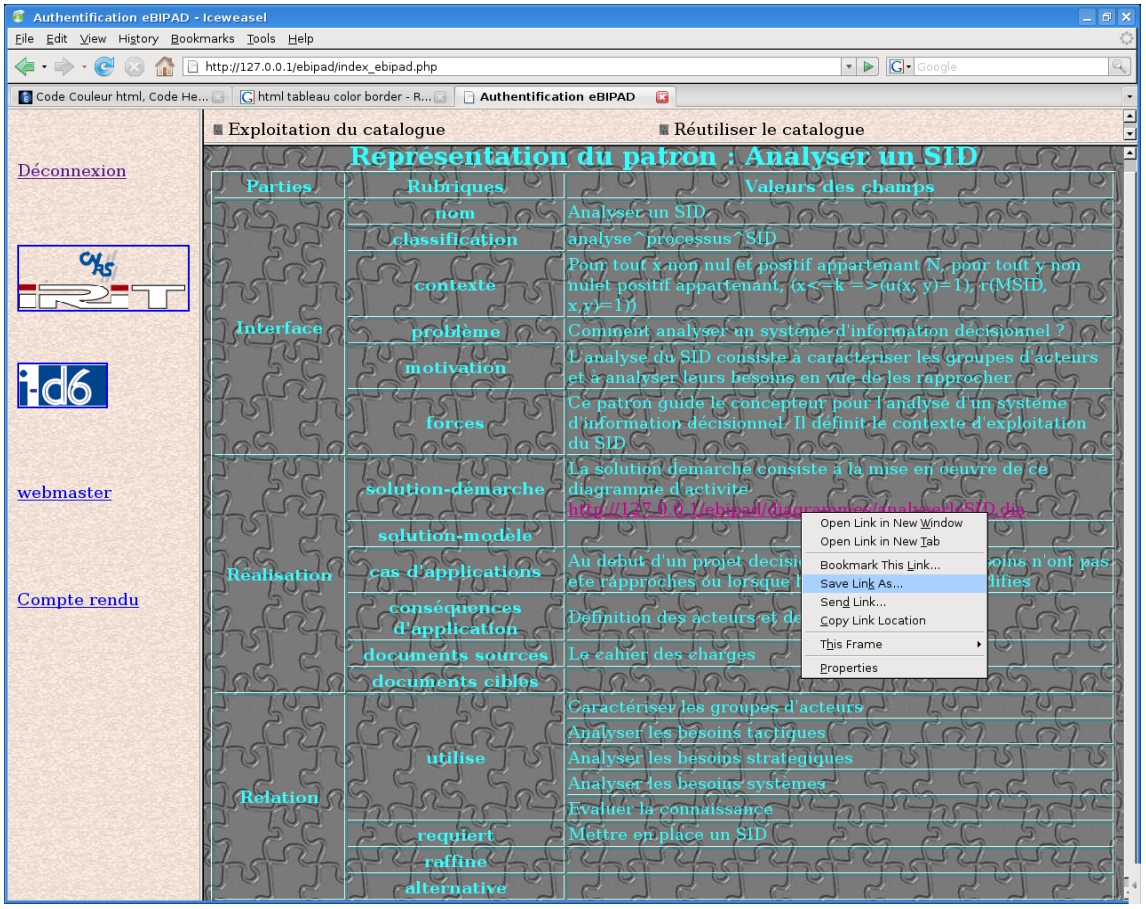


FIG. 7.16 – Sauvegarde de la solution-démarche

Cas d'utilisation « Rechercher un patron »

1. Identification
Titre : Rechercher un patron
Résumé : ce cas d'utilisation permet à l'ingénieur d'applications de recherche un patron du catalogue
Acteurs : Ingénieurs d'applications
Date de création : 15/05/05
Date de mise à jour : 16/05/05
Version : 0.4 Responsable : Estella Annoni

2. Description des scénarios
2.1. Pré-conditions :
2.2. Scénario nominal :

1. l'utilisateur choisit le nom du patron parmi la liste déroulante des noms de patrons,
2. eBIPAD affiche la représentation de base d'un patron.

2.2. Enchaînements alternatifs :

2.3. Enchaînements d'erreurs :
Pas d'erreur

3. Exigences non fonctionnelles
Concurrence : plusieurs utilisateurs peuvent recherche un même patron.

La fenêtre de l'outil eBIPAD présentant le résultat de la recherche du patron « Analyser un SID » est donnée dans la figure 7.10.

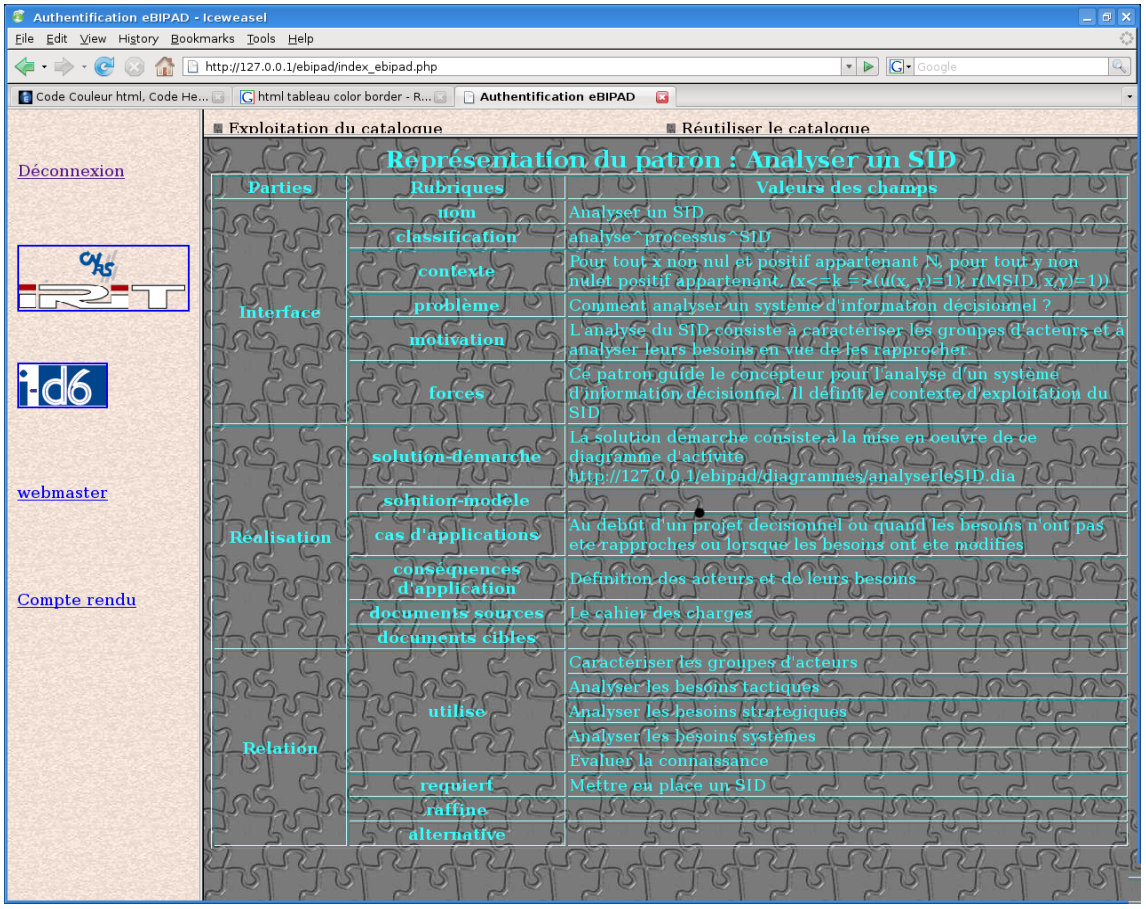


FIG. 7.17 – Présentation du résultat de la recherche du patron « Analyser un SID »

7.4.5 Fonctionnalités pour l'administrateur système

L'administrateur système veille au bon fonctionnement et à l'accès sécurisé de l'outil. Il réalise les tâches suivantes :

- ajouter un utilisateur,
- modifier un utilisateur,
- supprimer un utilisateur.

L'interface d'accueil d'un ingénieur d'applications est présenté à la figure 7.18.

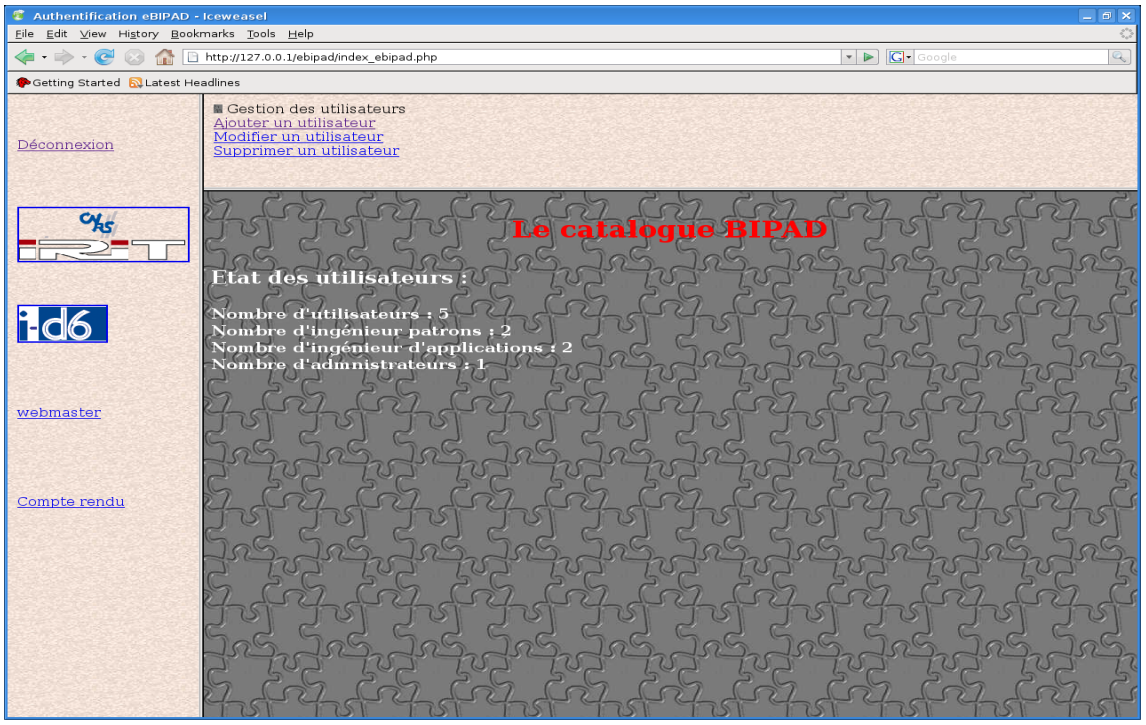


FIG. 7.18 – Interface d'accueil d'un administrateur système

Ces cas d'utilisation sont triviaux, nous ne les présentons donc pas dans cette thèse.

7.4.6 Base de patrons de eBIPAD

Cette base contient l'ensemble des patrons définis par l'ingénieur de patrons. Les patrons ont tous la même structure.

Spécifications de l'outil

Cette structure de base est hiérarchique et plus précisément elle est composée de 3 parties qui sont structurées en rubriques suivant le formalisme P-SIGMA étendu (cf. la représentation du patron ASID dans la base est définie dans les figures 7.19 et 7.20). Ces parties sont toujours les mêmes pour chaque patron, en l'occurrence « Interface », « Réalisation » et « Relation ». De même, les rubriques par partie et par patrons sont toujours les mêmes. Le nombre de rubriques n'est pas le même par partie. Une rubrique peut être obligatoire ou non. Le nombre de champs par rubrique varie suivant le patron.

Un champ est défini par son intitulé, sa description, son type et sa valeur. Les champs peuvent être de type : texte, document_texte, document_tableau, document_modélisation ou patron. Pour les champs de type texte, autrement dit de type simple, il n'y a pas de traitement spécial. Cependant, pour les champs complexes, la valeur stockée dans la base sera le chemin du fichier, alors que pour les patrons la valeur stockée sera le patron. Une rubrique est définie par son intitulé et sa description ; c'est aussi le cas d'une partie et d'un type_champ. Les liens inter-patrons sont gérés par la classe Type_champ. L'application gère aussi la structure qui est commune à l'ensemble des patrons. Cette structure, composée de parties et de rubriques, doit être modifiable. Un historique des traitements réalisés relatifs à la gestion des patrons sur la structure sera généré .

Parties	Rubriques	Champs	Valeurs	Types
INTERFACE	Sigle	sig_text	ASID	
	Identifiant	ind_tex	Analyser un SID	texte
	Classification	cla-text	analyse ^ processus ^ SID	texte
	Contexte	con_text	$\forall x \in N_+^*, \forall y \in N_+^*, (x \leq k, y \leq k \Rightarrow (u_s(x,y) = 1, r_s(MSID, x, y$ avec x : itération en cours du processus de développement du SID, y : nombre d'utilisations de ce patron au cours du processus complet k : nombre maximal d'itérations de la phase d'analyse u_s : la fonction qui indique les conditions dans lesquelles ce patron est utilisable	texte
	Problème	pro_text	Comment analyser un système d'information décisionnel ? L'analyse du SID correspond à l'étude de la mise en place du SID. Elle est réalisée afin d'évaluer la faisabilité du projet, en l'occurrence le rapprochement entre le besoins	texte
	Motivation	mot_text	L'analyse du SID consiste à caractériser les groupes d'acteurs et analyser leurs besoins en vue de les rapprocher.	texte
	Forces	for_text	Ce patron guide le concepteur pour l'analyse d'un système d'information décisionnel. Il définit le contexte d'exploitation du SID	texte
REALISATION	Solution-démarche	sol_dem_text	La solution consiste à la mise en oeuvre de ce diagramme d'activités. Analyser le SID Caractériser les groupes d'acteurs Analyser les besoins tactiques Analyser les besoins stratégiques Analyser les besoins système Evaluer la connaisance utilisateurs Evaluer la connaisance de l'environnement	texte
		sol_dem_dac		document_modélisation
	Solution-modèle	sol_mod_text	Au début d'un projet décisionnel ou quand les besoins n'ont pas été rapprochés ou lorsque que les besoins ont été modifiés	texte
	Cas d'applications	cas_app_text	Définition des acteurs et de leurs besoins	texte
	Conséquences d'application	con_app_text		texte
	Documents sources	doc_sou_text	Le cahier des charges	texte
	Documents cibles	doc_cib_text		texte

FIG. 7.19 – Représentation du patron « Analyser un SID » dans la base de eBIPAD - début

RELATION	Utilise	uti_1_pat	« Caractériser les groupes acteurs »	patron
		uti_2_pat	« Analyser les besoins tactiques »	patron
		uti_3_pat	« Analyser les besoins stratégiques »	patron
		uti_4_pat	« Analyser les besoins systèmes »	patron
		uti_5_pat	« Evaluer la connaissance»	patron
	Requiert	req_1_pat		patron
	Raffine	Raf_1_pat		patron
	Alternative	Alt_1_pat		patron

FIG. 7.20 – Représentation du patron « Analyser un SID » dans la base de eBIPAD - fin

L'outil présente les caractéristiques suivantes :
- il s'exécute depuis un navigateur web afin d'éviter une installation sur le poste du concepteur décisionnel,
- il est accessible en intranet et *via* Internet car les consultants décisionnels sont mobiles,
- les fichiers associés aux champs des patrons sont stockés sur un répertoire de l'arborescence du serveur où est installé eBIPAD,
- plusieurs utilisateurs doivent pouvoir se connecter à l'outil,
- assurer le principe de lecteurs-rédacteurs pour 1 rédacteur et N lecteurs,
- les tables qui décrivent la structure des patrons ne sont modifiables que par les ingénieurs de patrons. Ces tables sont TPARTIES et TRUBRIQUES,
- la table TUTILISATEURS n'est modifiable que par l'administrateur du système.

Schémas de la base de patrons

Le schéma conceptuel de la base DBEBIPAD est présenté dans la figure 7.21 et le dictionnaire de données associé est présenté dans la figure 7.22.

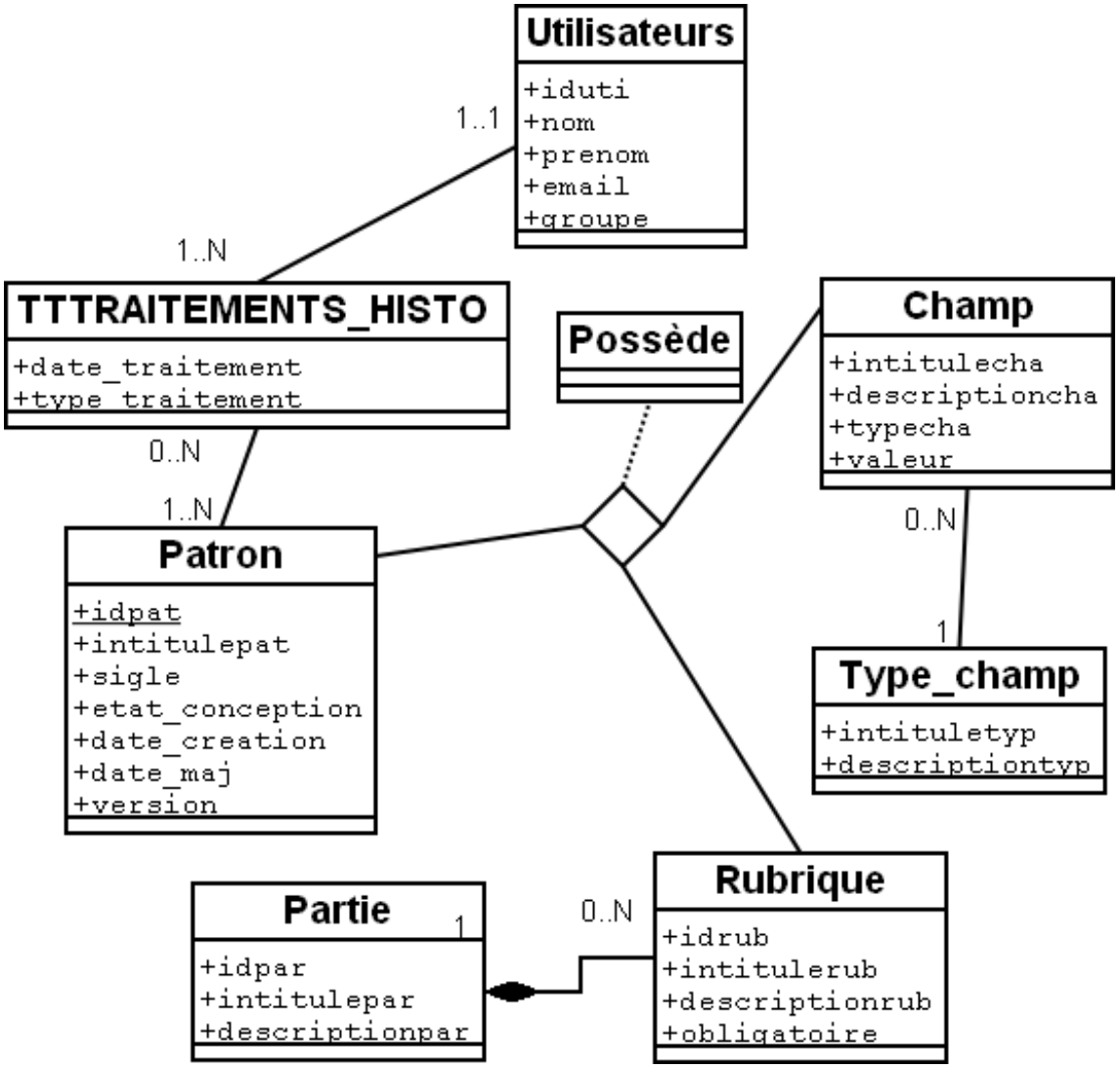


FIG. 7.21 – Diagramme de classes UML de la base de patrons

Mnémonique	Libellé	Type de données	Contraintes d'intégrité
idpat	Identifiant du patron	Entier	>0
intitulepat	Intitulé du patron	Chaîne (100)	Non nul
sigle	Sigle du patron	Chaîne (5)	Non nul
etat_conception	Etat de conception du patron	Enumération	{« en cours », « validé »}
date_creation	Date de création du patron	Date	Non nul
date_maj	Date de mise à jour du patron	Date	Non nul
version	Version du patron	Entier	Non nul
idpar	Identifiant de la partie	Entier	Non nul
intitulepar	Intitulé de la partie	Enumération	{« Interface », « Réalisation », « Relation »} Non nul
descriptionpar	Description de la partie	Chaîne (255)	Non nul
idtyp	Identifiant du type	Entier	Non nul
intituletyp	Intitulé du type	Enumération	{« Texte », « document_texte », « document_tableau », « document_modélisation », « patron »} Non nul
descriptiontyp	Description du type	Chaîne (255)	Non nul
idrub	Identifiant de la rubrique	Entier	Non nul
intitulerub	Intitulé de la rubrique	Chaîne (50)	Non nul
descriptionrub	Description de la rubrique	Chaîne (255)	
obligatoire	Caractère obligatoire du champ	Char(1)	{« 0 », « 1 »} (0 : non obligatoire, 1 : obligatoire)
idcha	Identifiant du champ	Entier	Non nul
intitulecha	Intitulé du champ	Chaîne (50)	
descriptionpcha	Description du champ	Chaîne (255)	
valeur	Valeur du champ	Chaîne (255)	
iduti	Identifiant de l'utilisateurs	Entier	Non nul
nom	Nom de l'utilisateur	Chaîne (50)	Non nul
prenom	Prénom de l'utilisateur	Chaîne (255)	Non nul
email	Email de l'utilisateur	Chaîne (100)	
groupe	Groupe de l'utilisateur	Chaîne (20)	Non nul

FIG. 7.22 – Dictionnaire des données de la base

Les noms des colonnes du dictionnaire des données signifient :
- Mnémonique : abréviation du nom de l'attribut,
- Libellé : libellé contenant la signification précise et le rôle de l'attribut,
- Type de donnée : type de l'attribut, soit : entier, réel, chaîne de caractères, date et énumération,
- Contraintes d'intégrité : liste des contraintes sur les valeurs possibles de l'attribut.

Le schéma de la base au niveau logique est défini comme suit :
TPATRONS (idpat, intitulepat, sigle, etat_conception, date_creation, date_maj, version)
TPARTIES (idpar, intitulepar, descriptionpar)
TTYPE_CHAMPS (idtyp, intituletyp, descriptiontyp)
TRUBRIQUES (idrub, intitulerub, descriptionrub, obligatoire, idpar#)
TCHAMPS (idcha, intitulecha, descriptioncha, idtyp#, valeur)
TUTILISATEURS (iduti, nom, prenom, email, groupe)
TPOSSEDE (idpat#, idrub#, idcha#)
TTRAITEMENTS_HISTO (idpat#, iduti#, date_traitement, type_traitement)

7.5 Evaluation de nos propositions

Dans cette section, nous présentons une synthèse des évaluations de nos propositions afin de mettre en avant les apports de leur utilisation aussi bien en terme d'avantages directes que de ceux générés par la correction des inconvénients rencontrés. Nous dressons cette synthèse à partir de cinq missions industrielles au cours desquelles nous avons appliqué nos propositions. Nous n'avons pas pu faire de validation quantitative de nos propositions car le domaine et le métier ainsi que les organisations clientes changeaient régulièrement.

7.5.1 Synopsis de nos propositions

Nos propositions sont liées aux trois domaines abordés durant nos travaux de recherche au regard de nos problématiques (cf. section 1.5). En ce sens, nous les récapitulons en fonction de ces domaines.

L'ingénierie des besoins des SID :

Modèle :
Nous proposons des **tableaux croisés** pour l'expression des besoins des groupes de décideurs tactiques et stratégiques par eux-mêmes, de manière autonome sans la participation des concepteurs décisionnels. Cette collecte complétée par celle des

schémas des sources de données permet de définir les données, en l'occurrence la statique du SID. Pour définir la dynamique, nous proposons des **graphes de propriétés** du SID. L'annotation de ces graphes permet de définir les traitements de dérivation et de préparation des données, autrement dit le contexte technique et le contexte décisionnel du SID. Puis, nous proposons de représenter de manière multidimensionnelle les besoins des décideurs tactiques, les besoins des décideurs stratégiques et les sources de données *via* un unique modèle de besoins appelé le **diagramme décisionnel**. Ce modèle présente la statique et la dynamique du SID dans un même schéma car il est une extension du diagramme de classes UML.

Démarche :
Nous proposons une **démarche de collecte des besoins** qui permet de sélectionner les structures de données d'expression des besoins et de guider le concepteur décisionnel pour l'expression de la dynamique du SID. Puis, nous proposons une **démarche de formalisation des besoins** en terme de données et de traitements basée sur un système de règles qui permet de passer de la structure représentant les besoins à notre diagramme décisionnel. Enfin, pour clore la phase d'analyse, nous proposons un **ensemble de règles pour la confrontation et le rapprochement des modèles de besoins** comparables des trois groupes d'acteurs à partir des poids des propriétés du SID.

L'ingénierie des SID :

Modèle :
Nous proposons un **modèle conceptuel multidimensionnel de données et de traitements** (MMDT) qui est une généralisation des modèles existants. L'unification et la généralisation des modèles existants permet de représenter les concepts reconnus et les spécificités des SID déjà énoncées ainsi que celles que nous avons constatées de par notre cadre industriel. Les spécificités relevées dans le cadre industriel sont les liens entre mesures et les traitements relatifs à tous les concepts multidimensionnels.

Démarche :
Nous proposons la démarche de développement globale, appelée le **Trident décisionnel** qui est une extension du processus de développement, appelé le diagramme en Y ou encore 2TUP [Roques 2004], par l'ajout de la branche relative aux besoins des décideurs stratégiques. Notre démarche repose sur un cycle de vie itératif incrémental avec deux points d'évaluation et une phase de prototypage. Nous proposons une démarche qui guide le concepteur décisionnel dans le choix de l'architecture du SID, avant la définition du schéma conceptuel du SID, après l'analyse. Cette dernière est basée sur des **critères de quantification de l'architecture** et un **processus automatique du choix de l'architecture**. Puis, nous proposons une **démarche pour le passage du modèle des besoins d'une part, au modèle conceptuel multidimensionnel de données et de traitements et d'autre part, au modèle entité-association**.

La réutilisation :

Modèle :
Nos modèles sont dévirés à partir de modèles orienté-objet afin de favoriser leur réutilisation. De manière plus spécifique, nous proposons un catalogue de composants réutilisables qui couvre l'analyse et la conception des SID. Le catalogue est composé de **patrons d'analyse** et de **patrons de conception** qui favorisent la réutilisation de notre démarche du Trident décisionnel. Ces patrons intègrent la gestion de la documentation *via* deux rubriques du formalisme.

Démarche :
Notre approche de développement est dévirée à partir d'une approche orienté-objet afin de favoriser la réutilisation. La réutilisation du catalogue est basée sur l'instanciation des patrons suivant les relations inter-patrons existantes. Par ailleurs, nous proposons de formaliser le contexte d'un patron afin de préciser les situations dans lesquelles le patron peut être utilisé et les raisons pour lesquelles il peut l'être.

7.5.2 Avantages de nos propositions

L'évaluation du tableau croisé a commencé lors de notre deuxième mission. Cette mission consistait à reprendre un SID existant et à le faire évoluer. Les acteurs tactiques, des contrôleurs de gestion, nous ont remis un document contenant l'ensemble des tableaux Microsoft Excel qu'ils utilisaient. Ce document nous a permis de déterminer rapidement et sans équivoque les besoins tels qu'ils ont été analysés lors de la construction initiale du SID. Cependant, pour les nouveaux besoins, la tâche fut plus difficile au début car les acteurs tactiques les avaient exprimés en langage naturel. Face à ces difficultés, nous leurs avons demandé des tableaux exprimant ces nouveaux besoins en leur fournissant la structure du notre tableau croisé que nous avions défini. La collecte de ces tableaux croisés, qui exprimaient les besoins que nous avions commencé à analyser nous a permis de faire apparaître les biais dus à l'interprétation technique des concepteurs décisionnels à partir d'une expression non structurée des besoins. Concernant les quatre autres missions où nous avons pu mettre en oeuvre ces tableaux, l'analyse de ces derniers était moins longue car nous avions gagné en compétence sur cette tâche, mais aussi grâce au fait que cette expression structurée étant maîtrisée par les décideurs, cela réduisait d'autant la durée de notre intervention sur le projet. Enfin, les erreurs d'interprétation ont été réduits par l'ajustement progressif des règles de transformation de ces tableaux croisés.

Pour un grand nombre de ces tableaux, seules les entêtes de lignes et de colonnes étaient renseignés. Ce constat concernant le remplissage des tableaux croisés a été fait sur les quatre autres missions. Nous avons aussi pu constater, à partir de ces cinq missions, que plus les utilisateurs maîtrisaient les outils d'exploitation du SID ou faisaient preuve d'une grande expertise de leur métier, plus les tableaux croisés

étaient remplis. Ainsi, les règles ont été basées sur l'expression minimale de ces tableaux, celle comprenant uniquement les entêtes de lignes et de colonnes.

L'évaluation du diagramme décisionnel est postérieure à celle du tableau croisé car initialement l'objectif principal de l'analyse des besoins était de lister les concepts multidimensionnels. Elle a été réalisée sur quatre missions. Le premier avantage procuré est une validation des besoins à partir d'une structure multidimensionnelle, proche de la vision des données par les utilisateurs. De plus, la structure simpliste du diagramme décisionnel, basée uniquement sur deux concepts, était accessible à un plus grand nombre d'acteurs, en particulier les acteurs stratégiques qui sont par nature synthétiques et directs. En effet, nous avons constaté une participation plus importante de leur part lors des validations de besoins. Ces dernières abordaient beaucoup plus de points que le projets où nous avions utilisé uniquement les tableaux croisés car les traitements liés au futur SID représentés sur le diagramme décisionnel suscitaient des interrogations. Ainsi, les problèmes liés aux traitements des SID, qui sont abordées classiquement après la conception du schéma du SID voire en début d'implantation, étaient définis dès la fin de l'analyse. De même, les demandes de complément d'information lors de l'implantation étaient moins fréquentes contrairement aux autres projets où nous n'avions pas utilisé le diagramme décisionnel. Cette comparaison ne repose pas sur des chiffres car seuls deux projets avaient une complexité et des acteurs tactiques d'expertise proches. Les autres projets et leurs acteurs, autres que les concepteurs décisionnels, n'étaient jamais les mêmes.

L'évaluation du graphe des propriétés principalement destiné aux concepteurs a été réalisée, en interne, à la société I-D6. Ce graphe a répondu au besoin de fil directeur pour l'interview des besoins utilisateurs exprimé par les concepteurs décisionnels. De plus, il nous a permis de justifier la modélisation des besoins en terme de traitements ou encore d'expliquer les besoins qui étaient à l'origine d'un traitement qui nécessitaient une prolongation de notre intervention pour une organisation cliente.

L'évaluation du modèle multidimensionnel de données et de traitements a été réalisée sur deux missions. Ces dernières ont mis en avant l'avantage et l'intérêt de la prise en compte des traitements lors de la conception, ce que ne permet pas le schéma étoile. La représentation pour chaque concept des traitements qui lui sont directement liés permet de mettre en avant des besoins qui n'ont pas été correctement définis. De plus, en ce qui les concepteurs décisionnels, le modèle a permis de mieux cerner les impacts d'un problème de données détecté pour une mesure donnée. Notre tâche de contrôle des données a augmenté en efficacité.

Les critères et le processus automatique pour le choix de l'architecture sont transparents à nos organisations clientes. L'évaluation pertinente et très pédagogique est celle faite auprès de l'un des consultants débutants issu d'une formation décisionnelle. Il a réussi à définir l'architecture adaptée à un projet interne à la société I-D6 et à mettre en place le SID lié. Les architectures, qui ont été définies suite

à cette proposition, sont jusqu'à maintenant évaluées comme satisfaisantes par les organisations clientes.

L'évaluation du catalogue de patron a démontré une facilité de réutilisation de notre méthode. En effet, lors de la mise en oeuvre de la méthode les concepteurs se reportent au patron lié à l'élément processus donné (étape, tâche). De plus, les concepteurs exploitent le fil directeur fourni par les patrons *via* leur formalisation qui intègre les prérequis définis et les tâches à réaliser. L'avantage qui a été le plus mentionné par les concepteurs décisionnels est la gestion de la documentation car l'application d'un patron par un concepteur implique la récupération des documents sources et la production des documents cibles.

7.5.3 Inconvénients rencontrés de nos propositions

En terme de modèles, l'inconvénient majeur que nous avons rencontré concerne le dictionnaire décisionnel. Nous avions proposé ce modèle à l'instar du dictionnaire de données classique avec des colonnes supplémentaires pour la définition des traitements du SID lors de la phase d'analyse. Cette proposition s'est relevée inadaptée et difficilement gérable dans le cadre de projets décisionnels car les SID ont la particularité de dériver à partir de systèmes sources suivant de nombreux traitements. A partir d'un essai qui a mis en avant une perte très importante de temps et de nombreuses difficultés de maintenance, nous avons proposé un autre modèle pour l'expression de la dynamique du SID, soit le graphe de propriétés.

Le cycle de vie initial de notre processus de développement a présenté un inconvénient. Il était inadapté dans le cas de projets ayant des délais de livraison très courts car nous avons défini notre processus en deux itérations minimales de la phase d'analyse. Ces deux itérations nécessaires pour rapprocher les besoins des différents acteurs représentent une charge importante dans la répartition des charges des projets. De ce fait, nous avons défini un cycle de vie avec un nombre variable d'itérations.

L'évaluation de notre processus de développement a révélé que l'apprentissage de la mise en oeuvre des ensembles de règles liés était gourmande en temps. Ces ensembles de règles permettent le passage de la collecte à la formalisation des besoins et le passage de la formalisation des besoins à la conception du schéma. Ils sont au nombre de six. Ils incluent chacun une dizaine de règles. Le besoin d'un outil pour la mise en oeuvre automatique des ensembles de règles a été exprimé. Une automatisation de cette tâche s'avère nécessaire.

Nous avons aussi rencontré un inconvénient dans l'utilisation des patrons car les flux liés à la réutilisation des patrons par concepteurs décisionnels ne sont pas gérés. Chaque concepteur gère la réutilisation de chaque patron pour tout projet auquel il

participe. La mise en commun des supports de réutilisation autres que les patrons et leur contenu n'est pas gérée.

7.6 Conclusion

Les projets décisionnels menés dans le cadre industriel ont permis de définir des problèmes à l'origine de l'échec de 40% des systèmes d'information décisionnels, plus précisément de la difficulté de 80% des SID à répondre aux besoins des utilisateurs [Schiefer et al. 2002]. Ces problèmes sont principalement liés à la phase d'analyse et à la phase de conception car ces phases impactent tout le processus d'ingénierie. De plus, de la qualité de l'analyse dépend la qualité du SID.

Ces projets décisionnels ont aussi contribué à l'amélioration des notre méthode au regard des besoins des concepteurs décisionnels de la société collaboratrice et des limites des méthodes de développement existants ainsi que des limites des modèles conceptuels. Nous nous sommes basés sur la diminution des retours de mise en production pour dresser ce constat. Ces améliorations ont été définies au cours de quatre itérations de la démarche qualité Roue de Deming.

Par ailleurs, pour favoriser l'utilisation systématique du catalogue de patrons, nous avons développé un outil eBIPAD. C'est une application Web qui permet la gestion et l'utilisation des patrons du catalogue afin de mettre en place des SID par réutilisation. eBIPAD fournit des fonctionnalités pour trois types d'acteurs : l'ingénieur de patrons, l'ingénieur d'applications et l'administrateur système. Ces acteurs peuvent réaliser respectivement des tâches d'organisation des patrons, des tâches de réutilisation des patrons et des tâches de gestion des acteurs.

L'évaluation de nos propositions a permis de mettre en avant certains avantages remarquables de nos propositions. L'expression structurée des besoins et la prise en compte continue de la dynamique du SID dès l'analyse se sont avérées des points fort de notre méthode pour la définition de SID répondant aux besoins des acteurs. De plus, la représentation sur le schéma du SID des liens entre les concepts et des impacts contribue à augmenter l'efficacité du contrôle de la fiabilité des systèmes et de la cohérence des données. Par ailleurs, les inconvénients que présentent encore la mise en oeuvre de notre méthode concernent le manque d'un outil de modélisation et le besoin d'évolution de l'outil de gestion des patrons. Ces inconvénients peuvent donc être résolus par le développement d'outils. Ainsi, bien que la mise en oeuvre de nos propositions présente ces inconvénients, elle permet de prendre en compte les spécificités des SID au cours de l'analyse et de la conception afin de définir des SID répondant aux besoins des acteurs par des concepteurs décisionnels de tous niveaux d'expertise.

Conclusion

Chapitre 8

Conclusion et perspectives

Table des matières

8.1 Conclusion **283**
8.2 Perspectives **285**
8.2.1 Approfondissement et continuité de la recherche réalisée . 285
8.2.2 Elargissement du domaine de recherche 286

Dans cette thèse, nous avons abordé la définition d'une méthode de développement des systèmes d'information décisionnels (SID) dans un contexte de réutilisation. Notre apport dans ce domaine est une démarche complète d'analyse et de conception du schéma des SID utilisant un modèle de besoins et un modèle conceptuel multidimensionnel de données et de traitements.

8.1 Conclusion

La solution que nous proposons dans cette thèse est constituée de deux parties. Dans la première partie, nous définissons une méthode de développement à partir des besoins des utilisateurs et des sources de données. Elle fournit un guidage continu de la phase d'analyse et de la phase de conception des SID basés sur des architectures variées. Dans la seconde partie, nous proposons un catalogue de patrons pour la mise en place de SID par réutilisation. Ce catalogue est associé à un outil qui facilite la gestion et la réutilisation des patrons.

Notre méthode de développement repose sur la démarche du Trident décisionnel qui est une extension de la démarche de développement des systèmes d'information classiques (SI), appelée « Diagramme en Y » ou encore « 2TUP » [Roques 2004], afin d'intégrer la dimension stratégique des SID. En plus, des besoins liés à la prise de décision pour un métier spécifique et des besoins liés aux systèmes existants, les besoins stratégiques liés à la prise de décision transversale à l'organisation doivent être pris en compte. De ce fait, nous considérons trois groupes d'acteurs au sein d'un projet décisionnel, en l'occurrence le groupe tactique, le groupe système et le groupe stratégique.

La méthode de développement commence par une phase d'analyse explicite et spécifique aux SID. Cette phase d'analyse repose sur :

- une structure de collecte des besoins des utilisateurs proche des tableaux de bord qu'ils manipulent au quotidien ; les utilisateurs des SID étant des décideurs. Elle est appelée tableau croisé. Elle permet de définir les données, en l'occurrence l'aspect statique des besoins des utilisateurs. Cette structure présente l'avantage de réduire le biais dû à l'interprétation technique de leurs besoins par les concepteurs décisionnels car ces utilisateurs peuvent exprimer seuls leurs besoins,
- un graphe de propriétés pour la collecte de l'aspect dynamique des besoins de tous les acteurs du SID suivant les deux contextes des SID : technique et décision. L'annotation de ce graphe permet de définir des traitements liés à la dérivation des données depuis les systèmes sources et à la préparation des données pour faciliter la prise de décision. Ce graphe présente l'avantage de guider les concepteurs décisionnels au cours des interviews utilisateurs et de leurs fournir un support pour la spécification de l'aspect dynamique du SID. De plus, l'expression en langage naturel des annotations favorise la compréhension et la validation par les utilisateurs,

- un modèle de besoins, proche de la vision multidimensionnelle des données par les utilisateurs, basé sur les deux principaux concepts du décisionnel « fait » et « dimension ». Il permet de représenter de manière unifiée aussi bien l'aspect statique que l'aspect dynamique des besoins liés à la dérivation et à la préparation des données. Il facilite la validation des besoins des utilisateurs par eux-mêmes avant la conception du SID,
- une démarche semi-automatique qui parallélise les tâches de l'analyse des trois groupes d'acteurs. Elle inclut deux phases de confrontation ; chaque confrontation consiste à comparer et à rapprocher deux modèles de besoins. La première entre celui des besoins tactiques et celui des besoins stratégiques afin définir le modèle des besoins des utilisateurs. La seconde entre le modèle des besoins utilisateurs et celui des besoins systèmes afin de définir le modèle des besoins confrontés du SID. Notre démarche favorise le gain de temps et elle prend en compte les besoins de tous les acteurs du SID suivant les priorités parmi eux.

A partir de ce modèle des besoins du SID, nous proposons une automatisation du choix de l'architecture adaptée au projet décisionnel. Cette dernière guide les concepteurs décisionnels face à l'hétérogénéité des architectures des SID. Ce processus est défini à partir de six critères qui sont évalués suivant les graphes de propriétés des groupes d'acteurs. Il permet de définir l'architecture en fonction des besoins des acteurs du projet mais, pas uniquement en fonction du marché des outils décisionnels. Cette architecture est composée de modules ayant des schémas multidimensionnels ou des schémas entité-association. Nous proposons donc deux ensembles de règles pour le passage du diagramme décisionnel à l'une de ces modélisations. Ces ensembles de règles présente l'avantage de guider les concepteurs décisionnels pour la définition du schéma conceptuel de tout module décisionnel qui compose l'architecture des SID.

La modélisation multidimensionnelle n'ayant pas de modèle reconnu, nous proposons un modèle conceptuel multidimensionnel généralisé. Ce modèle permet de représenter toutes les spécificités de la structure complexe des concepts multidimensionnels et de la cohérence des agrégations ainsi que les traitements liés à la dérivation et à la préparation des données. Il vise à répondre au manque de modèle standard via la représentation unifiée des aspects statique et dynamique et la prise en compte de l'ensemble de spécificités multidimensionnelles définies dans la littérature et rencontrées dans le cadre industriel. De plus, il explicite aussi bien les liens entre mesures d'un fait que les liens entre les paramètres d'une dimension.

Cette méthode est capitalisée dans un catalogue d'une trentaine de patrons représentés avec le formalisme P-SIGMA étendu. Cette extension du formalisme P-SIGMA que nous proposons intègre la gestion de la documentation pour assurer une traçabilité efficace et pour améliorer les échanges entre les concepteurs décisionnels. De plus, elle formalisme le contexte de réutilisation d'un patron en tenant compte des conditions dans lesquels ce patron est utilisable et les conditions dans lesquelles ce patron requiert un autre patron. Cette formalisation du contexte favorise une réutilisation systématique des patrons.

Pour simplifier l'utilisation du catalogue, nous avons développé un système de gestion de patrons permettant d'organiser et de réutiliser les patrons. Cet outil est basé sur une architecture client-serveur avec un client léger qui est un navigateur web.

Les mises en application de nos propositions au cours des missions menées au sein de la collaboratrice I-D6 ont permis de mettre en avant, suivant le processus qualité de la Roue de Deming, des limites et des inconvénients de nos propositions. Ils ont aussi permis de définir des points indispensables qu'une méthode doit vérifier afin d'assurer le développement de SID fiables. Le contexte des mises en application variant d'une organisation cliente à l'autre a permis de qualifier nos propositions dans des domaines et métiers variés.

8.2 Perspectives

Ce travail ouvre la voie vers différentes perspectives. Nous évaluons ces perspectives en considérant deux plans : celui de l'approfondissement et de continuité de la recherche réalisée et celui de l'élargissement du domaine de recherche.

8.2.1 Approfondissement et continuité de la recherche réalisée

Nous avons défini un modèle conceptuel multidimensionnel qui permet de représenter l'ensemble des spécificités liées à la structure complexe des concepts, à la cohérence de l'interrogation ainsi que les traitements du SID. Le passage au niveau logique implique une gestion particulière de ces spécificités et des traitements en fonction de la technologie du système d'implantation. Par exemple, les rôles multiples des dimensions, peuvent être appréhendés avec une seule structure de données ou deux structures de données identiques. Dans un cas, au niveau logique, la gestion de la structure est proche de la vision de la structure par les utilisateurs. Dans l'autre cas, les concepteurs doivent gérer ces structures de telle sorte que leur accès par les utilisateurs soit transparent. La définition de **notre modèle multidimensionnel au niveau logique et physique** représente un enjeu pour la prise en compte de toutes les spécificités multidimensionnelles tout au long du processus d'ingénierie.

De plus, la prise en compte de ces spécificités et de ces traitements lors de la définition des schémas de correspondances est un facteur déterminant pour la fiabilité et la pérennité du SID. Actuellement, ces schémas ne permettent de représenter que les traitements liés à la dérivation des données. Il serait donc intéressant de définir **un modèle de correspondances des SID qui représente toutes les spécificités des SID ainsi que les traitements de dérivation et de préparation des données**.

Actuellement, l'outil eBIPAD est basé sur une gestion non-suivie de la réutilisation des patrons et des résultats de cette réutilisation. Pour augmenter les possibilités de développements par réutilisation des SID, il est nécessaire d'ajouter une gestion des flux. Cette dernière suivrait le processus d'ingénierie par réutilisation et ces produits afin que les utilisations des patrons soient disponibles et réutilisables.

Notre démarche de développement est basée sur des ensembles de règles. L**a génération automatique du schéma conceptuel multidimensionnel du SID à partir des tableaux croisés des décideurs et des schémas entité-association des sources *via* le diagramme décisionnel** constitue un approfondissement intéressant des travaux présentés dans cette thèse.

8.2.2 Elargissement du domaine de recherche

Les perspectives d'élargissement de cette thèse s'articulent autour de trois axes :

Les modèles de besoins basés sur des ontologies métiers : face à la récurrence des métiers ou des domaines d'activités qui font l'objet de projets décisionnels au sein des organisations, il serait intéressant d'associer les ontologies définissant ces métiers. Ainsi, il serait possible de définir des diagrammes décisionnels génériques par domaine d'activité voire un diagramme décisionnel générique à toute l'organisation. D'une part, ces diagrammes décisionnels permettraient une adaptation plus pertinente du patron produit associé à ces diagrammes. D'autre part, ils permettraient de disposer de modèles de besoins par domaine d'activités.

La qualité des schémas de besoins ainsi que des schémas de données et de traitements [Vassiliadis et al. 2000; Maes and Poels 2006; Mazón et al. 2006b] : le SID est un système critique pour la réactivité d'une organisation. Il serait donc pertinent d'évaluer la qualité des schémas produits à partir de critères définis en fonction des besoins de l'organisation (tactiques, stratégiques et systèmes) mais aussi en fonction de son environnement. Ces propositions permettraient de confronter les modèles de besoins à partir du poids des besoins pour les acteurs mais aussi à partir de paramètres externes qui contraignent la réactivité d'une organisation.

L'ingénierie des SID à partir de besoins semi-structurés [Boussaid et al. 2006; Nassis et al. 2006; Ravat et al. 2007a] : notre méthode tient compte des besoins des utilisateurs exprimés sous forme de tableaux de chiffres ou des schémas entité-relation, il serait intéressant d'aborder la problématique de définition d'une méthode de développement des SID sous un autre angle, en l'occurrence à partir de besoins exprimés sous forme de documents semi-structurés. Ces propositions permettraient par exemple de répondre aux besoins des décideurs liés au domaine du médical car le dossier médical patient (DMP) est un document XML.

Annexe A

Diagrammes d'activités du choix de l'architecture

Dans cette annexe nous présentons 6 des 32 diagrammes d'activités qui décrivent la logique procédurale du choix de l'architecture décisionnelles. Plus précisément en fonction des six critères NCD (niveau de couverture des données), NTD (niveau de traitement des données), NCS (niveau de complexité des sources), NEE (niveau d'équipement décisionnel existant), NPA (niveau de pérennité de l'architecture) et NDS (niveau décisionnel souhaité).

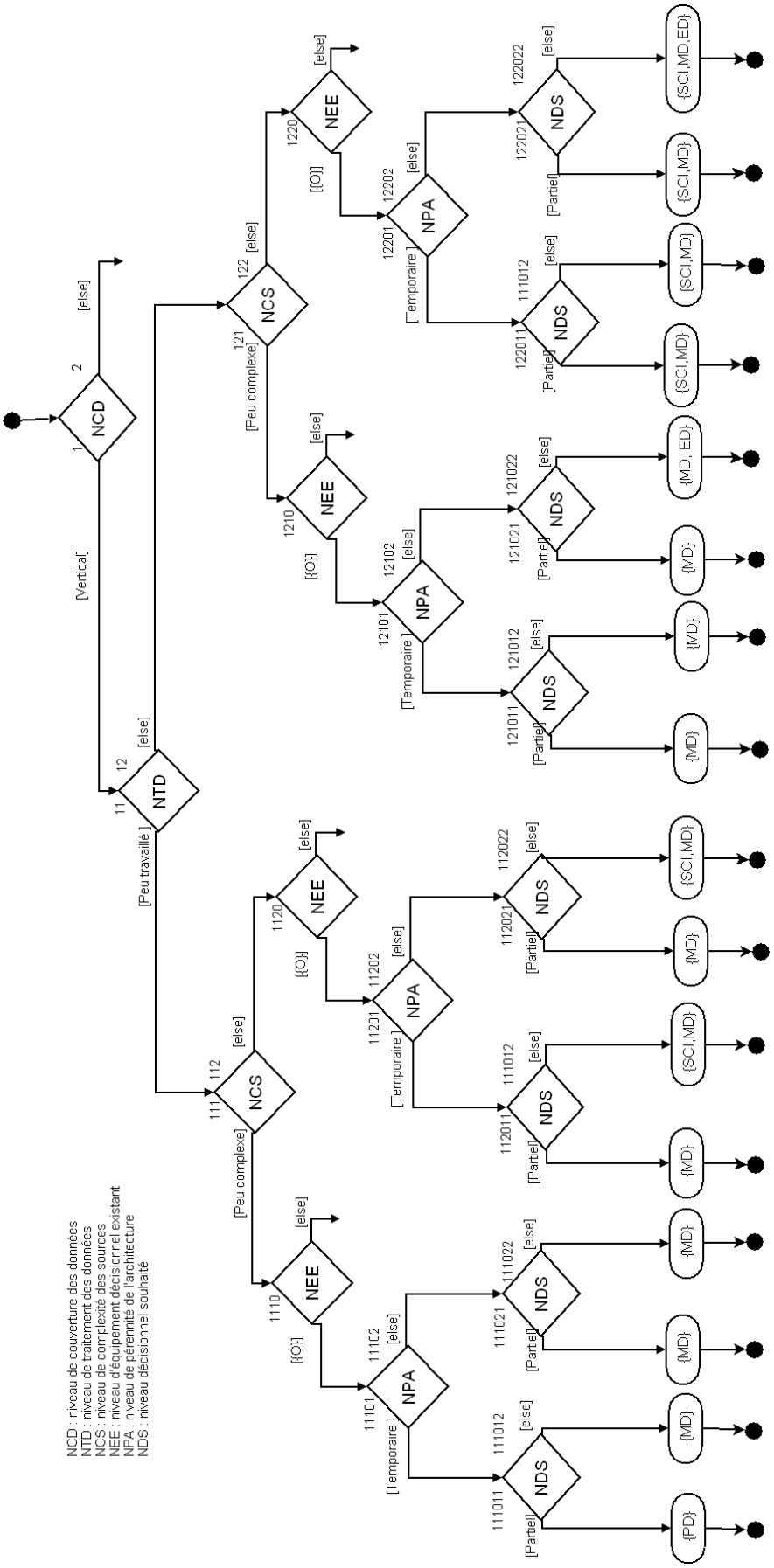


FIG. A.1 – Diagramme d'activités du choix de l'architecture à partir de NCD=Vertical et NEE={∅}

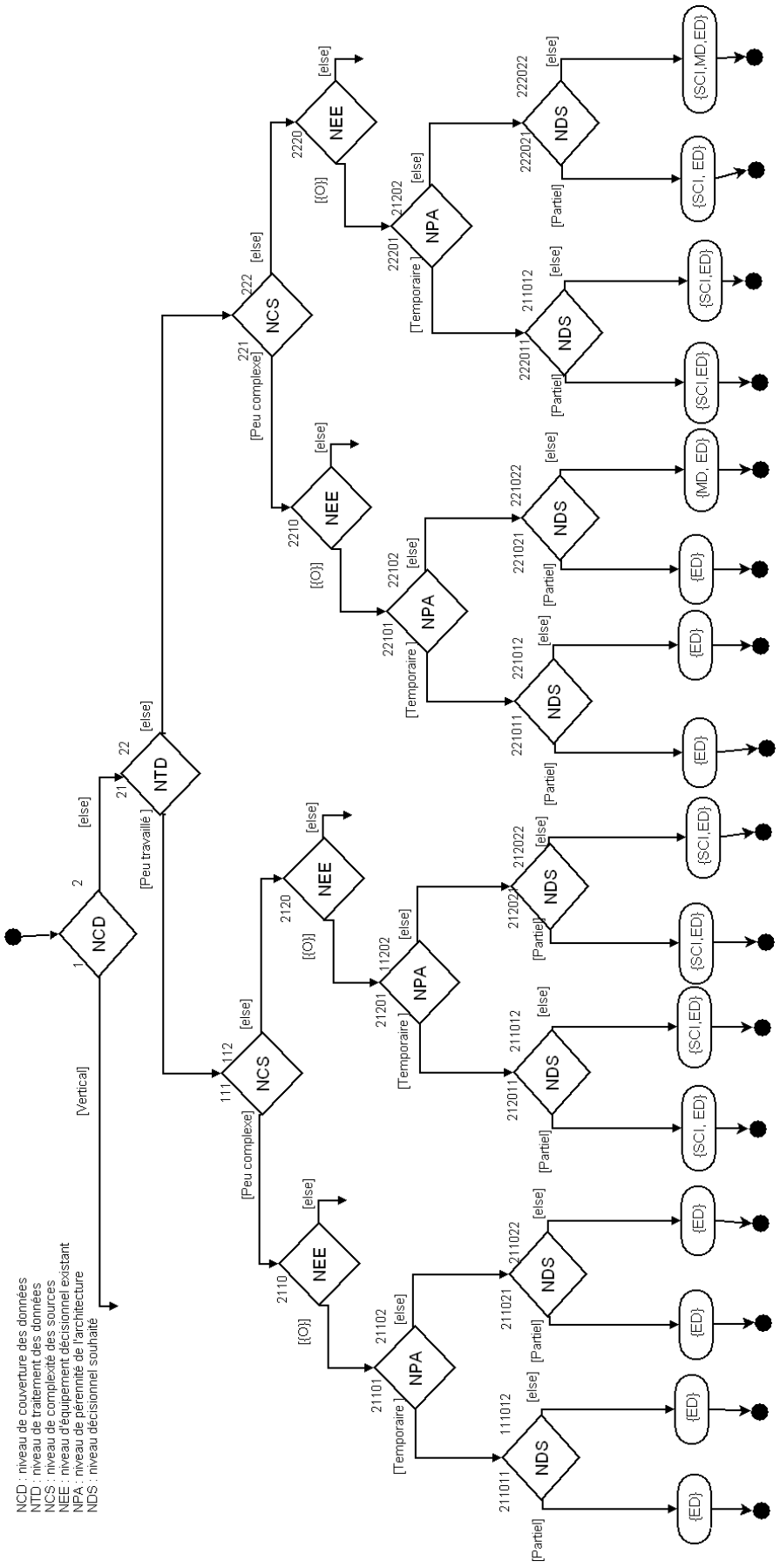


FIG. A.2 – Diagramme d'activités du choix de l'architecture à partir de NCD=Tranversal et NEE={∅}

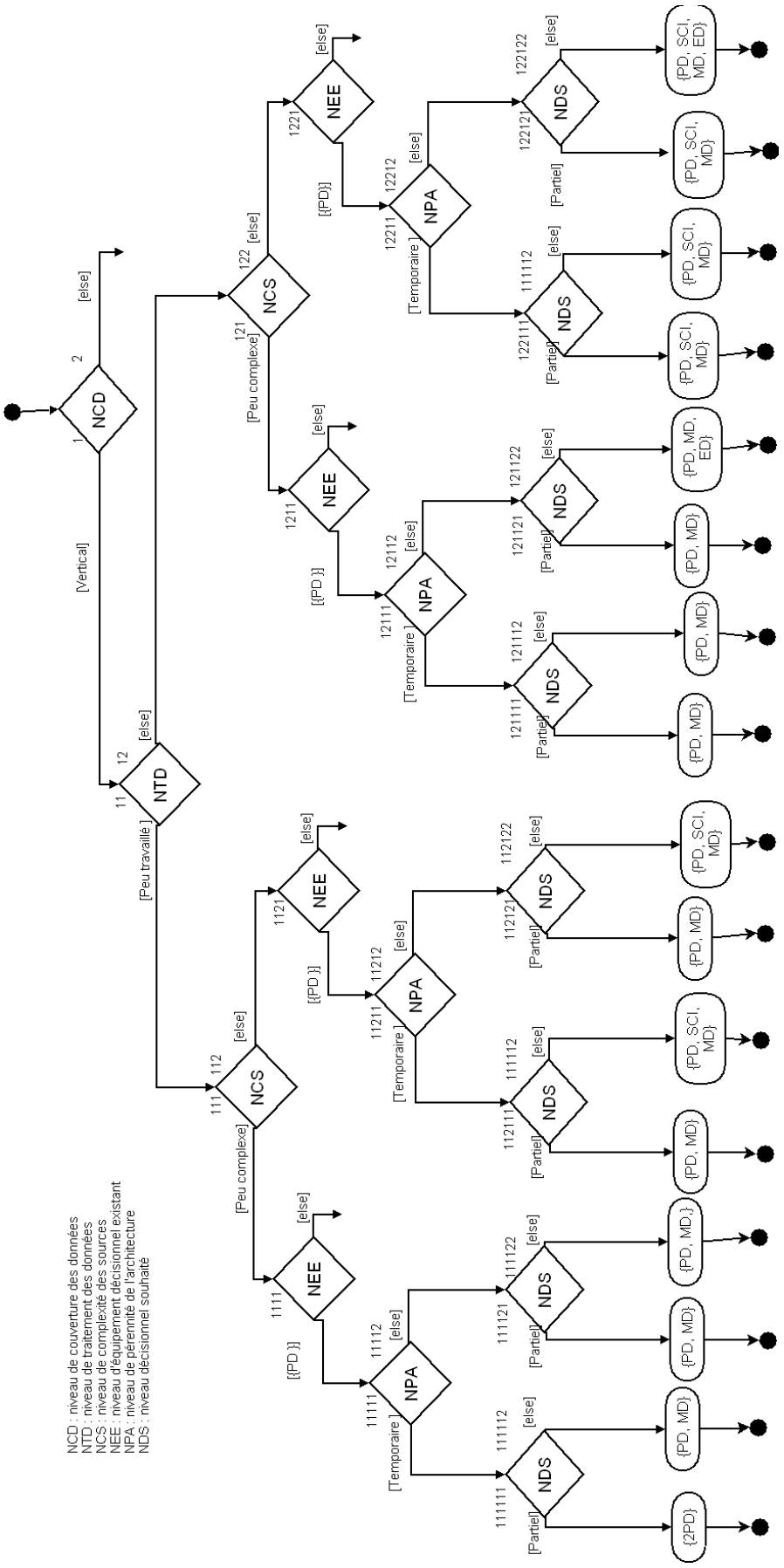


FIG. A.3 – Diagramme d'activités du choix de l'architecture à partir de NCD=Vertical et NEE={$PD^{1..N}$}

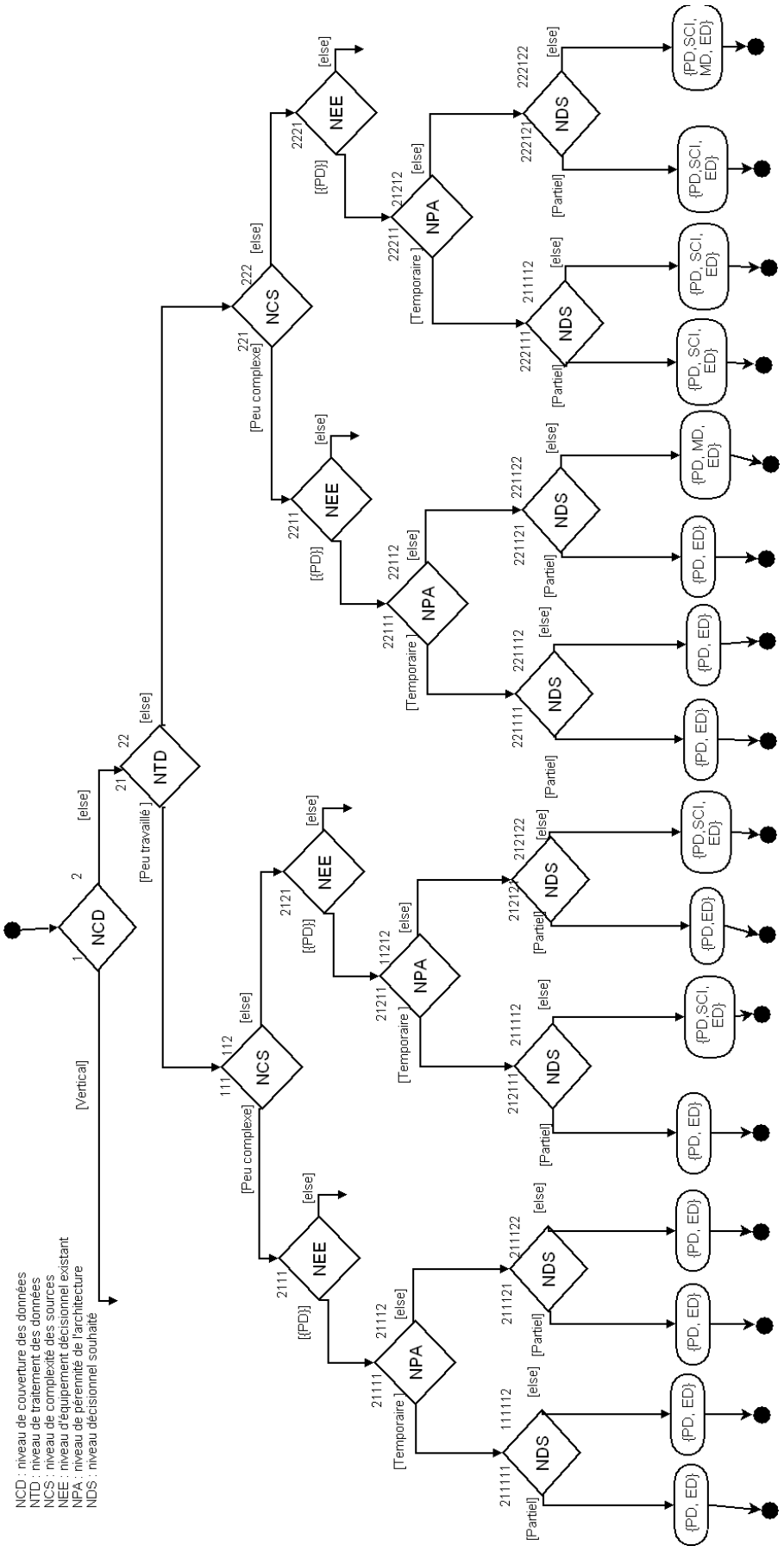


FIG. A.4 – Diagramme d'activités du choix de l'architecture à partir de NCD=Tranversal et NEE=$\{PD^{1..N}\}$

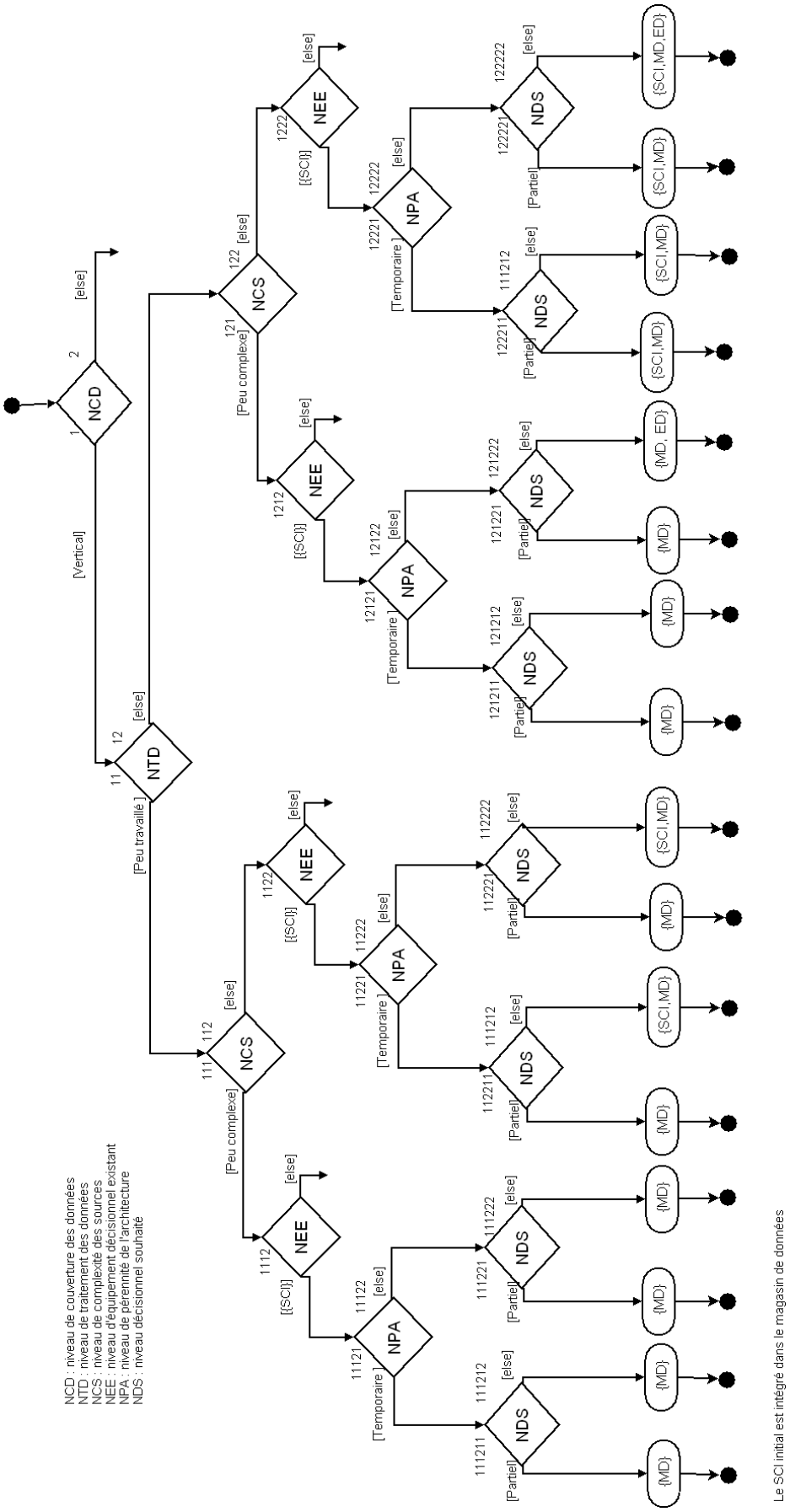


FIG. A.5 – Diagramme d'activités du choix de l'architecture à partir de NCD=Vertical et NEE=$\{SCI^{1..N}\}$

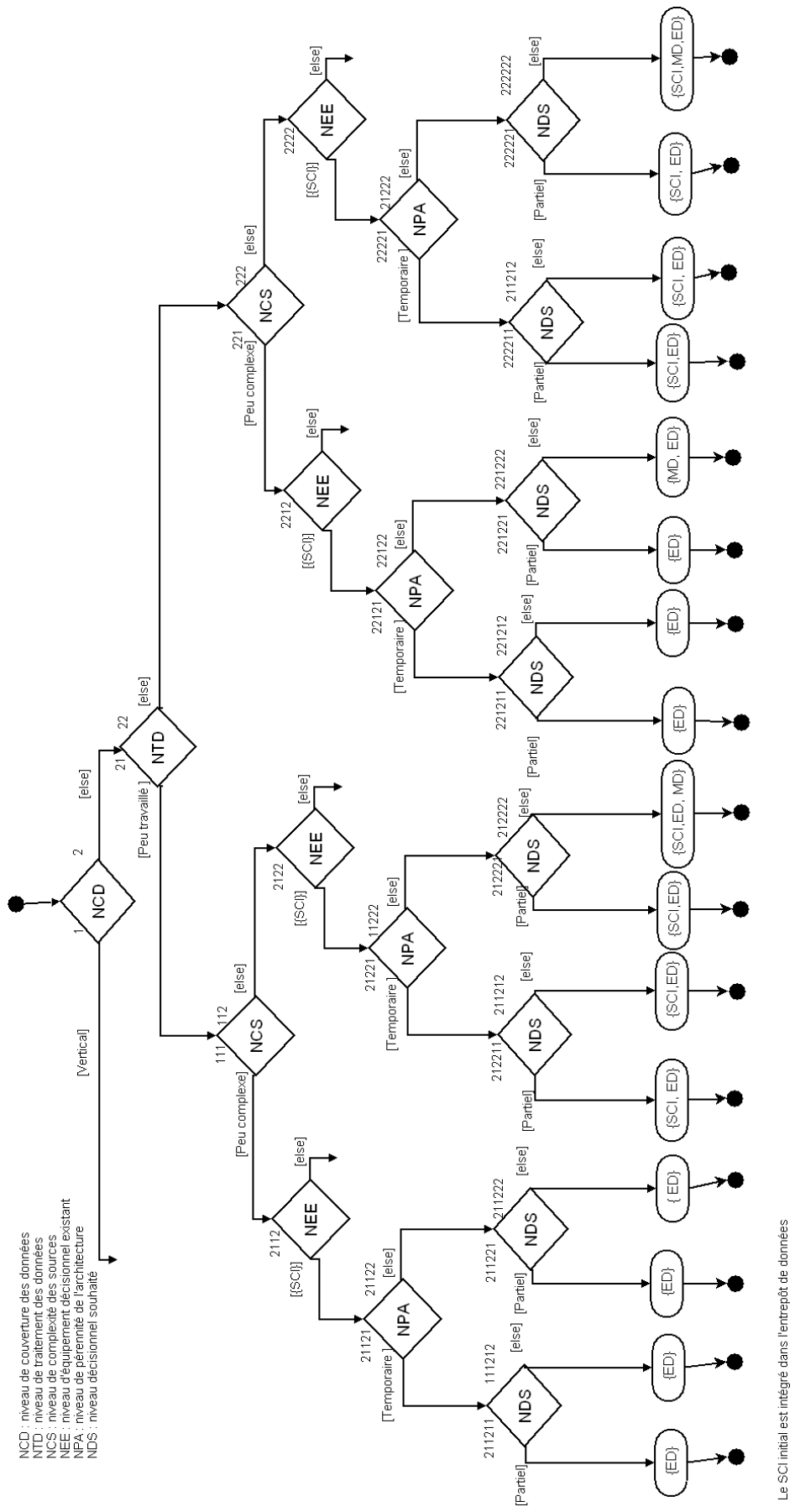


FIG. A.6 – Diagramme d'activités du choix de l'architecture à partir de NCD=Tranversal et NEE=$\{SCI^{1..N}\}$

Bibliographie

(1998a). *Actes du XVIème Congrès INFORSID, Montpellier, France, 12-15 mai, 1998.*

(1998b). *DOLAP '98, ACM First International Workshop on Data Warehousing and OLAP, November 7, 1998, Bethesda, Maryland, USA, Proceedings.* ACM.

(1999a). *DOLAP '99, ACM Second International Workshop on Data Warehousing and OLAP, November 6, 1999, Kansas City, Missouri, USA, Proceedings.* ACM.

(1999b). *Proceedings of the 15th International Conference on Data Engineering, 23-26 March 1999, Sydney, Austrialia.* IEEE Computer Society.

(2001). *Actes du XIXème Congrès INFORSID, Martigny, Suisse, 24-27 mai, 2001.*

(2004a). *Actes du XXIIème Congrès INFORSID, Biarritz, France, 25-28 mai, 2004.*

(2006). *Actes du XXIVème Congrès INFORSID, Hammamet, Tunisie, 31 mai - 4 juin, 2006.*

Abelló, A., Samos, J., and Saltor, F. (2001a). A framework for the classification and description of multidimensional data models. In Mayr et al. [2001], pages 668–677. 3540425276.

Abelló, A., Samos, J., and Saltor, F. (2002). Yam2 (yet another multidimensional model) : An extension of uml. In Nascimento et al. [2002], pages 172–181. 0769516386.

Abelló, A., Samos, J., and Saltor, F. (2006). Yam^2 : a multidimensional conceptual model extending uml. *Inf. Syst. journal*, 31(6) :541–567.

Alexander., C. (1977). *A Pattern Language.* Oxford University Press, New York, USA. 0195019199.

Annoni, E. (2005). Bipad : Une méthode d'analyse et de conception des systèmes d'information décisionnels. In *Colloque des Doctorants de l'Ecole Doctorale Informatique et Télécommunications EDIT*, pages 102–105.

Annoni, E., Ravat, F., and Teste, O. (2006a). Traitements à l'origine des systèmes d'information décisionnels. *Revue des Sciences et Technologies de l'Information*, 11(6) :115–143.

Annoni, E., Ravat, F., Teste, O., and Zurfluh, G. (2005a). Les systèmes d'informations décisionnels : une approche d'analyse et de conception à base de patrons. *Revue des Sciences et Technologies de l'Information, série ISI, «Méthodes Avancées de Développement des SI »*, 10(6) :81–106.

Annoni, E., Ravat, F., Teste, O., and Zurfluh, G. (2005b). Une méthode d'analyse et de conception des systèmes d'information décisionnels par réutilisation de patrons. *Conférence AIM Association Information and Management.*

Annoni, E., Ravat, F., Teste, O., and Zurfluh, G. (2006b). Automating the choice of decision support system architecture. In *Database and Expert Systems Applications*, volume 4080/2006, pages 75–84.

Annoni, E., Ravat, F., Teste, O., and Zurfluh, G. (2006c). Modélisation adaptée aux besoins utilisateurs dans le développement des systèmes d'information décisionnels. *Revue des Nouvelles Technologies de l'Information*, RNTI-B-2 (EDA'06) :23–38.

Annoni, E., Ravat, F., Teste, O., and Zurfluh, G. (2006d). Méthode de développement des systèmes d'information décisionnels : Roue de deming. In DBL [2006], pages 657–673.

Annoni, E., Ravat, F., Teste, O., and Zurfluh, G. (2006e). Towards multimensional requirement design. In *Data Warehousing and Knowledge Discovery, 8th International Conference, DaWak 2006, Krakow, Poland, September 4-8, 2006, Proceedings. Lecture Notes in Computer Science 4081 Springer 2006*, volume 4081/2006, pages 75–84.

Annoni, E., Ravat, F., Teste, O., and Zurfluh, G. (2007). Data and process analyses of data warehouse requirements. In *Software Engineering and Knowledge Engineering, 19th International Conference, SEKE 2007, Boston, USA, July 9-11, 2007*.

Atzeni, P., Chu, W. W., Lu, H., Zhou, S., and Ling, T. W., editors (2004). *Conceptual Modeling - ER 2004, 23rd International Conference on Conceptual Modeling, Shanghai, China, November 2004, Proceedings*, volume 3288 of *Lecture Notes in Computer Science*. Springer. 3540237232.

Avison, D. and Hanifa, S. (1997). *The Information systems development life cycle : a first course in information systems*. McGraw-Hill Publishing Co. 0077092449.

Barbier, F., Cauvet, C., Oussalah, M., Rieu, D., Bennasri, S., and Souveyet, C. (2004). Concepts clés et techniques de réutilisation dans l'ingénierie des systèmes d'information. *Revue L'Objet*, 10(1) :11–35.

Barros, S. (1995). *Maintenance logicielle : analyse d'impact, problématique et mise en oeuvre*. Thèse de doctorat, Université Henri Poincaré, Nancy I, Nancy, France.

Barros, S. (1997). *Analyse à priori des conséquences de la modification des systèmes logiciels : de la théorie à la pratique*. Thèse de doctorat, Université Paul Sabatier, Toulouse, France.

Basili, V., Caldiera, G., and Rombach, D. (1994). Goal/Question/Metric Paradigm . *Encyclopedia of Software Engineering*, 1 :528–532.

Beck, K. and Cunningham, W. (1987). Using pattern languages for object-oriented programs. Technical report. CR-87-43.

Blaschka, M., Sapia, C., and Höfling, G. (1999a). On schema evolution in multidimensional databases. In Mohania and Tjoa [1999], pages 153–164. 3540664580.

Blaschka, M., Sapia, C., Hofling, G., and Dinter, B. (1999b). An overview of multidimensional data models for olap. Technical report. Technical Report FR-1999-001.

Bonifati, A., Cattaneo, F., Ceri, S., Fuggetta, A., and Paraboschi, S. (2001). Designing data marts for data warehouses. *ACM Trans. Softw. Eng. Methodol.*, 10(4) :452–483.

Boussaid, O., Messaoud, R. B., Choquet, R., and Anthoard, S. (2006). X-warehousing : An xml-based approach for warehousing complex data. In Manolopoulos et al. [2006], pages 39–54.

Bouzeghoub, M., Fabret, F., and Matulovic-Broqué, M. (1999). Modeling the data warehouse refreshment process as a workflow application. In Gatziu et al. [1999], page 6.

Bruckner, R., List, B., and Schiefer, J. (2001). Developing requirements for data warehouse systems with use cases. pages 329–335. In Proceedings of the Seventh Americas Conference on Information Systems, AMCIS, Boston, Massachusetts, USA.

Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., and Stal, M. (1996). *Pattern-oriented software architecture : A system of patterns*. John Wiley & Sons, Chichester, UK. 0471958697.

Bushmann, F., Meunier, R., Rohnert, H., Sommerlad, P., and Stal, M. (1996). *A System of Patterns : Pattern-Oriented Software Architecture*. John Wiley, New York, USA. 0471958697.

Cabibbo, L. and Torlone, R. (1998). A logical approach to multidimensional databases. pages 183–197.

Cabibbo, L. and Torlone, R. (2000). The design and development of a logical system for olap. In Kambayashi et al. [2000], pages 1–10. 3540679804.

Carneiro, L. and Brayner, A. (2002). X-META : A methodology for data warehouse design with metadata management. In *Design and Management of Data Warehouses (DMDW)*., pages 13–22.

Cauvet, C., Gzara, L., Ramadour, P., and Rieu, D. (2000). Ingénierie des systèmes d'information produit : une approche méthodologique centrée réutilisation de patrons. *Revue l'OBJET*, 6(4) :445–468.

Cauvet, C., Rieu, D., Espinasse, B., Giraudin, J.-P., and Tollenaere, M. (1998). Ingénierie des systèmes d'information produit : une approche méthodologique centrée réutilisation de patrons. In DBL [1998a], pages 71–90.

Cavero, J. M., Piattini, M., and Marcos, E. (2001). Midea : A multidimensional data warehouse methodology. In *ICEIS*, pages 138–144.

Chrisment, C., Pujolle, G., Ravat, F., Teste, O., and Zurfluh, G. (2005b). Les entrepôts de données. In Zurfluh, G., editor, *Traité Informatique des Techniques de l'Ingénieur - H3870*, page 10. Techniques de l'Ingénieur.

Coad, P. (1992). Object-oriented patterns. *Commun. ACM*, 35(9).

Codd, E., Codd, S., and Salley, C. (1993). Providing olap to user-analysts : An it mandate. Technical report.

Conte, A., Fredj, M., Giraudin, J.-P., and Rieu, D. (2001). P-sigma : un formalisme pour une représentation unifiée de patrons. In DBL [2001], pages 67–86.

Coplien, J. O. (1992). *Advanced C++ Programming Styles and Idioms*. Addison-Wesley. 0201548550.

Deming, W. E. (1953). Techniques statistiques et commerce international. *Courrier de la Normalisation*, 109 :I–II.

Deneckère, R. (2001). *Approche d'extension de méthode fondée sur l'utilisation de composants génériques*. Thèse de doctorat, Université Paris I, Paris, France.

Diviné, M. (1982). *Parlez-vous merise ?* Editions Eyrolles, Paris, France. 2212041071.

Duong, P. (2000). *Spécification d'une démarche de développement à base de patterns*. PhD thesis. Mémoire de fin d'études de l'Institut de la Francophonie pour l'Informatique, Vietnam - Laboratoire LSR/INPG.

Ecudié, A. (1997). *Plate-forme de traçabilité : des concepts à l'implémentation*. Thèse de doctorat, Université Paul Sabatier, Toulouse, France.

Eder, J. and Welzer, T., editors (2003). *The 15th Conference on Advanced Information Systems Engineering (CAiSE '03), Klagenfurt/Velden, Austria, 16-20 June, 2003, CAiSE Forum, Short Paper Proceedings, Information Systems for a Connected Society*, volume 74 of *CEUR Workshop Proceedings*. CEUR-WS.org. 8643505498.

Embley, D. W., Olivé, A., and Ram, S., editors (2006). *Conceptual Modeling - ER 2006, 25th International Conference on Conceptual Modeling, Tucson, AZ, USA, November 6-9, 2006, Proceedings*, volume 4215 of *Lecture Notes in Computer Science*. Springer.

Feki, J., Ben-Abdallah, H., and Ben-Abdallah, M. (2006). Réutilisation des patrons en étoile. In DBL [2006], pages 687–701.

Fernandez, A. (2003). *Les nouveaux tableaux de bord des managers*. Editions d'Organisation, France. 2708128825.

Fowler, M. (2004). *UML 2.0*. Campus Press, Paris, France. 2744017132.

Franconi, E. and Sattler, U. (1999). A data warehouse conceptual data model for multidimensional aggregation. In Gatziu et al. [1999], page 13.

Gam, I. and Salinesi, C. (2006a). Analyse des exigences pour la conception d'entrepôts de données. In *INFORSID 2006, Hammamet (Tunisie), 31/05/06-03/06/06*, pages 1023–1038. INFORSID ed.

Gam, I. and Salinesi, C. (2006b). A requirement-driven approach for designing data warehouses. In *Requirements Engineering :Foundation for Software Quality (REFSQ)*.

Gamma, E., Helm, R., Johnson, R. E., and Vlissides, J. M. (1995). *Design Patterns, Elements of reusable Object-Oriented Software*. Addison-Wesley Publishing Company. 0201633612.

Gardarin, G. (2003. Première édition 1999). *Bases de données*. Eyrolles, Paris, France. 2212112815.

Gardner, S. R. (1998). Building the data warehouse. *Commun. ACM*, 41(9) :52–60.

Gatziu, S., Jeusfeld, M. A., Staudt, M., and Vassiliou, Y., editors (1999). *Proceedings of the Intl. Workshop on Design and Management of Data Warehouses, DMDW'99, Heidelberg, Germany, June 14-15, 1999*, volume 19 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Ghozzi, F. (2004). *Conception et manipulation de bases de données dimensionnelles à contraintes*. Thèse de doctorat, Université Paul Sabatier, Toulouse, France.

Ghozzi, F., Ravat, F., Teste, O., and Zurfluh, G. (2005). Méthode de conception d'une base multidimensionnelle contrainte. In *Revue des Nouvelles Technologies de l'Information - Entrepôts de Données et l'Analyse en ligne (EDA'05)*, volume RNTI-B-1, pages 51–70. Cépadues éditions.

Giorgini, P., Rizzi, S., and Garzetti, M. (2005). Goal-oriented requirement analysis for data warehouse design. In Song and Trujillo [2005], pages 47–56. 1595931627.

Giovinazzo, W. A. (2000). *Object-oriented data warehouse design : building a star schema*. Prentice Hall PTR, Upper Saddle River, NJ, USA. 0130850810.

Golfarelli, M., Maio, D., and Rizzi, S. (1998b). The dimensional fact model : A conceptual model for data warehouses. *Int. J. Cooperative Inf. Syst.*, 7(2-3) :215–247.

Golfarelli, M. and Rizzi, S. (1998a). Methodological framework for data warehouse design. In *DOLAP '98, ACM First International Workshop on Data Warehousing and OLAP, November 7, 1998, Bethesda, Maryland, USA, Proceedings*, pages 3–9. ACM.

Golfarelli, M. and Rizzi, S. (1998b). Methodological framework for data warehouse design. In DBL [1998b], pages 3–9.

Guzelian, G. (2004). Méthode de conception de système d'information : une approche orientée-composant. In *MAJECSTIC'04, MAnifestation des JEunes Chercheurs STIC, Calais, France*.

Gzara, L. (2000). *Les patterns pour l'ingénierie des systèmes d'information produit*. Thèse de doctorat, Institut national polytechnique, Grenoble, France, Grenoble, France.

Hameurlain, A., Cicchetti, R., and Traunmüller, R., editors (2002). *Database and Expert Systems Applications, 13th International Conference, DEXA 2002, Aix-en-Provence, France, September 2-6, 2002, Proceedings*, volume 2453 of *Lecture Notes in Computer Science*. Springer. 3540441263.

Husemann, B., Lechtenborger, J., and Vossen, G. (2000). Conceptual data warehouse modeling. In Jeusfeld et al. [2000], page 6.

Ibrahim, M. T., Küng, J., and Revell, N., editors (2000). *Database and Expert Systems Applications, 11th International Conference, DEXA 2000, London, UK, September 4-8, 2000, Proceedings*, volume 1873 of *Lecture Notes in Computer Science*. Springer. 3540679782.

Inmon, W. H. (1996). *Building the data warehouse (2nd ed.)*. John Wiley & Sons, Inc., New York, NY, USA. 0471141615.

Jarke, M., Lenzerini, M., Vassiliou, Y., and Vassiliadis, P. (2001). *Fundamentals of Data Warehouses*. Springer-Verlag New York, Inc., Secaucus, NJ, USA. 3540420894.

Jeusfeld, M. A., Shu, H., Staudt, M., and Vossen, G., editors (2000). *Proceedings of the Second Intl. Workshop on Design and Management of Data Warehouses, DMDW 2000, Stockholm, Sweden, June 5-6, 2000*, volume 28 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Johnson, R. E. (1992). Documenting frameworks using patterns. In *OOPSLA*, pages 63–76.

Jones, M. E. and Song, I.-Y. (2005). Dimensional modeling : identifying, classifying & applying patterns. In Song and Trujillo [2005], pages 29–38. 1595931627.

Kambayashi, Y., Mohania, M. K., and Tjoa, A. M., editors (2000). *Data Warehousing and Knowledge Discovery, Second International Conference, DaWaK 2000, London, UK, September 4-6, 2000, Proceedings*, volume 1874 of *Lecture Notes in Computer Science*. Springer. 3540679804.

Kimball, R. (1996). *The data warehouse toolkit : practical techniques for building dimensional data warehouses*. John Wiley & Sons, Inc., New York, NY, USA.

Kimball, R., Reeves, L., Thornthwaite, W., Ross, M., and Thornwaite, W. (1998). *The Data Warehouse Lifecycle Toolkit : Expert Methods for Designing*. John Wiley & Sons, Inc., New York, NY, USA. 0471255475.

Lakshmanan, L. V. S., editor (2002). *Design and Management of Data Warehouses 2002, Proceedings of the 4th Intl. Workshop DMDW'2002, Toronto, Canada, May 27, 2002*, volume 58 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Le Moigne, J. (1977). *La Théorie du système général : Théorie de la modélisation (4ième edition)*. Presses Universitaires de France, Paris. 2130465153.

Lee, M.-L., Tan, K.-L., and Wuwongse, V., editors (2006). *Database Systems for Advanced Applications, 11th International Conference, DASFAA 2006, Singapore, April 12-15, 2006, Proceedings*, volume 3882 of *Lecture Notes in Computer Science*. Springer.

Lenz, H.-J., Vassiliadis, P., Jeusfeld, M. A., and Staudt, M., editors (2003b). *Design and Management of Data Warehouses 2003, Proceedings of the 5th Intl. Workshop DMDW'2003, Berlin, Germany, September 8, 2003*, volume 77 of *CEUR Workshop Proceedings*. CEUR-WS.org.

List, B., Bruckner, R. M., Machaczek, K., and Schiefer, J. (2002). A comparison of data warehouse development methodologies case study of the process warehouse. In Hameurlain et al. [2002], pages 203–215. 3540441263.

List, B., Schiefer, J., and Tjoa, A. M. (2000). Process-oriented requirement analysis supporting the data warehouse design process - a use case driven approach. In Ibrahim et al. [2000], pages 593–603. 3540679782.

Luján-Mora, S. and Trujillo, J. (2003). A comprehensive method for data warehouse design. In Lenz et al. [2003b], pages 1.1–1.14.

Luján-Mora, S., Trujillo, J., and Song, I.-Y. (2006). A uml profile for multidimensional modelling in data warehouses. *Data & Knowledge Engineering*, 59(3) :725–769.

Luján-Mora, S., Vassiliadis, P., and Trujillo, J. (2004b). Data mapping diagrams for data warehouse design with uml. In Atzeni et al. [2004], pages 191–204. 3540237232.

Luján-Mora, S. (2005). *Data Warehouse Design with UML*. PhD thesis, Department of Software and Computing Systems, University of Alicante, Alicante, Espagne.

Maes, A. and Poels, G. (2006). Evaluating quality of conceptual models based on user perceptions. In Embley et al. [2006], pages 54–67.

Manolopoulos, Y., Pokorný, J., and Sellis, T. K., editors (2006). *Advances in Databases and Information Systems, 10th East European Conference, ADBIS 2006, Thessaloniki, Greece, September 3-7, 2006, Proceedings*, volume 4152 of *Lecture Notes in Computer Science*. Springer.

Mayr, H. C., Lazanský, J., Quirchmayr, G., and Vogel, P., editors (2001). *Database and Expert Systems Applications, 12th International Conference, DEXA 2001 Munich, Germany, September 3-5, 2001, Proceedings*, volume 2113 of *Lecture Notes in Computer Science*. Springer. 3540425276.

Mazón, J.-N., Trujillo, J., and Lechtenbörger, J. (2006b). A set of qvt relations to assure the correctness of data warehouses by using multidimensional normal forms. In Embley et al. [2006], pages 385–398.

Mazon, J.-N., Trujillo, J., Serrano, M., and Piattini, M. (2005). Designing data warehouses : from business requirement analysis to multidimensional modeling. In *13th IEEE International Requirements Engineering Conference Workshop on Requirements Engineering for Business Needs and IT Alignment (REBNITA*.

Meyer, B. (2000). *Conception et Programmation orientées objet*. Eyrolles . 2212091117.

Mohania, M. K. and Tjoa, A. M., editors (1999). *Data Warehousing and Knowledge Discovery, First International Conference, DaWaK '99, Florence, Italy, August 30 - September 1, 1999, Proceedings*, volume 1676 of *Lecture Notes in Computer Science*. Springer. 3540664580.

Moody, D. L. and Kortink, M. A. R. (2000). From enterprise models to dimensional models : a methodology for data warehouse and data mart design. In Jeusfeld et al. [2000], page 5.

Nascimento, M. A., Özsu, M. T., and Zaïane, O. R., editors (2002). *International Database Engineering & Applications Symposium, IDEAS'02, July 17-19, 2002, Edmonton, Canada, Proceedings*. IEEE Computer Society. 0769516386.

Nassis, V., Dillon, T. S., Rajagopalapillai, R., and Rahayu, J. W. (2006). An xml document warehouse model. In Lee et al. [2006], pages 513–529.

Paulk, M. C., Curtis, B., Averill, E., Bamberger, J., Kasse, T., Konrad, M., Perdue, J., Weber, C., and Withey, J. (1991). Capability maturity model for software. Technical Report CMU/SEI-91-TR-24 ADA240603.

Pedersen, T. B. and Jensen, C. S. (1999). Multidimensional data modeling for complex data. In DBL [1999b], pages 336–345.

Pedersen, T. B. and Jensen, C. S. (2005). Multidimensional databases. In Zurawski [2005], pages 1–13. 0849319854.

Phipps, C. and Davis, K. C. (2002). Automating data warehouse conceptual schema design and evaluation. In Lakshmanan [2002], pages 23–32.

Prakash, N. and Gosain, A. (2003). Requirements driven data warehouse development. In Eder and Welzer [2003]. 8643505498.

Prat, N. and Akoka, J. (2002). From uml to rolap multidimensional databases using a pivot model. In Pucheral [2002], page 24.

Pucheral, P., editor (2002). *18èmes Journées Bases de Données Avancées, BDA '02, 21-25 octobre 2002, Evry, Actes (Informal Proceedings)*.

Pujalte, V., Ramadour, P., and Cauvet, C. (2004). Recherche de composants réutilisables : une approche centrée sur l'assistance à l'utilisateur. In DBL [2004a], pages 211–227.

Rafanelli, M., editor (2003). *Multidimensional Databases : Problems and Solutions.* Idea Group. 1591400538.

Ralyté, J. (2001). *Ingénierie des méthodes à base de composants.* Thèse de doctorat, Université Paris 1, Sorbonne, Paris, France.

Ravat, F. and Teste, O. (2001). Modélisation et manipulation de données historisées et archivées dans un entrepôt orienté objet. In *Actes des 17ième Journées Bases de Données Avancées - BDA'01, Agadir (Maroc), 29/10/01-02/12/01*, pages 243–256. N. Mouaddib (ed.) Cépadues Editions.

Ravat, F., Teste, O., Tournier, R., and Zurfluh, G. (2007a). Integrating complex data into a data warehouse. In *SEKE International Conference on Software Engineering and Knowledge Engineering, Boston, 09/07/2007-11/07/2007*, http ://www.wspc.com.sg/. World Scientific Publishing.

Ravat, F., Teste, O., and Zurfluh, G. (2002). Langages pour bases multidimensionnelles : Olap-sql. *Revue des Sciences et Technologies de l'Information, série ISI-NIS (Ingénierie des Systèmes d'Information)*, 7(3) :11–38.

Ravat, F., Teste, O., and Zurfluh, G. (2006). Algèbre olap et langage graphique. In *INFORSID 2006, Hammamet (Tunisie), 31/05/06-03/06/06*, pages 1039–1054. INFORSID ed.

Ravat, F., Teste, O., and Zurfluh, G. (2007b). Algebraic and graphic languages for olap manipulations. In *International Journal of Data Warehousing and Mining, Idea Group.*

Rieu, D. (1999). *Ingénierie des systèmes d'information : bases de données, bases de connaissances et méthodes de conception.* Habilitation à diriger des recherches, Institut National Polytechnique de Grenoble, Grenoble, France.

Rizzi, S., Abelló, A., Lechtenbörger, J., and Trujillo, J. (2006). Research in data warehouse modeling and design : dead or alive ? In Song and Vassiliadis [2006], pages 3–10. 1595935304.

Rolland, C. (1993). Adapter les méthodes à l'objet : Challenges et embûches. In *Actes des journées de synthèse Méthodes d'analyse et de conception orientées objet des systèmes d'information, AFCET.*

Rolland, C. and Flory, A. (1990). La conception des systèmes d'information : etat de l'art et nouvelles perspectives. In *INFORSID*, pages 3–40.

Rolland, C. and Prakash, N. (2000). From conceptual modelling to requirements engineering. *Ann. Software Eng.*, 10 :151–176.

Roques, P. (2003). *UML in Practice - The Art of Modeling Software Systems Demonstrated Through Worked Examples and Solutions.* John Wiley & Sons, Inc., New York, NY, USA. 0470848316.

Roques, P. (2004). *UML 2 en action : De l'analyse des besoins à la conception J2EE*. Editions Eyrolles, Paris, France. 2212114621.

Sánchez, A., Cavero, J. M., de miguel, A., and Martínez, P. (1999). Idea : A conceptual multidimensional data model and some methodological implications. In *CIICC (1)*, pages 307–318, Cancun, México.

Sapia, C., Blaschka, M., Hofling, G., and Dinter, B. (1999). Extending the e/r model for the multidimensional paradigm. In *ER '98 : Proceedings of the Workshops on Data Warehousing and Data Mining*, pages 105–116, London, UK. Springer-Verlag. 3540656901.

Schiefer, J., List, B., and Bruckner, R. M. (2002). A holistic approach for managing requirements of data warehouse systems. In *In Proceedings of 8th Americas Conference on Information Systems, AMCIS, Dallas, TX, USA,*, pages 77–87.

Semmak, F. (1998). *Réutilisation de composants de domaine dans la conception des systèmes d'information*. Thèse de doctorat, Université Paris I, Paris, France.

Sen, A. and Sinha, A. P. (2005). A comparison of data warehousing methodologies. *Commun. ACM*, 48(3) :79–84.

Song, I.-Y. and Trujillo, J., editors (2005). *DOLAP 2005, ACM 8th International Workshop on Data Warehousing and OLAP, Bremen, Germany, November 4-5, 2005, Proceedings*. ACM. 1595931627.

Song, I.-Y. and Vassiliadis, P., editors (2006). *DOLAP 2006, ACM 9th International Workshop on Data Warehousing and OLAP, Arlington, Virginia, USA, November 10, 2006, Proceedings*. ACM. 1595935304.

Soussi, A., Feki, J., and Gargouri, F. (2005). Approche semi-automatisée de conception de schémas multidimensionnels valides. In *Revue des Nouvelles Technologies de l'Information - Entrepôts de Données et l'Analyse en ligne (EDA'05)*, volume RNTI-B-1, pages 71–89. Cépadues éditions.

Sowa, J. F. (1998). Conceptual graph standard and extension. In *ICCS '98 : Proceedings of the 6th International Conference on Conceptual Structures*, pages 3–14, London, UK. Springer-Verlag. 3540647910.

Srivastava, J. and Chen, P.-Y. (1999). Warehouse creation-a potential roadblock to data warehousing. *IEEE Transactions on Knowledge and Data Engineering*, 11(1) :118–126.

Tardieu, H., Rochfeld, A., and Colletti, R. (1983). *La méthode Merise : Principe et outils (Tome 1)*. Editions d'Organisation, France. 2708105515.

Teste, O. (2000). *Modélisation et Manipulation d'Entrepôts de Données Complexes et Historisées*. Thèse de doctorat, Université Paul Sabatier - Toulouse, Toulouse, France.

Theodoratos, D., editor (2002). *DOLAP 2002, ACM Fifth International Workshop on Data Warehousing and OLAP, November 8, 2002, McLean, VA, Proceedings*. ACM. 1581135904.

Theodoratos, D., Hammer, J., Jeusfeld, M. A., and Staudt, M., editors (2001). *Proceedings of the 3rd Intl. Workshop on Design and Management of Data Warehouses, DMDW'2001, Interlaken, Switzerland, June 4, 2001*, volume 39 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Torlone, R. (2003). Conceptual multidimensional models. In *Multidimensional Databases : Problems and Solutions*, pages 69–90.

Trujillo, J., Luján-Mora, S., and Medina, E. (2002). The gold model case tool : An environment for designing olap applications. In *ICEIS*, pages 699–707.

Trujillo, J. and Palomar, M. (1998). An object-oriented approach to multidimensional database conceptual modeling. In DBL [1998b], pages 16–21.

Tryfona, N., Busborg, F., and Christiansen, J. G. B. (1999). Starer : A conceptual model for data warehouse design. In DBL [1999a], pages 3–8.

Tsois, A., Karayannidis, N., and Sellis, T. K. (2001). Mac : Conceptual data modeling for olap. In Theodoratos et al. [2001], page 5.

Vassiliadis, P., Bouzeghoub, M., and Quix, C. (2000). Towards quality-oriented data warehouse usage and evolution. *Inf. Syst.*, 25(2) :89–115.

Vassiliadis, P. and Sellis, T. K. (1999). A survey of logical models for olap databases. *SIGMOD Record*, 28(4) :64–69.

Vassiliadis, P., Simitsis, A., Georgantas, P., Terrovitis, M., and Skiadopoulos, S. (2005). A generic and customizable framework for the design of etl scenarios. *Inf. Syst.*, 30(7) :492–525.

Vassiliadis, P., Simitsis, A., and Skiadopoulos, S. (2002a). Conceptual modeling for etl processes. In Theodoratos [2002], pages 14–21. 1581135904.

Winter, R. and Strauch, B. (2003). A method for demand-driven information requirements analysis in data warehousing projects. In *HICSS*, page 231.

Zurawski, R., editor (2005). *The Industrial Information Technology Handbook*. CRC Press. 0849319854.

Webographie

Crosby, P. (2006). Site de Philip Crosby. http://www.philipcrosby.com/pca/index.html.

I-D6 (2007). I-d6 société spécialisée dans le décisionnel et collaboratrice dans le cadre de la thèse cifre de numéro de convention 766/2003 préparée par estella annoni. http://www.i-d6.com.

IDC (2003). Le marché français de l'infogérance : quels investissements des entreprises en 2003 ? cabinet de statistiques. http://www.idc.com/france/about/cp_infogerance2003.jsp.

IDC (2004a). Cabinet de statistiques. le marché français du décisionnel. http://www.idc.com/france/research/decisionnel06.jsp.

IDC (2004c). Les clés d'une politique décisionnelle réussies. deuxième édition du décisionnel, idc, paris, 28 septembre 2004. http://www.artesi.artesi-idf.com/public/article.tpl?id=7717.

IDG (2005). Entreprises et décisionnel : Etat des lieux, objectifs et perspectives. http://www.decisio.info/Entreprises-et-decisionnel.html, Les résultats de l'enquête sont indiqués dans le document http://www.decisio.info/IMG/pdf/Enquete_CIO_SAS_230605.pdf.

Inmon, B. (1997). The data warehouse budget. http://www.dmreview.com/article_sub.cfm?articleId=1315, DM Review Magazine Publisher.

INSEE (2007). Institut national de la statistique et des etudes economiques. http://www.insee.fr/fr/nom_def_met/definitions/html/balance-paiements.htm.

Janiaux, P. (2006). Décisionnel et gestion de la performance : L'administration passe la vitesse supérieure. ZDnet http://www.zdnet.fr/entreprise/gestion-finances/decisionnel/0,50007174,39364358,00.htm.

Juran, J. (2006). Site de Joseph Juran. http://www.juran.com/.

Microsoft (2006b). Sql server integration services. http://www.microsoft.com/sql/technologies/integration/default.mspx.

OMG (2003). Cwm. http://www.omg.org/technology/documents/formal/cwm_2.htm.

Oracle (2006). Oracle warehouse builder user's guide 10g release 1. http://www.oracle.com/technology/products/warehouse/index.html.

Rosé, P. (2005). Outsourcing : Nouveaux modes d'infogerance. IDG http://www.cio-online.com/actualites/lire-quels-nouveaux-modes-d-infogerance-1.html, Les résultats de l'enquête sont indiqués dans le document http://www.idg.fr/wp/downloadfiles.asp?file=CIO_Infogerance_21sept2005.pdf.

Rouzeré, A.-M. (2007). Emploi : l'optimisme l'emporte pour trois informaticiens sur quatre. http://www.cio-online.com/actualites/lire-emploi-l-optimisme-l-emporte-pour-trois-informaticiens-sur-quatre-547.html.

Sunopsis (2006). Sunopsis etl for data warehousing. http://www.sunopsis.com/corporate/us/products/sunopsis/snps_dc.htm.

Zerbib, C. and Delsol, E. (2001). En 2001, le turnover reste vissé autour de 20 %, porté par les ingénieurs débutants. 01 Informatique (numero 1642), http://www.01net.com/article/155778.html.

Liste des tableaux

1.1 OLTP versus OLAP 7

4.1 Tableau croisé type 84
4.2 Tableau croisé de la valeur vénale des immobilisations par catégorie au cours des trois dernières années 84
4.3 Tableau croisé de la valeur vénale des immobilisations supérieure 1000 par catégorie au cours des trois dernières années 85
4.4 Propriétés « technique » du SID 87
4.5 Propriétés « décision » du SID 88
4.6 Signatures des opérations associées aux traitements du SID 93
4.7 Niveaux de consolidation et fonctions d'agrégation associées 94
4.8 Concept d'informativité et propriétés associés aux traitements du SID par groupe 95
4.9 Tableau croisé de la valeur vénale des biens du fabricant "BELL" et d'état "utilisé" pour les trois dernières années 100
4.10 Tableau de bord de la valeur vénale par catégorie et par état au cours l'année 2005 101
4.11 Tableau croisé synthétique des immobilisations sur le dernier trimestre 106
4.12 Matrice des faits candidats 112

5.1 Combinaisons valides d'architectures hétérogènes et complètes 134
5.2 Propriétés par critère de quantification 136
5.3 Opération et concept d'informativité par traitement 143
5.4 Table d'archivage du projet immobilisations 161
5.5 Traitements par classe multidimensionnelle 163

6.1 Patron « Tableau croisé » (TC) 211
6.2 Tableau croisé du volume turbiné par ouvrage physique au cours de quatre dernières semaines pour un exploitant 224
6.3 Tableau croisé des ouvrages physiques au cours du dernier mois . . . 224
6.4 Tableau croisé du montant de la redevance par ouvrage fonctionnel hydroélectrique et par payeur pour le dernier mois 225
6.5 Tableau croisé synthétique de la RFM sur le premier trimestre 2007 . 231
6.6 Matrice des faits candidats du projet RFM 235
6.7 Propriétés par critère de quantification 239

Table des figures

1.1 Les briques de la pyramide du système de gestion 4
1.2 L'environnement du SID 6
1.3 Nos publications 13

2.1 Une modélisation multidimensionnelle du domaine d'activités immobilisations 20
2.2 Un exemple de cube de données avec un fait et trois dimensions . . . 21
2.3 Les autres concepts multidimensionnels avec notre formalisme inspiré de [Golfarelli et al. 1998b] 22
2.4 Un schéma en constellation 24
2.5 Comparatif des modèles conceptuels multidimensionnels 33
2.6 Comparatif des méthodes de développement de SID 44
2.7 Comparatif des méthodes d'analyse de SID 53

3.1 Dichotomie des espaces de stockage 66
3.2 Le diagramme de classes UML des groupes d'acteurs décisionnels . . 70
3.3 Vue d'ensemble de la démarche du Trident décisionnel 72
3.4 Les trois principaux types itérations 74

4.1 Zoom sur la phase d'analyse de notre Trident décisionnel 78
4.2 Catégorisation des propriétés CSD des SID 81
4.3 Diagramme d'activités UML de la phase d'analyse du Trident décisionnel 82
4.4 Graphe général de propriétés 89
4.5 Diagramme décisionnel type 91
4.6 Graphe de propriétés CSD du groupe tactique du projet immobilisations 102
4.7 Diagramme décisionnel associé au tableau 4.2 103
4.8 Diagramme décisionnel associé au tableau 4.9 104
4.9 Diagramme décisionnel associé au tableau 4.10 104
4.10 Diagramme décisionnel du groupe tactique du projet immobilisations 105
4.11 Graphe de propriétés CSD du groupe stratégique du projet immobilisations 107
4.12 Diagramme décisionnel associé au tableau 4.11 109
4.13 Modèle entité-association de l'application des immobilisations 110
4.14 Graphe de propriétés CSD du groupe système du projet immobilisations 111
4.15 Diagramme décisionnel du groupe système du projet immobilisations 117
4.16 Diagramme décisionnel utilisateur du projet immobilisations 121
4.17 Diagramme décisionnel du SID du projet immobilisations 125

5.1 Zoom sur la phase de conception de notre Trident décisionnel 129
5.2 Les différents modules de l'architecture décisionnelle 133
5.3 Diagramme du choix de l'architecture à partir de NCD=Vertical et NEE={∅} 138
5.4 Diagramme du choix de l'architecture à partir de NCD=Tranversal et NEE={∅} 139
5.5 La dimension « Biens » avec ses deux hiérarchies 146
5.6 La dimension « Biens » avec ses deux hiérarchies, ces deux éléments terminaux dans un schéma en constellation et la dimension dégénérée No_contrat représentée explicitement 147
5.7 La représentation implicite de la dimension dégénérée « No_contrat » et la représentation du fait dégénéré « Commissions » 148
5.8 Modélisation conceptuelle du fait « Immobilisations » et des liens entre ses mesures 152
5.9 Hiérarchie des mesures de la comptabilité européenne (cf. INSEE [INSEE 2007]) 152
5.10 La représentation implicite du fait « Immobilisations » et des liens entre ses mesures 153
5.11 La représentation implicite du fait « Ventes » avec l'attribut mesure indépendant « CA » et l'attribut dimension dégénérée « No_contrat » 154
5.12 Modèle multidimensionnel de données et de traitements (MMDT) . . 157
5.13 Modèle multidimensionnel de données et de traitements du domaine Immobilisations avec les impacts de traitements 160
5.14 Diagramme décisionnel du SID du projet immobilisations (rappel) . . 167
5.15 Schéma conceptuel multidimensionnel du magasin de données du projet immobilisations 170
5.16 Schéma conceptuel entité-association de l'entrepôt de données du domaine des immobilisations 173

6.1 Diagramme de collaboration partiel de notre catalogue 181
6.2 Diagramme de classes UML du formalisme P-SIGMA 182
6.3 Formalisme P-SIGMA étendu 184
6.4 Diagramme de classes UML du formalisme P-SIGMA étendu 187
6.5 Patron « Mettre en place un SID » (MSID) 188
6.6 Diagramme de collaboration de patrons d'analyse 189
6.7 Patron « Analyser un SID » (ASID) 191
6.8 Patron « Caractériser las groupes d'acteurs » (CGA) 192
6.9 Patron « Définir le groupe tactique » (DGT) 193
6.10 Patron « Définir le groupe stratégique » (DGS) 194
6.11 Patron « Définir le groupe système » (DGSY) 195
6.12 Patron « Analyser les besoins tactiques » (ABT) 196
6.13 Patron « Collecter les besoins utilisateurs » (CBU) 197
6.14 Patron « Formaliser les besoins utilisateurs »- début (FBU) 198
6.15 Patron « Formaliser les besoins utilisateurs »- fin (FBU) 199
6.16 Patron « Interroger les acteurs du SID » (IAS) 200
6.17 Patron « Analyser les besoins stratégiques » (ABS) 201
6.18 Patron « Analyser les besoins systèmes » (ABSY) 202

6.19 Patron « Collecter les besoins systèmes » (CBSY) 203
6.20 Patron « Formaliser les besoins systèmes »- début (FBSY) 204
6.21 Patron « Formaliser les besoins systèmes »- suite (FBSY) 205
6.22 Patron « Formaliser les besoins systèmes »- fin (FBSY) 206
6.23 Patron « Evaluer la connaissance »- début (ELC) 207
6.24 Patron « Evaluer la connaissance »- fin (ELC) 208
6.25 Patron « Fiche de groupe » (FG) 209
6.26 Patron « Fiche signalétique » (FS) 210
6.27 Patron « Graphe de propriétés » (GP) 211
6.28 Patron « Diagramme décisionnel » (DD) 212
6.29 Diagramme de collaboration de patrons de conception 213
6.30 Patron « Concevoir un SID » (CSID) 214
6.31 Patron « Choisir l'architecture » (CAD) 215
6.32 Patron « Concevoir les modules décisionnels » (CMD) 216
6.33 Patron « Transformer le DD en MMDT »- début (TDM) 217
6.34 Patron « Transformer le DD en MMDT »- fin (TDM) 218
6.35 Patron « Transformer le DD en modèle E-A »- début (TDE) 219
6.36 Patron « Transformer le DD en modèle E-A »- fin (TDE) 220
6.37 Modèle multidimensionnel de données et de traitements généralisé MMDT 221
6.38 Graphe de propriétés CSD du groupe tactique du projet RFM 226
6.39 Diagramme décisionnel associé au tableau 6.2 227
6.40 Diagramme décisionnel associé au tableau 6.3 228
6.41 Diagramme décisionnel associé au tableau 6.4 229
6.42 Diagramme décisionnel du groupe tactique du projet RFM 230
6.43 Graphe de propriétés CSD du groupe stratégique du projet RFM . . 232
6.44 Diagramme décisionnel associé au tableau 6.5 233
6.45 Modèle entité-association de l'application RFM 234
6.46 Graphe de propriétés CSD du groupe système du projet RFM 235
6.47 Diagramme décisionnel du groupe système du projet RFM 236
6.48 Diagramme décisionnel utilisateur du projet RFM 237
6.49 Diagramme décisionnel du SID du projet RFM 238
6.50 Schéma conceptuel multidimensionnel du magasin de données du projet RFM 240
6.51 Légende du schéma conceptuel multidimensionnel du magasin de données du projet RFM 241

7.1 La démarche qualité Roue de Deming 249
7.2 Notre mise en oeuvre de la démarche qualité Roue de Deming 251
7.3 Diagramme statique de l'outil eBIPAD suivant [Roques 2004] 252
7.4 Fenêtre de connexion à l'outil eBIPAD 253
7.5 Diagramme de cas d'utilisation de l'outil eBIPAD 254
7.6 Diagramme de paquetages de l'outil eBIPAD 255
7.7 Interface d'accueil de l'outil eBIPAD d'un ingénieur de patrons 256
7.8 Diagramme de séquence système du scénario nominal d' " Ajouter un patron " 257
7.9 Fenêtre de saisie du nom du patron 259

7.10 Saisie du nom du nouveau champ 261
7.11 Définition du type du champ et de la valeur du champ 261
7.12 Affichage du patron après l'ajout de tous les champs 262
7.13 Interface d'accueil d'un ingénieur d'applications 263
7.14 Choix du nom patron 265
7.15 Représentation du patron avec le lien de l'adresse du diagramme UML actif 266
7.16 Sauvegarde de la solution-démarche 267
7.17 Présentation du résultat de la recherche du patron « Analyser un SID »268
7.18 Interface d'accueil d'un administrateur système 269
7.19 Représentation du patron « Analyser un SID » dans la base de eBI-PAD - début 271
7.20 Représentation du patron « Analyser un SID » dans la base de eBI-PAD - fin 272
7.21 Diagramme de classes UML de la base de patrons 273
7.22 Dictionnaire des données de la base 274

A.1 Diagramme d'activités du choix de l'architecture à partir de NCD=Vertical et NEE={$\emptyset$} 289
A.2 Diagramme d'activités du choix de l'architecture à partir de NCD=Tranversal et NEE={$\emptyset$} 290
A.3 Diagramme d'activités du choix de l'architecture à partir de NCD=Vertical et NEE={$PD^{1..N}$} 291
A.4 Diagramme d'activités du choix de l'architecture à partir de NCD=Tranversal et NEE={$PD^{1..N}$} 292
A.5 Diagramme d'activités du choix de l'architecture à partir de NCD=Vertical et NEE={$SCI^{1..N}$} 293
A.6 Diagramme d'activités du choix de l'architecture à partir de NCD=Tranversal et NEE={$SCI^{1..N}$} 294