

Modélisation adaptée aux besoins utilisateurs dans le développement des systèmes d'information décisionnels

Estella Annoni*, Franck Ravat*
Olivier Teste*, Gilles Zurfluh*

*Université Paul Sabatier
IRIT Equipe SIG-ED,
F-31062 Toulouse Cedex 9, France
{annoni, ravat, teste, zurfluh}@irit.fr

Résumé. La démocratisation des systèmes d'information décisionnels (SID) nécessite le développement de méthodes de conception. Contrairement aux modèles de systèmes d'information (SI) qui n'ont pas pour objet d'être compris par les utilisateurs, les modèles des SID doivent être exploitables par les analystes et les décideurs. Parmi les méthodes d'ingénierie des SID qui ont été proposées, rares sont celles qui explicitent la tâche d'analyse des besoins.

Pour ces raisons, nous proposons une démarche de collecte et de formalisation des besoins des utilisateurs du SID en utilisant des modèles proches de leur vision des données. A partir des besoins spécifiés sous forme de tableaux multidimensionnels, nous proposons des extensions de la modélisation objet afin de formaliser les besoins en terme de données et de traitements dans le contexte multidimensionnel.

1 Introduction

Les utilisateurs des systèmes d'information décisionnels (SID) expriment leurs besoins via des énoncés ou des résultats d'analyses tels que des tableaux à plusieurs dimensions Soussi et al. (2005) et non des scénarii. Les méthodes d'ingénierie des besoins qui utilisent généralement des scénarii s'avèrent inappropriées. De plus, ces utilisateurs ont une vision des données différente de celle des utilisateurs du SI et ils manipulent des données consolidées et non des données brutes. Les méthodes de SI ne sont donc pas adaptées Golfarelli et Rizzi (1999). La conception de méthodes permettant une analyse fiable et complète des besoins liés aux SID est donc cruciale¹ Bruckner et al. (1999).

Ainsi, la problématique étudiée dans cet article est de collecter et de formaliser les besoins des utilisateurs des SID avec des structures et des modèles adaptés aux caractéristiques de leurs besoins en vue d'une conception et d'un développement univoques. De plus, la démarche d'analyse des besoins doit intégrer les traitements comme le recommande Luján-Mora (2005). Force est de constater que rares sont les méthodes de SID qui utilisent des formalismes proches

¹ C'est le cas de la société I-D6 www.i-d6.com collaboratrice dans le cadre de la thèse CIFRE de numéro de convention 766/2003 préparée par Estella Annoni.

de la vision des utilisateurs des SID pour représenter leurs besoins d'information Kimball (1996). De plus, la modélisation des besoins qu'elles proposent n'intègre pas la dynamique des besoins, les traitements relatifs à l'information.

Dans cet article, nous proposons donc une démarche de collecte et de formalisation des besoins décisionnels. Le modèle sur lequel repose notre démarche est une extension du diagramme de classes UML au domaine du décisionnel permettant de caractériser les données et les traitements au plus tôt dans le processus d'ingénierie.

L'article est organisé suivant sept sections. Nous présentons l'état de l'art des méthodes d'analyses des besoins décisionnels dans la section 2. Le contexte de nos travaux est dressé dans la section 3. Puis, nous décrivons dans la section 4 notre démarche d'analyse des besoins décisionnels, plus précisément la tâche de collecte et la tâche de formalisation des besoins utilisateurs. Les règles de structuration des besoins (information et traitements) sont explicitées en section 5. Un extrait d'une application dans l'industrie est présenté en section 6. Nous concluons par une synthèse des caractéristiques de notre proposition et quelques perspectives de recherche.

2 Etat de l'art des méthodes d'analyse des besoins dans l'ingénierie des SID

Parmi les nombreuses propositions de méthodes d'ingénierie des SID, un nombre restreint traite des problèmes liés à l'analyse des besoins des utilisateurs. Ces méthodes peuvent être classifiées suivant trois approches : ascendante, descendante et mixte.

L'approche *ascendante*, orientée par les sources de données, définit le système à partir du schéma des sources de données Golfarelli et Rizzi (1998), Cabibbo et Torlone (1998) et Moody et Kortink (2000). Comme son nom l'indique, elle repose uniquement sur les sources. Les méthodes basées sur cette approche ne soulèvent donc pas les problèmes liés aux besoins.

La seconde approche *descendante*, orientée utilisateurs, définit le système à partir des besoins utilisateurs. La méthode proposée par Kimball (1996) définit le système à partir de l'expression informelle des besoins des utilisateurs. L'auteur n'utilise pas de formalisme particulier. De plus, il propose de nombreux exemples sans donner une démarche formelle de la phase d'analyse. Les auteurs Akoka et al. (2001) élaborent des diagrammes de classes UML à partir de l'expression informelle des besoins par les utilisateurs. L'inconvénient de cette méthode est qu'elle utilise, comme la précédente, l'expression informelle des besoins. Cet inconvénient couplé à la méconnaissance du domaine par le concepteur décisionnel implique une spécification souvent trop technique Bruckner et al. (1999). Ces méthodes n'utilisent donc pas un formalisme pour la représentation des besoins.

L'objet d'un SID étant de fournir l'information aux utilisateurs en vue d'une prise de décision ; les deux approches précédentes ne traitent pas des deux aspects du SID. Pour remédier à cette insuffisance, l'approche *mixte* a été proposée. Elle a fait l'objet de nombreux travaux. Mais, nous nous intéressons à ceux qui définissent une démarche d'analyse des besoins utilisateurs.

Les auteurs Bonifati et al. (2001) collectent les besoins des utilisateurs à partir d'interviews afin de déterminer les buts et les objectifs de l'entreprise en utilisant le paradigme GQM (Goal/Question/Metrics). Cette méthode utilise un formalisme mais la définition et l'analyse

des buts sont des tâches informelles comme le précisent les auteurs. De plus, elle ne modélise pas les traitements relatifs aux besoins.

Les méthodes mixtes proposées par Ghozzi et al. (2005) et Soussi et al. (2005) utilisent des structures proches de la vision des données par les utilisateurs. La première proposition utilise un pseudo-langage, mais celui-ci ne permet pas d'exprimer toutes les spécificités des besoins utilisateurs tels que le type des données et les contraintes sur la structure du SID.

La seconde proposition repose sur une expression tabulaire complète mais complexe. La formalisation des besoins suggérée ne tient compte que du niveau statique des besoins. Tous les traitements liés à l'information que souhaitent manipuler les utilisateurs ne sont pas collectés. Ainsi, nous nous basons sur les apports de deux ces dernières propositions et sur la modélisation objet afin de proposer une démarche formelle d'analyse des besoins complète et adaptée aux SID intégrant le niveau statique et le niveau dynamique du SID.

3 Contexte de nos travaux

Nous classons les besoins exprimés par les différents intervenants dans 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, soient, 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'entreprise.

Nous constatons un nombre important d'intervenants et de types d'intervenants sur les projets décisionnels. Comme Bruckner et al. (1999), nous classons les acteurs du SID en trois groupes :

- *le groupe Utilisateurs* exprime les besoins analytiques, fonctionnels (règles de gestion, règles de calcul) ainsi que non fonctionnels (ergonomie d'outil),
- *le groupe Equipement* 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,
- *le groupe Pilotage* exprime les besoins analytiques, fonctionnels, non fonctionnels mais aussi des besoins liés à la stratégie de l'entreprise.

La prise en compte des besoins de ces trois types d'acteurs est un élément capital pour le succès de l'intégration du SID dans l'organisme. La confrontation et le rapprochement de ces différents besoins sont indispensables pour une analyse complète, rigoureuse et satisfaisante des SID. C'est la raison pour laquelle nous confrontons les besoins des trois types d'acteurs dès la phase d'analyse Annoni et al. (2005).

Par ailleurs, et contrairement aux SI qui utilisent des modèles de données non familiers à l'utilisateur du SID Kimball (1996), les SID reposent sur des modèles qui doivent être compris par les utilisateurs. Les utilisateurs des SID sont des décideurs et des analystes ; ils ont une vision multidimensionnelle de l'information. La représentation multidimensionnelle consiste à

Immobilisation. Valeur vénale		Temps. Année		
		2003	2004	2005
Catégories. Famille	Catégories. Sous-famille			
Immatériel	Logiciels	115 999,26	69 059,42	173 170,09
	Progiciels	8 958,61	111 429,62	63 769,32
Postes utilisateurs	Ecrans	1 147,43	3 445,14	4 624,12
	PC	3 601,85	5 319,81	7 420,95
	Terminaux	1 985,47	1 798,06	1 291,34

TAB. 1 – Evolution de la valeur vénale de l'immobilisation par catégorie au cours des trois dernières années.

représenter les sujets d'analyse comme des points dans un repère à plusieurs axes d'analyse. Autrement dit, les faits sont étudiés suivant les dimensions du repère de l'analyse. Les faits sont caractérisés par les mesures et les dimensions sont décrites par des paramètres. Les modèles pour la représentation de ces besoins d'information² doivent donc avoir une structure proche des modèles de données multidimensionnelles Golfarelli et Rizzi (1998).

Afin de ne pas manipuler des expressions informelles des besoins des utilisateurs, nous préconisons que les concepteurs décisionnels exploitent des exemplaires des tableaux utilisés par les acteurs du SID tel que le tableau 1. Un intérêt de ce tableau est la commodité de l'expression des restrictions 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 et les analystes qui sont des acteurs non informaticiens.

Ce tableau explicite une autre spécificité des acteurs des SID qui est l'expression des besoins sous forme d'analyses, de tableaux multidimensionnels et non de scénarii. Les méthodes d'analyse des SI et d'ingénierie des besoins sont principalement basées sur des scénarii, des cas d'utilisations UML Jacobson (1992). Elles s'avèrent donc inadaptées dans le domaine de l'ingénierie des SID.

Forts de ce constat, nous nous sommes inspirés de structures permettant de spécifier la propriété multidimensionnelle des données pour la représentation des besoins. De même, pour la formalisation de ces besoins, nous utilisons des extensions du diagramme de classes UML. Ces extensions répondent aux besoins de structuration des données suivant le modèle en étoile et plus globalement, le modèle en constellation Kimball (1996). La modélisation objet, nous permet d'une part, de traiter séparément l'information et les traitements et d'autre part, de favoriser la réutilisation au sein de notre méthode.

4 Analyse des besoins utilisateurs

Notre méthode de développement des SID a été validée dans le cadre de la thèse CIFRE présentée dans la note 1. Elle suggère l'analyse en parallèle des besoins des trois groupes d'acteurs (cf. section 3) suivie d'une confrontation de ces derniers afin de clore ou de réitérer

²Nous distinguons les termes données et information. Les données représentent les éléments pertinents extraits des sources alors que l'information est le résultat de la transformation des données. C'est l'élément qui sera stocké dans le SID et restitué aux utilisateurs. Les besoins des utilisateurs sont exprimés en terme d'information et non de données.

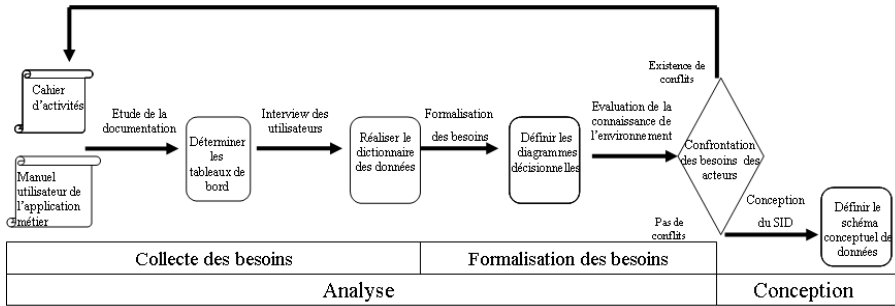


FIG. 1 – Processus de collecte et de formalisation des besoins utilisateurs.

les tâches liées à l'analyse Annoni et al. (2005). Cette confrontation se réalise dans un cadre fixé lors de l'analyse des besoins du groupe Pilotage. Cependant, pour des raisons de place, nous ne présentons pas l'analyse des besoins du groupe Pilotage ainsi que de ceux du groupe Equipement. L'analyse des besoins utilisateurs se déroule au cours d'un cycle de vie itératif incrémental avec un nombre d'itérations variable suivant les résultats de la phase de confrontation Annoni et al. (2005).

Nous nous focalisons donc sur les tâches liées à l'analyse des besoins du groupe Utilisateurs. La figure 1 présente notre démarche d'analyse des besoins utilisateurs. Notre démarche se décompose en deux tâches séquentielles : la 'Collecte des besoins utilisateurs' et la 'Formalisation des besoins utilisateurs'.

4.1 Collecte des besoins utilisateurs

Au début de la collecte, il convient de définir le périmètre du projet et en l'occurrence les domaines concernés. Nous définissons un domaine comme un secteur d'activité, une classe d'utilisateurs d'une entreprise. Par exemple, le domaine des immobilisations regroupe des utilisateurs qui ont en commun la gestion des biens de l'entreprise. Ces utilisateurs sont pour certains liés au métier des achats et pour d'autres au métier des frais généraux. Ainsi, dans le cas où le projet recouvre plusieurs domaines, la démarche est répétée pour chaque domaine.

La tâche de collecte se poursuit par l'étude des documents des activités des utilisateurs tels que les cahiers d'activités du domaine et les manuels utilisateurs des applications métiers utilisés à la société I-D6. Ces documents contiennent les tableaux multidimensionnels qui sont utilisés pour l'analyse et l'évaluation des indicateurs de leurs activités. A partir de cette documentation, un ensemble de tableaux multidimensionnels est recueilli. Dans le cas où les tableaux multidimensionnels ne sont pas directement disponibles, les utilisateurs spécifient leurs besoins avec des requêtes-types définies suivant notre pseudo-langage *BIQUERY*. A l'aide de la grammaire du pseudo-langage, les faits, les dimensions, les mesures et les paramètres sont automatiquement identifiés. Le langage définit l'ordre suivant :

ANALYSER $\{S.C_r\}^+$	– informations sur le(s) fait(s) de l'analyse,
QUAND $\{Cond(S.C_r_s)\}^+$	– informations sur la(les) mesure(s) du (des) fait(s),
EN FONCTION $\{A_i.E_{k_{A_i}}\}^+$	– informations sur la(les) dimension(s) de l'analyse,

S.C _r	A _j .E _l		
	Val1 _{E_lA_j}	...	Valm _{E_lA_j}
A _i .E _k			
Val1 _{E_kA_i}	{ValC _{rS} (Val1 _{E_kA_i} , Val1 _{E_lA_j})}	...	{ValC _{rS} (Val1 _{E_kA_i} , Valm _{E_lA_j})}
Val2 _{E_kA_i}	{ValC _{rS} (Val2 _{E_kA_i} , Val1 _{E_lA_j})}
Valn _{E_kA_i}	{ValC _{rS} (Valn _{E_kA_i} , Val1 _{E_lA_j})}	...	{ValC _{rS} (Valn _{E_kA_i} , Valm _{E_lA_j})}

TAB. 2 – Tableau multidimensionnel type.

POUR {Cond(A_i.E_{kA_i})}⁺ – informations sur le(s) paramètre(s) de(s) dimension(s).

Le passage d'une requête-type à un tableau multidimensionnel est simple avec des données fournies par les utilisateurs. Nous considérons la forme tabulaire simple et générale où les besoins sont analysés suivant 1 à 2 dimension(s) (cf. tableau 2) ; les autres dimensions sont fixées, pour un paramètre donné, à une valeur donnée. La forme tabulaire associée à la requête-type ci-dessus est représentée par le tableau 2.

Explications du tableau multidimensionnel type où les variables i, x, r, n, p, q sont des entiers :

- A_i est la i-ième dimension (1 ≤ i ≤ 2). Elle est composée de paramètres E_{kA_i} (0 ≤ k ≤ n). Le k-ième paramètre a plusieurs valeurs : Valx_{E_kA_i} (1 ≤ x ≤ p). L'explication est la même pour A_j,
- S est le fait présenté dans ce tableau multidimensionnel. Il est composé de mesures C_{rS} (1 ≤ r ≤ q). La mesure a plusieurs valeurs ValC_{rS}(Valx_{E_kA_i}, Valw_{E_lA_j}) pour la x-ième valeur de E_{kA_i} et la w-ième valeur de E_{lA_j}.

Par exemple, l'expression en langage naturel de l'exigence analytique³ 'Analyser l'évolution de la valeur vénale des immobilisations par famille et sous-famille des catégories pour les trois dernières années' se traduit sous forme de requête-type comme suit :

ANALYSER Immobilisations.Valeur vénale – Fait Immobilisations avec sa mesure
 EN FONCTION Catégories.Famille – Dimension Catégories avec ses paramètres
 , Catégories.Sous-famille – Dimension Catégories avec ses paramètres
 , Temps.Année – Dimension Temps avec son paramètre Année
 POUR Temps.Année DANS (2003,2004,2005) – Restriction sur le paramètre Année

Au cours de ces précédentes tâches, le concepteur décisionnel a collecté explicitement l'information que souhaitent manipuler les utilisateurs : le niveau statique des besoins. Il convient maintenant de collecter les traitements liés l'information, l'aspect dynamique des besoins. Pour cela, nous interviewons les utilisateurs sur la base d'un questionnaire admis au sein de la société I-D6. Nous avons fait le choix de ne pas utiliser de langage de spécification tel que le langage Z et ses dérivés car 2/3 des groupes d'acteurs du projet SID sont des non-informaticiens et ils participent à la validation de la spécification des besoins.

Au delà des questions relatives aux exigences analytiques du futur SID, ce questionnaire permet de caractériser les besoins fonctionnels. Par exemple, il permet de caractériser la fré-

³Par analogie aux exigences fonctionnelles qui doivent être garanties par les systèmes d'information, nous appelons exigences analytiques, les analyses qui doivent être possibles via le SID.

quence d'édition des tableaux multidimensionnels, les informations sur les différents historiques à conserver ou encore les informations sur les archives. Ce questionnaire aborde aussi les traitements liés aux besoins de types analytiques et fonctionnels tels que le calcul, l'historisation, le rafraîchissement, l'archivage.

A partir des réponses à ces questions, le concepteur réalise un dictionnaire dont la structure est inspirée du classique dictionnaire des données Merise Tardieu et al. (2000). En plus des colonnes standards (Mnémonique, Description, Type, Contraintes, Règles de calcul), nous ajoutons les colonnes liées aux traitements de rafraîchissement, d'historisation et d'archivage. Ces nouvelles colonnes sont désignées par les noms des traitements associés. Ce dictionnaire ne contient pas les données mais, l'information que les utilisateurs souhaitent analyser ainsi que les traitements. Nous le désignons par le terme *Dictionnaire Décisionnel*.

Le dictionnaire décisionnel a pour intérêt la spécification de l'information et des principaux traitements associés, mais il n'a pas pour rôle d'être compris par les utilisateurs. Il importe donc de formaliser les besoins utilisateurs en termes d'information et de traitements avec un modèle proche de la vision multidimensionnelle des utilisateurs.

4.2 Formalisation des besoins utilisateurs

Au regard des caractéristiques des besoins utilisateurs, nous avons pour objectif de proposer une formalisation qui a les propriétés suivantes :

- elle intègre les spécificités des besoins analytiques,
- elle permet d'exprimer les traitements liés aux besoins fonctionnels (alimentation, stockage et présentation du SID),
- elle est proche de la vision multidimensionnelle des utilisateurs en s'inspirant du schéma en étoile et en constellation,
- elle sépare l'information des traitements de par la modélisation objet,
- elle supporte l'évolutivité des besoins en matière de traitements sur l'information.

Notre formalisme, présenté à la figure 2, est une extension du diagramme de classe UML au domaine multidimensionnel. Il permet de représenter l'information ainsi que les traitements associés au niveau des faits, des dimensions, des mesures et des paramètres avec les concepts de classes et d'attributs. Nous proposons le concept d'informativité afin d'exprimer qu'une information fait l'objet d'un traitement ETL. Ce concept englobe toutes les propriétés spécifiques de l'information que les utilisateurs souhaitent manipuler. Les symboles des propriétés d'informativité sont :

- c : l'information est calculée,
- h : l'information doit être historisée,
- * : l'information doit être rafraîchie,
- a : l'information doit être archivée.

Nous définissons les signatures des méthodes associées aux traitements de calcul, de rafraîchissement, d'historisation et d'archivage comme suit :

- Historiser(p, c, cond) : spécifie le processus d'historisation appliqué au concept décisionnel de période de l'historisation p, de contrainte d'historisation c et de condition de l'historisation cond,

- Archiver(p, c, cond, fct) : spécifie le processus d’archivage appliqué au concept décisionnel de période de l’archivage p, contrainte d’archivage c, condition de l’archivage cond et à la fonction d’agrégation fct,
- Rafraîchir(cond, m) : spécifie le processus de rafraîchissement appliqué au concept décisionnel de condition de rafraîchissement cond et mode de rafraîchissement m,
- Calculer($\{v_i\}^+$) : spécifie le calcul de l’information à partir des paramètres v_i .

Avec ce concept, il est possible d’exprimer le processus d’historisation qui peut concerner les différents éléments de l’analyse (classe, attribut) Teste (2000). Pour cela, il importe de définir un comportement relatif aux attributs des classes. Cependant, UML ne permet pas de définir des méthodes au niveau de l’attribut d’une classe. Face à un problème de création de relations entre attributs, Luján-Mora et al. (2004) considèrent les attributs comme un deuxième type de classes. Cette solution n’est pas adaptable à notre problème dans le sens où comme le précisent les auteurs, il n’est pas possible de définir des attributs au niveau des attributs. De plus, la définition des méthodes au niveau des attributs a un sens dans le cas où l’on considère la classe à laquelle l’attribut appartient. En effet, les mesures et les paramètres de l’analyse n’ont pas d’existence propre ; ils ont un sens uniquement en tant que composant d’un fait ou d’une dimension.

Ainsi, dans l’optique du schéma en étoile, du schéma en constellation intégrant les traitements relatifs à l’information Kimball (1996), nous proposons d’étendre le diagramme de classes UML par la spécification des propriétés d’informativité au niveau des attributs, à côté des symboles de visibilité des informations. Nous associons le stéréotype <<attribut>> à la méthode définie au niveau de l’attribut. Dans la signature de la méthode, nous ajoutons l’attribut comme premier paramètre. Ainsi, il est possible de définir et de distinguer les méthodes de classe et les méthodes d’attribut. Le modèle résultant est appelé *Diagramme Décisionnel (DD)* présenté à la figure 2. Il est composé de deux ensembles de classes : classe-fait et classe-dimension :

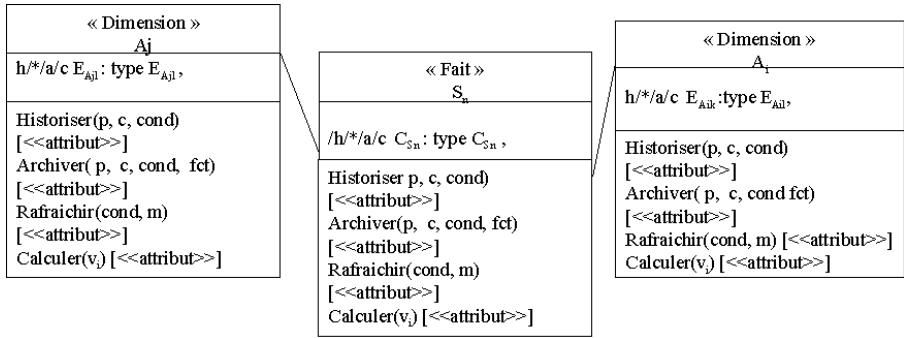
- Classe-fait S : la classe associée au fait traité dans le tableau multidimensionnel. Elle est composée d’attributs qui correspondent aux mesures. La r-ième mesure C_{r_S} est de type $\text{Type}C_{r_S}$,
- Classe-dimension A_i : la classe associée à la i-ième dimension suivant laquelle le fait S est analysé. Elle est composée d’attributs qui correspondent aux paramètres. Le k-ième paramètre est $E_{k_{A_i}}$ de type $\text{Type}E_{k_{A_i}}$.

Le lien entre une classe-fait et une classe-dimension exprime la relation *ÊTRE FONCTIONNÉ*. Le fait peut être analysé uniquement suivant les dimensions qui lui sont liées. La connaissance de toutes les dimensions permet de déterminer le fait.

Pour passer d’un tableau obtenu lors de la collecte des besoins au DD, il faut d’appliquer des règles de structuration en tenant compte du dictionnaire décisionnel et du domaine analysé.

5 Règles de structuration des besoins utilisateurs

Afin de formaliser le besoin exprimé par le tableau 2, le concepteur décisionnel applique les règles de structuration de l’information. Les règles de structuration se composent des règles de transformation des tableaux multidimensionnels type en DD, des règles syntaxiques des DD et des règles de fusion de ces différents DD. Le concepteur applique d’abord les règles de

FIG. 2 – *Diagramme décisionnel.*

transformation des tableaux multidimensionnels en DD, puis les règles syntaxiques et enfin les règles de fusion.

Plus précisément, pour un même domaine, il applique les règles de transformation liées à l'environnement. Puis, pour chaque tableau-type, il applique les règles de transformation relatives à l'information, puis celles relatives aux traitements. Le concepteur continue la conception de chaque DD par l'application des règles syntaxiques. Enfin, pour un même domaine, il applique les règles de fusion.

- Règles de transformation de l'environnement du projet Ex
 - Information EI :
 - EI1 : un diagramme décisionnel est associé à tout tableau-type ,
 - EI2 : tout tableau multidimensionnel spécifiant les besoins doit avoir une structure proche de celle du tableau-type (tableau 2),
 - Règles de transformation du fait S_x :
 - Information SI :
 - SI1 : une classe-fait avec le stéréotype <<fait>> est associée à tout fait,
 - SI2 : un attribut est associé à toute mesure,
 - SI3 : les propriétés d'informativité $\acute{n}c\acute{z}$, $\acute{n}h\acute{z}$, $\acute{n}*\acute{z}$, $\acute{n}a\acute{z}$ sont associées à toute mesure dont respectivement la colonne $\acute{n}R\grave{e}gle\ de\ calcul\acute{z}$, $\acute{n}Historisation\acute{z}$, $\acute{n}Rafra\acute{c}hissement\acute{z}$, $\acute{n}Archiver\acute{z}$ du dictionnaire décisionnel est renseignée,
 - Traitements SP :
 - SP1 : une méthode est associée à toute propriété d'informativité,
 - SP2 : une méthode d'attribut est associée à toute mesure qui possède des arguments différents pour un traitement donné.
 - Règles de transformation des dimensions A_i :
 - Information AI :
 - AI1 : une classe-dimension avec le stéréotype <<dimension>> est associée à toute dimension,
 - AI2 : un attribut $\acute{n}Id\acute{z}$ est associé à chaque classe-dimension,
 - AI3 : un attribut est associé à tout paramètre,
 - AI4 : les propriétés d'informativité $\acute{n}c\acute{z}$, $\acute{n}h\acute{z}$, $\acute{n}*\acute{z}$, $\acute{n}a\acute{z}$ sont associées à tout para-

mètre dont respectivement la colonne *Règle de calcul*, *Historisation*, *Rafraîchissement*, *Archivage* du dictionnaire décisionnel est renseignée,

- Processus AP :
 - AP1 : une méthode est associée à toute propriété d’informativité,
 - AP2 : une méthode d’attribut est associée à tout paramètre qui possède des arguments différents pour un traitement donné.

Les règles syntaxiques permettent de vérifier la cohérence des DD.

- 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 la propriété d’informativité d’historisation sur l’exercice précédent pour l’analyse des tendances,
 - SDI4 : à tout paramètre est associée la propriété d’informativité d’historisation sur l’exercice précédent pour l’analyse des tendances,
 - SDI5 : si un paramètre Id possède la propriété d’informativité *n*z* alors le fait lié possède la propriété d’informativité et la méthode associée est la même,
- Processus SDP :
 - SDP1 : si la propriété d’informativité porte sur tous les attributs de la classe et avec les mêmes paramètres alors la méthode liée est spécifiée au niveau de la classe,
 - SDP2 : si une des mesures du fait de l’analyse possède les propriétés d’informativité liée à l’historisation *nhz* ou à l’archivage *naz* alors toutes les dimensions liées doivent posséder aussi cette propriété. De plus, la période et la condition de la méthode associée doivent être au moins égales à celles de la mesure.

A partir de ces DD, le concepteur décisionnel applique les règles de fusion afin d’obtenir un DD en étoile ou en constellation. Le type de DD (étoile ou constellation) varie suivant les possibilités de mise en commun des dimensions et des faits, autrement dit suivant la logique de l’entreprise. Pour la définition des règles de fusion, nous prenons comme hypothèse que la sémantique est fiable.

- FUS : Regrouper les DD ayant la même classe-fait et des classes-dimensions en commun,
 - FUS1 : Fusionner les classes-dimensions par ajout des attributs et des méthodes,
 - FUS2 : Fusionner les classes-faits par ajout des attributs et des méthodes,
- FDS : Regrouper les DD ayant des classes-faits différents et des classes-dimensions en commun,
 - FDS1 : Fusionner les classes-dimensions par ajout des attributs et des méthodes.

Ainsi les besoins des utilisateurs sont formalisés avec un modèle permettant d’exprimer la structure et la dynamique du SID et sous un format dérivé de la vision des données par les décideurs et des analystes. Afin de justifier les propriétés de notre démarche, nous présentons dans la section suivante un extrait d’une mise en oeuvre au sein d’un projet industriel.

6 Application

Considérons l’exemple d’une société qui souhaite mettre en place un SID pour générer automatiquement les tableaux multidimensionnels de l’activité des immobilisations de biens. Le projet couvre un seul domaine, celui des immobilisations. Après l’étude de leur cahier

	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
Total dépréciation	9 757 215,25	9 895 456,38	10 823 340,82

TAB. 3 – Evolution des indicateurs de performances clés sur le dernier trimestre.

d'activités et du manuel utilisateur de leur application transactionnelle de gestion des biens, les tableaux multidimensionnels utilisés sont les tableaux 1, 3.

Faute de place, nous n'avons pas présenté le tableau de l'évolution de la valeur vénale par biens au cours des années 2004 et 2005 que nous considérons dans notre exemple. Ce tableau permet de définir le fait \acute{n} Immobilisations \acute{z} avec sa mesure \acute{n} Valeur_vénale \acute{z} et les dimensions \acute{n} Temps \acute{z} et \acute{n} Biens \acute{z} . Ainsi, suite à l'analyse des tableaux multidimensionnels et à l'interview des utilisateurs, nous obtenons le dictionnaire décisionnel représenté à la figure 3. L'application

Mnémonique	Description	Type	Contraintes	Règles de calcul	Historisation	Rafraîchissement	Archivage
No_imm	Numéro du bien	Entier	>0		Y>y-3 and Y<y		année, Y>y-5 and Y<y-1 sum
V_assurance	Valeur de l'assurance du bien	Entier			Y>y-3 and Y<y		année, Y>y-5 and Y<y-1 sum
Modele	Modèle du bien	Chaîne			Y>y-3 and Y<y		année, Y>y-5 and Y<y-1 sum
Saison	Indicateur saisonnier du bien	Booléen			Y>y-3 and Y<y	y<>Y merge	année, Y>y-5 and Y<y-1 sum
Valeur_vénale	Valeur vénale d'un bien	Réel	>0	valeur_achat - amortissement	Y>y-3 and Y<y		année, Y>y-5 and Y<y-1 sum
Total_dépréciation	Total dépréciation d'un bien	Réel			Y>y-3 and Y<y		année, Y>y-5 and Y<y-1 sum
Amortissement	Amortissement d'un bien	Réel			Y>y-3 and Y<y		année, Y>y-5 and Y<y-1 sum
Sous_famille	Sous famille de la catégorie	Chaîne	<<Logiciels, Progiciels, Ecrans, PC, Terminaux>>		Y>y-3 and Y<y	y<>Y merge	année, Y>y-5 and Y<y-1 sum
Famille	Famille de la catégorie	Chaîne	<<Immateriel, Poste_utilisateurs>>		Y>y-3 and Y<y	y<>Y merge	année, Y>y-5 and Y<y-1 sum

FIG. 3 – Dictionnaire décisionnel du projet.

des règles pour le tableau 1 donne :

- Règle E11 : il y a 3 expressions de besoins donc 3 diagrammes décisionnels seront créés,

Modélisation adaptée aux besoins décisionnels

- Règle EI2 : le tableau multidimensionnel des KPIs est restructuré en un tableau à une dimension avec 3 mesures qui sont : \acute{n} Valeur_vénale \acute{z} , \acute{n} Amortissement \acute{z} , \acute{n} Total_dépréciation \acute{z} ,
- Règle SII : la classe-fait \acute{n} Immobilisations \acute{z} est associée au fait \acute{n} Immobilisations \acute{z} ,
- Règle SI2 : l'attribut \acute{n} Valeur_vénale \acute{z} est associé à la mesure \acute{n} Valeur_vénale \acute{z} ,
- Règle SI3 : les propriétés d'informativité \acute{n} cz, \acute{n} hz, \acute{n} az sont associées à la mesure \acute{n} Valeur_vénale \acute{z} car elle est calculée, historisée, archivée,
- Règle SP1 : toutes les méthodes sont au niveau de la classe car elles ne sont pas spécifiques à la seule mesure du fait :
 - Historiser(annee, $Y > y - 3$ and $Y < y$) : historisation des deux années précédant l'année courante avec y : période courante et Y : période historisée,
 - Archiver(annee, NULL, $Y > y + 5$ and $Y < y - 1$, sum) signifie que la valeur vénale doit être archivée pour la troisième période et la quatrième période antérieures, en faisant la somme,
- Règle SP2 : la méthode \acute{n} Calculer(Valeur_vénale, Valeur_achat, Total_dépréciation) {attribut} \acute{z} est déclarée au niveau de l'attribut car elle indique que l'attribut est calculée à partir de la \acute{n} Valeur_achat \acute{z} et du \acute{n} Total_dépréciation \acute{z} ,
- Règle AI1 : les classes-dimensions \acute{n} Catégories \acute{z} et \acute{n} Temps \acute{z} sont associées aux dimensions \acute{n} Catégories \acute{z} et \acute{n} Temps \acute{z} ,
- Règle AI2 : l'attribut \acute{n} Id \acute{z} est ajouté à chaque classe-dimension \acute{n} Temps \acute{z} et \acute{n} Catégories \acute{z} ,
- Règle AI3 : les attributs de la classe-dimension \acute{n} Temps \acute{z} sont \acute{n} Jour \acute{z} , \acute{n} Mois \acute{z} , \acute{n} Trimestre \acute{z} , \acute{n} Semestre \acute{z} et \acute{n} Année \acute{z} . Les attributs de la classe-dimension \acute{n} Catégories \acute{z} sont \acute{n} Famille \acute{z} et \acute{n} Sous-famille \acute{z} ,
- Règle AI4 : tous les attributs des classes-dimensions \acute{n} Temps \acute{z} et \acute{n} Catégories \acute{z} doivent être historisés, rafraîchis et archivés,
- Règle AP1 : Les méthodes \acute{n} Historiser \acute{z} et \acute{n} Archiver \acute{z} sont définies au niveau des classes-dimensions car il n'y a pas de spécificités pour un paramètre donné. Les paramètres \acute{n} Sous-famille \acute{z} et \acute{n} Famille \acute{z} de la dimension \acute{n} Catégories \acute{z} doivent être rafraîchis chaque année. La méthode \acute{n} Rafraîchir($y < > Y$, merge) signifie que la classe-dimension \acute{n} Catégories \acute{z} doit être rafraîchie chaque année avec y : année courante et Y : année de rafraîchissement. Les explications pour les méthodes \acute{n} Historiser \acute{z} et \acute{n} Archiver \acute{z} sont similaires à celles des classes-faits.

On obtient alors le diagramme décisionnel associé au tableau 1.

En appliquant les règles de transformation au tableau multidimensionnel 3 comme ci-dessus, on obtient le diagramme décisionnel représenté à la figure 5.

Lors de la vérification des règles syntaxiques à la fin de la conception des DD, il n'y a pas d'incohérence détectée. Le résultat de l'application des règles de transformation et des règles syntaxiques est un ensemble de 3 DD. Il convient de les fusionner à l'aide des règles associées. Tous les DD ont la même classe-fait : \acute{n} Immobilisations \acute{z} . On applique donc uniquement les règles FUS :

- Règle 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 d'un DD à l'autre,
- Règle FUS2 : La fusion des classes-faits par ajout des attributs modifie les attributs du fait \acute{n} Immobilisations \acute{z} . La classe-fait \acute{n} Immobilisations \acute{z} du DD présenté à la figure 5

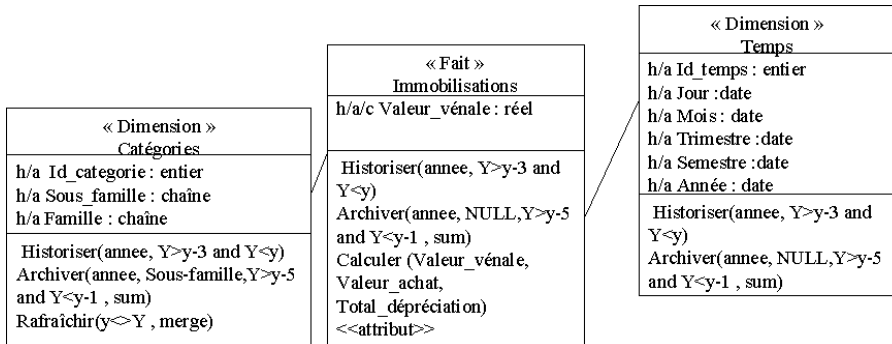


FIG. 4 – Diagramme décisionnel du tableau 1.

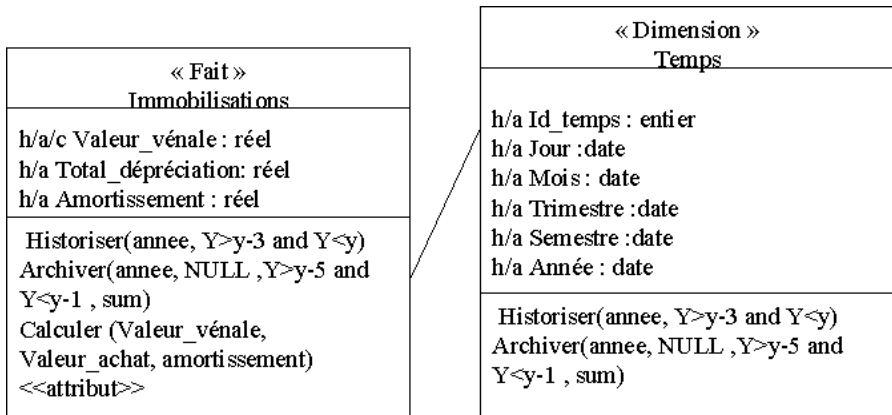


FIG. 5 – Diagramme décisionnel du tableau 3.

contient trois mesures (\hat{n} Valeur_vénale \hat{z} , \hat{n} Amortissement \hat{z} , \hat{n} Total_dépréciation \hat{z}) alors que celle présentée à la figure 4 contient uniquement la mesure \hat{n} Valeur_vénale \hat{z} . La classe-fait résultante contient donc ces 3 mesures.

La fusion des 2 diagrammes décisionnels donne le diagramme décisionnel suivant car un seul domaine d'activité est concerné.

Par la suite, le diagramme décisionnel doit être évalué par rapport aux résultats de l'analyse des besoins Pilotage et ceux de l'analyse des besoins Equipement. Dans le cas où l'évaluation révèle qu'il y a des problèmes d'adéquation entre les trois types de besoins, une nouvelle itération commence alors pour chaque processus d'analyse de besoins en tenant compte des retours de la confrontation.

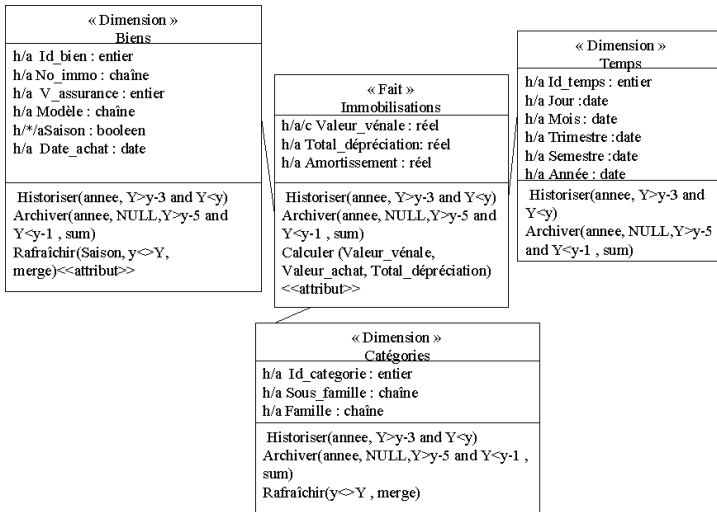


FIG. 6 – Diagramme décisionnel du projet.

7 Conclusion

Les besoins des utilisateurs se placent au centre de tout système d'information, y compris de SID Kimball et al. (1998). L'appropriation du SID par les utilisateurs est parmi les facteurs déterminant son succès. Cette importance est plus accentuée dans un contexte décisionnel car l'objectif du système est de fournir un outil d'aide à la décision permettant à l'entreprise de faire face à une concurrence accrue. L'analyse des besoins des trois groupes d'acteurs est une étape essentielle car elle est en amont de tout le développement du SID.

La représentation des besoins avec une structure proche de la vision des acteurs décisionnels et qui fournit des informations tant analytiques que fonctionnelles permet de réduire les interprétations possibles des besoins. De même, la formalisation avec les diagrammes décisionnels (DD) facilite la confrontation des besoins des différents acteurs. Notre formalisation des besoins permet aussi d'exprimer les traitements. Ainsi, nous abordons les problèmes des traitements ETL dès la phase d'analyse. De plus, notre démarche est facile à mettre en oeuvre car elle repose sur un seul modèle et un ensemble de règles de structuration. Les règles peuvent servir de base à une automatisation du processus d'analyse.

Nous souhaitons poursuivre nos travaux par l'analyse des besoins des groupes Pilotage et Equipement. Afin de valider nos propositions, nous réfléchissons au développement d'un outil pour la transformation des tableaux multidimensionnels en DD. A partir de l'étude des problèmes liés à l'analyse de ces deux types de besoins décisionnels, il sera possible de traiter les inadéquations entre les besoins des trois types d'acteurs décelées lors de la confrontation. Par ailleurs notre démarche n'intègre pas l'expression des hiérarchies dans la phase d'analyse, nous continuerons donc nos travaux par la prise en compte de cette caractéristique des besoins décisionnels.

Références

- Akoka, J., I. Comyn-Wattiau, et N. Prat (2001). From uml to rolap multidimensional databases using a pivot model. Technical report.
- Annoni, E., F. Ravat, O. Teste, et G. Zurfluh (2005). Les systèmes d'informations décisionnels : une approche d'analyse et de conception à base de patrons. *revue RSTI série ISI, n°Méthodes Avancées de Développement des SI* 10(6).
- Bonifati, A., F. Cattaneo, S. Ceri, A. Fuggetta, et S. Paraboschi (2001). Designing data marts for data warehouses. *ACM Trans. Softw. Eng. Methodol.* 10(4), 452–483.
- Bruckner, R., B. List, et J. Schiefer (1999). Developing requirements for data warehouse systems with use cases. AMCIS.
- Cabibbo, L. et R. Torlone (1998). A logical approach to multidimensional databases. *Lecture Notes in Computer Science* 1377, 155–162.
- Ghozzi, F., F. Ravat, O. Teste, et G. Zurfluh (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, pp. 51–70. Cépadués éditions.
- Golfarelli, M. et S. Rizzi (1998). 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*, pp. 3–9. ACM.
- Golfarelli, M. et S. Rizzi (1999). Designing the data warehouse : key steps and crucial issues. *Journal of Computer Science and Information Management* 2(3).
- Jacobson, I. (1992). *Object Oriented Software Engineering : A Use Case Driven Approach*. Wokingham, UK : Addison-Wesley.
- Kimball, R. (1996). *The data warehouse toolkit : practical techniques for building dimensional data warehouses*. New York, NY, USA : John Wiley & Sons, Inc.
- Kimball, R., L. Reeves, W. Thornthwaite, M. Ross, et W. Thornwaite (1998). *The Data Warehouse Lifecycle Toolkit : Expert Methods for Designing*. New York, NY, USA : John Wiley & Sons, Inc.
- Luján-Mora, S., P. Vassiliadis, et J. Trujillo (2004). Data mapping diagrams for data warehouse design with uml. In *ER*, pp. 191–204.
- Luján-Mora, S. (2005). *Data Warehouse Design with UML*. Ph. D. thesis, Department of Software and Computing Systems, University of Alicante.
- Moody, D. L. et M. A. R. Kortink (2000). From enterprise models to dimensional models : a methodology for data warehouse and data mart design. In *Design and Management of Data Warehouses*, pp. 5.
- Soussi, A., J. Feki, et F. Gargouri (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, pp. 71–89. Cépadués éditions.
- Tardieu, H., A. Rochfeld, et R. Colletti (2000). *Méthode Merise (La) : principes et outils*. Editions d'Organisation.

Teste, O. (2000). *Modélisation et Manipulation d'Entrepôts de Données Complexes et Historisées*. Ph. D. thesis, Paul Sabatier University.

Summary

The widespread use of Decision Support Systems (DSS) requires the development of design methods for DSS. Unlike Information systems (IS) models which have no ambition of being understood by users, DSS models must be usable by data analysts and decision makers. Among the proposed DSS engineering methods, few have an implicit task of user requirement analysis.

For such reasons, we propose our method of collection and formalization for DSS users requirements, using models close to their vision of data. From these requirements presented as multidimensional tables, we propose extensions of the object modeling to formalize user requirements in terms of data and treatments in the multidimensional context.