

MÉLANIE LAMBERT

**DÉVELOPPEMENT D'UNE APPROCHE POUR
L'ANALYSE SOLAP EN TEMPS RÉEL:
ADAPTATION AUX BESOINS DES ACTIVITÉS
SPORTIVES EN PLEIN AIR**

Mémoire présenté
à la Faculté des études supérieures de l'Université Laval
dans le cadre du programme de maîtrise en Sciences géomatiques
pour l'obtention du grade de maître ès sciences (M.Sc.)

Département des sciences géomatiques
FACULTÉ DE FORESTERIE ET DE GÉOMATIQUE
UNIVERSITÉ LAVAL
QUÉBEC

2006

Résumé

Au cours des dernières années, différents types de travaux ont été réalisés indépendamment au sein du même centre de recherche (Centre de Recherche en Géomatique de l'Université Laval). Parmi ceux-ci, on retrouve des travaux axés sur l'acquisition et le traitement des données spatiales en sport de plein air d'une part, et des travaux axés sur l'exploration et l'analyse des données spatiales avec une solution SOLAP d'autre part. L'exploitation conjointe de ces travaux permettait de répondre à de nouvelles attentes et plus particulièrement à une nouvelle application : l'évaluation et l'analyse de la performance d'athlètes pratiquant un sport extérieur grâce à des données calculées à partir d'observations GPS.

En effet, suite à des observations GPS, la position, la vitesse et l'accélération de l'athlète peuvent être calculées précisément. Cependant, aucun logiciel ne permettait d'analyser rapidement et facilement les nouvelles données recueillies. Pourtant, les entraîneurs d'athlètes de haut niveau désirent obtenir des données sur les performances actuelles, de façon rapide et exacte, pour ainsi adapter immédiatement leur entraînement et favoriser le succès de l'athlète. Or, la technologie SOLAP offre aux utilisateurs une interface cliente très intuitive pour l'analyse spatio-temporelle. Cependant, son fonctionnement ne permettait pas d'ajouter rapidement de nouvelles données obtenues à partir d'observations GPS.

Cette recherche visait alors à développer une approche répondant à des besoins d'analyse SOLAP en temps réel retrouvés dans certaines applications et plus particulièrement dans le sport de haut niveau. Nous avons aussi vérifié qu'une solution SOLAP utilisée dans le domaine de la gestion des entreprises pour faciliter les prises de décision peut être transposée dans celui de l'analyse de la performance des athlètes. Pour ce faire, un SOLAP **juste-à-temps**, baptisé SOLAP-SPORT, a été développé dans le cadre de ce projet de recherche.

Avant-Propos

Mon intérêt pour la recherche a débuté suite à un emploi d'été au Centre de Recherche en Géomatique (CRG) de l'Université Laval. Je voudrais alors remercier mon directeur de recherche, le Dr. Yvan Bédard, pour m'avoir donné l'opportunité de travailler au sein de son équipe de recherche et pour m'avoir dirigé pendant mes études au deuxième cycle. J'aimerais aussi remercier mon codirecteur, le Dr. Rock Santerre spécialiste en GPS, pour ses précieux conseils. Merci aussi au Dr. Thierry Badard pour ses commentaires constructifs suite à la lecture de ce mémoire.

Merci aussi à toute l'équipe du CRG et plus particulièrement à Frédéric Hubert et Sonia Rivest de m'avoir accordé plusieurs heures de leur temps pour répondre à mes innombrables questions et de m'avoir donné de bons conseils. Un merci tout spécial à Karine Bédard qui a été mon soutien moral et souvent ma source de motivation au cours de la rédaction de ce mémoire.

Merci à Jean-Philippe Veilleux pour avoir accepté de réaliser lui aussi sa maîtrise sur ce projet de recherche et pour sa disponibilité tout au long de ce projet. Merci aussi au soutien financier Centre National Multisport Montréal (CNMM) d'avoir cru en ce projet de recherche.

Je tiens aussi à remercier le Dr. Normand Teasdale professeur au département de kinésiologie, Robert Tremblay entraîneur national de patinage de vitesse ainsi que les professionnels du département de kinésiologie pour leur disponibilité. Merci aussi au Fond Québécois de Recherche sur la Nature et les Technologies (FQRNT) de m'avoir fourni un soutien financier au cours de mes études graduées.

Enfin, merci à ma famille et plus particulièrement à mon conjoint, Daniel, qui m'ont encouragée et soutenue tout au long de mes études.

Table des matières

INTRODUCTION	1
Mise en contexte	1
Problématique	6
Objectifs	6
Méthodologie	7
Description des chapitres du mémoire	10
1 REVUE DES CONCEPTS TEMPORELS POUR LA GÉOMATIQUE	
DÉCISIONNELLE	12
1.1 Concept du temps réel	13
1.1.1 Définition d'un système temps réel	13
1.1.2 Caractéristiques d'un système temps réel	14
1.2 Concept du temps avec les systèmes transactionnels	15
1.2.1 SGBD traditionnel	15
1.2.2 SGBD temps réel	17
1.3 Aide à la décision traditionnelle	18
1.3.1 Technologie OLAP traditionnelle	18
1.3.2 Entrepôt de données traditionnel	20
1.3.3 Principe de l'agrégation de données	22
1.3.4 Actualisation des données d'un cube OLAP	25
1.4 Aide à la décision temps réel	26
1.4.1 Technologie OLAP temps réel	27
1.4.2 Entrepôt de données temps réel	29
1.4.3 Actualisation rapide des agrégations	32
1.5 Conclusion	36
2 LE TEMPS RÉEL DANS UN CONTEXTE SOLAP	38
2.1 Technologie SOLAP traditionnelle	38
2.1.1 Définition	38
2.1.2 Agrégation spatiale	40
2.2 Adaptation du SOLAP pour des besoins temps réel	41
2.2.1 Concepts temporels	41
2.2.2 Catégories de SOLAP	43
2.2.3 SOLAP TR vs SOLAP JAT	49
2.3 Conclusion	55
3 IMPLANTATION D'UN OUTIL POUR L'ANALYSE DE LA PERFORMANCE	56
3.1 Évaluation d'opportunité	56
3.1.1 Besoins généraux	57
3.1.2 Besoins spécifiques	59
3.2 Inventaire et analyse de l'existant	61
3.2.1 Données existantes	61
3.2.2 Logiciels existants	62
3.3 Solution répondant aux besoins	63
3.4 Conception des cubes d'analyse	65
3.4.1 Dimensions descriptives	65

3.4.2	Dimension spatiale.....	67
3.5	Architecture SOLAP-SPORT	74
3.5.1	Module transactionnel.....	75
3.5.2	Module multidimensionnel.....	78
3.6	Fonctionnement du SOLAP-SPORT	83
3.6.1	Partie transactionnelle.....	84
3.6.2	Partie multidimensionnelle	85
3.7	Tests de performance et solution retenue	94
3.8	Présentation et validation du prototype SOLAP-SPORT	98
3.9	Conclusion	102
CONCLUSION.....		104
Travaux futurs.....		106
BIBLIOGRAPHIE.....		108
Annexe A	Inventaire des données spatiales disponibles.....	114
Annexe B	Dimensions et mesures	117
Annexe C	Information implicite et explicite sur un SOLAP	124
Annexe D	Diagrammes SOLAP-SPORT.....	126
Annexe E	Format des fichiers XML dans le SOLAP-SPORT	128
Annexe F	Diagrammes SOLAP-SPORT après optimisation	133
Annexe G	Comparaison entre une analyse réalisée sur un SIG et sur un SOLAP	135

Liste des tableaux

Tableau 1-1 : Besoins en entrepôt temps réel.....	30
Tableau 3-1 : Tests de performance réalisés avec le cube.....	95
Tableau 3-2 : Tests de performance réalisés avec le cube TECHNIQUE.....	97
Tableau C-1 : Nombre de cancers pour les régions.....	124
Tableau C-2 : Nombre de cancers pour les sous-régions.....	125

Liste des figures

Figure 1 : Étapes d'implantation d'une technologie décisionnelle.....	3
Figure 2 : Étapes de la méthodologie.....	8
Figure 3 : Cycle de vie d'un SIRS [Bédard 2002].....	9
Figure 1-1 : Temps de réponse d'un système.....	14
Figure 1-2 : Architecture d'un ROLAP.....	19
Figure 1-3 : Modèle multidimensionnel représentant les ventes de polices d'assurance d'une compagnie.....	21
Figure 1-4 : Cube nombre de polices d'assurance vendues.....	24
Figure 1-5 : Peuplement progressif.....	31
Figure 1-6 : Niveaux et membres de la dimension PRODUIT.....	34
Figure 2-1 : SOLAP temps réel, temps quasi-réel, temps propice, juste-à-temps et traditionnel.....	48
Figure 2-2 : Cas d'utilisation d'un SOLAP temps réel.....	49
Figure 2-3 : Cas d'utilisation SOLAP juste-à-temps.....	50
Figure 2-4 : Diagramme d'activités d'un SOLAP temps réel.....	51
Figure 2-5 : Diagramme d'activités d'un SOLAP juste-à-temps.....	52
Figure 2-6 : Diagramme de séquence avancé d'un SOLAP temps réel.....	53
Figure 2-7 : Diagramme de séquence d'un SOLAP juste-à-temps.....	54
Figure 3-1 : Visualisation de l'endroit exact sur la piste où le résultat (vitesse moyenne (m/s)) a été atteint.....	58
Figure 3-4 : Dimensions et mesures du cube PATINAGE DE VITESSE – ENTRAÎNEMENT.....	66
Figure 3-5 : Découpage de la piste en quatre niveaux hiérarchiques.....	68
Figure 3-6 : Niveau Piste de la dimension spatiale DÉCOUPAGE PISTE;.....	69
Figure 3-7 : Positions instantanées de la dimension spatiale.....	71
Figure 3-8 : Niveau alternatif Corridor de la dimension spatiale latente DÉCOUPAGE PISTE.....	72
Figure 3-9 : Exactitude du corridor occupé par l'athlète.....	73
Figure 3-10 : Architecture générale du prototype SOLAP-SPORT.....	74
Figure 3-11 : Interface graphique du système source.....	76
Figure 3-12 : Modèle physique de la base de données transactionnelle (BDT).....	77
Figure 3-13 : Modèle physique du marché de données appartenant à la base de données multidimensionnelle (BDM).....	79
Figure 3-14 : Interface cliente SOLAP avec le cube PATINAGE DE VITESSE-TECHNIQUE.....	82
Figure 3-15 : Sac à dos porté par l'athlète.....	84
Figure 3-16 : Exemple d'un format IGES représentant un segment atomique.....	88
Figure 3-17 : Indexation à partir des sous-segments de la piste.....	90
Figure 3-18 : Tour de piste.....	91
Figure 3-19 : Corrélation possible entre les pics de vitesse et la position de l'athlète dans son trajet.....	101
Figure B-1 : Dimensions et mesures pour le patinage de vitesse.....	117
Figure B-2 : Dimensions et mesures pour le cyclisme.....	118
Figure B-3 : Dimensions et mesures pour le ski de fond.....	119
Figure B-4 : Dimensions et mesures pour le ski alpin.....	120
Figure B-5 : Dimensions et mesures pour le canoë-kayak.....	121
Figure B-6 : Dimensions et mesures du cube TECHNIQUE.....	122

Figure B-7 : Dimensions et mesures du cube MÉTÉO	123
Figure C-1 : Carte affichant le nombre de cancers dans les régions.....	125
Figure D-1 : Diagramme d'activités SOLAP-SPORT	126
Figure D-2 : Diagramme de séquence SOLAP-SPORT	127
Figure F-1 : Diagramme d'activités SOLAP-SPORT après optimisation du prototype	133
Figure F-2 : Diagramme d'activités SOLAP-SPORT après optimisation du prototype	134

INTRODUCTION

Mise en contexte

Acquisition des données

De nos jours, plusieurs techniques existent pour acquérir des données à référence spatiale : la photogrammétrie, la topométrie, la télédétection, la géodésie (dont les GNSS (*Global Navigation Satellite Systems*)), etc. Le système de positionnement par GPS (*Global Positioning System*) est une technologie GNSS qui permet de localiser précisément les objets situés à la surface de la Terre en tout temps (24 heures sur 24), en tous lieux et peu importe les conditions météorologiques. Toutefois, la précision des données à référence spatiale recueillies avec ce système varie selon le type d'observations GPS offert par le récepteur (mesures de phase, mesures de pseudodistance, mesures de fréquence Doppler), la géométrie des satellites GPS lors des observations et le type de positionnement GPS (positionnement absolu ou relatif) utilisé [Santerre 2001]. Le système GPS est alors utilisé dans une diversité de domaines d'application exigeant des niveaux de précision différents (*ex.*: armée, santé, transport) et où sa fonction ne se limite plus au simple positionnement.

Des travaux [Maltais & Santerre 1999; Lambert 2002; Lambert & Santerre 2004; Veilleux, Lambert et al. 2004] réalisés au Centre de Recherche en Géomatique (CRG) de l'Université Laval sous la supervision du Dr. Rock Santerre et en collaboration avec le département de kinésiologie de l'Université Laval visaient justement à utiliser le système GPS pour une application bien particulière : mesurer et analyser la performance des athlètes pratiquant un sport extérieur. D'autres travaux [Miller, Miller et al. 2000; Skaloud & Merminod 2000; Larsson 2003] visant aussi l'analyse de la performance des athlètes à partir de mesures GPS, ont aussi été réalisées à l'extérieur de l'Université Laval.

Les travaux de Mathieu Lambert [Lambert 2002] ont démontré que des mesures sur la position et la vitesse peuvent être obtenues en temps réel, de façon précise (centimétrique), à l'aide d'un récepteur et d'une antenne GPS (portés par l'athlète), d'une station de

référence (un point géodésique) et d'un récepteur (installé sur la station de référence). Des mesures intéressantes ont été obtenus en patinage de vitesse, en canoë-kayak et en cyclisme et offraient une telle précision qu'il était possible, par exemple, de visualiser la fréquence des coups de patin (10 Hz) ou de rame sur un graphique de la vitesse en fonction du temps. Pour diffuser les données (position et vitesse) calculées à partir des observations GPS, un premier système offrant une interface de visualisation cartographique a été développé à partir d'un Système d'Information Géographique (SIG) [Lambert 2002]. Ce système permettait principalement de suivre le déplacement de l'athlète en temps réel. Un deuxième système a été développé au département de kinésiologie à partir du logiciel de traitement mathématique *Matlab* de la compagnie *The MathWorks* pour analyser plus en profondeur les valeurs sur la performance des athlètes obtenues.

Toutefois, ces deux systèmes possèdent des limites. Tout d'abord, ces systèmes gèrent difficilement le grand volume de données (position, vitesse, accélération) calculées à chaque époque lors d'observations GPS. En effet, des données très précises sont obtenues grâce à l'utilisation du GPS, mais l'entraîneur doit souvent attendre plusieurs heures, voire quelques jours pour analyser les nouvelles **données GPS**¹ ou pour obtenir des statistiques concrètes sur la performance d'un athlète. Ensuite, un SIG n'est pas adapté pour réaliser facilement des analyses sur les données, principalement des **analyses globales**² (cf. 0) [Proulx & Bédard 2004]. Il en est de même pour le logiciel *Matlab* qui, en plus, n'offre pas de volet cartographique, élément pourtant essentiel pour exploiter la référence spatiale des données à son plein potentiel. Enfin, les nombreuses fonctions offertes sur l'interface graphique de ces deux systèmes rendent les analyses très complexes à réaliser pour un non-initié.

¹ Pour alléger le texte, la simple expression « données GPS » sera dorénavant utilisée pour référer aux résultats (position, vitesse, accélération) calculés à partir d'observations GPS.

² Analyses ne concernant pas directement les données détaillées, c'est-à-dire un seul enregistrement dans la BD (ex. : la vitesse instantanée atteinte en un point), mais plusieurs enregistrements (ex. : la vitesse moyenne atteinte entre l'entrée et la sortie de la première courbe)

Exploration et l'analyse des données

Parallèlement au mécanisme d'acquisition de données, une famille d'outils, appelés systèmes d'analyse (systèmes décisionnels ou *Business Intelligence (BI)*), a été développée pour répondre à des besoins **d'analyse globale** sur les données. Ces outils sont principalement utilisés dans des organisations réalisant plusieurs opérations chaque jour (des milliers voire des millions) et possédant ainsi un grand volume de données. Un magasin de vêtements possédant plusieurs filiales, le ministère des Transports, ou une institution financière avec ses nombreuses succursales, sont des exemples d'organisation pouvant tirer profit des systèmes d'analyse lors de leurs prises de décision. Le système d'analyse OLAP (*On-Line Analytical Processing*) offre une interface graphique très intuitive, ce qui facilite l'analyse des données pour un non-initié. Ainsi, avec un tel système, les analystes et les décideurs peuvent réaliser des comparaisons dans le temps et construire des graphiques statistiques à l'aide de simples clics de la souris.

Le processus d'implantation d'une technologie décisionnelle se divise habituellement en trois grandes étapes [Basu 2003; Bédard 2003; Nelson 2004]: (1) la saisie des données, (2) l'intégration des données dans un entrepôt et (3) l'analyse à l'aide d'outils clients. Ces étapes sont illustrées sur la Figure 1.

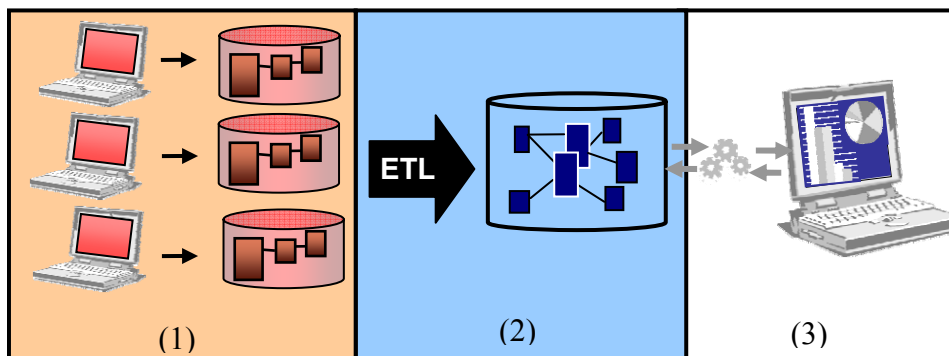


Figure 1 : Étapes d'implantation d'une technologie décisionnelle
 (1) la saisie des données; (2) l'intégration dans un entrepôt; (3) l'analyse à l'aide d'un outil client

(1) La saisie des données. La saisie des données se réalise principalement par des systèmes de type OLTP (*On-Line Transactional Processing*) ou systèmes transactionnels. Ces derniers facilitent aussi la mise à jour des données et assurent

l'intégrité de la base de données transactionnelle sous-jacente. Les systèmes de gestion de bases de données (SGBD) relationnelles, tels que *Microsoft SQL Server* ou *Oracle*, se spécialisent dans ce type d'opérations. Considérant qu'environ 80% des données possèdent une référence spatiale [Franklin 1992] et que cette information spatiale est souvent stockée mais difficilement gérée avec les SGBD, d'autres types d'outils sont apparus. Les SIG sont justement un exemple d'outil gérant plus efficacement la géométrie des données.

- (2) L'intégration dans un entrepôt.** Habituellement, lorsqu'une organisation désire interroger l'ensemble de ses données provenant de différentes sources, obtenir des valeurs sommaires et les comparer dans le temps, une structure centralisée nommée entrepôt de données est construite et exploitée. Des données spatiales et/ou non-spatiales sont alors extraites des systèmes transactionnels (les SIG, les SGBD, les chiffriers, etc.), intégrées et archivées dans l'entrepôt. Ce processus se nomme ETL pour *Extracting, Transforming and Loading*. Cette opération est plutôt lourde et prend typiquement plusieurs heures, notamment lorsque les données sont à agréger (résumer) dans le but de fournir des informations plus globales. Seule une copie des données sources est utilisée et conservée dans l'entrepôt. Bien qu'un entrepôt offre un accès rapide à l'information stratégique répondant à des besoins spécifiques, un outil d'analyse est souvent nécessaire pour en faciliter l'exploitation.
- (3) L'analyse à l'aide d'un outil client.** Les outils OLAP sont actuellement les plus performants pour l'analyse rapide et intuitive des données d'un entrepôt. De plus, des travaux effectués au Centre de Recherche en Géomatique (CRG) de l'Université Laval [Caron 1998] ont démontré qu'un système OLAP peut supporter l'analyse spatio-temporelle. Toutefois, pour tirer plein profit de la référence spatiale lors de la découverte de nouvelles connaissances ou la prise de décision, des améliorations importantes doivent être réalisées sur le système OLAP. Une description plus détaillée des avantages de l'exploitation de la référence spatiale est d'ailleurs présentée par [Bédard, Proulx et al. 2005]. En tenant compte de ces possibilités, une nouvelle famille d'outils nommée SOLAP (*Spatial On-Line Analytical Processing*) est apparue.

L'outil d'analyse SOLAP facilite la navigation à travers de très grandes bases de données spatio-temporelles (*ex.*: entrepôt de données spatiales). Il se distingue d'un outil OLAP principalement par ses méthodes d'exploitation de l'information spatiale des données. Il permet à la fois l'exploration facile et rapide des données dans l'espace et dans le temps. Selon les besoins de l'utilisateur, les résultats d'analyse sont affichés sur une carte, sur un graphique, sur un tableau ou sur les trois types d'affichage à la fois, et ceci de manière « synchronisée » ou non. Les applications SOLAP sont utilisées dans plusieurs domaines tels que la santé [Bédard & Henriques 2003], le transport [Rivest, Gignac et al. 2004], l'archéologie [Rageul 2004], et bien d'autres [Bédard et al. 2005].

Ainsi, les deux sections de la mise en contexte présentent deux types de travaux réalisés indépendamment au sein du même centre de recherche (CRG de l'Université Laval) : des travaux axés sur l'acquisition et le traitement des données spatiales en sport de plein air d'une part, et des travaux axés sur l'exploration et l'analyse des données spatiales avec une solution SOLAP d'autre part. Maintenant, l'exploitation conjointe de ces travaux permettrait de répondre à de nouvelles attentes et plus particulièrement à une nouvelle application : l'évaluation et l'analyse de la performance d'athlètes pratiquant un sport extérieur grâce à des données calculées suite à des observations GPS. Outre le fait que cette recherche a été réalisée pour permettre l'avancement du domaine géomatique (*ex.*: solution se rapprochant d'un SOLAP temps réel), elle doit aussi démontrer que les techniques de gestion, de diffusion et d'analyse des données à référence spatiales qu'on y retrouve peuvent être utilisées dans cette application sportive. Cette recherche a alors été réalisée dans le cadre d'une subvention offerte pour le projet « Utilisation du système de positionnement global (GPS) et des outils d'exploration et d'analyse OLAP/SOLAP pour l'évaluation et le suivi de sportifs de haut niveau en patinage de vitesse, cyclisme sur piste, et canoë-kayak » qui regroupe trois chercheurs principaux : le Dr. Yvan Bédard et le Dr. Rock Santerre du département de sciences géomatiques, ainsi que le Dr. Normand Teasdale, responsable du projet, Martin Simoneau et Vincent Cantin du département de kinésiologie et.

Problématique

Les observations GPS permettent de calculer la position, la vitesse et l'accélération de l'athlète de façon précise (au centimètre pour la position) [Veilleux et al. 2004], mais aucun logiciel ne permet à ce jour d'analyser rapidement et facilement les nouvelles données sur la performance obtenues. Pourtant, les entraîneurs désirent obtenir des données sur les performances actuelles des athlètes de façon rapide et exacte pour ainsi adapter immédiatement leur entraînement et favoriser le succès de l'athlète [Hay 1991]. De plus, les logiciels présentement offerts aux entraîneurs pour analyser la performance en sport de haut niveau sont très peu intuitifs et n'exploitent pas toujours la référence spatiale des données. Cette information est pourtant présente dans les données GPS (longitude, latitude et l'altitude) qui elles sont directement obtenues à partir d'observations GPS.

L'usage d'un outil SOLAP, outil très intuitif pour l'analyse spatio-temporelle, pourrait remédier à cette dernière lacune. Cependant, le fonctionnement d'un tel outil ne permet pas à l'heure actuelle d'ajouter et d'exploiter rapidement des données GPS. En effet, dans la majorité des applications, les nouvelles données sont ajoutées dans l'entrepôt à une fréquence hebdomadaire, mensuelle, voire même annuelle. De plus, les systèmes d'analyse SOLAP nécessitent souvent le pré-calcul des agrégations (*cf.* section 1.3.3). Or, ce processus n'est pas instantané suite à l'ajout de nouvelles données dans l'entrepôt. L'analyse des données n'est alors possible qu'en temps différé. Proposer alors une solution allant vers la constitution d'un outil SOLAP temps réel ou temps quasi-réel appliqué au domaine du sport et exploitant des données GPS était un défi intéressant et pertinent. En effet, cette solution SOLAP permettrait à un entraîneur d'analyser les nouvelles données GPS de façon simple, rapide et efficace.

Objectifs

L'objectif principal de cette recherche est de développer une approche répondant à des besoins d'analyse SOLAP en temps réel retrouvés dans certaines applications et plus particulièrement dans le sport de haut niveau. Bien que les systèmes de type SOLAP n'ont pas été construits au départ pour exploiter une source de données subissant des ajouts

fréquemment (source de données à faible latence), nous émettons l'hypothèse qu'un tel outil peut être utile dans ces circonstances.

Des objectifs secondaires ont également été dégagés :

1. Synthétiser les concepts théoriques servant à développer notre approche et qui sont relatifs au temps réel dans le monde de l'informatique et des bases de données décisionnelles;
2. Identifier les besoins propres à des sports extérieurs pour l'évaluation et l'analyse de la performance des athlètes. En raison de la contrainte imposée par le GPS, seulement des sports se pratiquant à l'extérieur doivent être choisis;
3. Identifier une approche permettant d'ajouter, dans des délais appropriés pour l'application, de nouvelles données spatiales (présentes dans les données GPS) et descriptives dans une base de données SOLAP;
4. Développer un prototype pour des applications sportives extérieures identifiées par l'équipe du Dr. Normand Teasdale du département de kinésiologie de l'Université Laval;
5. Vérifier si une solution SOLAP utilisée dans le domaine de la gestion des entreprises pour faciliter les prises de décision peut être transposée dans celui de l'analyse de la performance des athlètes.

Méthodologie

La méthodologie suivie au cours de cette recherche comporte huit étapes principales comme l'illustre la Figure 2. Les paragraphes suivants décrivent plus en détails chacune des étapes.

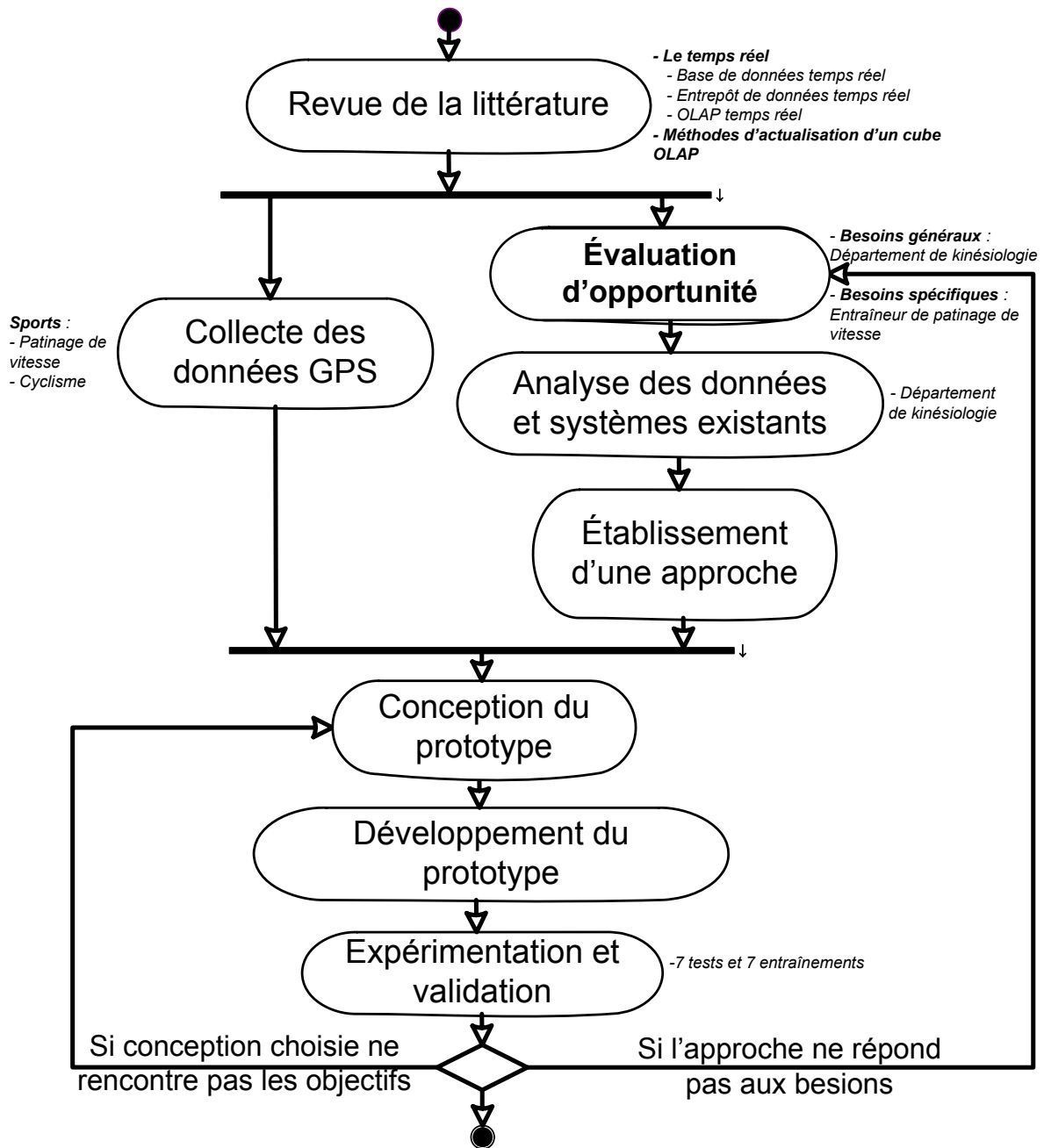


Figure 2 : Étapes de la méthodologie

La première étape de la méthodologie est la **revue de la littérature** sur différents concepts théoriques reliés aux bases de données décisionnelles (ex.: OLAP et SOLAP). Cette étape permet aussi de faire un état de l'art de tout ce qui existe actuellement en base de données décisionnelle pour répondre aux besoins temps réel des organisations (ex.: OLAP temps réel).

Les étapes suivantes de la méthodologie suivent celles du cycle de vie d'un système d'information à référence spatiale (SIRS) et sont directement reliées au volet applicatif de cette recherche.

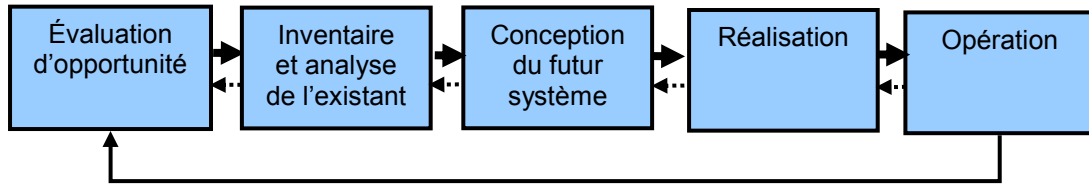


Figure 3 : Cycle de vie d'un SIRS [Bédard 2002]

Le lecteur est référé à [Bédard 2002] pour des détails sur chacune des étapes. L'**identification des besoins** (étape #1 du cycle de vie d'un SIRS) auprès des entraîneurs a d'abord été réalisée. Leurs besoins généraux et spécifiques en termes d'analyse ont été identifiés grâce aux recherches déjà accomplies au CRG dans le domaine du sport et à des rencontres avec le personnel du département de kinésiologie et un entraîneur national de patinage de vitesse (Robert Tremblay). Ensuite, un **inventaire des données et une analyse des systèmes existants** (étape #2 du cycle de vie d'un SIRS) au département de kinésiologie ont également été réalisés. Suite à ces deux étapes, l'**approche** la plus appropriée pour répondre aux besoins des entraîneurs en termes d'analyse de la performance des athlètes et pouvant être développée avec la technologie disponible a pu être déterminée.

La **collecte des données GPS** (enregistrement d'observations GPS et post-traitement) appartient habituellement à l'étape #4 du cycle de vie d'un SIRS, mais elle a été réalisée en parallèle par Jean-Philippe Veilleux candidat à la maîtrise sous la supervision du Dr. Rock Santerre. Au départ, des données en patinage de vitesse ont été recueillies sur l'anneau de glace Gaétan-Boucher situé dans la ville de Ste-Foy. Des données obtenues au cours de quatre séances d'entraînement différentes à l'hiver 2004 ont alors été utilisées pour notre recherche.

La **conception d'un prototype** pour mettre en œuvre notre approche (étape #3 du cycle de vie d'un SIRS) a débuté par la création de modèles conceptuels, étapes essentielles pour l'implantation d'un SOLAP traditionnel. Les fonctionnalités supplémentaires et reliées à

notre approche ont ensuite été modélisées à l'aide de diagrammes d'activités et de séquence. Le **développement du prototype** (étape #4 du cycle de vie d'un SIRS) a permis de mettre en place l'architecture. Pour ce faire, des logiciels disponibles au CRG comme *Oracle*, *SAS*, *SOLAP* ont été utilisés.

Enfin, lors de l'**expérimentation et la validation** du prototype (étape #5 du cycle de vie d'un SIRS), plusieurs tests ont été effectués. Si lors de l'expérimentation du prototype, la performance offerte ne permettait pas d'atteindre les objectifs, une nouvelle conception du prototype devait être trouvée. Si lors de la validation du prototype, celui-ci ne répondait pas aux besoins des entraîneurs, une nouvelle analyse des besoins devait être réalisée pour ainsi modifier l'approche choisie.

Description des chapitres du mémoire

Ce mémoire est structuré en trois chapitres. Le premier chapitre présente une synthèse de concepts temporels pour la géomatique décisionnelle. Tout d'abord, l'expression « temps réel » est définie. Ensuite, une revue des méthodes traditionnelles de gestion et d'exploitation des données propres aux systèmes décisionnels est réalisée. Au cours de cette revue, les nouveaux outils ou systèmes temps réel répertoriés dans ce domaine, comme l'aide la décision temps réel, entrepôt de données temps réel et OLAP temps réel, sont apportés et étudiés.

Le second chapitre rappelle les caractéristiques d'un outil SOLAP traditionnel, outil propre à la géomatique décisionnelle. Il apporte ensuite des approches *SOLAP* intégrant de nouveaux concepts associés à la temporalité : SOLAP temps réel, SOLAP temps quasi-réel, SOLAP temps propice (right time) ou SOLAP juste-à-temps.

Le troisième chapitre présente les étapes suivies lors de la détermination d'une approche visant à répondre aux besoins temps réel d'une application exploitant des données descriptives, mais aussi spatiales. Le cadre applicatif de notre travail est l'évaluation et l'analyse de la performance d'athlètes de haut niveau pratiquant un sport extérieur à partir de données (position, vitesse, accélération) obtenues suite à des observations GPS. Ainsi, ce chapitre débute par la présentation des besoins généraux et spécifiques des entraîneurs.

Un inventaire des données et des systèmes existants déjà dans ce domaine est ensuite fourni de manière à présenter les lacunes des solutions actuelles. Ce chapitre démontre aussi que l'approche retenue répond aux besoins d'analyse de l'application. Pour cela, la conception, l'architecture et le fonctionnement du prototype SOLAP-SPORT à développer sont effectués. Ensuite, une présentation et une analyse des résultats obtenus au cours de l'expérimentation du prototype et la solution finalement retenue pour atteindre les objectifs du système visé se retrouvent dans ce chapitre. Enfin, il montre les possibilités d'analyse sur l'interface utilisateur et démontre qu'une solution SOLAP permet d'analyser facilement et rapidement les performances des athlètes.

La conclusion de ce mémoire permet de faire un retour sur les objectifs que nous avons fixés au départ. Elle permet aussi d'identifier de nouvelles perspectives de recherche.

1 REVUE DES CONCEPTS TEMPORELS POUR LA GÉOMATIQUE DÉCISIONNELLE

Généralement, deux grandes catégories de systèmes existent dans les organisations : les systèmes **transactionnels** et **décisionnels**. Ces deux catégories de système répondent chacune à des besoins bien distincts. Cependant, pour répondre aux besoins analytiques temps réel de certains utilisateurs, un rapprochement des systèmes transactionnels et décisionnels est visé. Dès que de nouvelles données sont saisies et stockées dans la base de données transactionnelle, elles doivent être aussitôt diffusées sur le module client du système décisionnel (*ex.*: module client OLAP).

Le chapitre 1 présente les caractéristiques des systèmes transactionnels et décisionnels tout en apportant le nouveau concept temporel, le **temps réel**. Les études sur de nouvelles solutions décisionnelles dernièrement apparues, comme « l'aide à la décision temps réel », les « entrepôts de données temps réel » et l'« OLAP temps réel » permettra d'identifier des solutions pour la géomatique décisionnelle. En effet, la **géomatique décisionnelle** est apparue suite à l'utilisation des méthodes de gestion et d'exploitation des données propres aux systèmes décisionnels, plus particulièrement les systèmes OLAP, avec celles utilisées pour les données géospatiales [Bédard 2005].

La section 1.1 offre d'abord une définition de l'expression « temps réel ». Ensuite, les caractéristiques des systèmes transactionnels font l'objet de la section 1.2, alors que celles des systèmes décisionnels se retrouvent à la section 1.3. Enfin, le concept de l'aide à la décision temps réel est apporté à la section 1.4.

1.1 Concept du temps réel

La signification de l'expression « temps réel » varie selon les domaines d'application ou points de vue. Par exemple, les expressions « temps réel » et « temps de transaction » sont utilisées dans le domaine des bases de données pour distinguer le moment où un phénomène survient dans la réalité de celui où il est inscrit dans la base de données³ [Bédard & Chestein 1995]. L'expression « temps réel » est alors ici utilisée pour traduire l'expression anglaise *world time*.

Maintenant, l'expression « temps réel » est utilisée dans le domaine des bases de données pour qualifier certains systèmes (*ex.*: Système de Gestion de Bases Données (SGBD) *temps réel* [Duvallet & Mammeri 1999] et OLAP temps réel [Kennedy 2003]). Certaines précisions doivent alors être apportées sur la signification de cette expression. La section 1.1.1 définit le qualificatif « temps réel » en se basant sur un système temps réel. La section 1.1.2 explicite les caractéristiques d'un système temps réel.

1.1.1 Définition d'un système temps réel

Suite à la revue des définitions [Stankovic & Ramamritham 1988; Ginguay & Lauret 1993; Eriksson, Penker et al. 2004; Wikipédia 2005], un **système temps réel** est un système réactif qui accorde une grande importance à l'exactitude des résultats et qui doit respecter un temps de réponse minimal (contrainte temporelle). Le temps de réponse d'un système correspond au temps écoulé entre l'acquisition des données (entrée) et l'envoi des commandes à l'environnement (sortie) [Mabilleau 2001]. La Figure 1-1 illustre le temps de réponse d'un système (horloge rouge).

³ *Ex.* : maison incendiée. La date de l'incendie d'une maison est le *temps réel* et la date à laquelle l'information est inscrite dans la base de données est le *temps de transaction*.

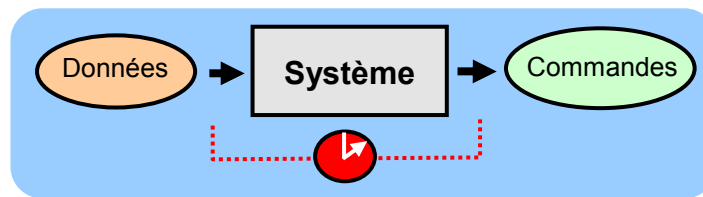


Figure 1-1 : Temps de réponse d'un système

Entre l'entrée et la sortie du système, un traitement de l'information (données) est réalisé et ce temps de traitement correspond principalement au temps de réponse du système. Dans la définition générale d'un système temps réel, aucune valeur maximale pour le temps de réponse n'est spécifiée puisqu'il doit simplement être acceptable pour l'application. La section 1.1.2 décrit les principales caractéristiques que doit posséder un système temps réel.

1.1.2 Caractéristiques d'un système temps réel

De façon générale, un système temps réel doit posséder les trois caractéristiques suivantes [Stankovic & Ramamritham 1988; Cooling 1991; Bui 1998; Eriksson et al. 2004]:

1. La **rapidité** est très importante, puisque le système doit exécuter des traitements en un certain laps de temps donné (inférieur à la contrainte temporelle de l'application).
2. La **réactivité** du système relativement à l'environnement. Un système temps réel est avant tout un système réactif. Les systèmes temps réel doivent réagir aux actions de l'utilisateur (environnement extérieur) souvent grâce à des déclencheurs internes présents dans le système.
3. La **fiabilité** du système ou la **tolérance aux fautes**. Prévoir le comportement du système en fonction de l'évolution de l'environnement doit être possible. Un système temps réel doit fournir des réponses exactes et respecter la contrainte temporelle.

Ainsi, le qualificatif « temps réel » est utilisé en informatique pour qualifier un système rapide, réactif et fiable. Lors de la présentation du nouveau concept SOLAP temps réel dans le chapitre 2, nous allons principalement utiliser cette définition et ces caractéristiques. La suite de ce chapitre fait une synthèse de l'utilisation actuelle de l'expression « temps réel » dans le domaine des bases de données transactionnelles et décisionnelles.

1.2 Concept du temps avec les systèmes transactionnels

Les systèmes transactionnels (OLTP : *On-Line Transactional Processing*) sont principalement utilisés pour la saisie, la recherche et la mise à jour des données d'une base de données. Ils assurent aussi l'intégrité de la base lors des transactions. La section 1.2.1 présente les SGBD traditionnels et la structure d'une base de données transactionnelle, alors que la section 1.2.2 définit un concept plus récent avec le SGBD temps réel

1.2.1 SGBD traditionnel

Les besoins transactionnels d'une organisation sont souvent satisfaits par un système de gestion de bases de données relationnelles (SGBD relationnel), comme *Access* ou *Oracle*. Un SGBD relationnel utilise le langage SQL (*Structured Query Language*) pour réaliser les opérations transactionnelles sur les données de la base de données.

Une **base de données transactionnelle** est généralement définie à partir d'un modèle conceptuel tel que le modèle *Entité/Relation*, *Objet/Relation* ou encore le modèle *UML de classes*. La base est ensuite créée à partir d'un modèle d'implantation dont la structure est fortement normalisée. La normalisation consiste à créer, dans la base de données, plusieurs tables liées entre elles grâce aux clés primaires et étrangères. Ainsi, en éliminant à son maximum la redondance à l'intérieur de la base de données, les opérations transactionnelles sont plus simples à réaliser, car une nouvelle transaction affecte souvent une seule table et un seul enregistrement de la base. Cet avantage est primordial lorsque des milliers, voire des millions, de transactions sont effectuées en une seule journée au sein d'une même organisation (*ex.*: institution financière).

La base de données transactionnelle d'une compagnie d'assurances, par exemple, possède une table CLIENTS, une table VENDEUR, une table CONTRAT D'ASSURANCE et elle permet la gestion des vendeurs, des clients, des contrats d'assurance choisis, etc. Lorsque la base contient seulement les données courantes, le SGBD supprime les données périmées et les remplace par des données à jour.

Un SGBD exploitant plutôt une **base de données temporelle** (parfois appelé SGBD temporel) [Tansel, Clifford et al. 1993] gère à la fois des données courantes⁴ et des données historiques⁵. En d'autres mots, il gère différentes versions de données correspondant chacune à une date ou une période particulière. Un SGBD qui gère à la fois l'aspect temporel et spatial des données (BD spatio-temporelle) permet en plus aux analystes de visualiser l'évolution des phénomènes géographiques [Bédard 1993]. Ce sont les SGBD spatio-temporels.

Les SGBD traditionnels assurent la gestion des données, mais aucun délai n'est imposé au système pour réaliser les opérations transactionnelles. Ces systèmes ne respectent aucune contrainte temporelle et ne peuvent ainsi répondre adéquatement aux besoins des applications en temps réel [Ramamritham 1993]. Un nouveau type de SGBD est alors défini pour tenir compte de la notion de temps réel. La section suivante définit les SGBD temps réel.

⁴ Les données courantes ou actuelles sont les données présentes dans le système transactionnel et sur lesquelles des mises à jour sont régulièrement réalisées. Elles sont les données les plus récentes.

⁵ Des données historiques sont des données associées à une période de temps bien précise (située avant la période courante) et qui permettent la gestion de l'historique dans une base de données.

1.2.2 SGBD temps réel

Les **systèmes de gestion de bases de données temps réel** (SGBDTR) seraient un amalgame entre les systèmes temps réel définis à la section 1.1 et les SGBD traditionnels de la section précédente [Duvallet & Mammeri 1999].

Les systèmes temps réel sont construits avec des algorithmes très puissants qui permettent de respecter efficacement les contraintes temporelles (*cf.* section 1.1.1). Toutefois, les SGBD sont plus efficaces que les systèmes temps réel pour gérer de grands volumes de données [Duvallet & Mammeri 1999]. Lors d'une nouvelle transaction, un SGBDTR respecterait l'intégrité des données, mais aussi la limite de temps fixée au départ pour effectuer cette transaction (contrainte temporelle).

Un SGBDTR posséderait alors les trois caractéristiques requises par un système temps réel (*cf.* section 1.1). Le SGBDTR doit être rapide (*rapidité*), réagir aux opérations de l'utilisateur (*réactif*), fournir des réponses exactes et respecter une contrainte temporelle (*fiabilité*).

En résumé, un SGBD traditionnel est souvent utilisé pour réaliser les opérations transactionnelles. Le SGBD temporel et spatio-temporel gère à la fois les données historiques et courantes. Le SGBD temps réel gère les données courantes et/ou les données historiques, mais doit respecter en plus des contraintes temporelles imposées aux transactions lors des accès aux données.

Toutefois, les bases de données transactionnelles exploitées par ces systèmes sont souvent implantées à partir d'un modèle rendant la base de données peu intuitive et peu performante pour l'analyse globale des données [Kimball 1997]. Par exemple, pour connaître le revenu total d'une compagnie d'assurances durant le dernier trimestre, le système relationnel devra réaliser plusieurs jointures entre les tables et agréger les coûts des polices trouvées. La structure transactionnelle est alors inadaptée pour répondre aux besoins analytiques des utilisateurs. Au contraire, les bases de données décisionnelles sont intuitives et performantes lors de l'exploitation des données. La section 1.3 présente certains outils décisionnels traditionnels.

1.3 Aide à la décision traditionnelle

Les outils décisionnels ont été développés principalement pour faciliter l'exploitation et l'analyse des données. L'une des technologies d'aide à la décision qui se distingue des autres par ses possibilités d'exploitation facile et rapide des bases de données temporelles très volumineuses est la technologie OLAP. Cette technologie offre généralement des temps de réponse en dessous de dix secondes pour que l'utilisateur conserve son flux de pensée lors de ses analyses [Marchand, Brisebois et al. 2004].

La section 1.3.1 rappelle d'abord le fonctionnement et l'architecture d'une technologie OLAP traditionnelle. La section 1.3.2 décrit ensuite l'entrepôt de données traditionnel et la section 1.3.3 explique le principe de l'agrégation de données. Enfin, la section 1.3.4 décrit les étapes nécessaires pour actualiser les données d'un système OLAP.

1.3.1 Technologie OLAP traditionnelle

L'objectif de cette section est de présenter les concepts généraux de la technologie OLAP. Le lecteur est référé à [Codd, Codd et al. 1993; OLAP Council 1995; Thomsen 1997; Rivest 2000] pour obtenir des détails supplémentaires.

Configurations

Lors de la construction d'une application OLAP, plusieurs types de **configurations** sont possibles (MOLAP – OLAP Multidimensionnel ; ROLAP – OLAP Relationnel ; HOLAP – OLAP Hybride). La suite de cette section s'attarde uniquement à la configuration ROLAP. En effet, cette configuration est la plus utilisée jusqu'à maintenant lors de l'implantation d'une solution OLAP spatial (SOLAP), concept propre à la géomatique décisionnelle (*cf.* section 2.1). De plus, l'approche ROLAP a été choisie par *Microsoft* [Kennedy 2003] qui offre, avec *SQL-Server Analysis Services*, l'OLAP temps réel (*cf.* section 1.4). Suite à l'importation des données à l'intérieur de la structure propriétaire MOLAP, les mises à jour ou plutôt l'actualisation des données (*cf.* section 1.4.3), est aussi plus difficile [Shoshani 1997].

Architecture

L'**architecture principale** d'une technologie ROLAP se divise en trois parties: (a) la base de données relationnelle (b) le serveur ROLAP et (c) le module client. [Bédard, Larrivée et al. 1997; Rivest 2000]. La Figure 1-2 présente ces trois parties de l'architecture d'un ROLAP.

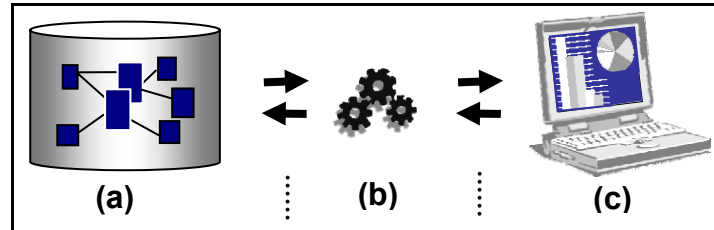


Figure 1-2 : Architecture d'un ROLAP.

(a) Base de données relationnelle, (b) Serveur ROLAP, (c) Client OLAP. Adaptée de [Rivest 2000]

(a) La base de données relationnelle d'une configuration ROLAP doit posséder une structure multidimensionnelle. Cette base de données est soit créée à partir de données détaillées provenant d'un entrepôt de données, soit l'entrepôt lui-même. Le concept d'entrepôt de données et la modélisation multidimensionnelle sont décrits à la section suivante (*cf.* section 1.3.2). La base de données d'une configuration ROLAP contient aussi les données agrégées. Le principe de l'agrégation de données est expliqué à la section 1.3.3. La structure multidimensionnelle et les données agrégées caractérisent un cube d'analyse OLAP.

(b) Le serveur ROLAP assure le lien entre la base de données et le module client. Il gère et exploite les données des cubes d'analyse OLAP à l'aide de requêtes SQL.

(c) Le module client OLAP offre une interface facile d'utilisation permettant aux utilisateurs d'interagir directement avec les données à l'aide de simples clics de la souris et sans nécessairement connaître un langage d'interrogation de bases de données.

Pour que les décideurs puissent exploiter l'ensemble des données concernant l'organisation, le module client doit exploiter une structure qui contient toutes les données de l'organisation, c'est-à-dire l'entrepôt. Si le module client doit répondre seulement à des

requêtes concernant une facette particulière de l'organisation, il peut exploiter seulement un marché de données. Un marché de données « *est une implantation localisée d'un entrepôt de données à usage unique* » [traduction libre [Devlin 1997]]. La section suivante définit un entrepôt de données traditionnel et explique la structure multidimensionnelle.

1.3.2 Entrepôt de données traditionnel

Un entrepôt de données « *est une collection de données portant sur des sujets touchant une organisation, intégrée, variant dans le temps, et non-volatile pour supporter le processus de prise de décision d'une organisation* » [traduction libre [Inmon 1996]]. L'entrepôt contient plusieurs versions d'un même phénomène (données historiques), car aucune mise à jour n'est réalisée sur les données (non-volatile).

Un entrepôt de données est, soit une base de données relationnelle, soit une base de données multidimensionnelle, soit un simple fichier (*flat file*)[Greenfield 2005]. Il peut aussi être à la fois une base de données relationnelle et multidimensionnelle. Lors de l'implantation d'un entrepôt dans une base de données relationnelle, la technique de conception la plus couramment utilisée est la **modélisation multidimensionnelle** (*Dimensional Modeling DM*) [Kimball 1997]. La structure multidimensionnelle est nécessaire lorsque l'entrepôt est exploité par un module client OLAP dans une configuration ROLAP.

Modélisation multidimensionnelle

Un modèle multidimensionnel se compose d'une table centrale nommée table des faits et de plusieurs tables nommées dimensions. La table des faits, comme son nom l'indique, contient des faits obtenus à la suite de la combinaison des dimensions (membres de dimension), ainsi qu'une ou plusieurs mesures. Une dimension est un thème ou un axe d'analyse et contient des membres alors qu'une mesure est une valeur quantitative sur laquelle portent les analyses [Kimball & Ross 2002].

Supposons que l'on possède des données sur les ventes de polices par une compagnie d'assurances. Pour cette application, les différentes formes de mesures pourraient être : le nombre de ventes, le total des ventes en dollars, le nombre de couvertures vendues, lesquelles seraient analysées selon différentes dimensions : le type de couverture, la catégorie d'âge des clients, les vendeurs, les jours de vente, les régions géographiques, etc. La Figure 1-3 présente le modèle multidimensionnel de cette compagnie d'assurances.

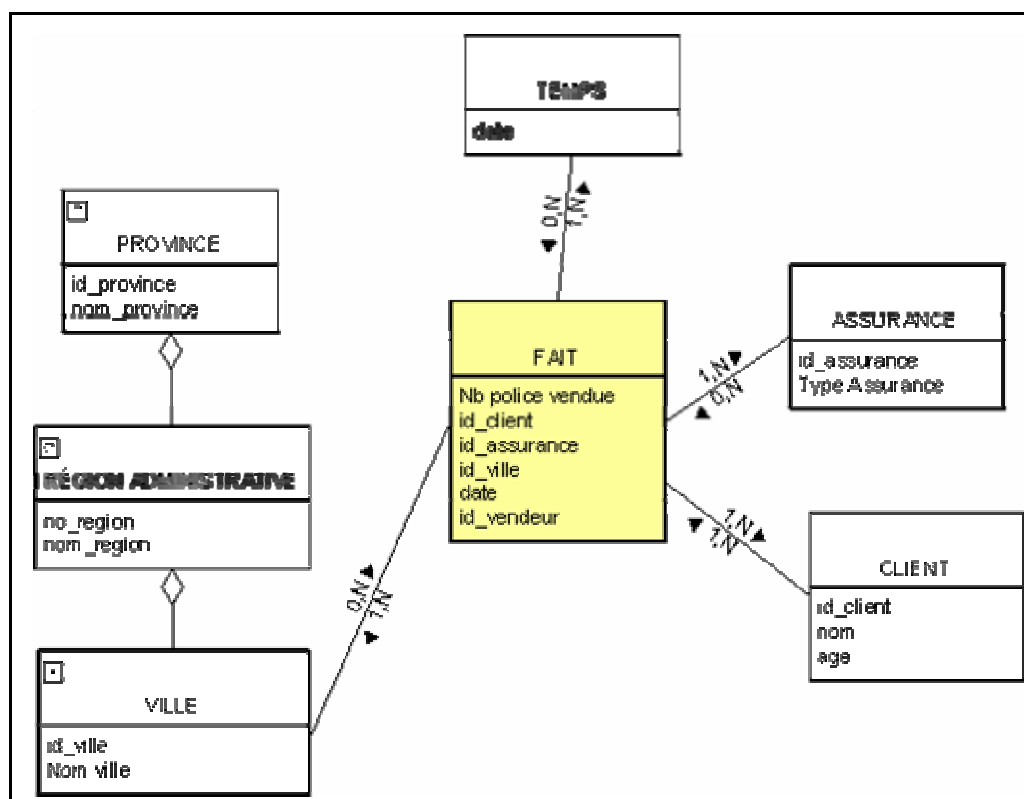


Figure 1-3 : Modèle multidimensionnel représentant les ventes de polices d'assurance d'une compagnie

Les dimensions TEMPS, ASSURANCE, CLIENT et DÉCOUPAGE ADMINISTRATIF et la table des faits FAIT_VENTE_ASSURANCE avec la mesure nombre de polices d'assurance vendue (NB POLICE VENDUE) forment le modèle multidimensionnel de la Figure 1-3. Un fait détaillé pourrait être : *Le 3 décembre 2005, le nombre de polices d'assurance-vie vendues à des clients âgés de 21 à 40 ans et habitant la ville de Matane est de 4 polices.*

Les dimensions sont souvent structurées selon plusieurs niveaux hiérarchiques. Par exemple, la dimension DÉCOUPAGE ADMINISTRATIF est représentée sur la Figure 1-3 par le

nom des villes (VILLE) au niveau hiérarchique inférieur, suivi du nom des régions administratives (RÉGION ADMINISTRATIVE) et enfin le nom des provinces (PROVINCE) au niveau supérieur. Cette structuration hiérarchique permet d'obtenir l'information à plusieurs niveaux de détails – niveaux de granularité – et de passer de l'un à l'autre plus facilement. Le membre BAS-ST-LAURENT appartient à la dimension DÉCOUPAGE ADMINISTRATIF au niveau hiérarchique RÉGION ADMINISTRATIVE.

Pour conserver les avantages du modèle multidimensionnel, le modèle physique de la base de données ressemble au modèle logique [Kimball & Ross 2002]. Contrairement à la **Modélisation Transactionnelle** (MT) (cf. section 1.2), le modèle logique multidimensionnel possède une structure dénormalisée, c'est-à-dire que la redondance des données est encouragée. La dénormalisation vise à faciliter l'exploitation de l'entrepôt et améliorer les performances.

Cependant, une base de données ROLAP contient aussi des données agrégées. Un cube d'analyse ROLAP est défini suite à la sélection des dimensions et des mesures (modèle multidimensionnel), mais il est construit grâce au pré-calcul et au stockage des agrégations. La section 1.3.3 explique le principe de l'agrégation de données.

1.3.3 Principe de l'agrégation de données

Une **agrégation** (un agrégat) est obtenue en effectuant une opération, souvent la somme, de plusieurs données détaillées [Lenz & Shoshani 1997; Kimball & Ross 2002]. La somme, le maximum, le minimum, le compte sont des fonctions agrégatives. Un exemple de requête agrégative est : *Quel est le nombre de polices d'assurance-voyage vendues à des clients âgés de moins de 20 ans pour tout le Bas-St-Laurent?* Cette requête se situe à un niveau de granularité supérieur à celui des données d'entrée de l'entrepôt, ne se réfère plus à des données détaillées et nécessite ainsi une opération d'agrégation.

Une méthode efficace pour accélérer le traitement des requêtes agrégatives est le pré-calcul et le stockage du résultat de ces agrégations dans la base de données [Shukla, Deshpande et al. 1996; Kimball & Ross 2002; Greenfield 2005]. Les résultats obtenus portent alors le nom de données agrégées/résumées.

Pour une configuration ROLAP, les agrégations pré-calculées sont enregistrées physiquement (matérialisées) dans la base de données relationnelle. Lorsque le résultat d'une requête agrégative est matérialisé dans la base de données, le serveur ROLAP le recueille directement et l'affiche rapidement sur le module client. Les temps de réponse deviennent alors excellents et prévisibles. Même les analystes les plus impatientes sont à chaque analyse satisfaits des temps de réponse offerts.

Combinaisons de membres

Tous les résultats de requêtes impliquant un membre d'un niveau hiérarchique supérieur dans au moins une dimension et nécessitant alors une opération d'agrégation peuvent être pré-calculés et enregistrés dans la BD.

Pour expliquer le principe de la combinaison de membres, prenons un exemple concret. Un cube, nommé NOMBRE DE POLICES D'ASSURANCE VENDUES, contient 3 dimensions : DÉCOUPAGE ADMINISTRATIF, TYPE D'ASSURANCE et AGE CLIENT. La Figure 1-4 présente les membres des niveaux hiérarchiques de chaque dimension.

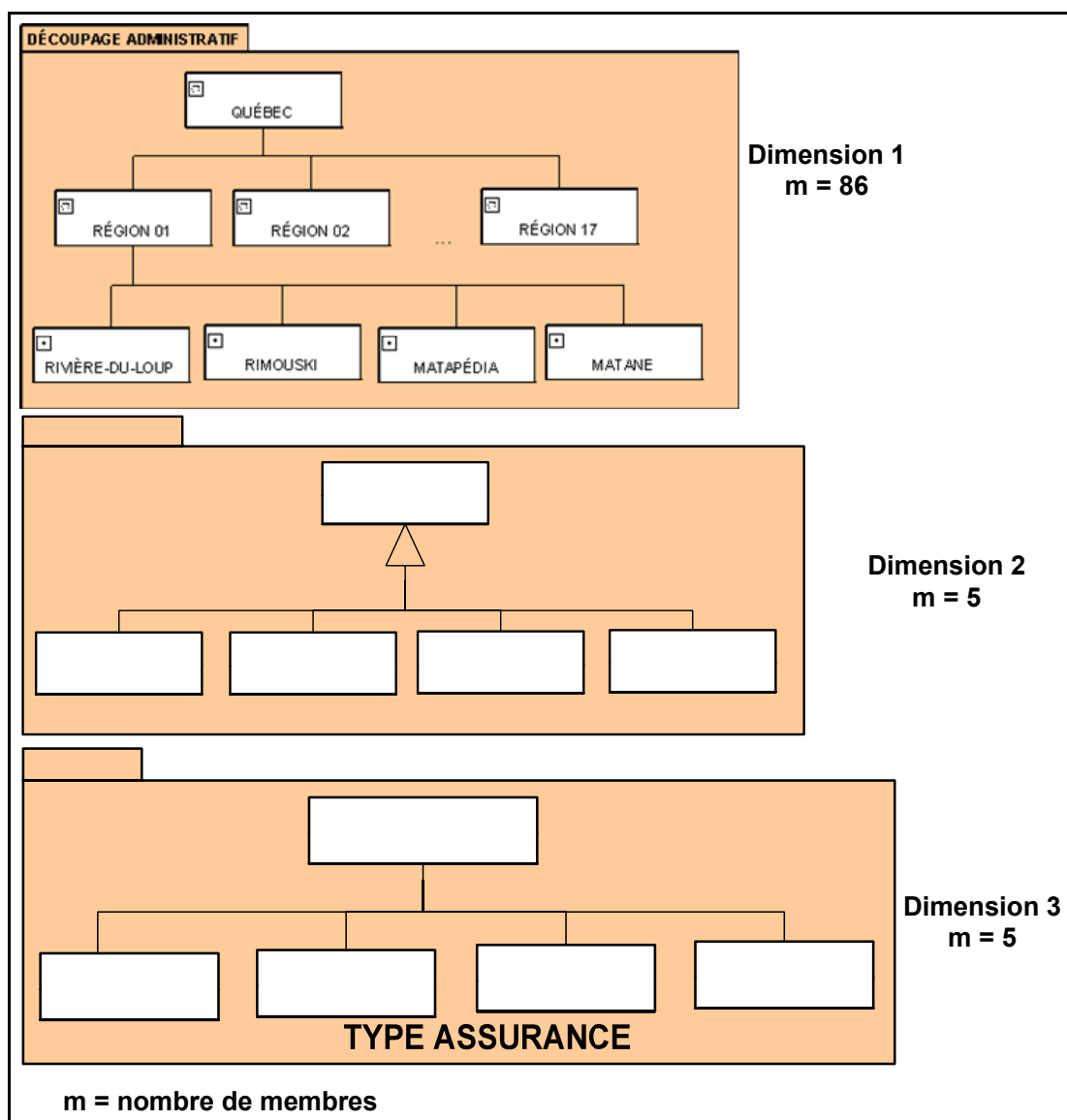


Figure 1-4 : Cube nombre de polices d'assurance vendues.

La méthode de modélisation utilisée se base sur [Lambert 2004]

ASSURANCE

La dimension ASSURANCE possède 5 membres (m) : Toutes les assurances (ASSURANCE), assurance-auto (AUTO), assurance-vie (VIE), assurance-maison (MAISON) et assurance-voyage (VOYAGE). Un exemple d'une combinaison de membres est : RÉGION 01, AUTO, 20 ANS ET MOINS. Pour le cube formé des trois dimensions illustrées sur la Figure 1-4, le nombre total de combinaisons de membres sera : $86 * 5 * 5 = 2150$ combinaisons. Il y a donc 2150 résultats qui peuvent être enregistrés dans la base de données (incluant les données détaillées de l'entrepôt).

AUTO

VIE

MAIS

AGE CLIENT

0 AN À 130 ANS

Le nombre total d'agrégations pré-calculées et stockées dans la base de données est souvent appelé la grandeur du cube (*size of Cube*) [Shukla et al. 1996]. Plus il y aura de résultats de combinaisons stockés dans la base de données, plus le cube sera volumineux.

Agrégations calculées « à la volée »

Selon certains auteurs, il est souvent indésirable de calculer à l'avance toutes les agrégations et les stocker à l'intérieur de la base de données [Pedersen & Nektaria 2001; Kimball & Ross 2002; Hung, Cheung et al. 2004]. Certains estiment que l'espace demandé pour stocker ces résultats est d'environ 200 à 500 fois celui demandé pour les données détaillées [Shukla et al. 1996; The OLAP Report 2005]. Par contre, les nombreux projets effectués au CRG de l'Université Laval (projet SOLAP) donnent des rapports beaucoup plus raisonnables (ex. 30 fois). Certains serveurs ROLAP (ex.: *SQL Serveur de Microsoft*) permettent l'exploitation d'une base de données relationnelle stockant seulement une partie des agrégations. Les agrégations non matérialisées, c'est-à-dire qui n'ont pas été pré-calculées et stockées dans la BD, sont calculées « à la volée ».

Périodiquement, de nouvelles données sont ajoutées dans les systèmes décisionnels. La section suivante présente les deux étapes nécessaires pour actualiser les données d'un cube d'analyse OLAP.

1.3.4 Actualisation des données d'un cube OLAP

Les données présentes dans l'entrepôt proviennent des bases de données transactionnelles (cf. section 1.2.1). Seule une copie de ces données se retrouve dans l'entrepôt. Toutefois, de nouvelles données sont saisies chaque jour avec les systèmes transactionnels. Ces données doivent alors être copiées régulièrement dans l'entrepôt. Ensuite, les agrégations pré-calculées sont actualisées pour que les analyses réalisées sur le module client OLAP tiennent compte de ces nouvelles données.

L'expression **rafraîchissement de l'entrepôt** sera utilisée dans ce mémoire pour signifier l'ajout de nouvelles données dans l'entrepôt (données d'entrée). Plus particulièrement, elle

signifiera l'ajout de nouveaux faits détaillés à l'intérieur de la table des faits ou de nouveaux membres dans les dimensions du modèle multidimensionnel (ex.: l'ajout du nouveau fait « *Le 4 septembre 2005, le nombre de polices d'assurance-vie vendues dans la ville de Matane à des clients âgés de 21 à 40 ans est de 3 polices* » dans la table FAIT_VENTE_ASSURANCE et du nouveau membre « *4 septembre 2005* » dans la dimension TEMPS).

L'expression **reconstruction des cubes** sera utilisée pour signifier l'actualisation des données agrégées suite au rafraîchissement de l'entrepôt de données. Par exemple, l'agrégation fournissant le nombre de polices vendues dans l'ensemble de la région administrative du BAS-ST-LAURENT doit nécessairement être recalculée suite à la vente de nouvelles polices dans la ville de MATANE pour que le nombre total de ventes soient représentatif de la réalité pour l'ensemble de cette région. Pour les systèmes ne supportant pas le calcul « à la volée » des agrégations, la reconstruction des cubes consiste à recalculer le résultat de toutes les combinaisons de membres possibles.

Ces deux traitements seront abordés plus en détail à la section 1.4.

Dans certaines organisations la fréquence de rafraîchissement de l'entrepôt est insuffisante [Langseth 2004]. En effet, pour les outils d'aide à la décision dit traditionnels, le rafraîchissement de l'entrepôt est réalisé périodiquement – soit à tous les mois, à toutes les semaines, voire plus rarement à tous les jours –. Les outils d'aide à la décision temps réel visent à satisfaire les besoins des organisations où les décisions doivent s'effectuer de plus en plus en temps réel. La section 1.4 définit ce nouveau concept retrouvé de plus en plus dans la littérature.

1.4 Aide à la décision temps réel

Un outil **d'aide à la décision temps réel** doit non seulement être facile d'utilisation et répondre rapidement aux requêtes des analystes, mais aussi fournir un accès à l'information courante concernant l'organisation en tout temps et en tous lieux [White 2002]. Pour demeurer compétitives, certaines organisations ne peuvent attendre une semaine voire une

seule journée pour analyser les données les plus récentes et réaliser des comparaisons avec les données historiques [Basu 2003].

Cependant, les besoins en données courantes (parfois appelées données temps réel) diffèrent d'une organisation à une autre. Pour certaines organisations, l'actualisation des données dans le système décisionnel devra se réaliser à toutes les secondes, alors que pour d'autres, une actualisation à toutes les minutes ou à toutes les heures, sera suffisante. En effet, il serait insensé d'implanter un outil d'aide à la décision temps réel pouvant ajouter de nouvelles données à toutes les minutes, si les camions de livraison quittent l'entreprise seulement deux fois par jour.

Certaines applications, comme la gestion des appels dans un centre de support, la détection des fraudes par un corps policier et la gestion des transactions dans les institutions financières tireraient profit d'un outil d'aide à la décision temps réel. Aussi, dans le domaine du commerce, plus particulièrement pour le prix des produits, ou encore, dans le monde de la bourse pour les valeurs des actions, les outils d'aide à la décision temps réel seraient certainement d'une grande utilité.

L'OLAP temps réel est un exemple d'outil d'aide à la décision temps réel. La section 1.4.1 présente l'outil OLAP temps réel. La section 1.4.2 décrit les entrepôts de données temps réel alors que la section 1.4.3 présente quelques méthodes pour actualiser rapidement les agrégations pré-calculées.

1.4.1 Technologie OLAP temps réel

Les systèmes OLAP ont principalement été développés pour exploiter des bases de données subissant rarement des ajouts (*ex.*: une fois par mois), comme les entrepôts de données traditionnels. Le traitement des agrégations pré-calculées est alors réalisé que périodiquement (*ex.*: mensuellement). Ainsi, les outils OLAP traditionnels sont plutôt inadaptés pour exploiter une source de données de courte latence.

En fait, suite à la saisie de nouvelles données à partir des systèmes transactionnels, deux principaux traitements doivent être réalisés avant qu'elles soient disponibles pour des

analyses sur les outils client OLAP: (1) le rafraîchissement de l'entrepôt ou du marché de données), et (2) la reconstruction des cubes.

(1) Rafraîchissement de l'entrepôt de données. Le premier traitement consiste à extraire les données des systèmes transactionnels, les intégrer et les archiver dans l'entrepôt. Ce processus se nomme ETL pour *Extracting, Transforming and Loading*. Un **entrepôt de données temps réel** (EDTR) est utilisé lorsque l'organisation désire que les données courantes soient présentes dans l'entrepôt. Le concept EDTR est expliqué à la section suivante (*cf.* section 1.4.2). Lorsque l'entrepôt possède une structure multidimensionnelle, l'opération d'archivage des données se traduit par l'ajout de nouveaux faits détaillés dans la table des faits et/ou de nouveaux membres dans les dimensions.

La plupart des dimensions d'un modèle multidimensionnel ne subiront que très rarement des ajouts de membres. Habituellement, les dimensions possèdent des membres pré-définis relativement statiques [Kimball & Ross 2002]. Par exemple, la dimension DÉCOUPAGE ADMINISTRATIF contiendra, au niveau RÉGION ADMINISTRATIVE, les 17 régions administratives du Québec. Un membre pourrait être ajouté à ce niveau seulement si une nouvelle région est créée, par exemple. Bien que la plupart des dimensions soient statiques, certaines dimensions risquent de subir des ajouts : nouvelle couverture dans la dimension ASSURANCE, nouveau produit dans la dimension PRODUIT, nouvelle maladie dans la dimension MALADIE. Ces dimensions sont qualifiées de dimensions changeantes lentes (*slowly changing dimension*) [Kimball & Ross 2002]. La dimension TEMPS se distingue des autres dimensions. Si le niveau de granularité de l'entrepôt est la journée, un nouveau membre (*ex.*: 21 septembre 2005) doit être ajouté à chaque nouvelle journée où une saisie de données est réalisée. L'ajout d'un nouveau membre dans une dimension changeante se traduit par l'ajout d'un ou plusieurs enregistrements dans une ou plusieurs tables selon le type de dimensions (étoile, parent-enfant ou flocon). Ces types de dimensions sont décrits dans [Rivest 2000].

(2) Reconstruction des cubes. Pour que les agrégations tiennent compte des nouvelles données ajoutées dans l'entrepôt, elles doivent être recalculées [Mumick, Dallan et al.

1997]. L'étape de reconstruction des cubes doit alors être enclenchée suite à l'archivage des nouvelles données dans l'entrepôt.

Microsoft SQL Server, l'un des principaux systèmes utilisés pour implanter une technologie OLAP, utilise une stratégie qui n'obligera pas le calcul de nouvelles agrégations suite à chaque rafraîchissement de l'entrepôt. L'**OLAP temps réel** permet de « *recupérer, organiser, regrouper et présenter rapidement les données dimensionnelles aux cubes et dimensions à chaque fois que les données de la source de données relationnelle sous-jacente changent et sans avoir besoin de commencer obligatoirement par le traitement du cube ou de la dimension.* » [Kennedy 2003] *Microsoft* est l'un des seuls actuellement qui utilise l'expression *OLAP temps réel*. Selon la définition, la configuration utilisée par OLAP temps réel est la configuration ROLAP. D'autres méthodes pour accélérer le calcul des nouvelles agrégations existent et sont présentées à la section 1.4.3.

1.4.2 Entrepôt de données temps réel

Aucune définition officielle n'existe actuellement dans la littérature, mais typiquement un **entrepôt de données temps réel** (EDTR) est un système qui représente les caractéristiques et la situation réelle de l'organisation. Par exemple, si une requête visant à analyser une facette particulière de l'organisation est lancée sur un EDTR, la réponse représentera l'état réel de l'organisation au moment même où la requête est envoyée [Levy 2005].

Ainsi, contrairement à la plupart des entrepôts de données traditionnels, un EDTR contient des données courantes (données temps réel) sur l'organisation. La fréquence de rafraîchissement joue alors un rôle prédominant dans un EDTR. Généralement, en se basant sur la littérature [Basu 2003; Langseth 2004; Nelson 2004; Levy 2005], un entrepôt qui subit des rafraîchissements plusieurs fois par jour et qui permet ainsi aux décideurs d'avoir accès aux données courantes de l'organisation est un EDTR.

Les besoins en EDTR

Cependant, les organisations ont-elles vraiment besoin d'un EDTR? En effet, si l'on vérifie le type d'analyse effectué avec les outils client OLAP (outils souvent utilisés pour exploiter un entrepôt de données), on réalise que la majorité des requêtes interrogent des données historiques et non des données courantes. Les requêtes sont habituellement : *Combien d'assurance-vie a été vendue durant les 2 derniers mois?* ou encore : *Combien d'assurance-vie a été vendue au cours de ce mois-ci l'année dernière?* Néanmoins, il est certainement intéressant pour les organisations de comparer leurs ventes actuelles avec celles passées : *Combien d'assurance-vie a été vendue au cours de ce mois-ci comparativement à ce mois-ci l'année dernière?* De plus, un sondage réalisé à la *TDWI World Conference in New Orleans* en 2003 démontrait que les organisations avaient l'intention d'augmenter graduellement la fréquence de rafraîchissement de leur entrepôt de données pour ainsi avoir accès à des données plus récentes lors de leurs analyses (cf. Tableau 1-1).

Fréquence de rafraîchissement de l'entrepôt de données	en 2003	Dans 18 mois
Mensuellement	41%	27%
Hebdomadairement	26%	29%
Quotidiennement	75%	72%
Plusieurs fois par jour	2%	14%
Quasi-temps réel	0%	10%

Tableau 1-1 : Besoins en entrepôt temps réel
(The Data Warehousing Institute and Giga Information Group, 2003)

Le Tableau 1-1 démontre qu'en 2003 seulement 2% des organisations réalisaient un rafraîchissement de l'entrepôt plusieurs fois par jour, mais qu'en 2005 cette valeur aurait augmenté à 14%.

Voici deux arguments qui justifient aussi l'utilisation d'un entrepôt de données temps réel [Basu 2003].

1. L'entrepôt de données traditionnel permet d'obtenir des états de l'entreprise à des périodes précises dans le passé (*ex.*: à chaque jour, à chaque semaine ou à chaque mois). Pourtant, certaines applications nécessitent une résolution temporelle plus petite (*ex.*: à chaque heure). Un entrepôt de temps réel permet à l'organisation de retrouver des variantes de son état à l'intérieur même d'une journée.
2. Lorsqu'un volume important de données est saisi dans les systèmes transactionnels au cours d'une même journée (*ex.*: secteur financier), le traitement ETL cause parfois certains problèmes. Le fonctionnement de l'entrepôt temps réel suggère plutôt des ajouts fréquents et réguliers au cours d'une même journée.

Solutions pour implanter un EDTR

Quelques solutions pour faciliter la mise en place d'un EDTR ont été suggérées [Raden 2003; Langseth 2004; The OLAP Report 2005]. Une première solution, le peuplement progressif, propose de peupler directement l'entrepôt de données à partir des systèmes transactionnels. Comme l'illustre la Figure 1-5, il n'y a plus d'étape intermédiaire de préparation entre les systèmes transactionnels et l'entrepôt de données.

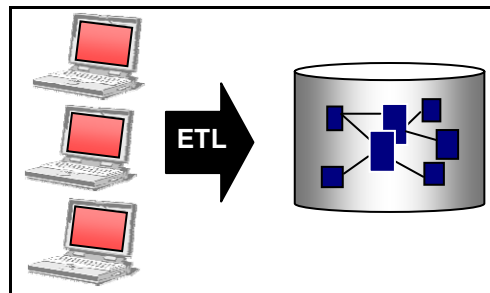


Figure 1-5 : Peuplement progressif

L'entrepôt contiendrait alors les données opérationnelles et non plus leur copie (comme le fait un *Operational Data Store* [Inmon, Imhoff et al. 1996]). Cependant, cette solution peut causer une perte de performance lors de l'exploitation de l'entrepôt par des outils d'analyse.

Une deuxième solution, le peuplement progressif et partitionnement est une variante de la première. Cette solution vise à ne pas peupler directement l'entrepôt de données avec les

données courantes, mais de peupler plutôt une base de données de transfert possédant exactement la même structure que l'entrepôt de données. Ou encore, ne pas peupler la table des faits utilisée dans l'entrepôt de données, mais une autre table des faits ayant la même structure dans l'entrepôt. La base de données de transfert (ou table des faits de transfert) contient seulement les données actuelles. Sur une base quotidienne, les données de la BD de transferts sont copiées et ajoutées dans l'entrepôt de données. L'entrepôt est alors automatiquement rafraîchi. La BD de transfert est vidée pour recevoir les données de la journée suivante. Cependant, lorsque la BD de transfert est volumineuse, plusieurs données doivent être copiées et transférées ce qui entraîne un temps de traitement parfois trop long. De plus, lors du calcul des agrégations, l'outil d'analyse doit pouvoir interroger les données dans la table statique et ensuite ajouter les données courantes de la BD de transfert.

En résumé, un EDTR contient les données les plus récentes, données qui sont rarement retrouvées dans un entrepôt traditionnel. Un entrepôt de données traditionnel peut aussi contenir les données courantes de l'organisation s'il permet des rafraîchissements plus fréquents (comme un *Operational Data Store*). En effet, une autre possibilité est de conserver l'architecture traditionnelle de l'entrepôt et y ajouter simplement les nouvelles données à une fréquence plus élevée, par exemple passer d'une fréquence hebdomadaire à une fréquence quotidienne ou bien d'une fréquence quotidienne à des ajouts à chaque heure. Cette solution permet de se rapprocher graduellement d'un EDTR et répondre aux besoins de plusieurs organisations. Suite au rafraîchissement de l'entrepôt, l'opération de reconstruction des cubes (calcul des nouvelles agrégations) doit être enclenchée. La section 1.4.3 présente quelques méthodes pour actualiser rapidement les agrégations pré-calculées.

1.4.3 Actualisation rapide des agrégations

Suite au rafraîchissement de l'entrepôt, les agrégations pré-calculées doivent être actualisées (reconstruction des cubes). Évidemment, les agrégations calculées « à la volée » tiendront automatiquement compte des nouveaux ajouts effectués dans l'entrepôt. Cependant, dans les applications exploitant un entrepôt de données volumineux, il est important de matérialiser certaines agrégations pour respecter le temps de réponse minimal (< 10 secondes) aux requêtes-client.

Première méthode

Une méthode pour accélérer la reconstruction des cubes suite au rafraîchissement de l'entrepôt vise à matérialiser seulement (1) les requêtes agrégatives les plus fréquentes et (2) les requêtes agrégatives les plus coûteuses en temps de calcul [Kimball & Ross 2002; Hung et al. 2004].

(1) Matérialiser les requêtes agrégatives les plus fréquentes. Au départ, les requêtes agrégatives qui risquent d'être les plus souvent demandées par l'utilisateur doivent être identifiées, car seul le résultat de ces requêtes sera stocké dans la BD. Ainsi, le temps de réponse minimal aux requêtes peut, dans la plupart des cas, être respecté. Les requêtes plus rares sont calculées « à la volée ». Ce procédé constitue une base d'apprentissage a priori. Des techniques d'intelligence artificielle (IA) [Russell & Norvig 2002; Luger 2004] peuvent ensuite être employées pour enrichir ce processus au cours de l'utilisation du système. Des requêtes non matérialisées dans la base de données, mais demandées fréquemment par l'utilisateur, sont alors automatiquement enregistrées dans la base de données et le résultat des requêtes n'étant pas ou plus demandées est alors supprimé.

(2) Matérialiser les requêtes agrégatives les plus coûteuses. Les requêtes agrégatives les plus coûteuses en temps de calcul doivent être identifiées. Le résultat de ces agrégations doit être stocké dans la base de données ce qui permet de diminuer le temps de réponse aux requêtes. Le calcul « à la volée » de ces requêtes demanderait probablement un temps d'attente trop long.

Par exemple, à la Figure 1-6, la dimension PRODUIT possède deux familles – NOURRITURE et BOISSON –. La famille nourriture contient cinq produits situées au niveau le plus détaillé de la dimension (VIANDE, POISSON, FRUIT DE MER, FRUIT et LÉGUME) et la famille BOISSON en regroupe trois (JUS, BOISSON GAZEUSE, ALCOOL). Dans ce cas-ci, l'agrégat concerne seulement cinq et trois produits pour obtenir le nombre de produits d'épicerie vendu selon les familles. Le résultat de ces deux agrégations est rapidement calculé « à la volée ».

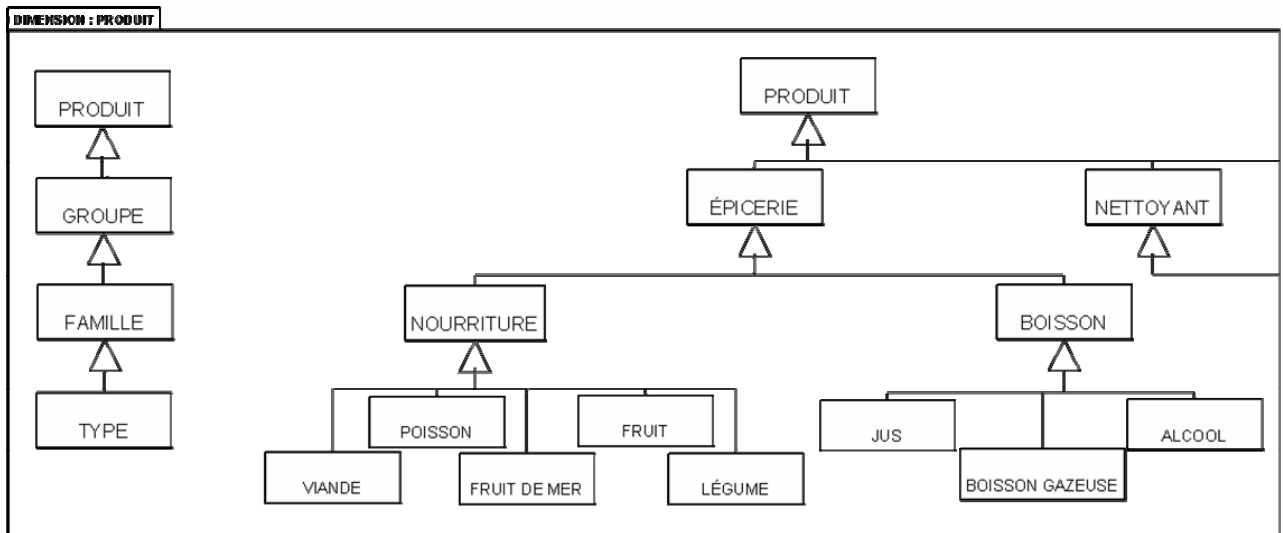


Figure 1-6 : Niveaux et membres de la dimension PRODUIT.
La méthode de modélisation utilisée se base sur [Lambert 2004]

Par contre, lorsque le niveau détaillé compte 10 000 produits, l'agrégation au niveau supérieur est évidemment plus longue à calculer, et il devient avantageux de pré-calculer et stocker le résultat dans l'entrepôt. Des méthodes d'IA pourraient ici aussi être utilisées pour identifier automatiquement les agrégations les plus longues à calculer.

La première méthode permet de diminuer le nombre de données agrégées. Ainsi, seulement quelques agrégations devront être recalculées suite au rafraîchissement de l'entrepôt. *Microsoft SQL Server* utilise une stratégie semblable lors de la construction de ses cubes.

Cette première méthode suppose évidemment que le système d'analyse supporte le calcul « à la volée » des agrégations. Dans une configuration ROLAP ne supportant pas le calcul « à la volée » des agrégations, le résultat de toutes les combinaisons de membres possibles (cf. section 1.3.3) doit alors être recalculé suite au rafraîchissement de l'entrepôt.

Deuxième méthode

Peu de méthodes sont suggérées dans la littérature pour faciliter la reconstruction des cubes lorsque le système d'analyse ne supporte pas le calcul « à la volée » des agrégations. Une des rares méthodes, *Lazy aggregates*, vise à identifier le moment critique où les agrégations

doivent être rafraîchies pour ne pas les recalculer à chaque nouvel ajout [Kiviniemi, Wolski et al. 1999]. Le moment critique est l'instant où les données ne sont plus représentatives de la réalité. Pour y parvenir, des seuils de tolérances sont établis et indiquent le moment exact où le calcul des nouvelles agrégations est nécessaire. En effet, maintenir les agrégations entièrement représentatives de la réalité en recalculant les nouvelles agrégations à chaque rafraîchissement de l'entrepôt est extrêmement difficile et parfois inutile. Par exemple, la vente d'un seul produit (*ex.*: un fruit) n'implique pas une grande différence sur le nombre total de produits d'épicerie vendus. De plus, les utilisateurs des systèmes OLAP recherchent souvent des ordres de grandeur, des tendances, des patrons ou des comportements lorsqu'ils exécutent leurs requêtes. Cependant, cette méthode ne peut être utilisée dans les applications nécessitant des valeurs précises et exactes.

Bref, en base de données décisionnelle, le qualificatif « temps réel » signifie que les décideurs ont accès aux données courantes (temps réel) en tout temps lors de leurs analyses. Il est d'abord utilisé pour signifier que de nouvelles données sont ajoutées plus d'une fois par jour dans l'entrepôt (entrepôt de données temps réel). Ensuite, il signifie que, suite au rafraîchissement de l'entrepôt, les nouvelles données doivent être traitées rapidement (OLAP temps réel).

Les systèmes décisionnels traités jusqu'à maintenant n'exploitent pas la référence spatiale des données à son plein potentiel. Pourtant, dans plusieurs applications, l'ajout d'un volet cartographique sur l'interface utilisateur de l'outil d'analyse permet d'obtenir de nouveaux résultats d'analyse plus rapidement et plus intuitivement. La section 2.1 traite alors d'un système décisionnel né d'un amalgame entre les outils OLAP et les systèmes d'information à référence spatiale, le Spatial OLAP (SOLAP).

1.5 Conclusion

Au début de ce chapitre nous avons défini l'expression « temps réel » à partir des définitions offertes dans le domaine informatique. Nous avons ensuite constaté que les systèmes transactionnels temps réel, plus particulièrement les SGBD temps réel, possèdent en général les mêmes caractéristiques qu'un système temps réel.

Cependant, dans le domaine des outils d'aide à la décision, le qualificatif « temps réel » vise principalement à distinguer les outils pouvant offrir les données les plus récentes sur l'organisation lors des analyses. Pour ce faire, le fonctionnement des entrepôts de données temps réel (EDTR) et des OLAP temps réel diffère de ceux dits traditionnels. Les EDTR contiennent les données courantes présentes dans les systèmes transactionnels principalement grâce à des rafraîchissements plus fréquents (plus d'une fois par jour). La fréquence d'actualisation au cours d'une journée est directement choisie en fonction des besoins en données courantes de l'application. L'expression « temps réel » ne qualifie alors pas un système nécessairement rapide et offrant des temps de réponse de l'ordre de la seconde [White 2002; Nelson 2004]. L'OLAP temps réel est défini comme étant un système optimisé pour exploiter une source de données latente et permettant aux utilisateurs d'avoir accès rapidement aux nouvelles données, mais aucune contrainte temporelle à respecter lors de l'exécution de ces opérations ne semble être spécifié.

Bref, l'utilisation actuelle du qualificatif « temps réel » en référence aux bases de données décisionnelles diffère de celle retrouvée en informatique et présentée à la section 1.1. L'utilisation de l'expression « temps réel » dans le domaine des outils d'aide à la décision est alors parfois abusive [White 2002; Raden 2003; White 2004]. En fait, le qualificatif *real time* (temps réel) devrait être parfois remplacé par celui de *right time* (temps propice) [Eckerson 2004; White 2004]. En effet, un outil d'aide à la décision temps réel fournit plutôt l'information exacte, aux bonnes personnes au bon moment.

C'est donc pour ces raisons que nous allons principalement nous baser sur la définition et les caractéristiques offertes à section 1.1 d'un système temps réel pour apporter le nouveau concept de SOLAP temps réel, ce qui permet de cadrer nos concepts dans le courant

principal de la terminologie informatique. De plus, les définitions précédentes de « EDTR » et « OLAP temps réel » sont très récentes, peu connues et davantage commerciales que scientifiques. La synthèse des concepts temps réel retrouvés dans le domaine de l'aide à la décision nous a tout de même permis d'identifier quelques techniques qui seront utilisés lors la conception et l'implantation d'un tel système.

Le deuxième chapitre présente différentes approches pour répondre aux besoins d'une application nécessitant d'analyser facilement et rapidement les nouvelles données spatiales et descriptives recueillies.

2 LE TEMPS RÉEL DANS UN CONTEXTE SOLAP

Le premier chapitre a réalisé une revue des concepts temps réel retrouvés dans le domaine des bases de données décisionnelles. Le chapitre 2 apporte de nouveaux concepts SOLAP visant à répondre aux besoins temps réel de certaines organisations. Tout d'abord, le SOLAP traditionnel est présenté à la section 2.1. Ensuite, les nouveaux concepts SOLAP pour répondre aux besoins temps réel de certaines applications, dont l'analyse de la performance des athlètes, sont apportés à la section 2.2.

2.1 Technologie SOLAP traditionnelle

Au cours des dernières années, de nouveaux systèmes pour l'exploration et la visualisation des données spatio-temporelles ont fait leur apparition. Les premiers essais visant à regrouper les avantages d'un Système d'Information Géographique (SIG) avec un outil OLAP ont été réalisés au Centre de Recherche en Géomatique (CRG) de l'Université Laval [Rivest 2000] ainsi qu'à l'Université Simon Fraser [Han, Koperski et al. 1997] et à l'Université du Minnesota [Lu, Kou et al. 2003]. Ces expérimentations ont donné naissance à l'**OLAP Spatial**, baptisé SOLAP par [Bédard et al. 1997].

La section 2.1.1 définit les outils OLAP Spatiaux. L'agrégation spatiale, concept propre aux OLAP Spatiaux, est expliquée à la section 2.1.2.

2.1.1 Définition

[Bédard 2004] définit l'outil SOLAP comme étant « *un logiciel de navigation rapide et facile dans les bases de données spatiales qui offre plusieurs niveaux de granularité* »

d'information, plusieurs thèmes, plusieurs époques et plusieurs modes de visualisation synchronisés ou non: cartes, tableaux et graphiques statistiques ».

Lors de l'implantation de l'outil SOLAP développé au Centre de Recherche en Géomatique de l'Université Laval, la configuration la plus utilisée jusqu'à maintenant est le ROLAP. Cette configuration exploite une base de données relationnelle conventionnelle facilitant ainsi la gestion de la géométrie des données. L'architecture générale d'un SOLAP utilisant une configuration ROLAP est alors formée (1) d'une base de données relationnelle; (2) d'un serveur SOLAP et; (3) d'un client SOLAP.

- (1) La **base de données relationnelle** contient des données descriptives et des données géométriques et possède une structure multidimensionnelle. Les données géométriques représentent les membres de la dimension dite spatiale. La base de données est formée de données détaillées provenant d'un entrepôt de données spatiales ou correspond à l'entrepôt lui-même. Elle contient aussi les données agrégées. La structure multidimensionnelle et les données (spatiales et non-spatiales) agrégées caractérisent un cube d'analyse SOLAP.
- (2) Le **serveur SOLAP** assure le lien entre le client SOLAP et la base de données relationnelle et gère les données des cubes SOLAP.
- (3) Le **module client SOLAP** possède en général les mêmes caractéristiques qu'un module OLAP mais offre en plus un volet spatial qui diffuse la géométrie de la dimension spatiale. Bien qu'il soit possible avec un module client OLAP d'obtenir le nombre de polices d'assurance vendues pour chacune des régions administratives du Québec, il est intéressant de pouvoir constater rapidement, par le biais d'une carte, que les régions où il y a le moins de ventes de polices se situent principalement dans l'Est de la province, par exemple. Constater des faits semblables est possible lorsqu'on dispose et visualise les données cartographiques, c'est-à-dire les données géométriques associées à la dimension spatiale du cube. De nombreux exemples d'applications SOLAP sont fournis dans [Bédard et al. 2005].

Dans le domaine de la recherche, d'autres formes d'OLAP spatiaux ont aussi été mises en place. L'OLAP spatial *MapCube* [Lu et al. 2003] est un système de visualisation de haute performance pour les données sur le trafic des voitures. L'entrepôt de données contient les dimensions TEMPS (niveaux : heure, jour, mois, année) et ESPACE (station, route, autoroute, région) avec les mesures VOLUME et OCCUPATION. Un autre système de type OLAP spatial, le *GeoMiner* [Han et al. 1997], offre une interface de visualisation interactive, facile d'utilisation pour la découverte de connaissances spatiales.

Une différence principale entre un outil OLAP et un outil OLAP Spatial porte sur le calcul des agrégations, notamment en tenant compte des particularités propres à l'aspect spatial des données.

2.1.2 Agrégation spatiale

L'agrégation spatiale est dérivée du principe de l'agrégation expliqué à la section 1.3.3. Lors du calcul des agrégats spatiaux, la valeur (mesure) d'un membre à un niveau général est obtenue en effectuant la somme (min, max, etc.) des membres du niveau plus détaillé. L'agrégation spatiale est basée sur un découpage territorial. Ce découpage territorial correspond à la dimension spatiale du cube et peut être fixe (*ex.*: découpage administratif de la province de Québec), ou variable (l'utilisateur dessine une zone rectangulaire sur l'interface cartographique). Les résultats d'agrégats spatiaux d'un cube possédant un découpage territorial fixe peuvent être calculés à l'avance pour chaque niveau hiérarchique de la dimension spatiale. Un découpage territorial variable exige que les agrégations soient à chaque fois calculées « à la volée » [Papadias, Kalnis et al. 2001].

Les OLAP Spatiaux utilisent la puissance des outils OLAP pour la navigation et l'exploration des bases de données spatiales et en plus, ils proposent des nouvelles méthodes d'exploitation de cette information spatiale. Le module client SOLAP permet l'exploration facile et rapide des données dans l'espace et le temps grâce à leur interface utilisateur intuitive. Cependant, les OLAP Spatiaux [Han et al. 1997; Lu et al. 2003; Bédard 2004] ne permettent pas à l'heure actuelle d'ajouter fréquemment et rapidement les

nouvelles données saisies au niveau des systèmes transactionnels. Les analyses ne tiennent alors pas compte des données actuelles et ne reflètent habituellement pas la situation réelle de l'organisation. Pourtant, si la requête suivante est lancée sur un module client SOLAP : *Quel est le nombre total de voitures à l'intérieur d'un district?* la réponse fournie par l'outil doit correspondre à la réalité. La section 2.2 apporte les nouveaux concepts SOLAP établis pour répondre aux besoins en temps réel des organisations.

2.2 Adaptation du SOLAP pour des besoins temps réel

La section 2.2.1 présente les différences entre les expressions « temps réel » (*cf.* section 1.1), « temps quasi-réel » et « juste-à-temps ». Ces définitions seront utilisées pour situer les nouveaux concepts : SOLAP temps réel, SOLAP temps quasi-réel, SOLAP temps propice et SOLAP juste-à-temps qui seront présentées à la section 2.2.2.

2.2.1 Concepts temporels

Temps réel vs temps quasi-réel

Dans le domaine informatique, l'expression **temps quasi-réel** est habituellement utilisée pour qualifier les systèmes qui ne répondent pas à toutes les exigences d'un *système temps réel* (*cf.* section 1.1). L'importance accordée à la contrainte temporelle est le principal critère qui distingue ces deux types de système (sévère versus souple)[Eriksson et al. 2004; Wikipédia 2005].

Les **systèmes temps réel** offrent dans toutes les occasions, sans exception, un temps de réponse respectant une contrainte temporelle bien précise. Ils ne tolèrent aucun dépassement. Les applications utilisant des systèmes temps réel requièrent souvent des

traitements en temps réel (temps réel dur⁶). Des exemples de systèmes temps réel sont les systèmes de contrôle aérien et les systèmes de contrôle automatique des trains.

Dans les **systèmes temps quasi-réel**, si la contrainte temporelle n'est pas respectée, une erreur n'est pas automatiquement générée par le système. En fait, la contrainte temporelle des systèmes temps quasi-réel n'est pas une valeur fixe et précise. Elle est plutôt une valeur approximative. Les applications utilisant des systèmes temps quasi-réel exigent souvent des traitements en temps quasi-réel (temps réel mou⁷). Des exemples de systèmes temps quasi-réel sont les systèmes téléphoniques et les jeux en réseau.

Prenons un exemple avec un guichet automatique. Il doit, dans la plupart des cas, offrir un temps de réponse acceptable pour le consommateur (inférieur à la contrainte temporelle), autrement le système n'est pas fiable et peut rendre le consommateur insatisfait. Si, lors d'une seule transaction, le système offre au client un temps de réponse 10%, 20% ou 50% moins rapide que prévue (supérieur à la contrainte temporelle), le problème est petit. Le guichet automatique est un exemple de système temps quasi-réel.

Juste-à-temps

L'office de la langue française définit l'expression **juste-à-temps** (JAT) ou *just-in-time* (JIT) dans un seul domaine d'application, la gestion. L'expression juste-à-temps signifie alors: « *Méthode de gestion de la production en flux tendu qui consiste à coordonner le système de production en fonction des commandes et non des stocks, en produisant ou en achetant la quantité juste nécessaire au moment précis où on en a besoin, et ce, à chaque étape du processus.* » [OLF 2005].

En se basant sur cette définition, l'expression juste-à-temps, dans le monde des bases de données décisionnelles, pourrait alors signifier : *Méthode de gestion des données qui vise à réduire les traitements et à permettre à l'utilisateur de choisir l'instant où l'ajout de*

⁶ Correspond à des traitements instantanés, c'est-à-dire que le décalage entre l'entrée des données de l'émission des résultats est de moins de 0,5 sec [OLF 1999]

⁷ Correspond à des traitements qui se réalisent avec un léger décalage (environ 0,5 sec) [OLF 1999]

nouvelles données (ou seulement une partie des données) sera réalisée pour ainsi les analyser qu'au moment où il en a besoin. Un système juste-à-temps fournit les nouvelles informations qu'au moment où les analystes en ont réellement besoin. S'il n'est pas nécessaire dans un domaine d'application que les nouvelles données soient ajoutées toutes les secondes, il est inutile de dépenser du temps, des efforts et des ressources pour mettre en place un tel système.

La section 2.2.1 nous a permis de constater que la principale différence entre un système temps quasi-réel et temps réel est l'importance qu'il accorde à la contrainte temporelle. Nous tiendrons compte de cette caractéristique pour distinguer les outils SOLAP temps réel et temps quasi-réel. Quatre nouveaux concepts sont donc apportés dans la section suivante : (1) SOLAP temps réel, (2) SOLAP temps quasi-réel, (3) SOLAP temps propice et (4) SOLAP juste-à-temps.

2.2.2 Catégories de SOLAP

Dans le fonctionnement global d'un outil SOLAP implanté selon une configuration ROLAP, deux principales étapes doivent être réalisées pour que de nouvelles données soient ajoutées dans les cubes d'analyse (1) Acquisition de données spatiales et non-spatiales et (2) Rafraîchissement de l'entrepôt et la reconstruction des cubes.

(1) Acquisition et traitement de données spatiales et non-spatiales. Les nouvelles données spatiales sont d'abord collectées et traitées avant d'être transférées dans une base de données. La saisie de toutes autres informations (données non-spatiales) est souvent réalisée à partir de systèmes transactionnels pour ensuite être transférées dans une base de données. Cette dernière est habituellement implantée selon un modèle transactionnel pour faciliter les opérations de saisie, de recherche et de mise à jour.

(2) Rafraîchissement de l'entrepôt et la reconstruction des cubes. Lorsque de nouvelles données sont saisies à partir des systèmes transactionnels et transférés dans la structure transactionnelle, elles doivent ensuite être intégrées et archivées dans une base de données possédant une structure multidimensionnelle (*ex.*: entrepôt de données spatiales). L'opération d'archivage dans la structure multidimensionnelle se traduit par

l'ajout de nouveaux faits dans la table des faits et de nouveaux membres dans les dimensions. Suite à ces opérations, les nouvelles agrégations des cubes d'analyse SOLAP peuvent être calculées.

La structure multidimensionnelle (entrepôt de données spatiales) exploitée par un serveur SOLAP doit contenir la géométrie (surface, ligne, point) de chaque membre de la dimension spatiale. La collecte de nouvelles données spatiales implique qu'un nouveau membre doit être ajouté dans la dimension spatiale, mais aussi qu'une nouvelle géométrie doit être ajoutée dans l'entrepôt. Même si une dimension spatiale (*ex.*: DECOUPAGE ADMINISTRATIF) subira que très rarement des ajouts de membres, certaines applications (*ex.*: projet en archéologie [Rageul 2004], projet en sport (*cf.* chapitre 3)), devront tout de même permettre des ajouts fréquents de membres dans la dimension spatiale (*ex.*: ajout d'un nouveau lot de fouilles archéologiques ou l'ajout d'un nouvel entraînement avec ses milliers de positions calculées).

Ces deux étapes présentes dans le fonctionnement global d'un outil SOLAP sont effectuées différemment selon la catégorie de SOLAP. L'étape de l'acquisition et du traitement des nouvelles données spatiales et non-spatiales relève davantage des systèmes transactionnels que des systèmes décisionnels et ne sera que légèrement abordée dans la description des catégories d'outils SOLAP. De plus, l'objectif de cette section n'est pas d'identifier la technologie ou la meilleure méthode pour rafraîchir l'entrepôt et reconstruire les cubes. Elle vise plutôt à définir les critères que chaque approche doit respecter ainsi que leur fonctionnement.

SOLAP temps réel (SOLAP TR)

Dans un **SOLAP temps réel** le rafraîchissement de l'entrepôt et la reconstruction des cubes sont réalisés immédiatement suite à l'acquisition de nouvelles données. Le SOLAP TR est donc un système réactif. Il fournit aussi des réponses exactes tout en respectant le temps de réponse pré-établi (contrainte temporelle). Cette contrainte est établie directement en fonction de la fréquence de l'acquisition des nouvelles données. Lorsque la fréquence est

très élevée (*ex.*: à toute les secondes) le rafraîchissement de l'entrepôt et la reconstruction des cubes devront nécessairement être des traitements en temps réel (<0,5 sec).

Le SOLAP temps réel pourrait être utilisé dans des applications comme la gestion du trafic aérien ou du trafic routier. Par exemple, un SOLAP TR pourrait représenter en temps réel la situation de congestion retrouvée sur le réseau routier et permettrait aux analystes d'exploiter facilement et rapidement les données, de visualiser les résultats de leurs requêtes sur des cartes et de diriger les véhicules médicaux sur les chemins les plus adéquats.

SOLAP temps quasi-réel (SOLAP TQR)

Un **SOLAP temps quasi-réel** est un système réactif qui fournit des réponses exactes, mais qui peut, en certaines occasions, ne pas respecter la contrainte temporelle. Les algorithmes ou la technologie utilisée ne permettent pas d'atteindre le niveau de prédictibilité exigé pour un SOLAP TR. Un SOLAP TQR est alors plus souple qu'un SOLAP TR. De plus, suite à l'acquisition des nouvelles données, les temps de traitement sont généralement plus longs que ceux offerts par un SOLAP TR. La performance plus faible du système peut se situer à deux niveaux : soit, le système gère moins efficacement les nouvelles données transmises (rafraîchissement de l'entrepôt), soit la reconstruction des cubes est trop exigeante. Toutefois, ces deux opérations doivent généralement être des traitements en temps quasi-réel (~0,5 sec) puisqu'elles sont réalisées immédiatement suite à l'acquisition de nouvelles données.

Une application possible de l'utilisation d'un SOLAP TQR est la compétition de cyclisme sur route. Les cyclistes portent un GPS qui transmet la position instantanée en temps réel à un ordinateur. Les nouvelles données acquises sont ensuite traitées, intégrées et transférées dans l'entrepôt de données (*étape 2*). Cette opération déclenche l'opération de reconstruction des cubes de données et les agrégations sont alors recalculées avec les données nouvellement recueillies (*étape 3*). Le traitement, débutant à *l'étape 1* et se terminant à *l'étape 3*, doit s'effectuer en un certain temps minimal (~0,5 sec), mais il est permis au système de parfois dépasser cette contrainte temporelle (ce que ne permet pas un

SOLAP TR). Ainsi, l'entraîneur ou même les spectateurs peuvent analyser les performances des athlètes en temps quasi-réel.

D'autres applications possibles seraient un SOLAP TQR présentant les températures, les précipitations ou les activités volcaniques sur l'ensemble du pays. Ou encore, un SOLAP TQR fournissant l'enfoncement des navires en fonction des marées, des courants, de leur vitesse, de leur forme et de leur poids.

SOLAP temps propice (SOLAP TP)

L'expression anglaise *right time* (ici traduit par l'expression « temps propice ») est utilisée pour qualifier un SOLAP dont l'étape du rafraîchissement de l'entrepôt et de la est simplement réalisée plus d'une fois par jour (*ex.*: à toutes les heures), mais de façon totalement automatique. Ainsi, même si de nouvelles données sont acquises, l'entrepôt ne sera pas immédiatement rafraîchi. La fréquence des rafraîchissements est directement établie en fonction des besoins de l'application. La durée des traitements dans un SOLAP TP doit simplement être inférieure à la fréquence des ajouts (*ex.*: moins d'une heure). Ainsi, le SOLAP TP exige généralement des temps de réponse moins rapide qu'un SOLAP TR ou un SOLAP TQR. Cependant, si l'application nécessite des ajouts à toutes les secondes, un SOLAP TR ou un SOLAP TQR doivent alors être utilisés.

SOLAP juste-à-temps (SOLAP JAT)

Le rafraîchissement de l'entrepôt et la reconstruction des cubes dans un SOLAP juste-à-temps sont réalisés seulement suite à la demande de l'utilisateur. Il est utilisé dans les applications ne réalisant pas l'acquisition de nouvelles données à une fréquence fixe et lorsque cette acquisition est généralement réalisée en temps différé. Le temps de traitement demandé pour le rafraîchissement de l'entrepôt et la reconstruction des cubes doit simplement satisfaire les besoins de l'application.

Reprenons l'application en cyclisme sur route. Considérons le cas où l'étape d'acquisition des données est réalisée en temps différé. Ce cas implique que les données GPS sont enregistrées à l'intérieur du récepteur situé sur le vélo et qu'elles sont transférées seulement

une fois la course terminée. Une fois les nouvelles données transférées, le SOLAP JAT ajoutera ces nouvelles données dans l'entrepôt seulement suite à la demande de l'utilisateur, mais de façon totalement automatique. Un SOLAP JAT peut alors être utilisé dans une application qui n'exige tout simplement pas que le rafraîchissement de l'entrepôt et la reconstruction des cubes soient réalisés immédiatement après l'acquisition des données. Dans l'exemple du vélo de route, l'athlète ne réalisera pas des analyses sur sa performance durant la compétition, de même que l'entraîneur puisqu'il lui est habituellement interdit de communiquer avec l'athlète durant la compétition.

L'expression « juste-à-temps » utilisée pour qualifier une approche SOLAP n'est pas synonyme de l'expression « à la volée ». Le SOLAP *juste-à-temps* développé dans le cadre de cette recherche calcule et matérialise les résultats d'agrégations dans la base de données (création de données persistantes). En effet, le fonctionnement actuel des outils SOLAP développés au CRG exige que toutes les agrégations du cube d'analyse soient pré-calculées et stockées dans la base de données relationnelle. Le calcul « à la volée » des agrégations nécessite à chaque fois le calcul des agrégations puisque le résultat n'est pas enregistré dans la base de données.

La Figure 2-1 positionne les cinq catégories de SOLAP : SOLAP TR, SOLAP TQR, SOLAP TP, SOLAP JAT et SOLAP traditionnel. Leur position est choisie en fonction du mode **d'acquisition de données spatiales et non-spatiales** et la fréquence du **rafraîchissement de l'entrepôt et la reconstruire les cubes** d'analyse du SOLAP.

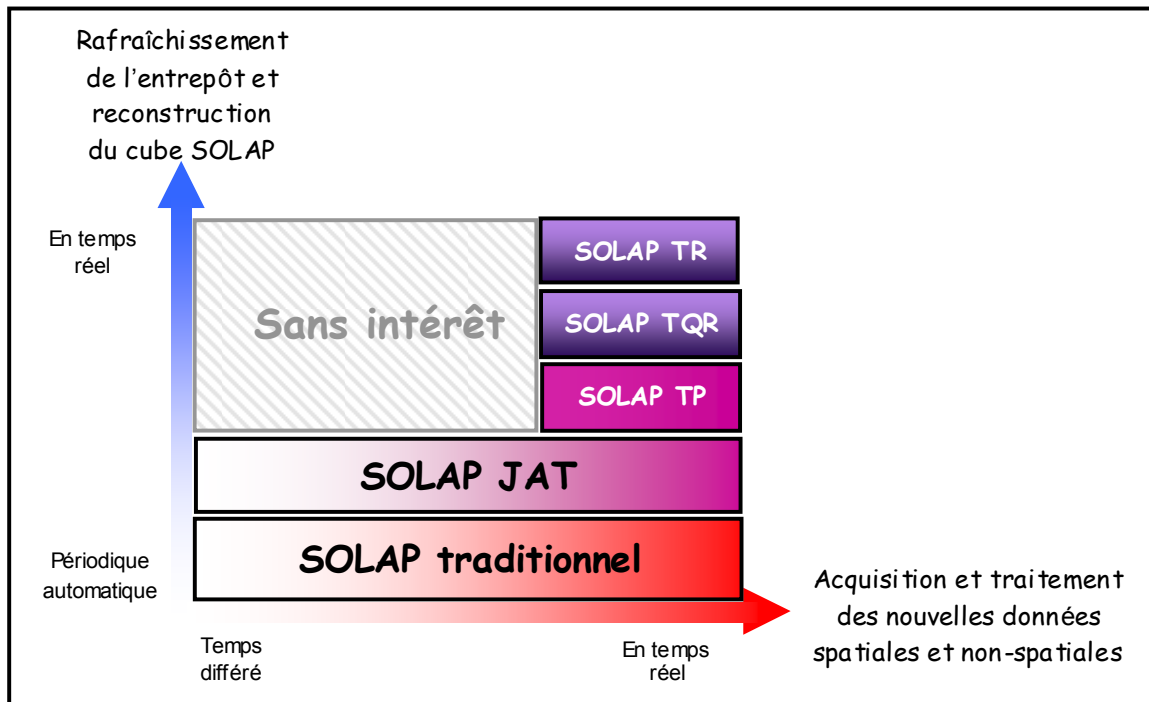


Figure 2-1 : SOLAP temps réel, temps quasi-réel, temps propice, juste-à-temps et traditionnel

Comme le montre le graphique de la Figure 2-1, selon le type de SOLAP, l'étape de **l'acquisition et le traitement des nouvelles données spatiales et non-spatiales** est soit réalisée au cours de l'événement (en temps réel et instantanément) soit après l'événement et l'acquisition totale des données (temps différé). Le meilleur exemple est encore le cyclisme sur route. Si les données GPS sont transférées au fur et à mesure dans la base de données pendant la course, l'acquisition et le traitement des données sont alors réalisés en temps réel. Si les données GPS sont transférées suite à la collecte complète des données (après la course), cette étape est réalisée en temps différé.

L'étape du **rafraîchissement de l'entrepôt et de la reconstruction des cubes** est soit réalisée pendant la collecte (en temps réel et instantanément), soit réalisée périodiquement (quotidien, mensuel, annuel). La section intermédiaire correspond aux situations où le rafraîchissement et la reconstruction des cubes sont effectués périodiquement, mais plus d'une fois par jour (SOLAP TP) ou en tout temps, mais seulement suite à la demande de l'utilisateur (SOLAP JAT).

La section suivante présente en détail les différences entre un SOLAP temps réel et un SOLAP juste-à-temps.

2.2.3 SOLAP TR vs SOLAP JAT

Cette section compare à l'aide de différents diagrammes (cas d'utilisation, diagramme d'activités et diagramme de séquence) le fonctionnement d'un SOLAP temps réel et du SOLAP juste-à-temps. Le cas d'utilisation permet de structurer les besoins des utilisateurs et les objectifs correspondants d'un système, alors que le diagramme d'activités montre l'activité et le fonctionnement du système.

La Figure 2-2 présente le cas d'utilisation pour un **SOLAP TR**.

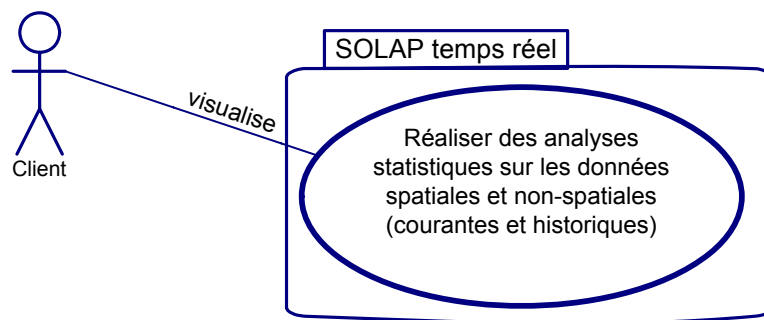


Figure 2-2 : Cas d'utilisation d'un SOLAP temps réel

Une seule catégorie d'acteur se retrouve sur la Figure 2-2 et entre dans le fonctionnement d'un SOLAP TR. L'acteur *CLIENT* réalise seulement des analyses statistiques sur des données spatiales et descriptives. Ces données sont soit des données nouvellement recueillies et immédiatement ajoutées dans la base de données (données courantes), soit des données recueillies et ajoutées auparavant. L'acteur *CLIENT* représente, par exemple, un entraîneur de vélo sur route qui analyse les nouvelles données spatiales et non-spatiales en temps réel au cours d'une compétition.

Le cas d'utilisation d'un **SOLAP juste-à-temps** contient aussi un seul acteur le *Client*. La Figure 2-3 présente le cas d'utilisation d'un SOLAP JAT.

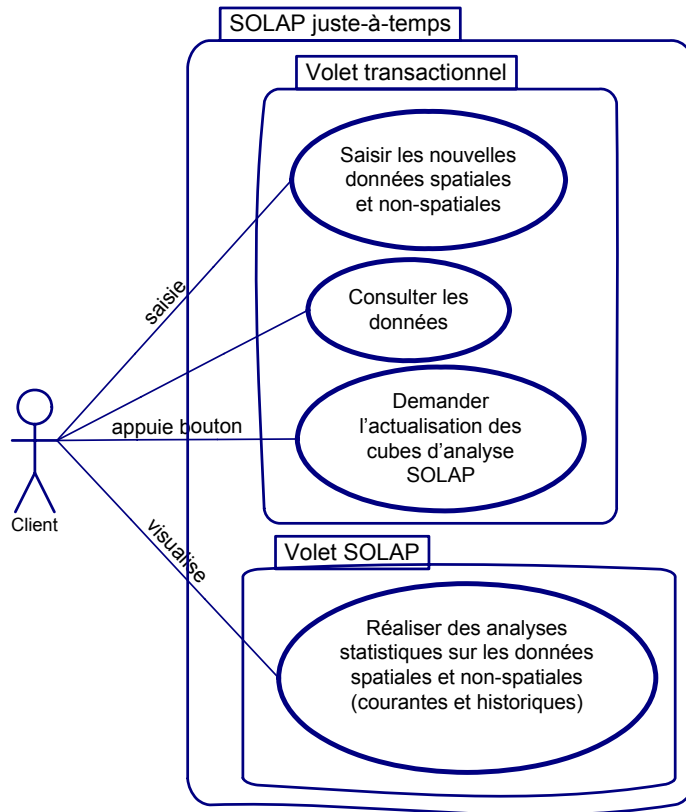


Figure 2-3 : Cas d'utilisation SOLAP juste-à-temps

Le cas d'utilisation du SOLAP JAT possède deux principaux paquetages : le premier représente le volet SOLAP et le deuxième le volet transactionnel. Ils sont utilisés pour répondre à des besoins bien distincts. Sur le volet transactionnel, l'acteur CLIENT peut saisir les données (*ex.*: données GPS nouvellement acquises, nom du coureur, données météorologiques, etc.). Il peut simplement les consulter ou alors demander d'ajouter les nouvelles données dans le système d'analyse SOLAP. Suite à cette opération, il lui sera possible de réaliser des analyses statistiques sur de nouvelles données à partir du volet SOLAP. Un SOLAP JAT permet au CLIENT de choisir le moment où le rafraîchissement de l'entrepôt et la reconstruction des cubes doivent être réalisés. Ainsi, le client analyse les données qu'au moment où il en a besoin.

Un **SOLAP temps réel** contient quatre couloirs d'activités : UTILISATEUR, CAPTEUR, MODULE TRANSACTIONNEL et MODULE MULTIDIMENSIONNEL. Ces couloirs sont illustrés sur la Figure 2-4 dans un diagramme d'activités.

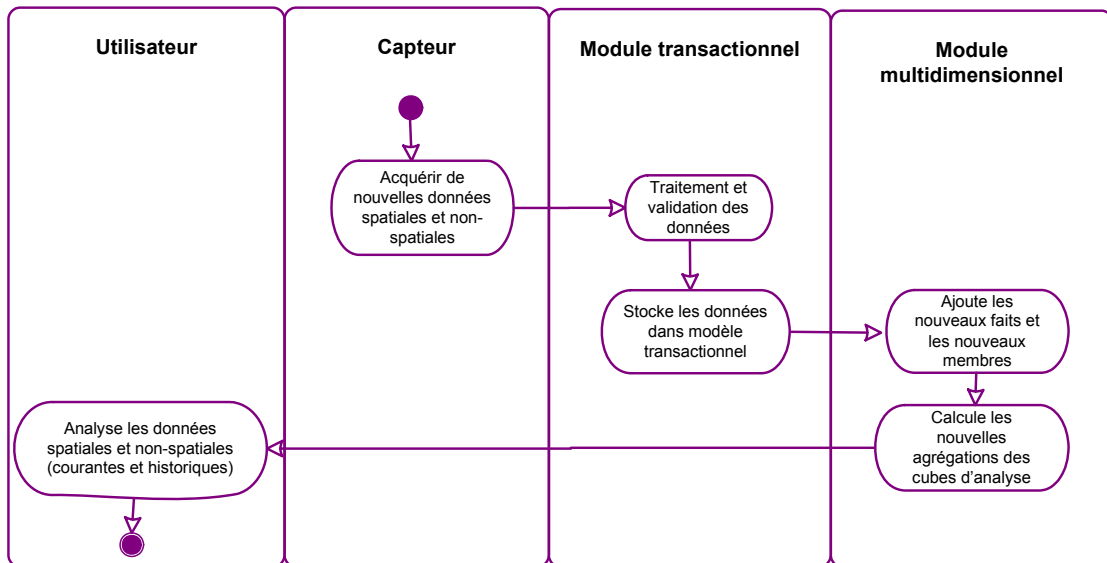


Figure 2-4 : Diagramme d'activités d'un SOLAP temps réel

Le CAPTEUR représente les systèmes qui collectent les nouvelles données et les transmettent au MODULE TRANSACTIONNEL. Dans l'exemple d'application du cyclisme sur route, l'acteur CAPTEUR peut représenter le récepteur GPS dont les observations permettront de calculer la position d'un cycliste. L'acteur CAPTEUR peut aussi correspondre à une station météorologique qui mesure la température, la vitesse et la direction du vent. Tout d'abord, les données sont traitées (si nécessaire) et validées avant d'être transférées dans la base de données pour de futures opérations transactionnelles. Ensuite, le MODULE MULTIDIMENSIONNEL archive les données dans le modèle multidimensionnel de l'entrepôt. L'opération d'archivage se traduit par l'ajout de nouveaux faits dans la table des faits et de nouveaux membres dans les dimensions changeantes. Les nouveaux faits sont directement calculés à partir des nouvelles données saisies par le module transactionnel. Enfin, le système recalcule les agrégations des cubes d'analyse. Ces actions sont exécutées séquentiellement par le SOLAP TR. Le quatrième couloir contient l'action de l'UTILISATEUR qui analyse les données (spatiales et non-spatiales) courantes et historiques.

Le diagramme d'activités d'un **SOLAP juste-à-temps** (cf. Figure 2-5) est plutôt formé de trois couloirs d'activité : L'UTILISATEUR, le MODULE TRANSACTIONNEL et le MODULE MULTIDIMENSIONNEL.

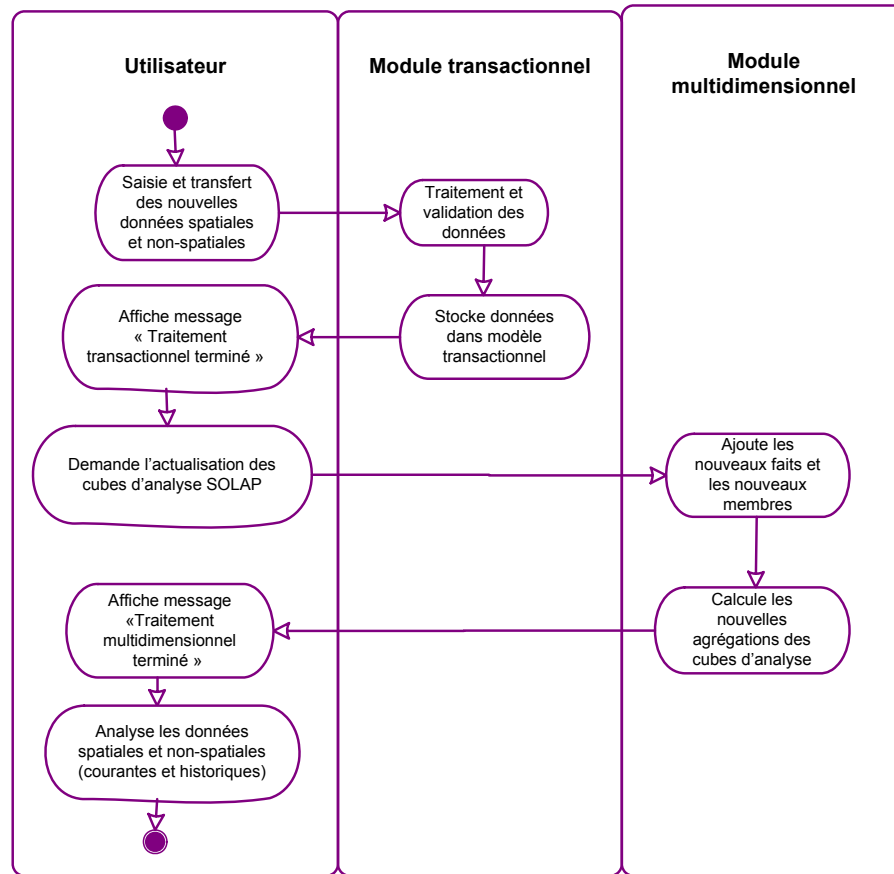


Figure 2-5 : Diagramme d'activités d'un SOLAP juste-à-temps

L'UTILISATEUR réalise à la fois la saisie des nouvelles données et l'analyse de ces données. Le SOLAP JAT permet à l'utilisateur de choisir le moment où le rafraîchissement de l'entrepôt et la reconstruction des cubes d'analyse sont réalisés. Un message est affiché à l'utilisateur lorsque les traitements sont terminés.

La Figure 2-6 présente le fonctionnement d'un **SOLAP temps réel**, à l'aide d'un diagramme de séquence.

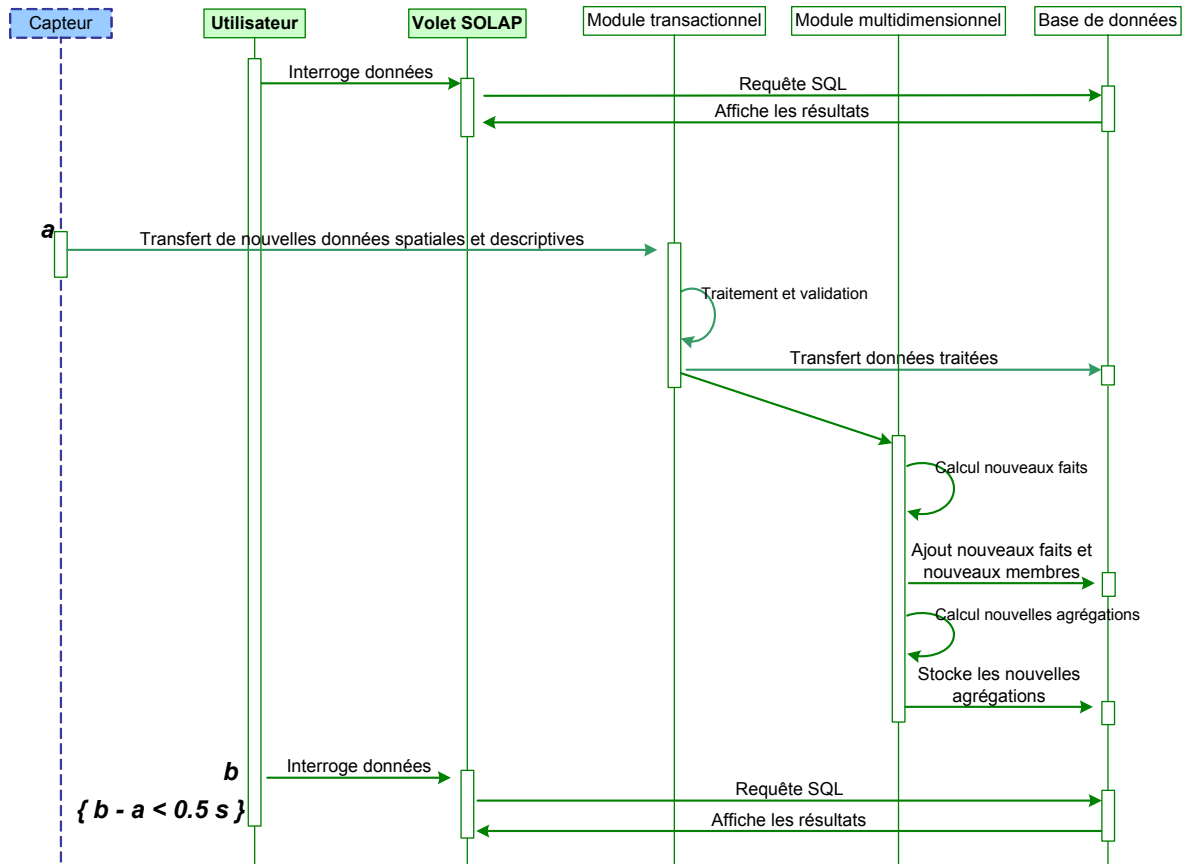


Figure 2-6 : Diagramme de séquence avancé d'un SOLAP temps réel

Le diagramme de séquence a pour but de montrer les relations entre les parties du système d'un point de vue temporel. La contrainte de temps attribuée au système pour une application donnée peut être ajoutée entre accolades sur le diagramme de séquence avancé [Douglass 1999]. Le diagramme de la Figure 2-6 montre qu'entre le moment où de nouvelles données sont enregistrées sur le récepteur (Instant a) et le moment où l'utilisateur peut interroger ces données (Instant b), pas plus de 0,5 seconde doivent s'écouler ($\{ b - a < 0,5 s \}$). Pour cet exemple, le système doit offrir en tout temps des temps de réponse inférieurs à 0,5 seconde.

Aucune contrainte de temps n'apparaît sur un diagramme de séquence de **SOLAP juste-à-temps** (Figure 2-7).

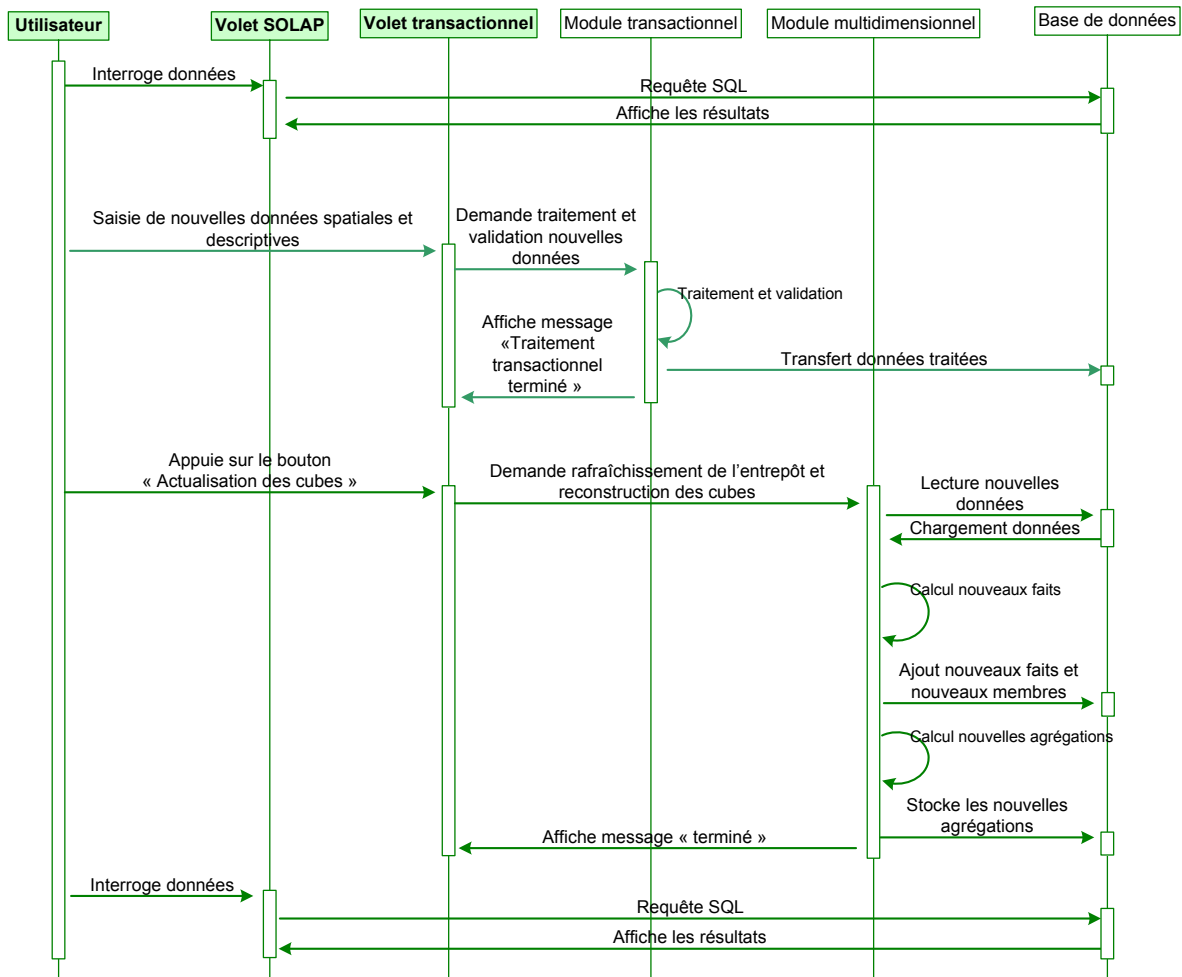


Figure 2-7 : Diagramme de séquence d'un SOLAP juste-à-temps

Deux volets se retrouvent sur le diagramme de séquence d'un SOLAP JAT, comparativement à celui du SOLAP TR. Le VOLET TRANSACTIONNEL permet principalement à l'utilisateur de saisir les nouvelles données descriptives et de transférer les données spatiales. Sur le VOLET TRANSACTIONNEL, on retrouve aussi un bouton permettant à l'utilisateur, une fois qu'il a saisi de nouvelles données, de lancer l'actualisation des cubes d'analyse SOLAP. Une contrainte temporelle peut être ajoutée entre le moment où l'utilisateur appuie sur le bouton et celui où il analyse les nouvelles données pour s'assurer de respecter les besoins de l'application.

En résumé, un SOLAP JAT doit simplement répondre aux besoins de l'application et fournir les données aux analystes que lorsqu'ils en ont besoin. Un SOLAP TR procédera plutôt à l'actualisation des cubes dès que de nouvelles données sont acquises. La principale différence entre un SOLAP TR et un SOLAP TQR se situe au niveau de l'importance accordée au respect de la contrainte temporelle. Dans les applications nécessitant un temps de réponse inférieur à 0,5 seconde et que cette valeur doit toujours être respectée par le système, un SOLAP TR doit être utilisé. Le fonctionnement du SOLAP TP se rapproche de celui des outils d'aide à la décision temps réel défini au chapitre 1. En effet, ils visent simplement à augmenter la fréquence des rafraîchissements des outils traditionnels pour ainsi permettre aux décideurs d'avoir accès aux données courantes de l'organisation.

2.3 Conclusion

Dans le chapitre 2, les SOLAP temps réel, temps quasi-réel, temps propice et juste-à-temps ont été définis ainsi que leur fonctionnement général a été présenté. Dans l'application visée par ce mémoire, c'est-à-dire l'analyse de la performance des athlètes pratiquant un sport extérieur, l'approche SOLAP temps réel permettrait d'ajouter en temps réel de nouvelles données GPS. Cependant, des ajouts plus fréquents seraient, pour cette application, suffisants. En effet, l'application vise principalement les périodes d'entraînement. De plus, l'étape d'acquisition et de traitement des données spatiales et non-spatiales pourrait seulement être réalisée en temps différé puisqu'un entraîneur utilisera que très rarement le système d'analyse SOLAP pendant l'entraînement de l'athlète. L'approche choisie pourrait alors permettre à l'entraîneur de choisir le moment où il désire ajouter de nouvelles données. Cependant, elle doit tout de même pouvoir ajouter les nouvelles données dans des délais raisonnables répondant aux besoins de l'application.

Le troisième et dernier chapitre présente en détail les besoins identifiés auprès des entraîneurs ainsi que l'approche choisie pour analyser la performance des athlètes.

3 IMPLANTATION D'UN OUTIL POUR L'ANALYSE DE LA PERFORMANCE

Le chapitre 2 a présenté différentes approches SOLAP permettant de répondre aux besoins d'analyse en temps réel de certaines applications. Le chapitre 3 présente les besoins des entraîneurs en termes d'analyse de la performance, l'approche choisie et le prototype développé dans le cadre de ce projet de recherche.

Tout d'abord, nous exposons à la section 3.1 les différents besoins des entraîneurs en termes d'outils d'analyse de la performance. Ensuite, un inventaire des données et des systèmes déjà utilisés pour ce type d'analyse par les entraîneurs et les professionnels de kinésiologie de l'Université Laval est présenté à la section 3.2. La section 3.3 justifie l'approche choisie, SOLAP JAT, pour répondre aux besoins des entraîneurs. La section 3.4 présente les détails sur la définition des cubes d'analyse pour l'application en sport. Cette étape est nécessaire lors de l'implantation de toutes applications SOLAP traditionnelles. L'architecture et le fonctionnement du prototype SOLAP-SPORT développé sont décrits aux sections 3.5 et 3.6. Enfin, le prototype a été testé et validé. Les résultats sont présentés aux sections 3.7 et 3.8.

3.1 Évaluation d'opportunité

De nombreuses méthodes pour améliorer et optimiser la performance des athlètes sont apparues dans le domaine du sport de compétition au cours des dernières années. Les entraîneurs utilisent maintenant de nouveaux systèmes (*ex. Personal Digital Assistants* aussi appelés *PDAs*) pour soutenir et faciliter leur travail ce qui changera probablement de façon significative l'image actuelle du sport et du jeu [Stresing 2004]. Un système pour analyser la performance des athlètes semblait alors approprié et susciter un intérêt chez les entraîneurs.

Cette section distingue deux types de besoins. Nous avons, dans un premier temps, les besoins généraux en termes d'outils d'analyse de la performance, que nous exposons en section 3.1.1. Dans un deuxième temps, les besoins plus spécifiques à une discipline, le patinage de vitesse sur longue piste, que nous exposons à la section 3.1.2.

3.1.1 Besoins généraux

Lors d'une rencontre au département de kinésiologie de l'Université Laval, cinq besoins généraux ont été identifiés : (1) Obtenir des données instantanées sur la performance des athlètes; (2) Archiver ces données; (3) Analyser facilement ces données; (4) Visualiser l'endroit exact où chaque performance a été atteinte; et (5) Analyser rapidement les nouvelles données recueillies. Ces besoins sont décrits en détails dans les paragraphes suivants. Les critères requis pour répondre à chacun de ces besoins sont aussi spécifiés.

- (1) Obtenir des données instantanées sur la performance.** Pour analyser en détail la performance d'un athlète, les entraîneurs doivent connaître sa position, sa vitesse et son accélération instantanées tout au long de l'entraînement. Actuellement, pour mesurer la performance des athlètes, les entraîneurs utilisent des chronomètres ou des cellules photoélectriques fournissant seulement des valeurs moyennes (*ex.*: vitesse moyenne). Les données calculées à partir d'observations GPS (données GPS) au cours de l'entraînement d'un athlète décrivent précisément sa performance (position, vitesse, accélération) en tous points de la piste (valeurs instantanées) [Lambert 2002; Lambert & Santerre 2004; Veilleux et al. 2004]. Grâce à la vitesse instantanée, l'entraîneur peut identifier, par exemple, des zones où la variation de la vitesse d'un athlète est trop grande. Ainsi, ce dernier peut apprendre à mieux doser ses efforts et obtenir de meilleures performances lors de compétitions. Pour analyser et comparer la performance des athlètes, la vitesse est effectivement le paramètre le plus utile et le plus représentatif [Schutz & Herren 2000].
- (2) Archiver les données sur la performance.** Les entraîneurs désirent archiver les données obtenues sur la performance pour effectuer des comparaisons dans le temps et ainsi suivre la progression de l'athlète. Présentement, les données sur la performance

ne sont que très rarement conservées. Une base de données, ou plutôt un marché de données (cf. section 1.3.2), outil optimal pour archiver de grand volume de données temporelles, assurerait une évaluation continue de la performance des athlètes.

- (3) **Analyser facilement les données sur la performance.** Il est primordial pour un entraîneur d'obtenir des résultats d'analyse facilement. Plusieurs d'entre eux n'ont qu'une faible connaissance de l'informatique et utiliseront le système si et seulement si son interface est simple, intuitive et facile d'utilisation. La navigation et les analyses sur le système doivent alors s'effectuer à l'aide de simples clics de la souris.
- (4) **Visualiser l'endroit exact où chaque performance a été atteinte.** Les entraîneurs désirent obtenir des résultats d'analyse pour des endroits spécifiques de la piste et visualiser, à l'instant même, l'endroit exact où ces résultats ont été atteints. Ainsi, ils peuvent identifier des pertes de vitesse trop importantes ou alors noter des progressions de la performance dans une section particulière de la piste (ex.: les courbes). La Figure 3-1 illustre un volet spatial indiquant, à l'aide d'une carte de la piste, l'endroit exact où l'athlète a atteint une vitesse de 6.52 m/s. Visuellement, on peut constater que l'athlète a atteint cette vitesse à l'entrée de la première courbe.

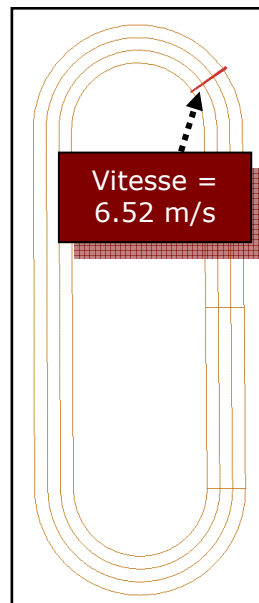


Figure 3-1 : Visualisation de l'endroit exact sur la piste où le résultat (vitesse moyenne (m/s)) a été atteint

Le système doit alors non seulement diffuser les données sur la performance d'un athlète, mais aussi indiquer, sur une carte, la ou les sections exactes sur la piste où elles ont été obtenues.

- (5) Analyser rapidement les nouvelles données recueillies sur la performance.** Les entraîneurs veulent obtenir les données sur la performance actuelle des athlètes de façon rapide et exacte pour ainsi adapter immédiatement leur entraînement et favoriser le succès de l'athlète. Le système doit alors afficher dans les heures qui suivent les données sur la performance recueillies lors du dernier entraînement et offrir des résultats d'analyse prenant en compte ces nouvelles données.

Afin de répondre à l'ensemble de ces besoins, une approche doit être identifiée pour ainsi faciliter les analyses des entraîneurs. Auparavant, les besoins spécifiques à un entraîneur de patinage de vitesse ont été identifiés et sont détaillés à la section suivante.

3.1.2 Besoins spécifiques

Suite à une rencontre avec M. Robert Tremblay, entraîneur de patinage de vitesse, des besoins spécifiques pour l'analyse de la performance des athlètes pratiquant ce sport ont pu être identifiés.

Tout d'abord, lors de son entraînement, un patineur de vitesse effectue habituellement plusieurs tours de piste et ne parcourt pas nécessairement une distance spécifique (*ex.*: 1500 m). De plus, au cours d'une même séance d'entraînement plusieurs athlètes effectuent des tours de piste. Le terme « entraînement » utilisé dans cette section, ainsi que dans les sections suivantes, fait alors référence à un ou plusieurs tours de pistes effectués successivement par un ou plusieurs athlètes.

Les besoins spécifiques d'un entraîneur de patinage de vitesse identifiés sont, lors d'un entraînement :

- Obtenir des valeurs, en terme de résultats d'analyse, pour la vitesse, l'accélération, la durée, la trajectoire, la cadence, le rythme cardiaque dans les situations suivantes :
 - en tous points de la piste (valeurs instantanées)
 - sur des sections particulières de la piste (valeurs moyennes)
 - sur un seul tour de piste (valeurs moyennes)
 - pour l'entraînement en entier (valeurs moyennes)
- Comparer les entraînements dans le temps: Année, Mois, Semaine, Jour, AM et PM
- Comparer les athlètes entre eux ou selon leur club, leur catégorie d'âge, leur sexe (féminin, masculin) ou leur niveau (provincial, national, international)
- Comparer les entraînements selon la technique adoptée par l'athlète : type de départ, entrée plus serrée dans les courbes (trajectoire), coup de patin plus long, etc.
- Comparer les entraînements selon certains critères mécaniques: type d'aiguisage des patins, vêtements utilisés, etc.
- Comparer les entraînements selon certains facteurs météorologiques présents au cours de la journée d'entraînement: la pression barométrique, la température, la direction et la vitesse du vent et la qualité de la glace
- Comparer les entraînements selon le type d'épreuve effectué (si cela se présente): 1000 m, 1500 m, 10000 m

Bref, les entraîneurs veulent obtenir facilement des données sur la performance des athlètes en tous points de la piste (valeurs instantanées), visualiser l'endroit où ces performances ont été atteintes et archiver l'ensemble de ces données pour des comparaisons temporelles. Ils doivent aussi pouvoir analyser rapidement les nouvelles données recueillies au cours d'un entraînement et non attendre plusieurs jours. Un entraîneur de patinage de vitesse veut plus particulièrement obtenir la performance de l'athlète pour des sections particulières de la piste, comparer les athlètes entre eux et comparer les entraînements selon différents facteurs.

Avant d'identifier ou d'entreprendre le développement d'une nouvelle approche répondant à l'ensemble de ces besoins, un inventaire des données et des systèmes existants a été réalisé. La section suivante présente cet inventaire.

3.2 Inventaire et analyse de l'existant

Un inventaire des données et des systèmes déjà utilisés par les entraîneurs ou au Département de kinésiologie pour l'analyse de la performance des athlètes a été réalisé. La section 3.2.1 contient l'inventaire des données recueillies par Mathieu Lambert au cours de sa maîtrise [Lambert 2002]. Les logiciels déjà utilisés pour l'analyse de la performance des athlètes sont décrits à la section 3.2.2.

3.2.1 Données existantes

Quelques données provenant des tests réalisés en patinage de vitesse par Mathieu Lambert [Lambert 2002] sur l'anneau de glace Gaétan-Boucher à Ste-Foy ont été récupérées. Les mesures GPS ont alors été réalisées en mode RTK-inversé à une fréquence de 10 Hz et représentaient les entraînements (tours de piste) de quelques athlètes. Finalement, ces données n'ont pas été utilisées dans le projet, puisqu'une meilleure résolution temporelle était recherchée (supérieur à 10Hz). Un récepteur GPS offrant une plus haute fréquence d'observation (20 Hz) a plutôt été utilisé pour recueillir la position, la vitesse et l'accélération de l'athlète [Veilleux et al. 2004].

Un relevé de l'anneau de glace Gaétan-Boucher était aussi disponible. Cependant, les corridors de la piste étaient absents. Puisqu'ils sont essentiels pour les analyses sur la technique de l'athlète (*ex.*: trajectoire empruntée), un nouveau relevé de l'anneau, avec cette fois les corridors, a été nécessaire. La collecte des données a été réalisée en mode RTK cinématique avec deux récepteurs bi-fréquence à 20Hz.

D'autres données obtenues lors d'essais en cyclisme au vélodrome Louis-Garneau, en canoë-kayak au Lac Beauport et en ski alpin étaient aussi présentes, mais elles n'ont pas été utilisées dans le projet. Éventuellement, ces données pourraient être utilisées pour analyser la performance des athlètes dans ces trois autres disciplines. L'inventaire des données est détaillé à l'0.

3.2.2 Logiciels existants

Une **interface de visualisation** a été développée à partir d'un Système d'Information Géographique (SIG) par Mathieu Lambert au cours de sa maîtrise [Lambert 2002]. Ce système permet de suivre sur un volet cartographique le déplacement en temps réel d'un athlète portant une antenne GPS.

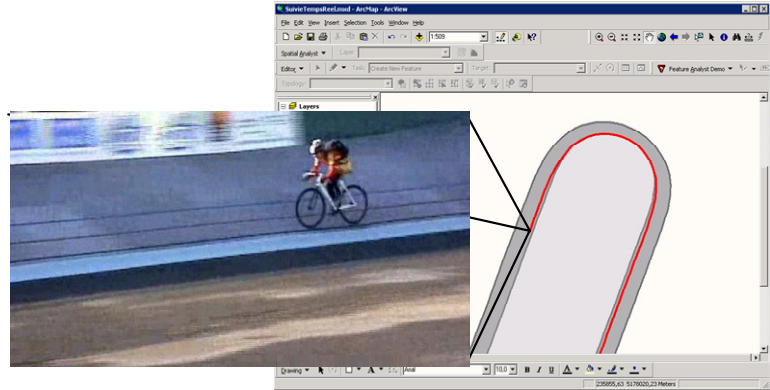


Figure 3-2 : Interface de visualisation

Des applications ont aussi été développées à partir du **logiciel de traitement mathématique Matlab** de *The MathWorks* au département de kinésiologie. Ces applications sont principalement utilisées

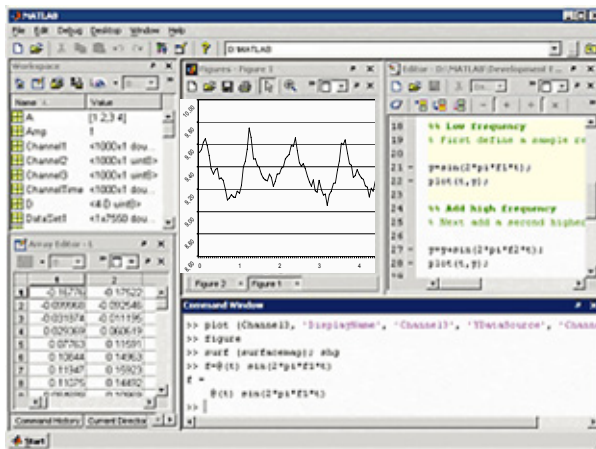


Figure 3-3 : Logiciel de traitement mathématique Matlab

pour obtenir des graphiques de la vitesse instantanée en fonction du temps à partir des données GPS recueillies. Plusieurs fonctions ont été ajoutées pour obtenir des statistiques précises sur les données. Différents algorithmes ont aussi été développés et intégrés dans ces applications.

En général, chacun de ces systèmes ne permet pas de répondre à l'ensemble des besoins identifiés chez les entraîneurs pour l'évaluation et l'analyse de la performance des athlètes (cf. section 3.1). Tout d'abord, les interfaces clientes sont très peu intuitives et offrent plusieurs fonctionnalités qui rendent les analyses de la performance assez complexes pour un non-initié. Notamment, les requêtes agrégatives, c'est-à-dire les requêtes permettant

d'obtenir la performance pour une section spécifique de la piste (*ex.*: première courbe, deuxième droit, etc.) ou pour une période de temps en particulier (*ex.*: pour le mois en cours, pour l'année, etc.) sont longues et difficiles à construire à partir d'un logiciel SIG et elles nécessitent souvent un expert (*cf.* 0). De son côté, le logiciel *Matlab* ne possède pas de volet cartographique pour référencer les valeurs sur la performance obtenue et ainsi rendre les analyses plus intuitives.

Ensuite, les fonctions d'analyse offertes sur l'interface d'un SIG ou sur un logiciel mathématique ne permettent pas à un entraîneur de comparer facilement et rapidement les entraînements dans le temps, selon différents facteurs (mécaniques, météorologiques, techniques) ou comparer les athlètes. Pour y parvenir plusieurs opérations doivent être réalisées et plusieurs résultats sur plusieurs fenêtres doivent être obtenus. De plus, avec le logiciel *Matlab*, il n'est pas possible pour un entraîneur d'analyser ou de comparer les trajectoires des athlètes, puisqu'il n'offre aucun volet cartographique.

Enfin, les systèmes existants permettent de réaliser des analyses complètes sur la performance des athlètes à partir de données datant de quelques semaines ou quelques jours. En effet, l'ajout de nouvelles données est long et difficile.

3.3 Solution répondant aux besoins

Pour répondre au premier besoin identifié chez les entraîneurs (*cf.* section 3.1.1), obtenir des données instantanées sur la performance, le système de positionnement par GPS semble tout indiqué (*cf.* besoin #1). En effet, les avantages du système GPS pour obtenir des données instantanées et précises sur la performance d'un athlète ont déjà été prouvés [Skaloud & Merminod 2000; Lambert 2002; Lambert & Santerre 2004; Veilleux et al. 2004].

Les outils d'aide à la décision, tels que les OLAP, sont optimisés pour exploiter des bases de données volumineuses (*cf.* besoin #2) et ils offrent un module client facile d'utilisation (*cf.* besoin #3). En effet, les « *outils décisionnels s'adressent à une clientèle d'utilisateurs moins spécialisée que les usagers des SIG et peuvent être utilisés par des non experts en géomatique ou en informatique* » [Proulx & Bédard 2004].

La technologie décisionnelle SOLAP permet à l'utilisateur de visualiser en plus la référence spatiale des données puisqu'elle possède un volet spatial (*cf.* besoin #4). Des tests ont été réalisés et ont prouvé qu'un système d'analyse SOLAP est plus convivial qu'un système transactionnel SIG pour l'analyse des données spatio-temporelles, particulièrement pour les données agrégées et pour obtenir des statistiques (*cf.* 0)[Proulx & Bédard 2004]. Sur le module client d'un outil SOLAP, l'utilisateur interagi directement avec les données en sélectionnant les thèmes d'analyse grâce aux dimensions descriptives (volet descriptif) ou alors en sélectionnant sur une carte les endroits d'intérêt (volet spatial).

Le module client SOLAP exploite des cubes définis en fonction des besoins spécifiques d'une application (*cf.* section 3.1.2). Ainsi, suite à la définition des dimensions (niveaux hiérarchiques et membres) et des mesures du cube d'analyse pour cette application un entraîneur de patinage de vitesse pourra obtenir sur un outil SOLAP des résultats très précis (*ex.*: la vitesse instantanée atteinte en un point) ou des valeurs moyennes (*ex.*: l'accélération moyenne dans la courbe, la vitesse moyenne pour les entraînements de la journée, de la semaine ou du mois). Il pourra comparer les athlètes et les entraînements effectués avec certains facteurs météorologiques (*ex.*: un vent du nord-est, une température de cinq degrés Celsius sous le point de congélation), facteurs mécaniques (*ex.*: utilisation des patins « Clap ») ou facteurs techniques (*ex.*: départ debout).

Bref, l'outil SOLAP traditionnel (*cf.* section 2.1) peut répondre à la majorité des besoins d'analyse d'un entraîneur. Cependant, il ne permet pas d'ajouter de nouvelles données rapidement (*cf.* besoin #5). En effet, la technologie SOLAP a été principalement développée pour exploiter une source de données latente (*ex.*: entrepôt de données spatiales). L'entraîneur ne peut alors analyser la performance actuelle de l'athlète. Le SOLAP JAT répond à ce dernier besoin. Le prototype développé au cours de cette recherche et baptisé SOLAP-SPORT est donc un *SOLAP* juste-à-temps (JAT). Il permet à l'entraîneur d'ajouter de nouvelles données sur la performance lorsqu'il en a besoin pour ses analyses. De plus, cet ajout est réalisé en un temps raisonnable pour l'application (environ 30 à 60 minutes le temps que l'athlète passe au vestiaire).

La section 3.4 présente les détails sur la définition des cubes d'analyse pour l'application en sport utilisées dans le prototype SOLAP-SPORT.

3.4 Conception des cubes d'analyse

L'outil SOLAP développé au CRG [Bédard 2005] est utilisé dans diverses applications (*ex.*: santé, archéologie, transport, etc.). Actuellement, le prototype SOLAP-SPORT permet de comparer seulement les entraînements réalisés sur l'anneau de glace Gaétan Boucher. L'interface utilisateur interroge simplement les données d'un cube d'analyse construit a priori (*cf.* section 2.1). C'est lors de la définition d'un cube d'analyse – dimensions, mesures et table des faits – que l'on tient compte de l'application. Dans ce projet de recherche, le cadre applicatif est l'analyse de la performance d'athlètes de haut niveau pratiquant des sports extérieurs (*ex.*: patinage de vitesse). Les dimensions et les mesures d'un cube d'analyse (*ex.*: le cube PATIN DE VITESSE) sont alors définies directement en fonction des besoins spécifiques de l'entraîneur (section 3.1.2).

Deux types de dimension se retrouvent dans un cube d'analyse SOLAP (*cf.* section 2.1.1): (1) Les dimensions descriptives et; (2) La dimension spatiale. Les dimensions descriptives sont fournies sur le volet descriptif OLAP alors que la géométrie de la dimension spatiale est affichée sur le volet spatial. Les dimensions et les mesures des cubes d'analyse propres à chaque sport extérieur doivent alors être sélectionnées et font l'objet de la section 3.4.1. Les niveaux hiérarchiques de la dimension spatiale doivent être choisis et construits a priori et sont abordés à la section 3.4.2.

3.4.1 Dimensions descriptives

L'une des étapes importantes lors de la mise en place d'un outil SOLAP est l'identification des dimensions thématiques à offrir sur l'interface cliente. Les dimensions correspondent à des thèmes d'analyse et sont directement choisies à partir des besoins spécifiques identifiés auprès du client (*cf.* section 3.1.2). L'0 contient la liste des dimensions possibles avec leurs niveaux hiérarchiques pour les sports suivants : patinage de vitesse, cyclisme, ski de fond, canoë-kayak et ski alpin. Seulement les dimensions pour l'application en patinage de

vitesse ont été sélectionnées avec un entraîneur de patinage de vitesse. Les dimensions retrouvées pour les autres applications sont des suggestions. En général, seulement les dimensions reliées aux facteurs techniques varient d'une application à l'autre.

La Figure 3-4 illustre le premier cube qui a été défini pour répondre aux besoins spécifiques d'un entraîneur de patinage de vitesse et il se nomme PATINAGE DE VITESSE-ENTRAÎNEMENT.

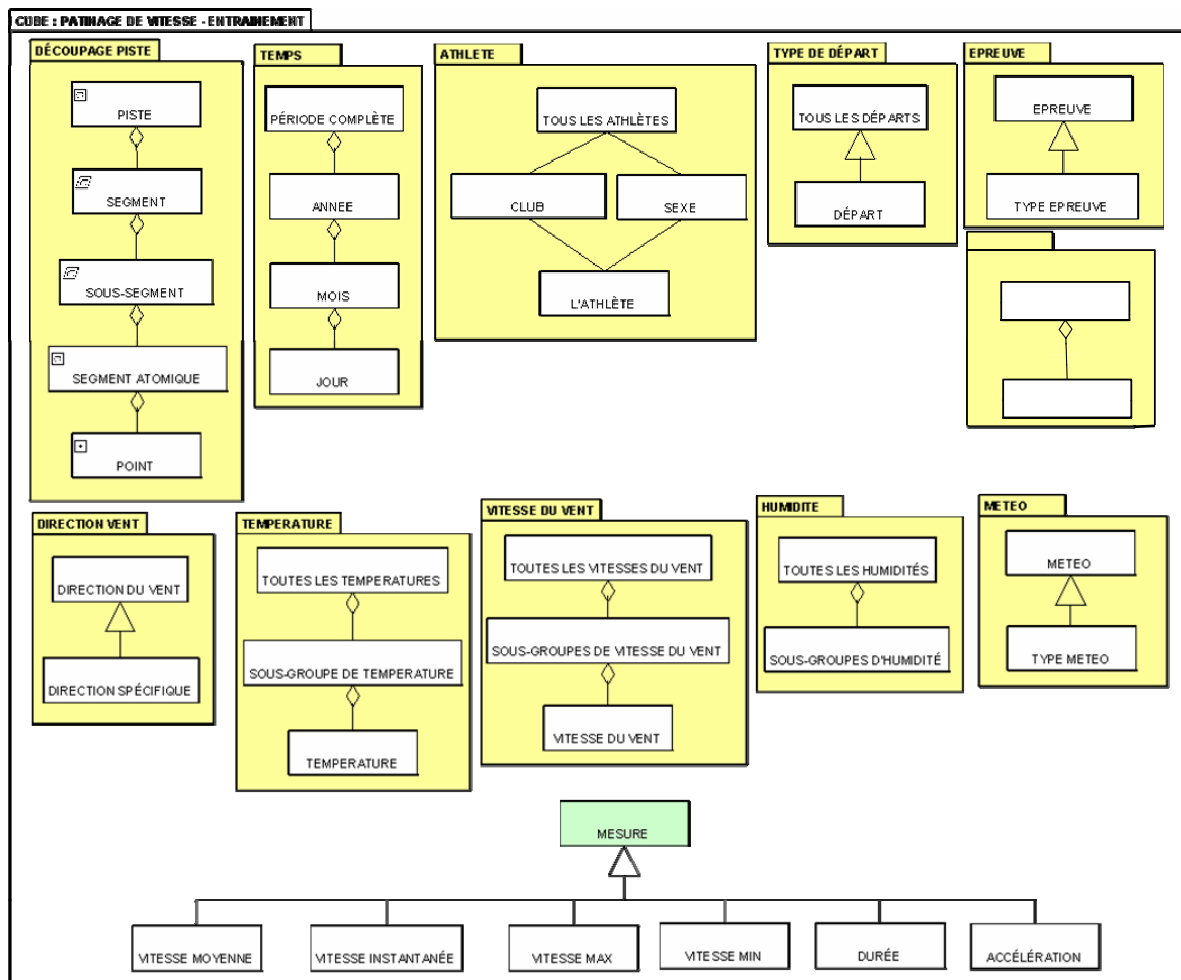


Figure 3-4 : Dimensions et mesures du cube PATINAGE DE VITESSE – ENTRAÎNEMENT

Ce cube a été défini avec les dimensions thématiques suivantes : TEMPS, ATHLÈTE, TYPE DE DÉPART, ÉPREUVE, TOUR, DIRECTION DU VENT, TEMPÉRATURE, VITESSE DU VENT, HUMIDITÉ et MÉTÉO. Le cube contient aussi la dimension spatiale DÉCOUPAGE PISTE (cf. section 3.4.2) et

les mesures suivantes : VITESSE INSTANTANÉE, VITESSE MOYENNE, VITESSE MAXIMUM, VITESSE MINIMUM, ACCÉLÉRATION et DURÉE.

L'exploitation des données via le SOLAP à partir du cube PATINAGE DE VITESSE-ENTRAÎNEMENT permet à l'entraîneur d'effectuer des analyses répondant à la plupart des besoins initialement identifiés. L'entraîneur peut comparer les entraînements dans le temps, selon les conditions météorologiques présentes ou le type de départ utilisé. Il peut aussi comparer les athlètes entre eux et selon leur niveau et leur sexe. Des données sur la performance (vitesse moyenne, vitesse max, vitesse min, durée, accélération) peuvent être obtenues pour un seul tour de piste ou pour des sections particulières de la piste grâce à la dimension spatiale.

La section 3.4.2 décrit en détail la dimension spatiale DÉCOUPAGE PISTE dont la géométrie de ses membres est exploitée à partir du volet spatial d'un outil SOLAP.

3.4.2 Dimension spatiale

La dimension spatiale possède deux types de membres (données spatiales): (1) les données spatiales latentes, et (2) les données spatiales non latentes.

(1) Données spatiales latentes. Les données spatiales latentes, c'est-à-dire les données qui ne changent pas ou peu, doivent être recueillies au départ et représentent le site où les entraînements ont lieu. Pour le cadre applicatif de ce projet de recherche et spécifiquement pour le patinage de vitesse, nous avons recueilli des données latentes à l'anneau de glace Gaétan-Boucher. Ces données caractérisent et positionnent la piste de patinage de vitesse. Les données latentes pour une application en cyclisme représenteraient la piste d'un vélodrome (*ex.*: le vélodrome Louis-Garneau), en canoë-kayak, les données positionnant un corridor sur un lac (*ex.*: au Lac Beauport), etc.

Si les entraîneurs désirent obtenir la performance de l'athlète pour des parties spécifiques de la piste, la dimension spatiale doit posséder plusieurs niveaux de granularité. Pour l'application en patinage de vitesse, la dimension spatiale latente nommée DÉCOUPAGE PISTE est formée de quatre niveaux hiérarchiques possédant des

membres latents (données latentes): (a) Piste, (b) Segment, (c) Sous-segment et (d) Segment atomique. Ces quatre niveaux hiérarchiques ainsi que leurs membres ont été définis suite à une rencontre avec un entraîneur de patinage de vitesse et peuvent être affichés sur le volet spatial du client SOLAP. Ils sont illustrés sur la Figure 3-5.

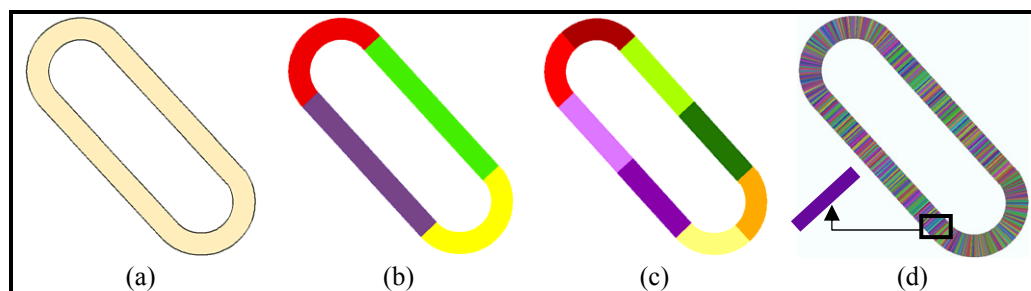


Figure 3-5 : Découpage de la piste en quatre niveaux hiérarchiques
(a) Piste; (b) Segment; (c) Sous-segment; (d) Segment atomique

Le niveau **Piste** (niveau (a) sur la Figure 3-5) fournit les données sur la performance pour l'ensemble de la piste (*ex.*: piste de 400 m pour le patin de vitesse). Pour s'assurer que toutes les données recueillies lors d'un entraînement soient considérées, une valeur supplémentaire de 2 m (1 m à l'intérieur et 1 m à l'extérieur) a été ajoutée à la largeur réelle de la piste. Cette particularité tient compte du fait que l'antenne GPS est située sur le dos de l'athlète et non à la hauteur du sol. La position de l'antenne GPS peut alors se retrouver en dehors de la piste lorsque l'athlète emprunte, par exemple, une courbe avec une position très penchée ou adopte un mouvement latéral très prononcé. Cette situation est expliquée en détail un peu plus loin dans cette section. La Figure 3-6 illustre la différence de largeur entre le niveau Piste et l'anneau de glace Gaétan-Boucher.

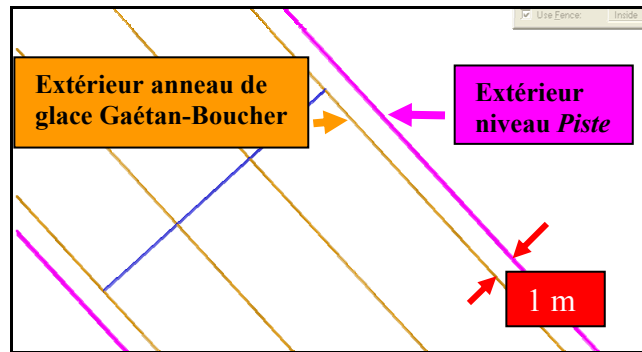


Figure 3-6 : Niveau Piste de la dimension spatiale DÉCOUPAGE PISTE;
Ajout d'une marge de 1 mètre de chaque côté de la piste

Une marge de 1 m de chaque côté de la piste a aussi été ajoutée sur la largeur des niveaux hiérarchiques inférieurs (segment, sous-segment et segment atomique) de la dimension spatiale.

Les **segments** (niveau (b) sur la Figure 3-5) permettent de séparer et distinguer les sections droites et les sections courbes de la piste : 1^{er} droit en vert; 1^{ère} courbe en rouge; etc. Le niveau *segment* de la dimension spatiale DÉCOUPAGE PISTE contient alors quatre membres. Deux **sous-segments** (niveau (c) sur la Figure 3-5) divisent les droits et les courbes de la piste (1^{ère} courbe (1/2) ; 1^{ère} courbe (2/2)). Le niveau sous-segment de la dimension spatiale contient alors huit membres.

Finalement, des **segments atomiques** (niveau (d) sur la Figure 3-5) divisent à nouveau les sous-segments. Il s'agit du niveau le plus détaillé disponible pour la dimension spatiale latente. L'anneau de glace Gaétan-Boucher a été divisé en 442 segments atomiques. Le niveau segment atomique contient alors 442 membres. Un segment atomique représente approximativement 1 m de distance sur la piste. Le GPS recueille des données à 20 Hz, c'est-à-dire à toutes les 0,05 secondes (1/20 sec), et la vitesse moyenne d'un patineur de vitesse est de 40 km/h environ (≈ 10 m/s), alors une position est habituellement enregistrée environ à chaque demi-mètre. Ainsi, la performance associée à un segment atomique correspond à la moyenne d'environ deux positions instantanées consécutives lorsqu'il n'y a pas une perte du signal GPS.

Grâce au découpage de la piste, que nous venons de présenter, un entraîneur de patinage de vitesse peut obtenir la vitesse, l'accélération, la durée pour la piste en entier ou seulement pour un segment, un sous-segment ou un segment atomique.

Le découpage de la piste présenté sur la Figure 3-5 est propre à l'anneau de glace Gaétan-Boucher (piste de 400m). Pour obtenir des données sur la performance obtenues à partir d'un GPS sur un anneau de glace différent (*ex.*: piste de 333.33m) ou pour un sport extérieur différent (*ex.*: cyclisme) un relevé et un nouveau découpage spatial doivent être réalisés (*ex.*: relevé et découpage du vélodrome Louis-Garneau). La définition des niveaux hiérarchiques de la dimension spatiale latente – piste, segment, sous-segment et segment atomique – demeure tout de même valide pour un anneau de glace différent ou pour tous autres anneaux moyennant quelques petits changements principalement au niveau le plus détaillé.

- (2) **Données spatiales non latentes.** Les données non latentes (membres non latents) de la dimension spatiale sont les positions instantanées observées par le GPS. Les positions instantanées sont, entre autres, essentielles pour visualiser la trajectoire de l'athlète lors de son entraînement. En effet, la trajectoire de l'athlète est un besoin identifié auprès de l'entraîneur de patinage de vitesse (*cf.* section 3.1). De plus, le GPS nous permet d'atteindre ce niveau de précision (valeur instantanée à 20 Hz) qui doit nécessairement être offert sur l'outil d'analyse. Les points sur la Figure 3-7 représentent l'ensemble des positions instantanées recueillies par le GPS pour le premier tour de piste effectué par un athlète lors de son entraînement.

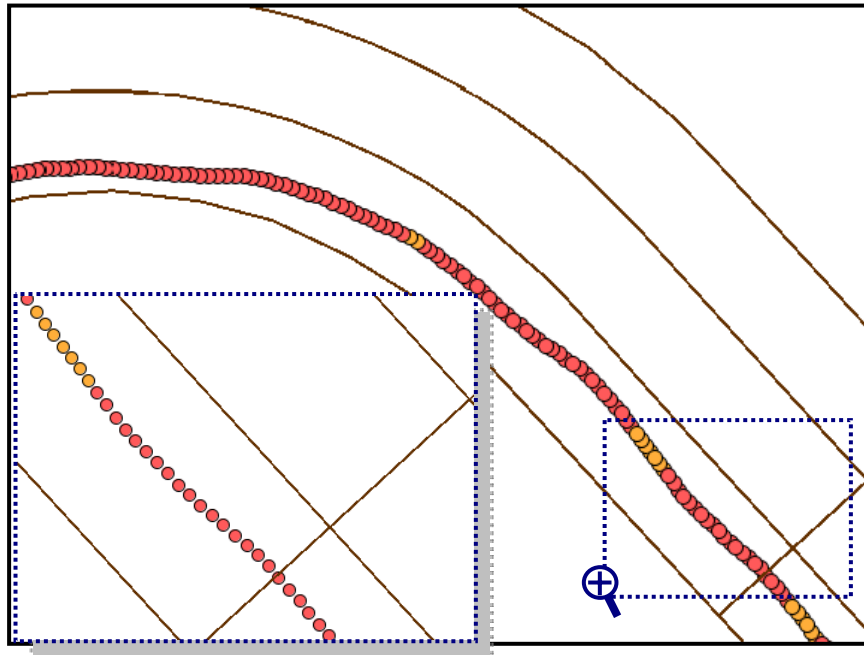


Figure 3-7 : Positions instantanées de la dimension spatiale
Découpage piste (Niveau Point)

Les positions instantanées sont donc représentées sur le volet spatial à l'aide de points. Cette géométrie ponctuelle représente le niveau le plus détaillé (niveau 5) de la dimension spatiale DÉCOUPAGE PISTE. Ce niveau est appelé le niveau **Point**.

Les niveaux hiérarchiques de la dimension spatiale latente divisent la piste sur sa longueur (droit, demi-droit, section d'un mètre). Les corridors de la piste divisent plutôt la piste sur sa largeur. Un niveau hiérarchique alternatif, **Corridor**, pourrait alors être ajouté dans la dimension spatiale latente. La Figure 3-8 montre les niveaux hiérarchiques de dimension spatiale avec cette fois le niveau hiérarchique alternatif Corridor.

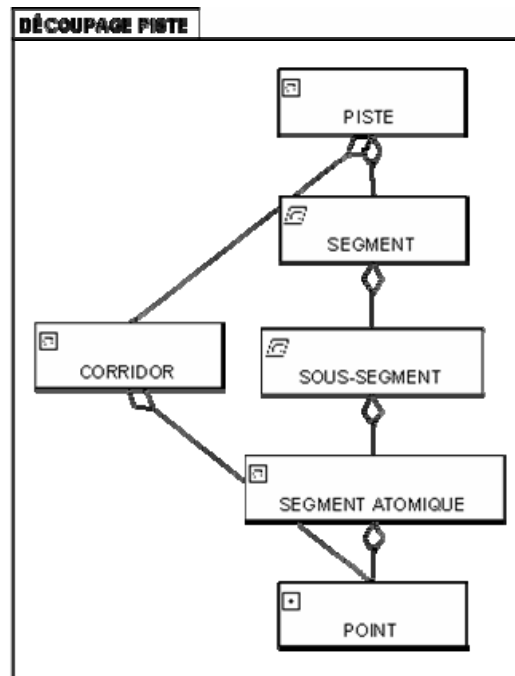


Figure 3-8 : Niveau alternatif Corridor de la dimension spatiale latente DÉCOUPAGE PISTE

Cependant, il est intéressant d'ajouter un niveau alternatif à une dimension spatiale seulement s'il apporte une information supplémentaire lors des analyses. Cette alternative lors de la définition d'un cube SOLAP ne se présente pas lors de la définition d'un cube OLAP. Contrairement aux outils OLAP, deux types d'information sont offerts dans un outil SOLAP : (1) information implicite, (2) information explicite. Cette différence entre les outils OLAP et les outils SOLAP est expliquée à l'0.

Le niveau alternatif **Corridor** n'a pas été ajouté dans la dimension spatiale puisque l'information visuelle, c'est-à-dire les points représentant chaque position instantanée de l'athlète, est suffisante pour identifier le corridor occupé et permettre à l'entraîneur d'effectuer des analyses comparatives. De plus, la position fournie par l'antenne GPS n'indique pas toujours le corridor réellement occupé par l'athlète. En effet, l'athlète porte l'antenne GPS sur son dos et cette dernière peut se retrouver dans un corridor qui n'est pas réellement celui occupé par l'athlète. La Figure 3-9 illustre une situation où le corridor indiqué par la l'antenne GPS ne correspond à celui réellement occupé par l'athlète.

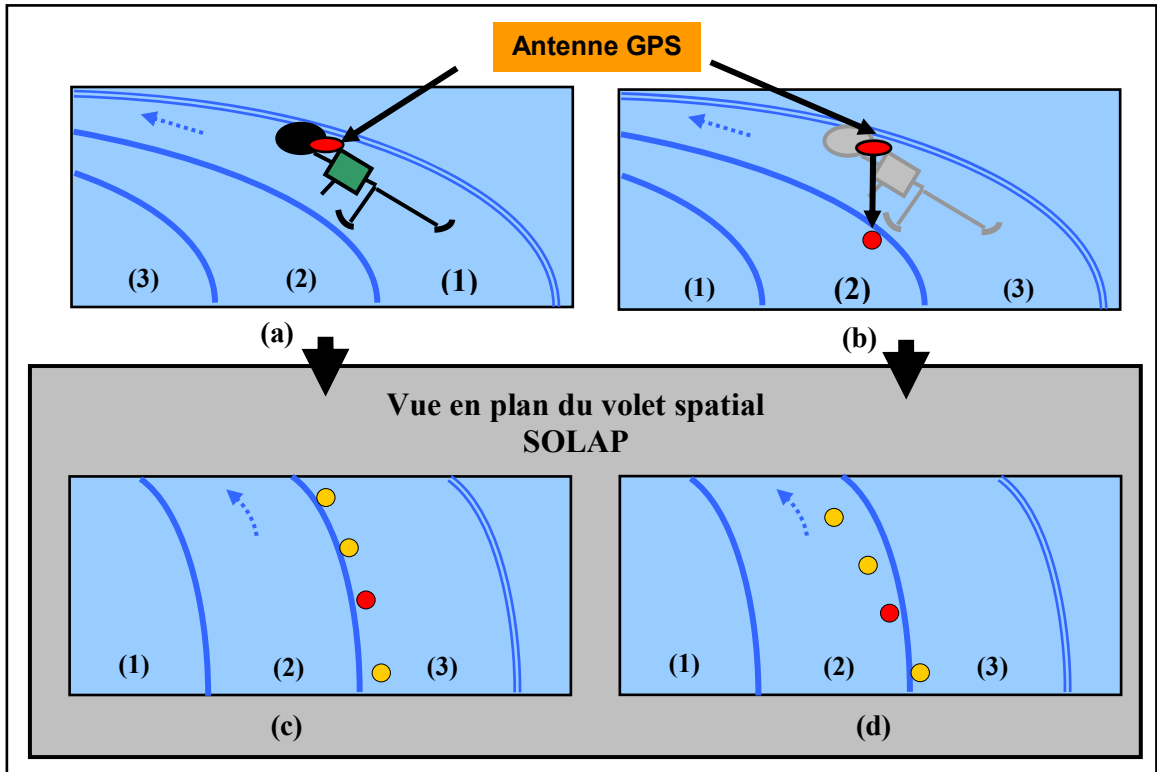


Figure 3-9 : Exactitude du corridor occupé par l'athlète

(a) Corridor réellement occupé par l'athlète; (b) Corridor indiqué par l'antenne GPS; (c) Position au niveau du sol de l'athlète (d) Position de l'antenne GPS

Le corridor #1 est celui réellement occupé par l'athlète sur l'exemple de la Figure 3-9 (partie a). Lors de son virage, l'athlète se penche vers l'intérieur de la courbe positionnant ainsi l'antenne GPS au-dessus du corridor #2 (partie b). Sur le volet spatial de l'interface cliente SOLAP, le point représentant la position de l'athlète sera alors situé dans le corridor #2 (partie c). La dimension spatiale ne peut alors posséder un niveau alternatif corridor en raison du risque d'inexactitude. En effet, lors du calcul des agrégations, les mesures du cube d'analyse (vitesse, accélération, etc.) seraient agrégées au niveau Corridor et les positions erronées entraîneraient des valeurs inexacts.

En conclusion, la dimension spatiale DÉCOUPAGE PISTE possède cinq niveaux de granularité : Piste, Segment, Sous-segment, Segment atomique et Point. Les niveaux hiérarchiques de la dimension spatiale offrent des valeurs moyennes (Piste, Segment, Sous-segment, Segment atomique) et instantanées (Point) sur la performance des athlètes.

Les analyses offertes sur le SOLAP sont intéressantes, mais le système doit nécessairement permettre l'ajout facile et rapide des nouvelles données recueillies au cours d'un entraînement. Étant donné que l'outil SOLAP traditionnel ne répondait pas à ce besoin, une nouvelle fonctionnalité a été développée. Le prototype nommé SOLAP-SPORT vise alors à offrir à un utilisateur (entraîneur) un outil d'aide à la décision spatial permettant d'analyser et d'ajouter rapidement (dans la journée même) les nouvelles données collectées au cours d'un entraînement. La section 3.5 décrit les composantes de l'architecture SOLAP-SPORT.

3.5 Architecture SOLAP-SPORT

L'architecture du SOLAP-SPORT décrite dans cette section, doit répondre à deux objectifs majeurs : (1) permettre l'ajout des données caractérisant un entraînement (données provenant du GPS et données descriptives) lorsque l'entraîneur le demande et (2) offrir ces nouvelles données dans des délais acceptables sur l'interface cliente SOLAP suite à cette demande.

Le prototype SOLAP-SPORT développé est un SOLAP juste-à-temps (*cf.* section 2.2). Il fournit les nouvelles données recueillies aux entraîneurs uniquement lorsqu'ils en ont besoin. Les données GPS et descriptives sont recueillies et transférées dans le prototype SOLAP-SPORT en temps différé. Le rafraîchissement de l'entrepôt et la reconstruction des cubes d'analyse sont lancés seulement suite à la demande de l'utilisateur. La Figure 3-10 montre les différents modules de l'architecture du SOLAP-SPORT.

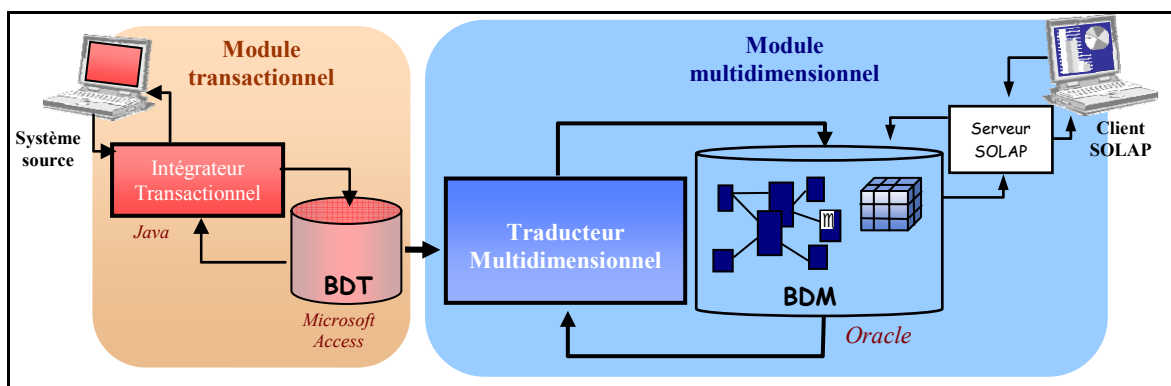


Figure 3-10 : Architecture générale du prototype SOLAP-SPORT

L'architecture générale de notre prototype montre l'utilisation de deux types de SGBD relationnel : *Microsoft Access* et *Oracle*. Le langage de programmation utilisé pour développer les composantes du prototype est le *JavaTM*. Ce langage a été choisi principalement parce que le client SOLAP a été développé dans le même langage de programmation, mais aussi pour la facilité du langage à établir des connexions et le nombre d'API (*Application Programming Interface*) disponibles.

Notre architecture se divise en deux principaux modules, le module transactionnel (*cf.* Figure 3-10 section rouge) et le module multidimensionnel (*cf.* Figure 3-10 section bleu). Ces deux modules sont décrits dans les sections suivantes.

3.5.1 Module transactionnel

Le but du module transactionnel est de permettre à un utilisateur de saisir et stocker les données caractérisant un entraînement, telles les données descriptives et les données calculées à partir des observations GPS (données GPS).

Ce module comprend trois composantes : (1) le système source; (2) l'intégrateur transactionnel; et (3) la base de données transactionnelle.

(1) Le **système source** offre une interface graphique facilitant la saisie des données caractérisant un entraînement pour une application sportive particulière. Pour l'application sportive en patinage de vitesse une interface utilisateur a été développée en *JavaTM* (Figure 3-11).

Figure 3-11 : Interface graphique du système source

Elle permet notamment de recueillir les données descriptives nécessaires aux analyses de la performance à partir du client SOLAP. Le nom de l'athlète, l'heure à laquelle il a débuté son entraînement (essentiel pour associer le groupe de positions GPS à un athlète), la météo, la température, la direction du vent, le type de départ utilisé et le type d'aiguillage des patins sont des exemples de données descriptives. L'utilisateur transfère les données GPS obtenues pour l'entraînement dans la base de données transactionnelle en identifiant, à partir de l'interface graphique, le fichier texte contenant ces mesures GPS. Cette interface peut être utilisée pour d'autres applications sportives moyennant quelques petites modifications (*ex.*: type de pneus utilisé pour l'application en cyclisme).

- (2) **L'intégrateur transactionnel** traite et intègre les données dans la base de données transactionnelle (BDT). Les données descriptives saisies par l'entraîneur sur l'interface graphique du système source et les données GPS sont traitées avant d'être intégrées dans la base de données transactionnelle. Les détails sont fournis à la section 3.6.

- (3) La **base de données transactionnelle (BDT)** possède une structure normalisée (cf. section 1.2.1) ce qui facilite l'intégration des données saisie à partir du système source. La Figure 3-12 illustre le modèle d'implantation de la BDT propre à l'application en patinage de vitesse.

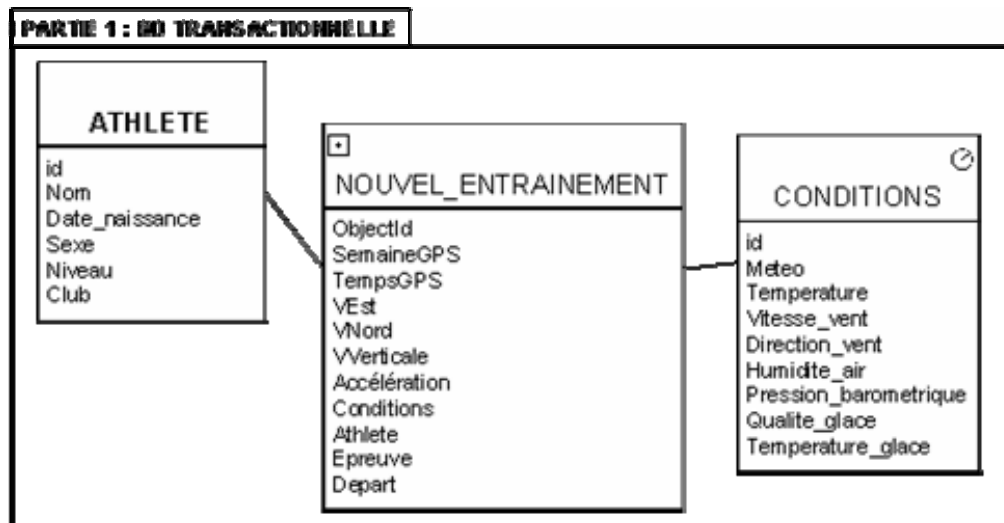


Figure 3-12 : Modèle physique de la base de données transactionnelle (BDT)

La table ATHLETE contient les informations relatives à un athlète tel son nom, sa date de naissance, son sexe, son niveau de compétition (provincial, national, international) et le nom du club auquel il appartient.

La table NOUVEL_ENTRAINEMENT contient les informations obtenues à partir des données enregistrées par le récepteur GPS : la semaine GPS ; le temps GPS (essentiel pour établir la correspondance entre un athlète et le groupe de positions GPS recueillies lors de son entraînement) ; les coordonnées cartographiques MTM (Transverse Mercator Modifiée) calculées à partir des coordonnées géographiques observées (latitude, longitude et altitude) et représentées par un pictogramme ponctuel sur la classe NOUVEL_ENTRAINEMENT de la Figure 3-12 ; les vitesses instantanées Est (V_{Est}), Nord (V_{Nord}) et Verticale ($V_{Verticale}$) ; et l'accélération instantanée. L'information sur la technique, tel le type de départ (Depart) utilisé, est directement inscrite dans la table. Un enregistrement dans la table CONDITION décrit les conditions météorologiques présentes lors de la journée d'entraînement. La qualité de la glace et sa température sont aussi inscrites dans cette table.

3.5.2 Module multidimensionnel

Le module multidimensionnel vise à faciliter l'analyse des données caractérisant un entraînement. L'architecture de ce module est basée sur la configuration ROLAP (cf. section 1.3.1) puisqu'elle est la plus utilisée jusqu'à maintenant lors de l'implantation d'une solution SOLAP. Le module multidimensionnel est formé de quatre composantes : (1) la base de données multidimensionnelles, (2) le traducteur multidimensionnel, (3) le serveur SOLAP et (4) le client SOLAP.

(1) La **base de données multidimensionnelle** (BDM) est structurée pour faciliter l'analyse et l'exploitation des données caractérisant un entraînement. La BDM est formée de données détaillées (**marché de données**) et de données agrégées des **cubes d'analyse**. Elle est implantée dans un SGBD relationnel puisque ce type de système caractérise une configuration ROLAP. Le SGBD sélectionné devait au départ supporter de grands volumes de données et nous avons ainsi choisi le SGBD relationnel de la compagnie *Oracle*.

Le **marché de données** est structuré selon un modèle multidimensionnel permettant la définition des cubes d'analyse (cf. section 1.3.2). Les données du marché de données sont seulement vouées à des fins de consultation et ne subissent aucune mise à jour, car l'historique des données est conservé. Dans notre cadre applicatif, le marché de données est construit pour répondre aux besoins d'une application sportive particulière (ex.: le patinage de vitesse). La Figure 3-13 montre la modèle d'implantation du marché de données pour l'application en patinage de vitesse.

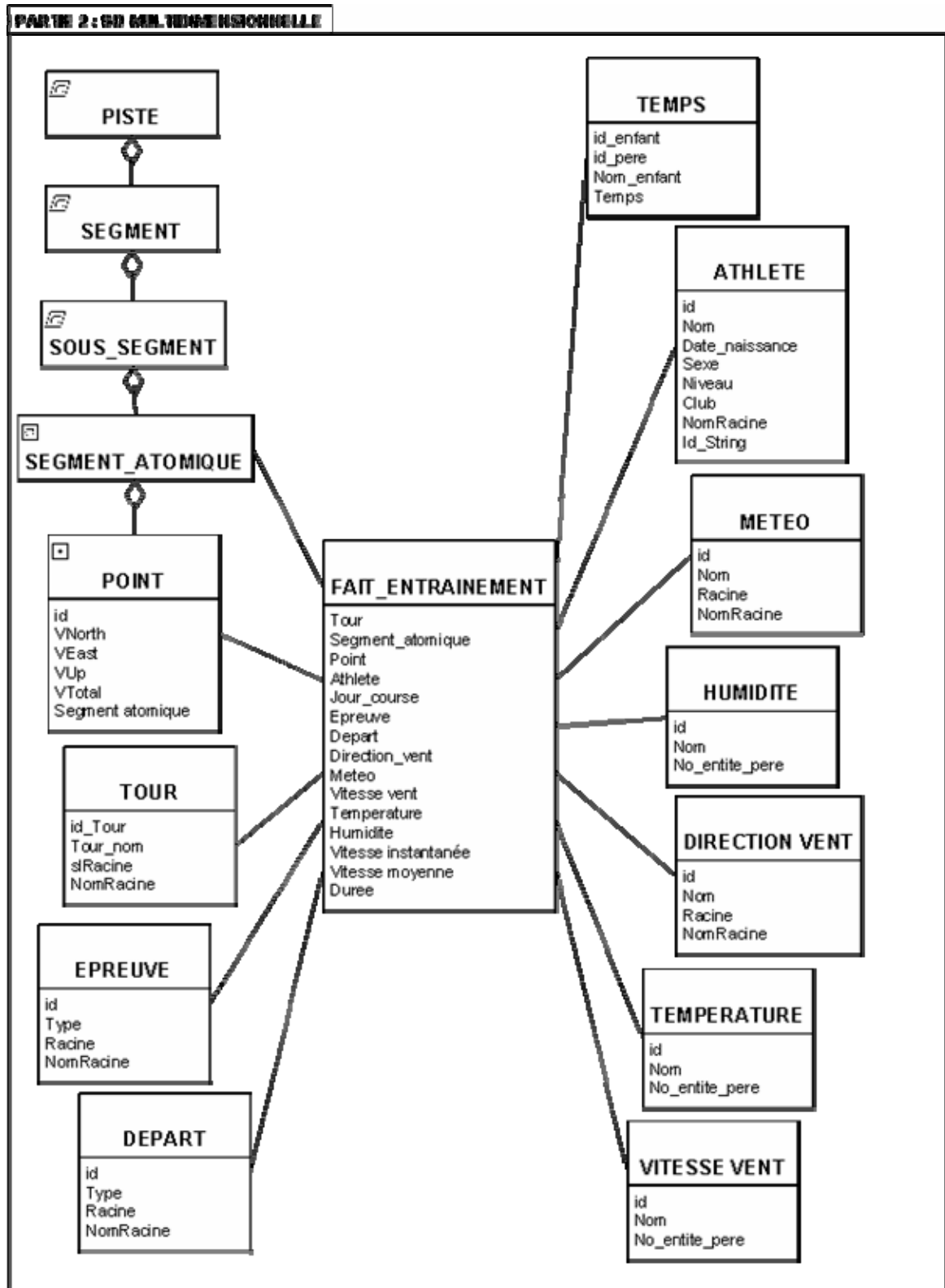


Figure 3-13 : Modèle physique du marché de données appartenant à la base de données multidimensionnelle (BDM)

La table des faits, FAIT_ENTRAINEMENT, est reliée à chacune des dimensions (EPREUVE, DÉPART, TEMPS, ATHLÈTE, METEO, HUMIDITÉ, DIRECTION DU VENT, TEMPÉRATURE, VITESSE_VENT) du modèle multidimensionnel. Cependant, deux liens existent entre la dimension spatiale DÉCOUPAGE PISTE. La dimension spatiale DÉCOUPAGE PISTE est illustrée sur la Figure 3-13 par ses cinq niveaux hiérarchiques, piste, segment, sous-segment, segment atomique et point. Un lien supplémentaire a été établi entre le niveau le plus détaillé de la dimension spatiale latente, segment atomique et la table des faits, car les agrégations sont calculées à partir de ce niveau. Cette particularité est expliquée à la section 3.6.2.

Les **cubes d'analyse**, essentiels pour l'analyse des données à partir d'un client SOLAP, sont construits à partir du marché de données et lors du calcul des agrégations (*cf.* section 1.3.3).

- (2) Le **traducteur multidimensionnel** assure le lien entre la BDT et la base de données multidimensionnelle (BDM). Il extrait les données présentes dans la table NOUVEL_ENTRAINEMENT de la BDT, transforme ces données pour les intégrer ensuite dans la base de données multidimensionnelle (BDM). Cette dernière opération se traduit par l'ajout de nouveaux faits dans la table des faits et/ou de nouveaux membres dans les dimensions du modèle multidimensionnel de la BDM. Le traducteur assure aussi la construction des cubes d'analyse (le calcul des nouvelles agrégations) suite à l'ajout de nouvelles données dans le marché de données. Étant donné que le SOLAP ne supporte pas présentement le calcul « à la volée » des agrégations, l'opération de reconstruction des cubes consiste à calculer le résultat de toutes les combinaisons de membres possibles (*cf.* section 1.3.3).
- (3) Le **serveur SOLAP** assure le lien entre le client SOLAP et les cubes d'analyses construits dans la BDM. Ces trois composantes correspondent en fait à l'architecture 3-tiers d'un SOLAP : un tiers CLIENT, un tiers SERVEUR D'APPLICATION (Serveur SOLAP) et un dernier tiers SERVEUR DE DONNÉES (BDM). Le serveur SOLAP met à la disposition d'un client, les données présentes et disponibles dans le serveur de données multidimensionnel (*Oracle* dans le cas présent). Ainsi, lorsqu'une requête est lancée sur

le client SOLAP, le serveur recueille le résultat dans les cubes d'analyse et l'affiche ensuite sur l'écran de l'utilisateur.

- (4) Le **client SOLAP** permet à l'utilisateur d'interagir directement avec les données présentes dans les cubes d'analyse. Sur l'interface cliente SOLAP, l'utilisateur construit ses analyses, lance ses requêtes et choisit le mode de représentation des résultats (graphique statistique, un tableau et/ou une carte).

Le client SOLAP développé au Centre de Recherche en Géomatique [Bédard et al. 2005] et commercialisé sous le nom *JMap Spatial OLAP Extension* par la société *KHEOPS Technologie*, est utilisé pour plusieurs applications. Il lit simplement les cubes d'analyse déjà définis (table des faits, dimensions et mesures) en fonction des besoins identifiés dans le cadre de l'application (*cf.* section 3.4) et construit (nécessite le calcul de toutes les agrégations a priori). Rappelons qu'un cube d'analyse SOLAP est défini suite à la sélection des dimensions et des mesures (modèle multidimensionnel), mais qu'il est construit grâce au pré-calcul et au stockage des agrégations. Le client SOLAP affiche un seul cube à la fois sur l'interface-cliente.

Un cube (*PATIN_DE_VITESSE_TECHNIQUE*, *cf.* section 3.7) pour des analyses sur les entraînements réalisés en patin de vitesse est ouvert sur l'interface de la Figure 3-14.

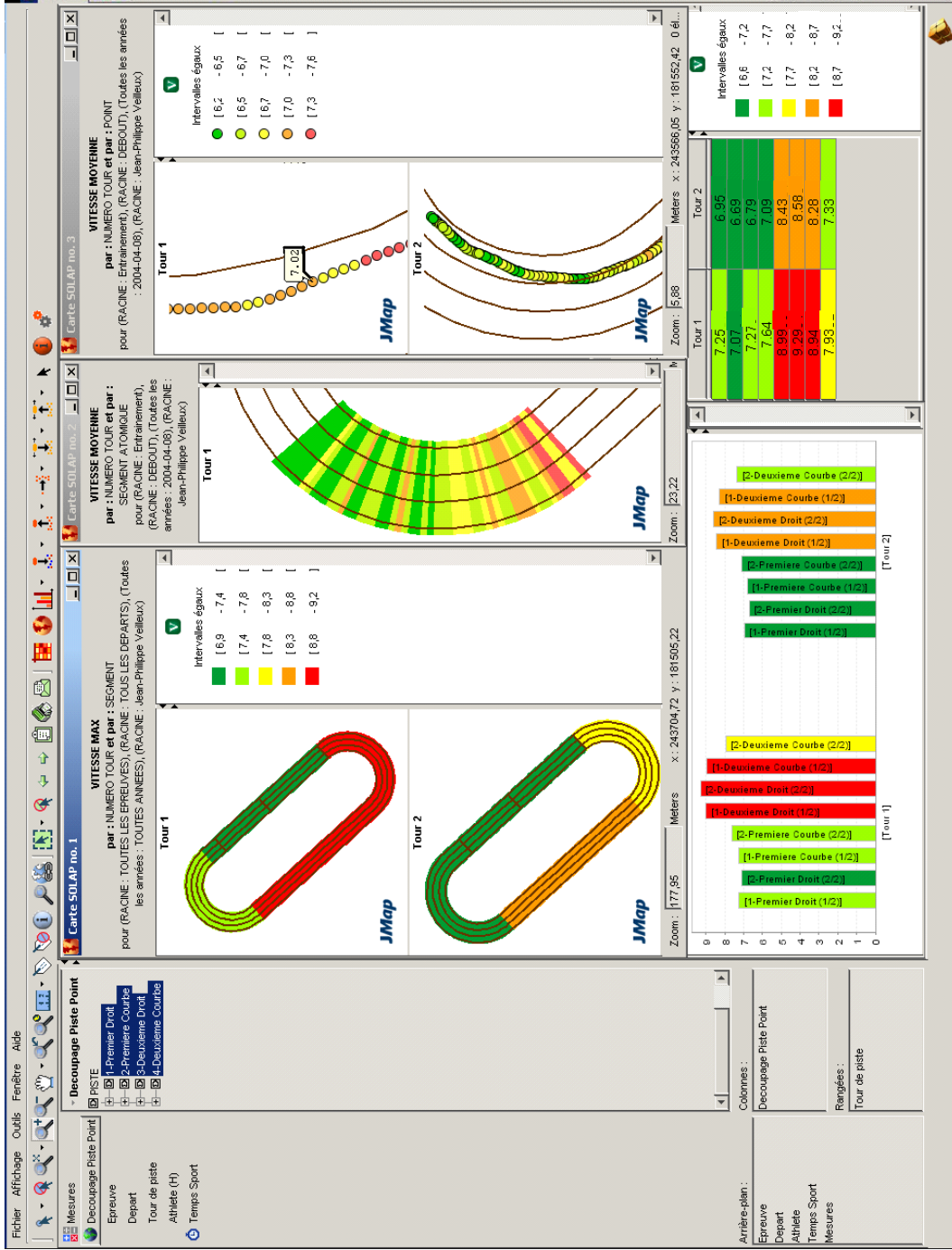


Figure 3-14 : Interface cliente SOLAP avec le cube PATINAGE DE VITESSE-TECHNIQUE
 Le choix, le nombre et la synchronisation des fenêtres ouvertes simultanément est au choix de l'utilisateur

Les six dimensions du cube (DÉCOUPAGE PISTE, ÉPREUVE, DÉPART, TOUR DE PISTE, ATHLÈTE et TEMPS) apparaissent sur la partie gauche de l'interface. L'entraîneur navigue à travers les données en sélectionnant un ou plusieurs membres sous les dimensions.

Les outils pour la navigation cartographique conventionnels (*zoom in*, *zoom out*, *pan*, etc.) et ceux pour la navigation SOLAP (forer, remonter, forer latéralement, etc.) sont disponibles en haut de l'interface. Tel que représenté sur la Figure 3-14 trois types d'affichage sont offerts sur l'interface-client SOLAP : tabulaire (vue en bas à droite), graphique statistique (vue en bas à gauche) et cartographique (les trois vues du haut). Les vues cartographiques sur la Figure 3-14 illustrent trois niveaux hiérarchiques de la dimension spatiale : niveau 2 – Segment (à gauche); niveau 4 – Segment atomique (au centre) et niveau 5 – Point (à droite) (*cf.* section 3.4.2).

La section suivante décrit en détail le fonctionnement du prototype SOLAP-SPORT à partir de l'acquisition des données jusqu'à leur analyse.

3.6 Fonctionnement du SOLAP-SPORT

Tel que mentionné au début de la section précédente le SOLAP-SPORT est un SOLAP juste-à-temps (*cf.* section 2.2.2). Il permet à l'entraîneur de choisir le moment où les nouvelles données doivent être ajoutées dans les cubes d'analyse et ainsi de les analyser que lorsqu'il en a besoin. L'étape de l'acquisition et du traitement des données spatiales et non-spatiales est présentement réalisée en temps différé.

Le fonctionnement du prototype SOLAP-SPORT est expliqué dans cette section en deux parties. La section 3.6.1 contient les détails du module transactionnel, alors que les étapes franchies dans le module multidimensionnel sont détaillées à la section 3.6.2. Actuellement, le prototype SOLAP-SPORT exploite seulement des données recueillies dans l'application en patinage de vitesse. Le fonctionnement sera alors expliqué en fonction de cette application, mais il est en général identique pour les autres applications sportives. Le

diagramme d'activités et le diagramme de séquence sont présents à l'0 et décrivent le fonctionnement global du prototype SOLAP-SPORT.

3.6.1 Partie transactionnelle

Lors de son entraînement, l'athlète porte un sac à dos dans lequel se trouve un récepteur GPS et une antenne GPS (*cf.* Figure 3-15).

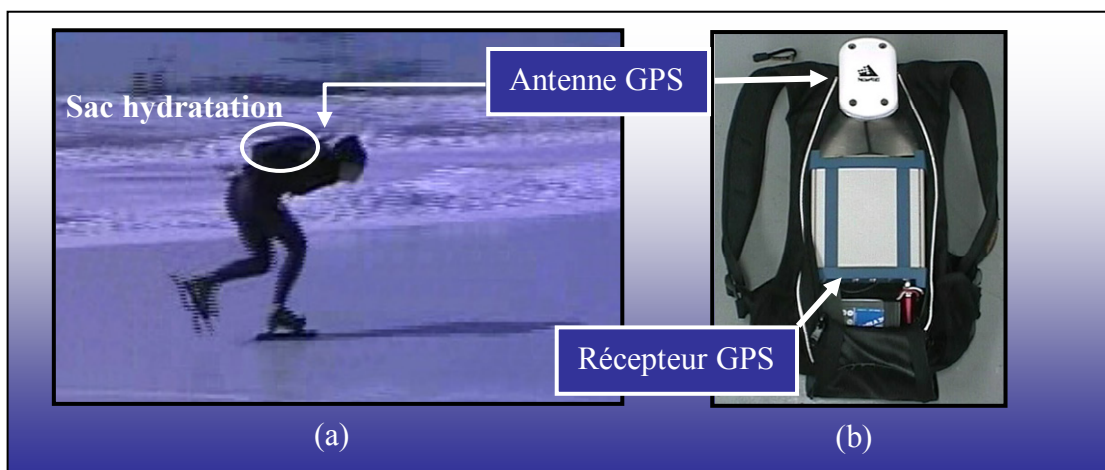


Figure 3-15 : Sac à dos porté par l'athlète

(a) L'athlète portant le sac à dos de type sac hydratation; (b) Le sac hydratation avec l'antenne GPS et le récepteur GPS

Une fois l'entraînement de l'athlète terminé (quelques tours de piste), l'entraîneur peut installer le sac à dos (sac hydratation) sur un deuxième athlète et poursuivre la collecte des données. Cependant, il doit noter l'heure exacte à laquelle le deuxième athlète a débuté son entraînement, information qu'il devra spécifier lors de la saisie des données sur le système source (*cf.* section 3.5.1). Après la séance d'entraînement, les observations GPS enregistrées dans le récepteur doivent alors être corrigées et filtrées. Le lecteur est référé à [Santerre 2001; Veilleux 2006] pour les détails sur les corrections et les filtres. Le fichier texte obtenu suite au post-traitement contient les données GPS traitées (temps GPS, position, vitesse et accélération, etc.). Ces données sont ensuite triées. Le triage des données GPS consiste à s'assurer que les données GPS sont seulement celles caractérisant réellement un entraînement (celles enregistrées après le début de l'entraînement jusqu'à sa

fin), puisque ces données seront celles diffusées sur le client SOLAP et utilisées pour le calcul des agrégations à différents niveaux

L'interface cliente du **système source** (cf. section 3.5.1) permet de recueillir l'information décrivant un entraînement (ex.: météo, technique, athlètes, etc.). C'est aussi à partir de cette même interface cliente que le fichier contenant les données GPS traitées, filtrées et triées est transféré dans le système.

L'intégrateur transactionnel traite et intègre les données dans la base de données transactionnelle (BDT) (cf. section 3.5.1). Étant donné qu'un fichier de données GPS contient habituellement plusieurs entraînements d'athlètes différents, l'un des traitements réalisés par l'intégrateur transactionnel est de lier l'athlète et les époques GPS représentant son entraînement. Cette liaison est possible seulement si l'entraîneur a saisi l'heure à laquelle l'athlète a débuté son entraînement sur le système source. Une fois le lien établi, il est inscrit physiquement dans la base de données pour les traitements subséquents. L'intégration consiste simplement à stocker les données traitées par le système dans la base de données transactionnelle.

L'utilisateur peut visualiser et vérifier les informations reliées à chaque entraînement stocké dans la BDT à partir des systèmes sources. Cependant, ce type de système n'est pas optimisé pour des analyses générales sur les données. Les données sont alors restructurées dans un modèle multidimensionnel pour être disponibles pour des fins d'analyse à partir d'un système SOLAP.

3.6.2 Partie multidimensionnelle

Lorsqu'un nouvel entraînement est saisi et inscrit dans la BDT, le système doit ajouter ces nouvelles données dans la BDM suite à la demande de l'utilisateur (entraîneur). Le rafraîchissement du marché de données doit s'effectuer le plus rapidement et efficacement possible. Les agrégations doivent aussi être recalculées (reconstruction des cubes) suite à l'ajout de nouvelles données dans le marché de données.

Deux processus sont alors réalisés par le **traducteur multidimensionnel** (*cf.* Figure 3-10) avant que l'entraîneur puisse analyser les performances des athlètes recueillies au cours de la journée : (1) le rafraîchissement du marché de données et (2) la reconstruction des cubes.

(1) Rafraîchissement du marché de données. Le premier processus vise à extraire les données récemment saisies avec les systèmes sources et caractérisant un nouvel entraînement de la BDT, les intégrer et les archiver dans le marché de données de la BDM. Quatre étapes sont en fait réalisées au cours du processus de rafraîchissement du marché de données : 1^{ère} Étape - Lecture de la structure BDT et BDM ; 2^e Étape - Lecture du niveau le plus détaillé de la dimension spatiale latente *DECOUPAGE DE LA PISTE* ; 3^e Étape – Lecture du nouvel entraînement dans la BDT ; 4^e Étape – Calcul des nouveaux faits ; 5^e Étape – Ajout des nouveaux faits et nouveaux membres dans la BDM.

Étape 1 Lecture de la structure BDT et BDM : Premièrement, le programme recueille dans un fichier XML (*eXtensible Markup Language*) la structure de la BDT et celle de la BDM (structure multidimensionnelle du marché de données). Le lien entre les deux bases est aussi fourni à l'intérieur du fichier XML. Ainsi, grâce à ces informations sur la structure les nouvelles données caractérisant un entraînement peuvent être extraites de la BDT et transférées dans la BDM. Un fichier DTD (*Document Type Definition*) décrivant la structure du fichier XML est fourni à l'0. On y retrouve aussi un extrait d'un fichier XML détaillant la structure de la base de données source (BDT) et celle de la base de données de destination (BDM) ainsi que le lien entre les deux.

Le format XML a été choisi, car on peut facilement mettre à jour les fichiers résultants dans ce format si la structure de l'une des bases de données est modifiée. Un programme pourrait éventuellement être développé pour créer automatiquement ce fichier XML suite à des modifications dans la structure de l'une des deux bases.

Étape 2 Lecture du niveau le plus détaillé de la dimension spatiale latente : La dimension spatiale latente correspond au découpage de la piste : Piste, Segment,

Sous-segment et Segment atomique (*cf.* section 3.4.2). Seulement les coordonnées des points délimitant le polygone représentant les segments atomiques (niveau le plus détaillé de la dimension spatiale latente) doivent être recueillies et chargées en mémoire. Ces coordonnées sont nécessaires pour retrouver des données sur la performance de l'athlète (*ex.*: vitesse moyenne, accélération moyenne) pour chaque segment atomique. Les agrégations seront calculées ensuite à partir de ce niveau hiérarchique.

En effet, les agrégations ne sont pas calculées à partir des valeurs instantanées retrouvées au niveau Point, niveau le plus détaillé du marché de données, car les membres de ce niveau sont trop nombreux (un point = un membre). De plus, à chaque entraînement, de nouveaux membres sont ajoutés à ce niveau (Point) alors qu'aucun membre n'est ajouté au niveau Segment atomique. Étant donné que le calcul des agrégations doit être lancé suite à chaque entraînement, une solution devait alors être trouvée pour diminuer le nombre de combinaisons possibles entre les membres et ainsi augmenter la vitesse de calcul des agrégations du cube. Cependant, cette stratégie implique que, lors des analyses sur le client SOLAP, les mesures sur chaque position instantanée (membre du niveau Point) sont offertes seulement lorsqu'au moins un membre du niveau le plus détaillé de chaque dimension est sélectionné. De plus, les mesures devaient alors nécessairement être calculées pour chaque segment atomique (niveau le plus détaillé de la dimension spatiale latente) lors du transfert des données dans le marché de données, puisque toutes les agrégations seraient ensuite calculées à partir des valeurs retrouvées au niveau Segment atomique.

Puisque les segments atomiques ont été créés avec le logiciel *Microstation* de *Bentley*, les données étaient initialement en format DGN. Pour faciliter la lecture des coordonnées délimitant les segments atomiques par notre programme, une transformation de format de données a été nécessaire. Le format *IGES* (<http://www.nist.gov/iges/>) a été choisi. Le format *IGES* est un format neutre qui permet le transfert des données dans un système différent. L'unité fondamentale de ce format est l'entité. Les entités sont soit géométriques ou non-géométriques.

Les entités géométriques sont des points, des courbes, des surfaces, des solides, ou le regroupement de ces formes. Ainsi, le format *IGES* fournit les coordonnées de chaque surface (segment atomique) dans un fichier texte et facilite la lecture de la géométrie du niveau le plus détaillé de la dimension spatiale latente. La Figure 3-16 présente un exemple de fichier IGES (.igs).

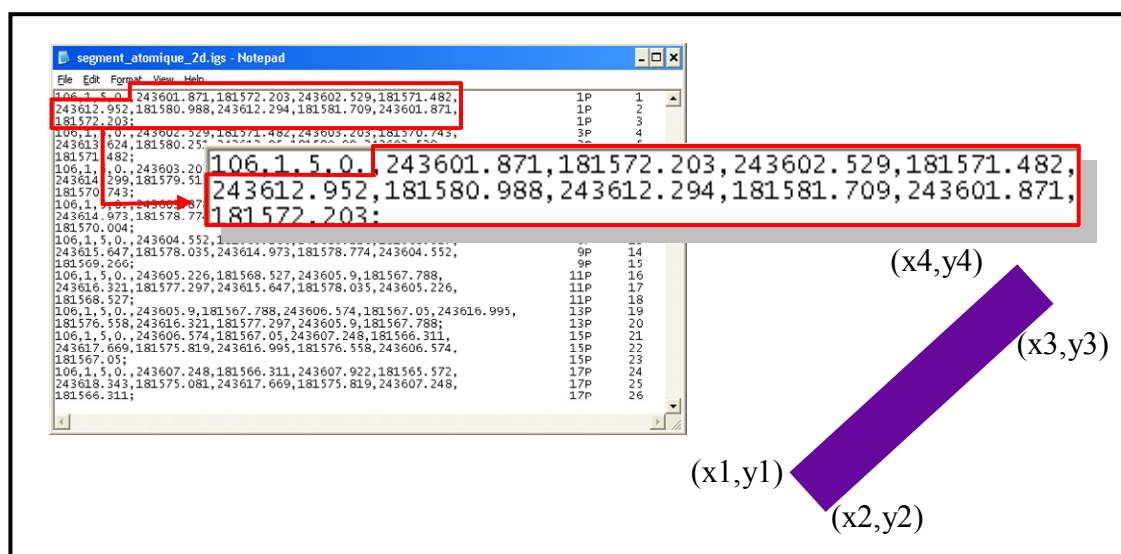


Figure 3-16 : Exemple d'un format IGES représentant un segment atomique

L'agrandissement de la Figure 3-16 correspond à un segment atomique dont les coordonnées des points délimitant cette surface débutent après le caractère « 0. » : $(x1,y1) = (243601.871, 181572.203)$; $(x2,y2) = (243602.529, 181571.482)$; $(x3,y3) = (243612.952, 181580.988)$; $(x4,y4) = (243612.294, 181581.709)$. Les coordonnées du dernier point sont identiques à celles du premier point ce qui permet de refermer la surface représentée sur la Figure 3-16. Une fois les coordonnées de segment atomique recueillies, sa géométrie est gérée en *Java*TM par l'api *JTS Topology Suite* [VIVID Solutions 2005] permettant ainsi de réaliser des opérations spatiales (ex : intersection).

Étape 3 Lecture d'un nouvel entraînement dans la BDT : Dans la table NOUVEL_ENTRAINEMENT de la base de données transactionnelle (cf. Figure 3-12), le programme récupère les coordonnées cartographiques MTM (données spatiales

non latentes), les vitesses instantanées correspondantes (V_{Nord} , V_{Est} , $V_{\text{Verticale}}$) et les informations saisies par l'entraîneur caractérisant la journée de l'entraînement (*ex.*: nom athlète, l'heure du début de son entraînement, les conditions météorologiques, le type de départ utilisé, etc.) et les charge en mémoire. Ces informations seront utilisées pour construire les nouveaux faits et peupler ainsi la table des faits.

Étape 4 Calcul des nouveaux faits : Pour construire les nouveaux faits quelques traitements sont nécessaires. Les faits doivent être obtenus au niveau de granularité du marché de données.

Calcul vitesse totale instantanée

Pour construire les nouveaux faits, la vitesse instantanée (V_{Totale}) de chaque position instantanée doit auparavant être calculée à l'aide du théorème de Pythagore :

$$V_{\text{Totale}} = \sqrt{V_{\text{North}}^2 + V_{\text{East}}^2 + V_{\text{Up}}^2}$$

Équation 3-1 Calcul de la vitesse instantanée totale à l'aide du théorème de Pythagore

Identification du segment atomique

Chaque position instantanée doit ensuite être associée au segment atomique dans lequel elle se situe (recueilli et chargé en mémoire à la 2^e Étape) de la piste à l'aide de l'opérateur *within* de l'API *JTS Topology Suite*. L'opération *within* indique si un objet géographique (ici la position instantanée) se situe à l'intérieur ou non d'un autre objet géographique (ici un segment atomique). Ainsi, chaque position instantanée est nécessairement associée à un segment atomique. Toutefois, un segment atomique peut contenir plus d'une position instantanée. La moyenne des quantités sur la performance (vitesse, accélération) associés à chacune des positions instantanées situées dans un même segment atomique et appartenant au même tour de piste est calculée et attribuée au segment atomique. Ce résultat

permet ensuite de calculer les agrégations (construction du cube) directement à partir de ce niveau de détail (segment atomique). Pour accélérer la recherche du segment atomique contenant la position instantanée, une indexation des segments atomiques avec les sous-segments de la piste a été réalisée (Figure 3-17).

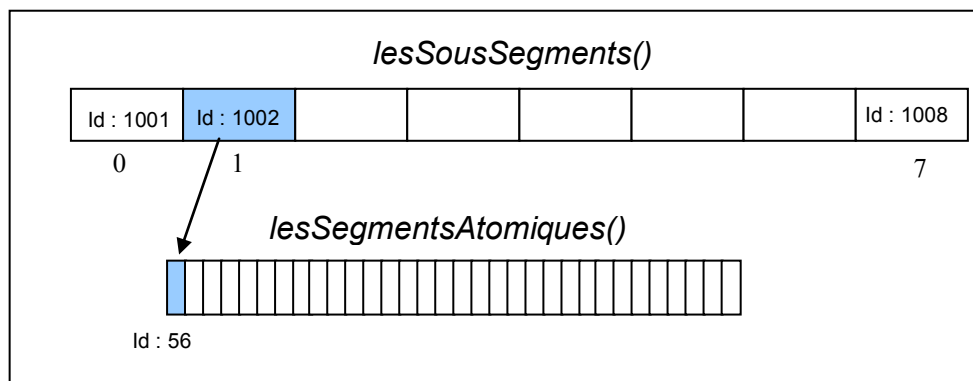


Figure 3-17 : Indexation à partir des sous-segments de la piste

Le vecteur `lesSousSegments()` de la Figure 3-17 contient tous les identifiants des sous-segments de la piste. Chaque sous-segment est associé à un vecteur (`lesSegmentsAtomiques()`) contenant tous les identifiants des segments atomiques qu'il contient. Cette indexation a permis d'accélérer d'un facteur cinq les temps de traitement pour retrouver le segment atomique contenant la position instantanée traitée. Une fois que toutes les époques d'un même tour (*cf.* paragraphe suivant) sont retrouvées pour un segment atomique, la vitesse moyenne, l'accélération moyenne et la durée sont calculées à partir des mesures instantanées.

Division en tour

Évidemment, le groupe de positions GPS de la BDT doit être divisé en tour de piste (Tour#1, Tour#2, etc.) et ce, en fonction de chaque athlète entraîné. La ligne utilisée pour diviser l'entraînement en tour, Ligne-tour, est indiquée en rouge sur la Figure 3-18 section (a).

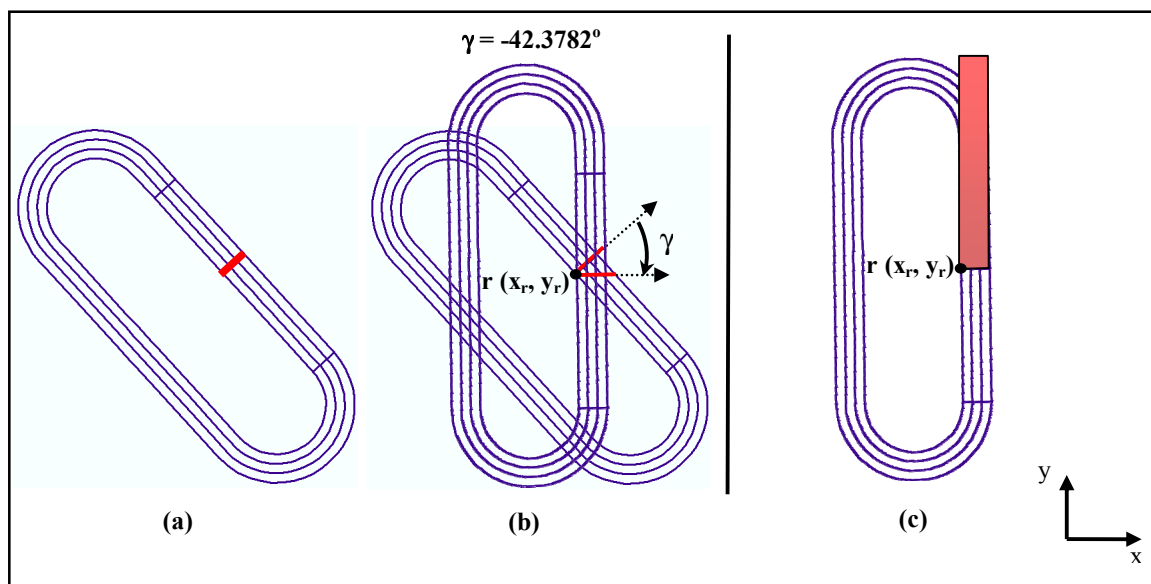


Figure 3-18 : Tour de piste

(a) Ligne-tour utilisée pour diviser les entraînements en tour de piste (Tour#1, Tour#2, etc.);
 (b) Rotation de la piste de -42.3782 degrés autour de l'axe de z sur le point r ; (c) zone de positionnement possible d'un point dont l'abscisse et l'ordonnée sont supérieures à celles du point de rotation

Pour identifier plus facilement si une position instantanée appartient à un nouveau tour, la piste est tout d'abord orientée verticalement grâce à une rotation négative de 42.3782° autour de l'axe des Z sur le point de rotation r (Figure 3-18 section (b)). Cet angle de rotation a été calculé directement dans logiciel *Microstation* de la compagnie *Bentley*. Une fois la rotation effectuée, il suffit de vérifier si la position instantanée traitée possède une abscisse (x_p) et une ordonnée (y_p) supérieures à celles du point de rotation r ($x_p > x_r$ Et $y_p > y_r$). Si c'est le cas, la position instantanée se situe nécessairement dans la zone illustrée en rouge sur la Figure 3-18 section (c). Si la position précédente était située sous la Ligne-tour (y_p précédent $< y_r$), la position instantanée traitée appartient alors à un nouveau tour de piste.

Pour réaliser la rotation, la matrice de rotation autour de l'axe des Z avec un angle de rotation γ (γ) de -42.3782° a été utilisée (Équation 3-2).

$$R_z(\gamma) = \begin{bmatrix} \cos(\gamma) & \sin(\gamma) & 0 \\ -\sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Équation 3-2 : Matrice de rotation autour de l'axe des z

Cependant, avant d'appliquer la rotation avec cette matrice de rotation, la piste doit être ramenée, par translation, à l'origine du système de coordonnées auquel elle appartient (MTM). L'extrémité intérieure de la Ligne-Tour (point de rotation r) a été utilisée pour positionner la piste à la coordonnée (0,0). Une translation de -243640.89 m en X et de -181530.92 m en Y a alors été nécessaire. Ces valeurs correspondent respectivement à l'inverse de l'abscisse (x_r) et à celle de l'ordonnée (y_r) du point de rotation.

Étape 5 Ajout des nouveaux faits et nouveaux membres dans la BDM : Les performances (vitesse, accélération, durée) calculées pour chaque segment atomique sont utilisés pour peupler le marché de données : (1) de nouveaux faits sont ajoutés dans la table des faits et dans certains cas; (2) de nouveaux membres sont ajoutés dans les dimensions changeantes.

- (1) Un nouveau fait pourrait être : Le 11 mars 2005, lors d'un entraînement, l'athlète Sébastien Fleury a obtenu, au cours de son 2^e tour de piste, à l'entrée de la courbe (segment atomique #44) une vitesse moyenne de 7.3 m/s.
- (2) Les dimensions changeantes sont les dimensions qui peuvent subir des ajouts de membres à une fréquence élevée ou faible. La dimension TEMPS est une dimension changeante à haute fréquence, car des membres y sont ajoutés fréquemment. En effet, un nouveau membre, la date de l'entraînement (ex.: 11 mars 2005), est ajouté dans la dimension TEMPS à chaque nouvel ajout de données. Une dimension changeante à faible fréquence est la dimension ATHLÈTE.

Une fois le rafraîchissement du marché de données terminé, le traducteur multidimensionnel lance immédiatement le processus de **reconstruction des cubes** d'analyses.

- (2) **Reconstruction des cubes d'analyse.** Afin d'effectuer la reconstruction des cubes d'analyse, il est nécessairement de connaître, au préalable, la définition (dimensions, leurs niveaux hiérarchiques et les mesures du cube) de chaque cube d'analyse présent dans la BDM. Cette information est recueillie dans un fichier au format XML. Ce format est déjà utilisé dans la littérature [Jensen, Møller et al. 2001; Niemi, Niinimäki et al. 2002; Lambert 2004] pour définir la structure des cubes d'analyse. L'0 contient le fichier *Cubes.dtd* et un extrait du fichier *Cubes.xml* décrivant la structure des cubes d'analyse.

Une fois l'information recueillie, de nouvelles agrégations sont automatiquement calculées. Le résultat des requêtes agrégatives est alors pré-calculé et stocké à l'intérieur de la BDM. Les données agrégées résultantes accélèrent les temps de réponse lors des analyses sur le client SOLAP. Actuellement, le SOLAP nécessite que tous les résultats de requêtes possibles (combinaisons de membres) (*cf.* section 1.3.3) soient pré-calculés et stockés dans la BD.

Pour le calcul des agrégations, le logiciel *SAS* (www.sas.com) a été utilisé. Ce logiciel particulièrement puissant calcul tous les résultats de combinaisons de membres possible d'un cube d'analyse et les stocke ensuite dans la BDM (données agrégées). Les combinaisons de membres possibles sont fournies dans un fichier généré automatiquement par notre programme à partir des informations recueillies dans le fichier *Cubes.xml*.

Calculer le résultat de toutes les combinaisons de membres possibles n'est cependant pas une solution optimale, car elle demande des temps de traitement assez longs (souvent quelques heures). Les méthodes de reconstruction rapide des cubes SOLAP présentées à la section 1.4.3 bien qu'intéressantes, ne pouvaient être utilisées dans le cadre de l'application. Les prochains paragraphes expliquent les raisons.

La méthode visant à matérialiser seulement les requêtes les plus fréquentes et les requêtes les plus coûteuses en temps de calcul ne peut être suivie puisque l'outil SOLAP utilisé ne permet pas encore le calcul des agrégations « à la volée ». En effet, le logiciel SOLAP exploite actuellement une configuration ROLAP exigeant que toutes les agrégations soient stockées dans la base de données. Ainsi, les résultats de toutes les combinaisons de membres possibles doivent être pré-calculés.

Dans le cadre de la présente recherche, la méthode *Lazy aggregates* qui consiste à établir des seuils de tolérance à l'intérieur du système aurait pu être utilisée. Les seuils de tolérance déterminent le moment où il est significatif de recalculer le résultat des requêtes agrégatives à partir de l'entrepôt de données nouvellement rafraîchi. Ils sont basés sur les différences qu'apporterait un niveau très détaillé (*ex.*: jour) à un niveau plus général (*ex.*: mois). En effet, si les données d'un seul entraînement n'apportent pas de différences significatives au niveau supérieur (*ex.*: semaine), il n'est pas nécessaire de recalculer les agrégats. Cependant, l'un des besoins des entraîneurs était d'obtenir rapidement des valeurs exactes sur la performance d'un athlète (*cf.* section 3.1.1). Il devenait alors inapproprié d'utiliser cette méthode qui est plus dédiée à des applications où l'utilisateur recherche des ordres de grandeur.

Les temps de calcul des agrégations suite à l'ajout de nouvelles données dans la BDM doivent être évalués pour s'assurer que le prototype répond aux besoins des entraîneurs. La section présente les tests de performance réalisés.

3.7 Tests de performance et solution retenue

Le prototype SOLAP-SPORT est un SOLAP juste-à-temps (*cf.* section 2.2.2). Le traducteur multidimensionnel sera alors lancé seulement suite à la demande de l'utilisateur. Ce traitement peut être lancé en tout temps.

Des tests de performance ont été effectués avec le cube PATINAGE DE VITESSE-ENTRAÎNEMENT pour vérifier si le traducteur multidimensionnel du système était fonctionnel et offrait des temps de réponse acceptables. Le cube PATINAGE DE VITESSE-ENTRAÎNEMENT est défini à partir des 11 dimensions illustrées sur la Figure 3-4. Les

mesures offertes par ce cube sont la VITESSE INSTANTANÉE, la VITESSE MOYENNE, la VITESSE MINIMALE, la VITESSE MAXIMALE, l'ACCÉLÉRATION et la DURÉE. Pour vérifier la performance du module multidimensionnel, sept entraînements ont été ajoutés à partir du système source.

Comme le montre le Tableau 3-1 pour l'entraînement #1 et l'entraînement #5, le temps de rafraîchissement du marché de données (cf. section 3.6.2) est assez rapide (8 et 61 secs). Cependant, bien que logiciel *SAS* utilisé soit très puissant, la reconstruction des cubes (cf. section 3.6.2) demande quelques heures.

Entraînement	Nouvel Entraînement	Rafraîchissement du marché de données	Table des faits	Reconstruction du cube	Grandeur du cube
Id	nb enr.	sec	nb enr.	hh:mm:ss	nb enr.
1	2198	8	864	1:54:38	6 894 720
...					
5	22699	61	17239	3:13:00	94 312 300
...					

Tableau 3-1 : Tests de performance réalisés avec le cube
PATINAGE DE VITESSE-ENTRAÎNEMENT
avec *Processeur Intel Pentium 4 CPU 3,60 GHz; Mémoire vive 3,12 GB*

Cette performance offerte par le logiciel *SAS* pour la reconstruction du cube est tout de même bien supérieure à celle obtenue avec un algorithme que nous avons développé en *Java*TM. Le temps de reconstruction du cube demandait alors plusieurs jours.

Inévitablement, au fur et à mesure que de nouveaux entraînements sont ajoutés, le temps de réponse augmente et devient inacceptable pour l'application. Bien que le SOLAP-SPORT soit un SOLAP juste-à-temps, ce qui permet aux entraîneurs de choisir le moment où le traducteur multidimensionnel est lancé, une attente de quelques minutes est acceptable (ex.: le temps que l'athlète passe au vestiaire). Cependant, une attente de plusieurs heures risque de provoquer une non-utilisation du système. Une solution doit alors être trouvée.

Des avenues de solutions étaient : développer un algorithme pour le calcul des agrégations plus puissant, réaliser des traitements en parallèle ou modifier l'aspect conceptuel des cubes d'analyse. Le logiciel *SAS* est actuellement l'un des logiciels les plus puissants pour

réaliser la reconstruction d'un cube d'analyse (calcul des agrégations). Développer un algorithme plus puissant était impensable. La solution retenue est alors la suivante.

Solution retenue

La solution retenue se divise en deux parties : (1) Définition de plus petits cubes d'analyse; et (2) Construction d'un cube ACTUEL et un cube HISTORIQUE.

(1) Définition de plus petits cubes. La première partie de la solution trouvée touche l'aspect conceptuel des cubes d'analyse. Elle consiste à définir de plus petits cubes, c'est-à-dire des cubes formés d'un nombre de dimensions moins élevé. Chaque petit cube redéfini ne satisfait pas tous les besoins spécifiques d'analyse identifiés auprès de l'entraîneur (*cf.* section 3.1.2), mais offre les analyses qui seront fait probablement le plus souvent réalisées. Un cube TECHNIQUE et un cube MÉTÉO ont alors été définis. Le cube TECHNIQUE permet de comparer les différentes techniques utilisées lors des entraînements alors que le cube MÉTÉO offre plutôt des analyses comparatives selon les conditions météorologiques présentes. Les deux cubes permettent de comparer les athlètes et comparer les entraînements dans le temps, selon le numéro de tour ou la section de la piste. Les dimensions des cubes TECHNIQUE et MÉTÉO se retrouvent à l'0.

Les résultats obtenus lors des tests de performance sont maintenant plus satisfaisants. Le Tableau 3-2 présente les résultats avec le cube TECHNIQUE (6 dimensions). Maintenant la reconstruction du cube demande seulement quelques minutes.

Entraînement	Nouvel Entraînement ds BDT	Rafraîchissement du marché de données	Table des faits	Reconstruction du cube	Grandeur du cube
Id	nb enr.	sec	nb enr.	hh:mm:ss	nb
1	2 198	7.812	864	0:01:18	63 840
2	5 660	16.874	2 975	0:01:20	187 818
3	9 181	21.141	5 647	0:01:27	306 186
4	13 528	37.546	10 419	0:01:34	465 170
5	22 699	48.109	17 239	0:01:39	612 238
6	28 359	62.452	26 298	0:01:49	789 066
7	37 540	80.717	38 127	0:02:07	1 161 430

Tableau 3-2 : Tests de performance réalisés avec le cube TECHNIQUE
avec Processeur Intel Pentium 4 CPU 3,60 GHz; Mémoire vive 3,12 GB

Évidemment, le temps de reconstruction des cubes est directement proportionnel au nombre de faits dans la table des faits. Donc, au fur et à mesure que de nouveaux entraînements sont ajoutés dans le marché de données, le temps de reconstruction du cube augmente. Même si la vitesse des processeurs double presque tous les 18 mois [Moore 1965] et que les mémoires principales et secondaires, qui étaient un frein important au développement d'applications représentant beaucoup de calculs et contenant de grands volumes de données, ne sont presque plus un problème [Sadeg 2004] une alternative devait être trouvée.

(2) Création d'un cube ACTUEL et un cube HISTORIQUE. La deuxième partie de la solution retenue vise à accélérer les temps de reconstruction des cubes d'analyse en définissant un cube contenant seulement les entraînements effectués au cours de la journée (*ex.*: CUBE TECHNIQUE_ACTUEL). Un deuxième cube pour les données d'archives présentes dans le marché de données (*ex.*: CUBE TECHNIQUE_HISTORIQUE) est aussi construit. La solution peuplement progressif et partitionnement présentée à la section 1.4.2 nous a ici inspirée. Les données sur les performances recueillies au cours de la journée sont directement transférées par le traducteur multidimensionnel dans un modèle multidimensionnel de transfert possédant la même structure que le marché de données. Un cube ACTUEL est alors construit (calcul des agrégations) à partir des données de ce modèle. Ainsi, l'entraîneur analyse très rapidement les données nouvellement recueillies grâce au cube ACTUEL puisque peu de données sont traitées lors de sa reconstruction. L'entraîneur peut demander la reconstruction du cube

HISTORIQUE à partir des données présentes dans le marché de données durant la nuit, puisqu'il demande évidemment des temps de traitement plus longs (quelques heures).

Suite à la mise en place de ces solutions, le SOLAP-SPORT demeure un SOLAP juste-à-temps. Le SOLAP-SPORT permet à l'entraîneur de choisir le moment où il désire rafraîchir les cubes d'analyse en fonction de ce qu'il veut analyser et cette opération peut être réalisée en tout temps. Il peut ainsi choisir, à n'importe quel moment, si le cube ACTUEL et/ou le cube HISTORIQUE doivent être construits. Le fonctionnement du SOLAP-SPORT utilisant cette solution pour optimiser la performance des traitements est décrit par un diagramme d'activités et de séquence à l'0.

La section 3.8 fournit des éléments permettant de valider l'approche adoptée pour l'élaboration du prototype SOLAP-SPORT.

3.8 Présentation et validation du prototype SOLAP-SPORT

Lors d'une présentation du prototype au Département de kinésiologie à l'automne 2004, l'interface cliente générale du SOLAP ainsi que les dimensions et les mesures offertes par le SOLAP-SPORT ont pu être validés.

Premièrement, les professionnels du Département de kinésiologie ont appréciés l'interface générale d'un outil SOLAP puisqu'elle est facile d'utilisation et très intuitive. Deuxièmement, le volet spatial de l'interface utilisateur leur permettait de visualiser l'endroit exact où les données sur la performance ont été atteintes. Ces deux aspects répondaient en effet à deux de leurs besoins généraux identifiés dès le départ (*cf.* section 3.1).

Les dimensions thématiques, la dimension spatiale et les mesures offertes sur l'interface cliente SOLAP devaient répondre aux besoins d'analyse d'un entraîneur de patinage de vitesse (*cf.* section 3.1.2). Aucun entraîneur de patinage de vitesse n'a pu assister à cette rencontre. Les professionnels de kinésiologie ont validé les dimensions et les mesures choisies qui avaient au départ été sélectionnées lors d'une rencontre avec l'entraîneur. Lors de cette rencontre, le niveau le plus détaillé de la dimension spatiale était le niveau Segment

atomique. Ils ont alors souligné l'importance d'avoir accès au niveau le plus détaillé offert par le GPS (niveau Point) et l'intérêt de rajouter ce niveau dans la dimension spatiale DÉCOUPAGE DE LA PISTE. En effet, les valeurs instantanées sont essentielles pour les analyses de la performance (cf. section 3.1.1) et les positions instantanées essentielles pour décrire la trajectoire d'un athlète (cf. section 3.1.2). D'autres besoins se sont manifestés comme avoir la possibilité d'obtenir des graphiques de la vitesse instantanée en fonction du temps. Maintenant, la performance obtenue à chaque position instantanée (ex.: vitesse instantanée) est offerte sur le client SOLAP et elle peut être affichée sur un graphique (ex.: vitesse instantanée en fonction de chaque position instantanée).

Pour montrer les possibilités de l'outil SOLAP exploitant un cube défini et construit pour l'application en patinage de vitesse, prenons un exemple concret. Un entraîneur désire comparer les performances de deux athlètes, François et Jean-Philippe, obtenues lors des entraînements du mois d'Avril de l'année 2004. Il désire comparer la vitesse moyenne atteinte dans une section bien précise de la piste, la première moitié du 2^e droit (niveau Sous-segment) et seulement pour le 2^e tour de piste. Pour ce faire, l'entraîneur doit ouvrir le cube `PATIN_DE_VITESSE_TECHNIQUE_HISTO` et sélectionner sur l'interface cliente les membres suivants :

- Vitesse moyenne sous le bouton **Mesures**
- 2^{eme} Droit (1/2) sous le bouton **Découpage Piste**
- Entraînement sous le bouton **Épreuve**
- Tour#2 sous le bouton **Tour de Piste**
- François et Jean-Philippe sous le bouton **Athlète**
- Avril sous le bouton **Temps**

Ensuite, il suffit d'appuyer sur le bouton « Carte » pour obtenir les résultats. Le résultat de cette analyse a été obtenu avec 21 clics de la souris sur l'interface utilisateur SOLAP. Ce nombre peut paraître élevé, mais le même résultat d'analyse nécessite près de 60 clics avec un logiciel SIG classique. De plus, un seul type d'opération, la sélection de membres, doit être réalisé sur l'interface cliente SOLAP. Or, sur une interface SIG, plusieurs fonctions différentes doivent être utilisées (*Definition Query, Query Builder, Select By Attributes, Select By Location, Statistics, etc.*) et nécessitent une bonne connaissance du logiciel. Les

détails de cette analyse réalisée sur le client SOLAP versus sur le SIG *ArcGIS* de la compagnie *ESRI* sont présents à l'0.

Supposons, que l'entraîneur désire maintenant approfondir son analyse et obtenir plus de détails. Sur le client SOLAP, il obtient la vitesse moyenne pour chaque segment atomique composant le demi-droit en forant (outil forer) sur la dimension spatiale directement à partir du volet cartographique (1 seul clic). Avec un logiciel SIG, cette variante nécessite plusieurs étapes supplémentaires.

Sur le client SOLAP, l'entraîneur obtient rapidement ses résultats d'analyse à l'aide de simples clics souris pour effectuer des sélections et obtenir l'affichage de différentes vues (tabulaires, graphiques, cartes). Il peut alors conserver son flux de pensée lors de ses analyses et tirer des conclusions qui n'auraient pu être possibles autrement. Par exemple, comme le montre la Figure 3-19, il semble y avoir une corrélation entre la position de l'athlète et les pics de vitesse.

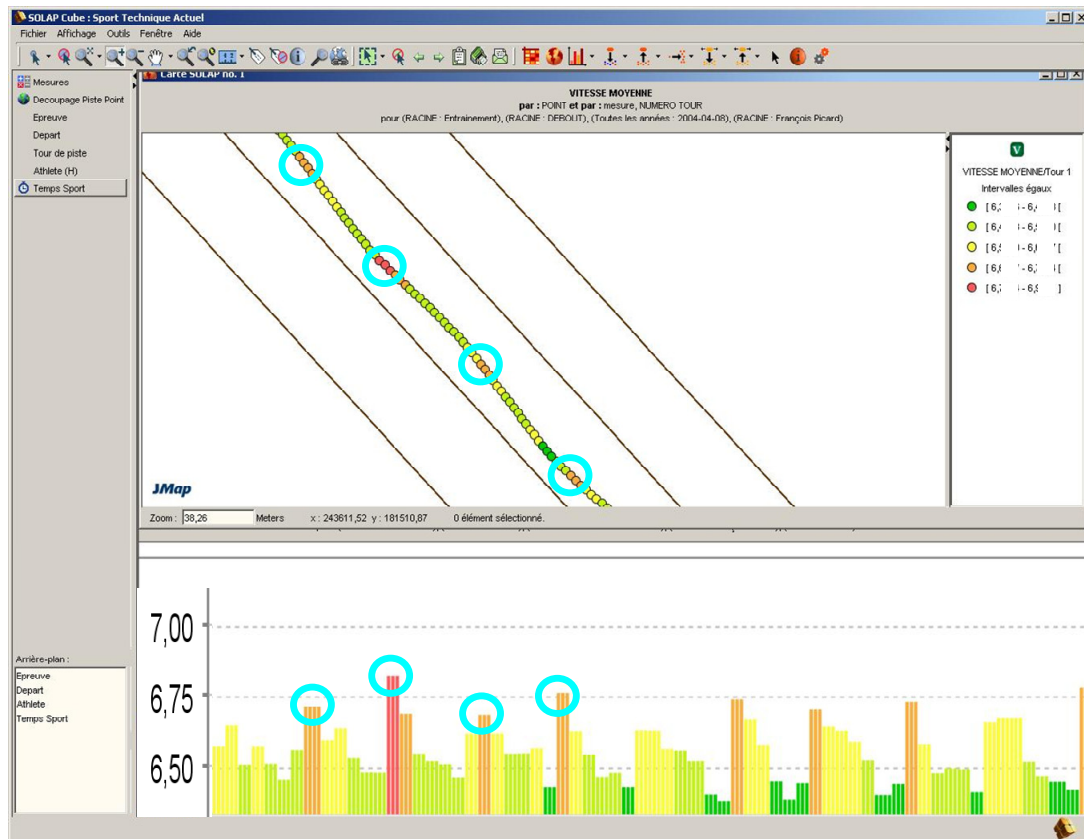


Figure 3-19 : Corrélation possible entre les pics de vitesse et la position de l'athlète dans son trajet

Les vitesses maximales (points rouges ou points oranges) sont toujours atteints dans un creux ou une crête du parcours sinusoïdal de l'athlète affiché sur la vue cartographique (vue du haut sur la Figure 3-19). Les vues de la Figure 3-19 nous permettent alors de confirmer en seulement quelques secondes que les pics de vitesse retrouvés sur un graphique de la vitesse instantanée en fonction du temps semblent bien correspondre au coup de patin de l'athlète. Cette corrélation permet à l'entraîneur d'analyser visuellement la technique de l'athlète.

L'ajout de nouveaux membres et de nouveaux faits dans le système est certainement le besoin le plus important aux yeux du personnel de kinésiologie. Le SOLAP-SPORT, tel qu'il fonctionne actuellement, offre cette possibilité. En effet, il permet l'ajout de nouveaux entraînements (nouveaux faits) et de nouveaux membres dans les dimensions changeantes (TEMPS et ATHLÈTE) facilement et suffisamment rapidement pour l'application (cf. section 3.6).

Cependant, le prix d'un tel système d'analyse n'est actuellement pas abordable pour un seul entraîneur (40 000 \$ au total environ). L'une alternative à l'achat du logiciel pourrait être la location du système d'analyse à des moments stratégiques de l'année d'entraînement.

3.9 Conclusion

Le Chapitre 3 a permis d'identifier les besoins généraux des entraîneurs pour l'analyse de la performance des athlètes pratiquant un sport extérieur. Les besoins spécifiques à un entraîneur de patinage de vitesse ont aussi été identifiés ainsi que les données et les systèmes existants inventoriés. Ces deux étapes nous ont permis de constater qu'un nouveau système d'analyse devait être sélectionné. En considérant les besoins généraux identifiés chez les entraîneurs et les besoins spécifiques d'analyse d'un entraîneur de patinage de vitesse, la solution SOLAP est présentement la plus appropriée pour l'analyse de la performance des athlètes de haut niveau moyennant quelques adaptations ou l'ajout de fonctionnalités.

Le prototype SOLAP-SPORT offre des possibilités d'analyse bien supérieures à celles qui étaient offertes par les systèmes existants (*cf.* section 3.2). L'outil SOLAP se démarque entre autre par sa capacité à sa diffuser facilement et rapidement différentes valeurs statistiques localisées à des niveaux agrégées ou très détaillées. L'interface cliente SOLAP est très intuitive rendant ce type d'analyses plus faciles à réaliser pour un non-expert. De plus, elle suscite de nouveaux besoins inconnus jusqu'alors. En effet, l'interface cliente permet à l'entraîneur de visualiser la performance d'un athlète autrement. Elle lui permet aussi de se concentrer sur son analyse en cliquant simplement sur les thèmes d'intérêt. Ainsi, il peut explorer rapidement de nouvelles avenues en comparant des sections de la piste, des tours, des techniques, des périodes de l'année, des athlètes, des températures, etc. L'exploration interactive des données sur la performance des athlètes offerte par un outil SOLAP peut alors permettre la découverte de nouvelles connaissances spatio-temporelles ou de nouvelles corrélations.

Le prototype permet même à l'entraîneur de choisir le moment où il désire actualiser les cubes d'analyse SOLAP. Les nouvelles données recueillies au cours d'une journée

d'entraînement sont ensuite ajoutées dans un délai raisonnable (quelques minutes ou dans certains cas quelques heures). Cependant, pour y parvenir, une stratégie a été adoptée pour diminuer les temps de reconstruction des cubes d'analyse et ainsi mieux répondre aux besoins des entraîneurs.

En effet, le SOLAP-SPORT est un SOLAP juste-à-temps et doit fournir les nouvelles données lorsque l'entraîneur en a réellement besoin. Suite à une demande d'actualisation des cubes SOLAP, les temps de traitement doivent être acceptables pour les besoins de l'application.

L'approche SOLAP juste-à-temps est, dans le cadre de cette application, adéquate. Toutefois, dans l'éventualité que les données seraient collectées en temps réel au cours de l'entraînement et transférées au fur et à mesure (*ex.*: à toutes les secondes) dans la base de données, il serait intéressant que le rafraîchissement de l'entrepôt et la reconstruction des cubes SOLAP soient réalisés immédiatement. Ce fonctionnement nécessiterait une approche SOLAP TR ou SOLAP TQR puisque les temps de traitement devraient tourner autour de 0,5 seconde.

Le fonctionnement actuel du SOLAP-SPORT permet d'effectuer des analyses de performance sur des athlètes pratiquant d'autres sports extérieurs (*ex.*: cyclisme, canoë-kayak) en modifiant simplement la dimension spatiale latente, *Découpage piste*. Pour comparer des entraînements réalisés dans une même discipline (*ex.*: patinage de vitesse), mais sur des pistes différentes (*ex.*: Anneau de glace Gaétan-Boucher à Ste-Foy et l'Ontario *Speed Skating Oval* à Lakefield), les données doivent être transférées dans un système de coordonnées unique permettant leur superposition. Cependant, les dimensions des pistes doivent être identiques. Par exemple, un anneau de glace - style olympique (333.33m) n'a pas les mêmes dimensions qu'un anneau de glace – départ en groupe (400 m).

CONCLUSION

L'objectif principal de cette recherche était d'établir une approche répondant à des besoins d'analyse SOLAP en temps réel retrouvés dans certaines applications et plus particulièrement dans le sport de haut niveau. Différentes approches pour y parvenir ont alors été présentées (SOLAP TR, SOLAP JAT, etc.), mais une seule a été retenue et développée dans le cadre de cette recherche.

Le premier chapitre a tout d'abord synthétisé les concepts théoriques relatifs au temps réel dans le monde de l'informatique et des bases de données décisionnelles. En effet, des outils d'aide à la décision temps réel, tels que les entrepôts temps réel et les OLAP temps réel, ont récemment été développés pour répondre à des besoins analytiques temps réel. Le fonctionnement de ces outils nous a inspirés lors du développement de notre approche.

Cependant, pour répondre aux besoins analytiques des applications possédant des données à référence spatiale, un outil SOLAP est habituellement plus approprié. L'évaluation et l'analyse de la performance des athlètes de haut niveau pratiquant un sport extérieur à partir de mesures GPS est justement un exemple de ce type d'application. Le chapitre 2 a donc permis de revoir les caractéristiques d'un SOLAP traditionnel et ensuite d'apporter les nouveaux concepts de SOLAP temps réel, SOLAP temps quasi-réel, SOLAP temps propice (*right time*) ou SOLAP juste-à-temps.

Le chapitre 3 a permis d'identifier des besoins généraux et des besoins spécifiques auprès d'entraîneurs d'athlètes de haut niveau en termes d'analyse de la performance. Ensuite, une analyse des systèmes existants nous a confirmé qu'une nouvelle approche devait être établie. Entre autres, les systèmes existants ne permettaient pas aux entraîneurs d'analyser rapidement les nouvelles données recueillies sur la performance des athlètes. Cependant, les connaissances actuelles et la technologie utilisée ne nous ont pas permis de mettre en place un SOLAP TR ou TQR. De plus, l'acquisition et le traitement des données dans le cadre du projet ont été réalisés en temps différé, c'est-à-dire que les données sont transférées une fois l'entraînement terminé. Un SOLAP TR ou TQR aurait plutôt permis à l'entraîneur d'analyser la performance de l'athlète ou de communiquer à l'athlète un

changement de stratégie pendant son entraînement. Bien qu'un SOLAP TR ou TQR est l'approche rêvée dans le sport de haut niveau, un SOLAP JAT peut tout de même répondre aux besoins des entraîneurs principalement lors des entraînements. Un tel SOLAP JAT a été retenu et développé.

L'architecture et le fonctionnement du prototype SOLAP-SPORT ont ensuite été présentés. Les nouvelles données recueillies (données GPS et données descriptives) seront éventuellement saisies et transférées par l'entraîneur à partir du système transactionnel. Les nouvelles données sont ensuite ajoutées dans les cubes d'analyse seulement suite à la demande de l'utilisateur. Pour actualiser plus rapidement les cubes d'analyse, l'entraîneur peut choisir d'actualiser seulement le cube contenant les données de la présente journée. Cette stratégie nous permet de mieux respecter les critères de l'application et ainsi obtenir un SOLAP JAT. Cependant, la mise en place du SOLAP-SPORT actuel doit respecter certains critères.

1. Les données sources ajoutées dans le système doivent posséder une géométrie ponctuelle.
2. La dimension spatiale est formée de membres majoritairement statiques, c'est-à-dire formés de données latentes. Ainsi, le prototype supporte seulement des ajouts de nouveaux membres au niveau hiérarchique le plus détaillé, le niveau Point, qui représente les positions de l'athlète.
3. Les analyses offertes sur l'interface utilisateur sont assez simples. En fait, elles sont les mêmes que celles offertes dans toutes les autres applications SOLAP.
4. Enfin, l'approche suggérée, SOLAP JAT, doit suivre une stratégie en deux étapes. Premièrement, la collecte, le transfert, le traitement et le triage des données GPS doivent être réalisés (le transfert est actuellement réalisé en temps différé et le post-traitement ainsi que le triage des données GPS demande l'interaction d'un opérateur). Deuxièmement, le rafraîchissement de l'entrepôt et la reconstruction des cubes d'analyse doivent être effectués. La reconstruction des cubes est l'étape la plus exigeante en temps de traitement.

Le développement du prototype SOLAP-SPORT nous a tout de même permis de constater qu'une solution habituellement utilisée pour supporter le processus de prise de décision des entreprises peut être utilisée pour analyser la performance en sport de haut niveau. Plusieurs avantages de l'utilisation d'un tel outil dans ce type d'application peuvent aussi être identifiés. Premièrement, les données sur la performance sont maintenant archivées et un entraîneur peut ainsi suivre la progression de l'athlète. Il peut alors adapter les entraînements au cours de la saison et tenter d'optimiser la performance de l'athlète. Deuxièmement, le volet spatial du module client SOLAP, permet à un entraîneur de visualiser la trajectoire empruntée par l'athlète ou encore la longueur de ses coups de patin. En ajoutant un graphique de la vitesse en fonction du temps à cette information visuelle, l'entraîneur peut identifier rapidement certaines lacunes dans la technique de l'athlète (*ex.*: fluctuation de la vitesse plus grande dans les sections droites) ou simplement comparer celle adoptée par deux athlètes. Troisièmement, de nouvelles corrélations peuvent même être trouvées (*exemple fictif mais plausible*: Un athlète obtient de meilleures performances lorsqu'il utilise les patins « Claps » seulement lorsque la température de la glace est inférieure à 10° Celsius).

Travaux futurs

Étant donné que le calcul des agrégations est actuellement l'étape la plus coûteuse en temps de traitement pour actualiser les cubes d'analyse SOLAP de nouvelles technologies ou de nouveaux algorithmes pourraient être identifiés ou développées pour accélérer la reconstruction des cubes. Cependant, les agrégations calculées « à la volée » tiennent immédiatement compte des nouveaux faits ajoutés dans la BDM et ne nécessitent alors pas l'opération de reconstruction des cubes. Des travaux futurs de recherche pourront alors être réalisés dans cette direction. Le système d'analyse SOLAP doit nécessairement supporter le calcul « à la volée » de certaines agrégations pour ainsi se rapprocher des performances exigées par un SOLAP TR ou TQR. De plus, des stratégies pour réduire le coût de maintenance des cubes SOLAP en matérialisant seulement les agrégations les plus fréquentes et celles les plus coûteuses en temps de traitement pourront alors être utilisées (*cf.* section 1.4.3). Pour offrir un SOLAP TR, les observations GPS enregistrées à une

fréquence de 20 Hz devront être traitées et validées en moins d'une demi-seconde. Les observations GPS devront nécessairement être enregistrées en temps réel à la station de référence (RTK inversé [Lambert 2002]). Enfin, pour favoriser l'utilisation d'un tel système par des entraîneurs, ce dernier devra nécessairement être plus abordable au niveau de coût.

Aucune donnée surfacique ou linéaire n'est actuellement ajoutée dans le prototype. Or, certaines applications nécessiteront que de nouveaux membres spatiaux possédant une géométrie de types linéaire ou surfacique puissent être ajoutés rapidement dans un SOLAP TR, SOLAP TQR, SOLAP TP ou SOLAP JAT (*ex.*: application en archéologie pour l'ajout de nouveaux lots dont la géométrie est surfacique). Ces approches devront alors permettre l'ajout des données géométriques autres que ponctuelles.

BIBLIOGRAPHIE

Basu, R., 2003, *Challenges of Real-Time Data Warehousing*. DM Review, www.dmreview.com.

Bédard, Y., 1993, *La prochaine génération de logiciel SIG: les SIG dynamiques*, Revue de l'ordre des Arpenteurs-géomètres du Québec, Vol.19, No.5, p.25-26.

Bédard, Y., 2002, *Base de données spatiales*, Notes de cours, Université Laval, Département des sciences géomatiques.

Bédard, Y., 2003, *Notions avancées de base de données spatiales*, Notes de cours, Université Laval, Département des sciences géomatiques.

Bédard, Y., 2004, *De la géomatique transactionnelle à la géomatique décisionnelle*, Colloque Géomatique 2004 - Un choix stratégique! Montréal, 27-28 octobre.

Bédard, Y., 2005, *Développement d'applications d'analyses géographiques en ligne (Spatial On-Line Analytical Processing: l'émergence du premier logiciel générique SOLAP)*, Université de Provence, Marseille, 8 juin.

Bédard, Y., 2005, *La géomatique décisionnelle, une nouvelle approche pour mettre en valeur les données géographiques*, École des Mines de Paris, Pôle Cindyniques, Sophia-Antipolis, France, 7 juin.

Bédard, Y. & Chestein, Y. v., 1995, *La gestion du temps dans les systèmes de gestion de données localisées: état actuel et avenues futures*, Colloque international Géomatique V, Association Canadienne des Sciences Géomatiques, Montréal, 9-10 novembre, p.21-33.

Bédard, Y. & Henriques, W., 2003, *Technologies de pointe de l'information pour la surveillance de la santé environnementale*, Environnement et santé publique, Vol.Chapitre 36, p.27.

Bédard, Y., Larrivée, S., Proulx, M.-J., Caron, P.-Y. & Létourneau, F., 1997, *Gespatial Data Warehousing: Positionnement technologique et stratégique*, Rapport préparé pour le Centre de recherche de la défense de Valcartier, Université Laval, 79p.

Bédard, Y., Proulx, M. J. & Rivest, S., 2005, *Enrichissement du OLAP pour l'analyse géographique: exemples de réalisation et différentes possibilités technologiques*, 1re Journée Francophone sur les Entrepôts de Données et l'Analyse en ligne (EDA 2005), Lyon, France, 10 juin.

Bui, M. D., 1998, *Conception et modélisation objet des systèmes temps réel*, 341p.

- Caron, P. Y., 1998, *Étude du potentiel de OLAP pour supporter l'analyse spatio-temporelle*, Mémoire de M. Sc., Université Laval, Département des sciences géomatiques, Ste-Foy, 132p.
- Codd, E. F., Codd, S. B. & Salley, C. T., 1993, *Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate*. Hyperion white papers, 20p., <http://www.hyperion.com>.
- Cooling, J. E., 1991, *Software Design for Real-time Systems*, International Thomson Computer Press, 504p.
- Devlin, B., 1997, *Data Warehouse: from Architecture to Implementation*, 432p.
- Douglass, B. P., 1999, *Real-Time UML Second Edition*, 368p.
- Duvallet, Z. & Mammeri, B. S., 1999, *Les SGBD temps réel*, Technique et scientifique informatiques, Vol.18, No.5, Mai, p.479-516.
- Eckerson, W., 2004, *The "Soft Side" of Real-Time BI*, from www.DMReview.com.
- Eriksson, H., Penker, M., Lyons, B. & Fado, D., 2004, *UML 2 Toolkit*, Wiley Publishing Inc., 511p.
- Franklin, C., 1992, *An Introduction to Geographic Information Systems: Linking Maps to Databases*, Database, April, p.13-21.
- Ginguay, M. & Lauret, A., 1993, *Dictionnaire d'informatique, 5e*, Paris, MASSON, 371p.
- Greenfield, L., 2005, *The Data Warehousing Information Center*, from www.dwinfocenter.org.
- Han, K., Koperski & Stefanovic, N., 1997, *GeoMiner: A System Prototype for Spatial Data Mining*, Proc. 1997 ACM-SIGMOD Int'l Conf. on Management of Data(SIGMOD'97), Tucson, Arizona, May, p.553-556.
- Hay, G. J., 1991, *Reaction to Performance feedback: Advances in Biomechanics*, Enhancing Human Performance In Sport (book), Vol.25 of The Academy Papers, p.33-37.
- Hung, E., Cheung, D. W. & B.Kao, 2004, *Optimization in Data Cube System Design*, Journal of Intelligent Information Systems, Vol.23, No.1, July, p.17-45.
- Inmon, W. H., 1996, *Building the Data Warehouse, 2nd Edition*, John Wiley & Sons, Inc, 401p.
- Inmon, W. H., Imhoff, C. & Battas, G., 1996, *Building the Operational Data Store*, John Wiley & Sons, 315p.

Jensen, M. R., Møller, T. H. & Pedersen, T. B., 2001, *Specifying OLAP Cubes on XML Data*, Journal of Intelligent Information Systems, Vol.17, No.2-3, Decembre, p.255-280.

Kennedy, D., 2003, *La réalité de OLAP Temps réel*, Microsoft SQL Server™ 2000 Analysis Services MSDN, Microsoft Corporation, from www.microsoft.com.

Kimball, R., 1997, *A dimensional Modeling Manifesto*, DMBS Magazine, Vol.10, No.9, August 1997, p.59.

Kimball, R., 1997, *Relocating the ODS: Moving the Operational Data Store Will Solve a Number of Problems (Column: Data Warehouse Architect)*, DMBS Magazine, Vol.10, No.13, December, p.12.

Kimball, R. & Ross, M., 2002, *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling, 2nd Edition*, John Wiley & Sons, Inc., 464p.

Kiviniemi, J., Wolski, A., Pesonen, A. & Arminen, J., 1999, *Lazy Aggregates for Real-Time OLAP*, Proceedings of the First International Conference on Data Warehousing and Knowledge Discovery, September, p.165-172.

Lambert, M., 2002, *Utilisation du GPS pour l'analyse de performances sportives*, Mémoire de M. Sc., Université Laval, Département des sciences géomatiques, Ste-Foy, 144p.

Lambert, M., 2004, *Méthodes de suivi des analyses exploratoires dans un SOLAP*, Rapport Final présenté dans le cadre du cours SCG-62058 – Sujet spécial, Université Laval, Département des sciences géomatiques, 57p.

Lambert, M. & Santerre, R., 2004, *Performance Monitoring with RTK GPS*, GPS World, Vol.15, No.2, 1er février, p.14-23.

Langseth, J., 2004, *Real-Time Data Warehousing: Challenges and Solutions*, DSSResources.COM, from <http://dssresources.com>.

Larsson, P., 2003, *Global Positioning System and Sport-Specific Testing*, Sports Medicine, Vol.33, No.15, p.1093-1101.

Lenz, H. & Shoshani, A., 1997, *Summarizability in OLAP and Statistical Data Bases*, In Proceedings of SSDBDM, p.39-48.

Levy, E., 2005, *What is real-time data warehousing?* from www.dmreview.com.

Lu, C. T., Kou, Y., Wang, H., Shekhar, S., Zhang, P. & Liu, R., 2003, *Two Web-based Spatial Data Visualization and Mining Systems: Mapcube & Mapview*, International Workshop on Next Generation Geospatial Information, Cambridge (Boston), Massachusetts, Oct. 19-21, 12p.

Luger, G., 2004, *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*, Addison-Wesley, 928p.

Mabilleau, P., 2001, *Système en temps réel*, Notes de cours, Université de Sherbrooke, Département de génie électrique et de génie informatique, 12p.

Maltais, L. & Santerre, R., 1999, *Suivi par satellites de la performance des patineurs de vitesse*, Géomatique, Revue de l'Ordre des arpenteurs-géomètres du Québec, Vol.26, No.3, p.5-7.

Marchand, P., Brisebois, A., Bédard, Y. & Edwards, G., 2004, *Implementation and evaluation of a hypercube-based method for spatio-temporal exploration and analysis*, International Society for Photogrammetry and Remote Sensing (ISPRS) Journal, Vol.59, No.1-2, p.6-20.

Miller, C. C., Miller, M. M. & Agnew, J., 2000, *A Novel GPS-Based Training Device to Improve Track and Cross Country Training Effectiveness*, Proceeding of the ION GPS 2000, Vol.19-22 Septembre, p.1694-1702.

Moore, G. E., 1965, *Cramming More Components Onto Integrated Circuits*, Electronics, 19 avril 1965, p.4.

Mumick, I. S., Dallan, Q. & Mumick, B. S., 1997, *Maintenance of data cubes and summary tables in a warehouse*, Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data, June, p.100-111.

Nelson, G. B., 2004, *Real Time Decision Support: Creating a Flexible Architecture for Real Time Analytics*, ThatWave Technologies, Cary, North Carolina, Vol.109, No.29, p.1-12.

Niemi, T., Niinimäki, M., Nummenmaa, J. & Thanisch, P., 2002, *Constructing an OLAP cube from distributed XML data*, Proceedings of the 5th ACM international workshop on Data Warehousing and OLAP, November, p.22-27.

OLAP Council, 1995, *OLAP and OLAP Server Definitions*, OLAP Council, from <http://altaplana.com>.

OLF 2005, from www.granddictionnaire.com.

Papadias, D., Kalnis, P., Zhang, J. & Tao, Y., 2001, *Efficient OLAP Operations in Spatial Data Warehouses*, Proc. of the Int. Symposium in Spatial and Temporal Databases (SSTD), Redondo Beach, CA, p.443-459.

Pedersen, T. B. & Nektaria, T., 2001, *Pre-aggregation in Spatial Data Warehouses*, Advances in Spatial and Temporal Databases, 7th International Symposium, SSTD 2001, Springer, Redondo Beach, July 12-15, p.460-478.

Proulx, M.-J. & Bédard, Y., 2004, *Le potentiel de l'approche multidimensionnelle pour l'analyse de données géospatiales en comparaison avec l'approche transactionnelle des SIG*, Colloque Géomatique 2004 - Un choix stratégique! Montréal, 27-28 octobre.

Raden, N., 2003, *Real time: get real: start by discarding your current concepts of ETL. (Data Warehouse Designer)*. Intelligent Enterprise, Vol.6, No.11, June 30, p.16.

Raden, N., 2003, *Real time: get real; take the idea of a real-time data warehouse with a grain of salt, then realize the possibilities. (Data Warehouse Designer)*. Intelligent Enterprise, Vol.6, No.10, June 17, p.22.

Rageul, N., 2004, *Développement d'une application d'exploration de données géospatiales comme support à la fouille archéologique*, Mémoire pour l'obtention du diplôme d'ingénieur, INSA Strasbourg, Strasbourg, 59p.

Ramamritham, K., 1993, *Real-time databases*, J. of Distributed and Parallel Databases, Vol.1, No.2, p.199-226.

Rivest, S., 2000, *Investigation des modes d'intégration physique entre un serveur de base de données multidimensionnelle et un SIG*, Essai de M.Sc., Université Laval, Département des sciences géomatiques, Ste-Foy, 84p.

Rivest, S., Gignac, P., Charron, J. & Bédard, Y., 2004, *Développement d'un système d'exploration spatio-temporelle interactive des données de la Banque d'information corporative du ministère des Transports du Québec*, Colloque Géomatique 2004 - Un choix stratégique! Montréal, 27-28 octobre.

Russell, S. J. & Norvig, P., 2002, *Artificial Intelligence: A Modern Approach, 2nd Edition*, Hardcover, 905p.

Sadeg, B., 2004, *Contributions à la gestion des transactions dans les SGBD temps réel*, Habilitation à diriger des recherches, Université du Havre, Le Havre, 149p.

Santerre, R., 2001, *Positionnement par satellites GPS*, Notes de cours, Université Laval, Département des sciences géomatiques.

Schutz, Y. & Herren, R., 2000, *Assessment of speed of human locomotion using a differential satellite global positioning system.*, Med Sci Sports Exerc., Vol.32, No.3, p.642-646.

Shoshani, A., 1997, *OLAP and Statistical Databases: Similarities and Differences*, In Proceedings of the Sixteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, Tuscon, Arizona, May, p.185-196.

Shukla, A., Deshpande, M., Naughton, J. F. & Ramasamy, K., 1996, *Storage Estimation for Multidimensional Aggregates in the Presence of Hierarchies*, In Proceedings of VLDB, p.522-531.

Skaloud, J. & Merminod, B., 2000, *DGPS-Calibrated Accelerometric System for Dynamic Sports Events*, ION GPS 2000, Salt Lake City, UT, September 19-22, 7p.

Stankovic, J. & Ramamritham, K., 1988, *Tutorial on Hard Real-Time Systems*, IEEE Computer Society Press, p.611-618.

Stresing, D., 2004, *PDA's change the game*, LAPTOP, August 2004, p.126-131.

Tansel, A., Clifford, J., Gadia, V., Segev, A. & Snodgrass, R., 1993, *Temporal databases: theory, design and implementation*, Redwood city, California, The Benjamin/Cummings Publishing Company, 633p.

The OLAP Report, 2005, *Database explosion*, February 11, 2005, from www.olapreport.com.

Thomsen, E., 1997, *OLAP solutions: Building Multidimensional Information Systems*, John Wiley & Sons, 608p.

Veilleux, J.-P., 2006, *Détermination précise des vitesses et des accélérations d'un athlète par mesures GPS prises à haute fréquence*, Dépôt initial janvier 2006, Université Laval, Département des sciences géomatiques, Ste-Foy, 123 p.

Veilleux, J.-P., Lambert, M., Santerre, R. & Bédard, Y., 2004, *Utilisation du système de positionnement par satellites (GPS) et des outils d'exploration et d'analyse SOLAP pour l'évaluation et le suivi de sportifs de haut niveau*, Colloque Géomatique 2004 - Un choix stratégique! Montréal, Canada, 27-28 octobre.

VIVID Solutions, 2005, from www.vividsolutions.com/JTS/jts_frame.htm.

White, C., 2002, *Intelligent Business Strategies: Near Real-Time and Automated Decision Making*, DMReview, October, from www.dmreview.com.

White, C., 2004, *Now is the Right Time for Real-Time BI*, DMReview, from www.dmreview.com.

Wikipédia, 2005, *Wikipédia, L'encyclopédie libre*, from <http://fr.wikipedia.org>.

ANNEXE A - Inventaire des données spatiales disponibles

1. Liste complète des données spatiales

- Patinage de vitesse (compilées par Mathieu Lambert)
- Cyclisme (compilées par Mathieu Lambert)
- Canoë-kayak (compilées par Mathieu Lambert)
- Ski alpin (compilées par Martin Simoneau, Vincent Cantin, Mathieu Lambert)

- Informations descriptives sur les jeux de données spatiales

1- Identification du document

Nom de la collection	Anneau Gaétan-Boucher (400 m)	Vélodrome Louis-Garneau + MNT	Canoë-kayak Lac Beauport	Ski alpin
Nom du document	Patinage : int.shp, ext.shp, lines.shp	Vélo : Fond.shp, haut.shp, lines.shp, pnts3d, ahtin, nrhtin	Corridor	Ski : Poly.shp, thmz1.shp, slope
Thèmes	-Limite extérieure de l'anneau; -Limite intérieure de l'anneau; -Ligne de départ	- La côte d'azur (limite de fond (intérieur)) - Limite du haut (extérieur) - Lignes corridors - Points 3D - TIN		- Polygone composé des limites de la piste - Pente en % de la piste
Méthode de captage ou de conception	Récepteurs GPS-RTK	Récepteurs GPS-RTK		Récepteurs NovAtel bi-fréquence
Type de support (analogique, numérique)	Numérique	Numérique	Numérique	Numérique
Lieu d'archivage ou fichier	Au groupe GPS – CRG, sur Ordinateur de Mathieu Lambert	Au groupe GPS – CRG, sur Ordinateur de Mathieu Lambert	Au groupe GPS – CRG, sur Ordinateur de Mathieu Lambert	Au groupe GPS – CRG, sur ordinateur de Mathieu Lambert
Structure de données (vecteur topologique, raster,...)	Vecteur non-topologique	Vecteur non-topologique	Vecteur non-topologique	Vecteur non-topologique
Format de données (dgn, arcview, dxf, ascii, gif, ...)	arcview	arcview	arcview	arcview

2- Identification du responsable

	Anneau Gaétan-Boucher (400 m)	Vélodrome Louis-Garneau	Canoë Lac Beauport	Ski alpin
Fonction	Mathieu Lambert, Maîtrise en Géomatique	Mathieu Lambert, Maîtrise en Géomatique	Mathieu Lambert, Maîtrise en Géomatique	Mathieu Lambert, Martin Simoneau et Vincent Cantin,
Nom organisation	Département des sciences géomatiques	Département des sciences géomatiques	Département des sciences géomatiques	Département des sciences géomatiques et Département de kinésiologie

3- Identification de la référence temporelle du document ou de l'entité

	Anneau Gaétan-Boucher (400 m)	Vélodrome Louis-Garneau	Canoë Lac Beauport	Ski alpin
Date de publication ou d'acquisition	07/12/2001	29/09/2001	21/10/2001	04/2002
Fréquence de mise à jour	aucune	aucune	aucune	aucune

4- Identification de la référence spatiale du document ou de l'entité

	Anneau Gaétan-Boucher	Vélodrome Louis-Garneau	Canoë Lac Beauport	Ski alpin
Couverture territoriale (latitude N-S, longitude E-O)	<u>Approximative</u> Longitude ouest: -71°18'07" Latitude nord: 46°46'16" Longitude est: -71°18'00" Latitude sud: 46°46'11"	<u>Approximative</u> Longitude ouest: -71°24'10" Latitude nord: 46°44'19" Longitude est: -71°24'08" Latitude sud: 46°44'16"		<u>Approximative</u> Longitude ouest: -70°56'18" Latitude nord: 47°05'35" Longitude est: -70°05'20" Latitude sud: 47°05'20"
Datum horizontal	NAD83	NAD83	NAD83	NAD83
Datum vertical	inconnu			
Ellipsoïde géodésique (Clark1866, GRS80, WGS84,...)	GRS80	GRS80	GRS80	GRS80
Système de coordonnées (géographique, planaire,...)	planaire	planaire	planaire	planaire
Système de projection (UTM, MTM, Lambert, ...)	MTM	MTM	MTM	MTM
Unité de mesure (mètre, pied,...)	mètre	mètre	mètre	mètre
Précision	Au cm (RTK)	Au cm (RTK)	Au cm (RTK)	
Coordonnées du point de référence utilisé	Géodésique	ϕ : 46°46'15.0000 λ : 71°18'03.4000 h : 118.000 m	<i>Centre de la piste</i> ϕ : 46°44'17.5445 λ : 71°24'09.0344 h : 17.210 m	ϕ : 46°57'16.67898 λ : 71°16'13.97631 h : 252.958 m
	Cartographique (MTM-Zone 7)	X : 243629.635m Y : 5181564.227m	X : 235831.245m Y : 5178021.676m	X : 246151.524m Y : 5201971.541m

5- La position recueillie lors de l'entraînement d'un athlète

	Anneau Gaétan-Boucher (400 m)	Vélodrome Louis-Garneau	Canoë Lac Beauport	Ski alpin
Méthode de captage ou de conception	Récepteur GPS-RTK inversé	Récepteur GPS-RTK inversé	Récepteur GPS-RTK inversé	Différentielle
Type de récepteur	Scorpio, Thales Navigation	Scorpio, Thales Navigation	Scorpio, cie Thales Navigation	NovAtel
Emplacement de l'antenne	Sur le casque de vélo	Sur le casque de vélo	À l'arrière du canoë	Sur un casque de descente
Distance max entre récepteur et mobile	119 m	42 m	100 m	
Max PDOP obtenu	3.2	4.2	2.5	
Max erreur	composante verticale du vecteur	0.37 mm	0.11 mm	9 mm
	vitesse	0.3 mm/s	0.4 mm/s	0.2 mm/s
Coordonnées du point de référence utilisé	Géodésique	ϕ : 46°46'15.0000 λ : 71'18'03.4000 h : 118.000 m	<i>Centre de la piste</i> ϕ : 46°44'17.5445 λ : 71'24'09.0344 h : 17.210 m	ϕ : 46°57'16.67898 λ : 71'16'13.97631 h : 252.958 m
	Cartographique (MTM)	X : 243629.635 m Y : 5181564.227 m	X : 235831.245 m Y : 5178021.676 m	X : 246151.524 m Y : 5201971.541 m

ANNEXE B - Dimensions et mesures

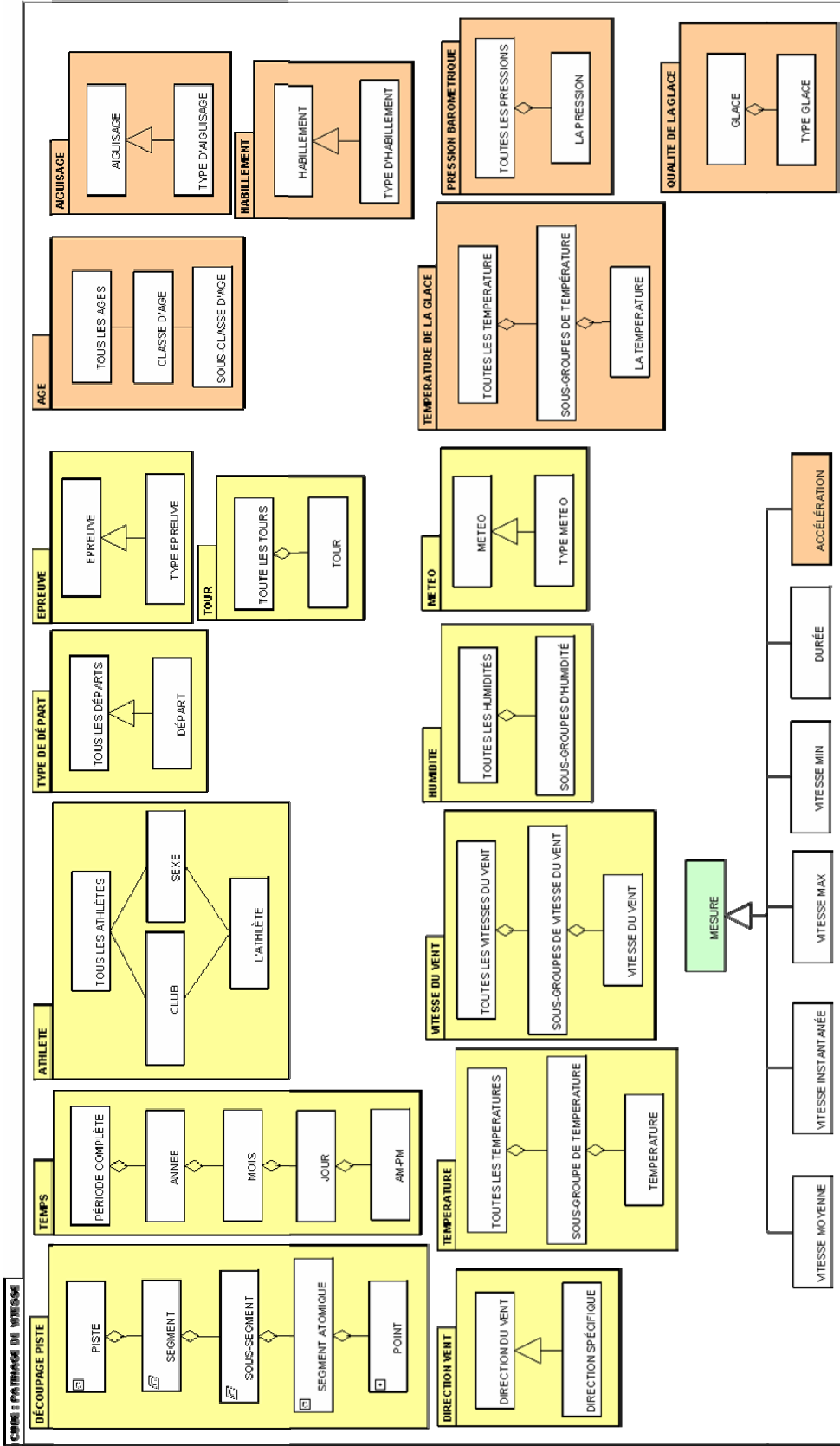


Figure 0-1 : Dimensions et mesures pour le patinage de vitesse

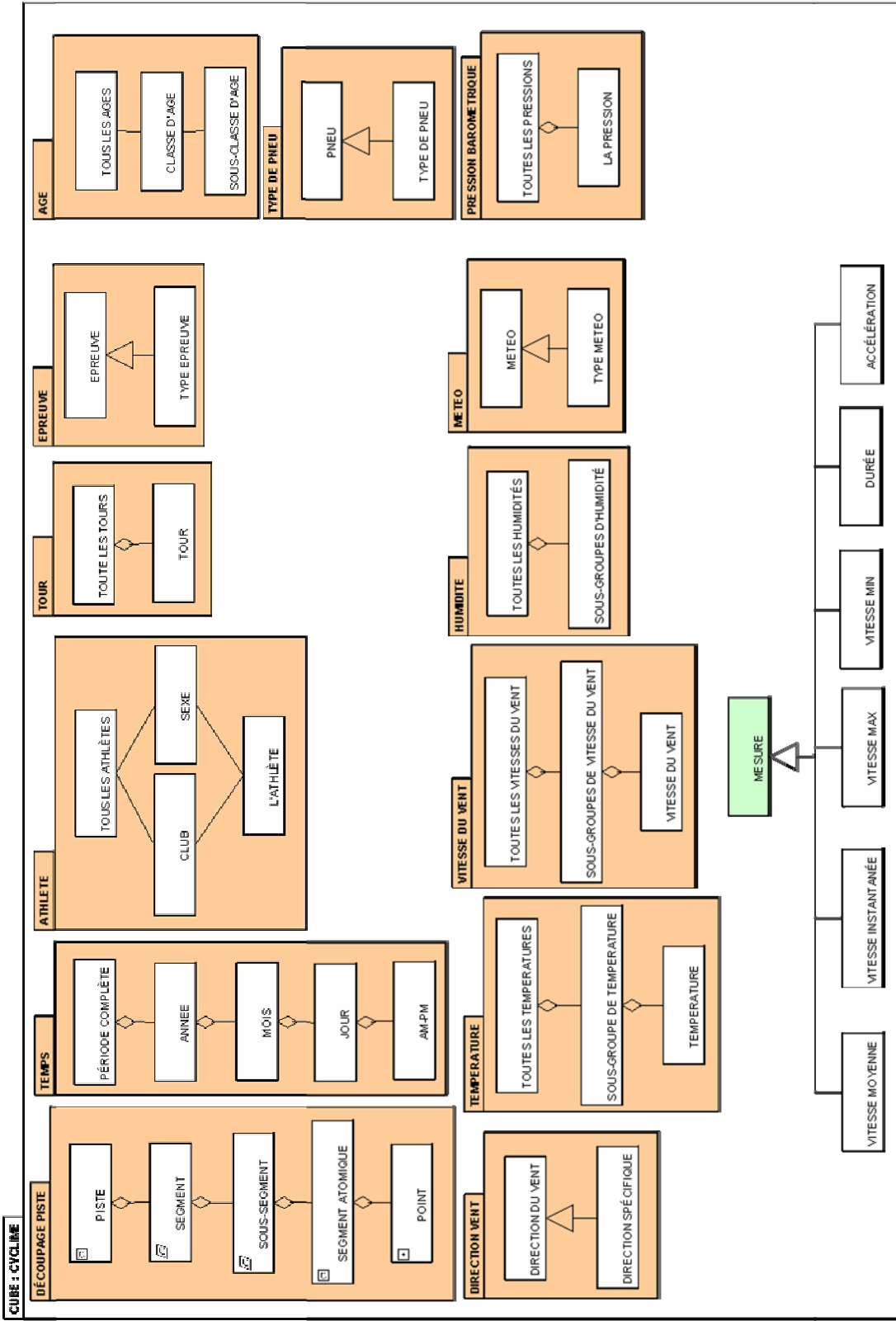


Figure 0-2 : Dimensions et mesures pour le cyclisme

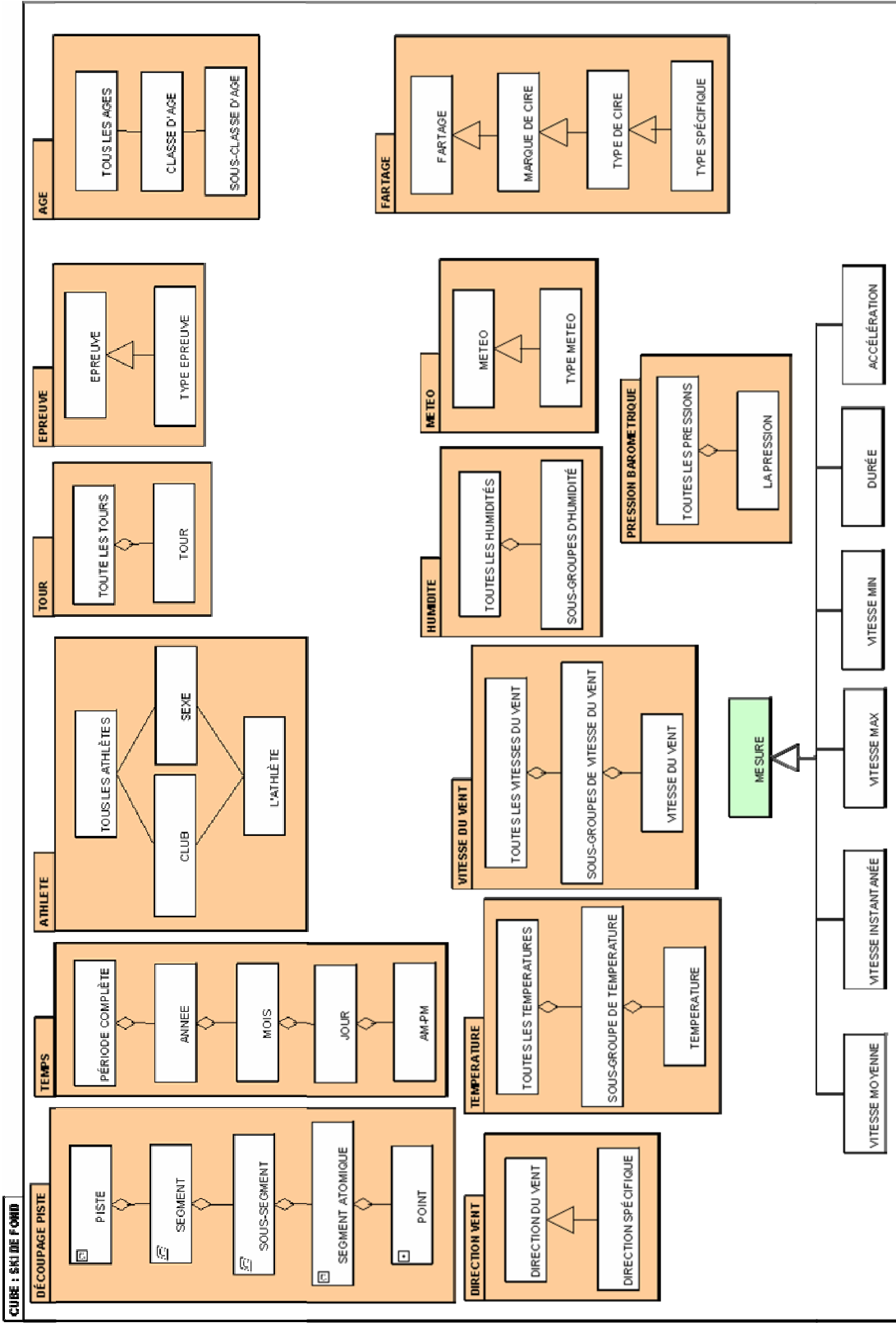


Figure 0-3 : Dimensions et mesures pour le ski de fond

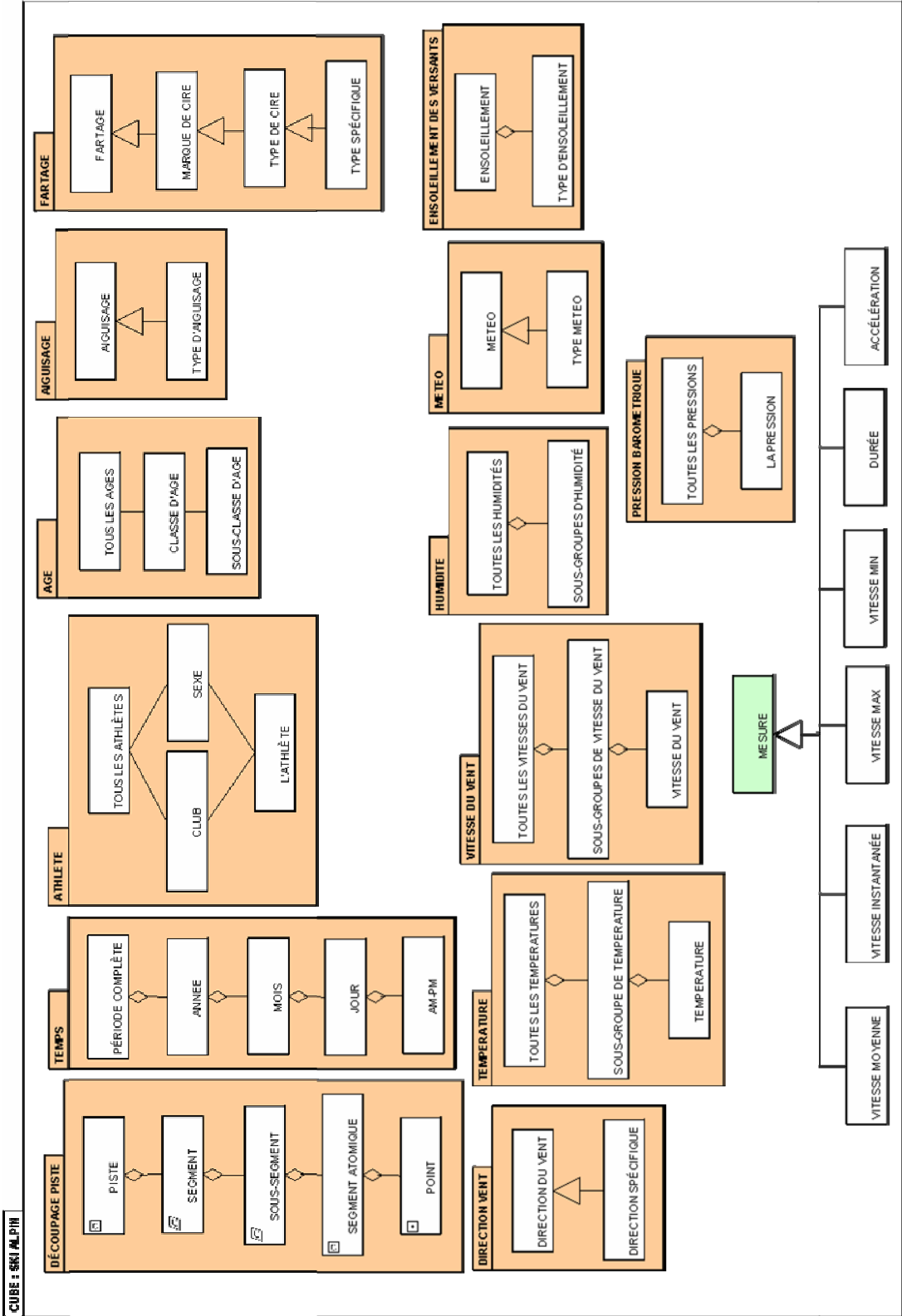


Figure 0-4 : Dimensions et mesures pour le ski alpin

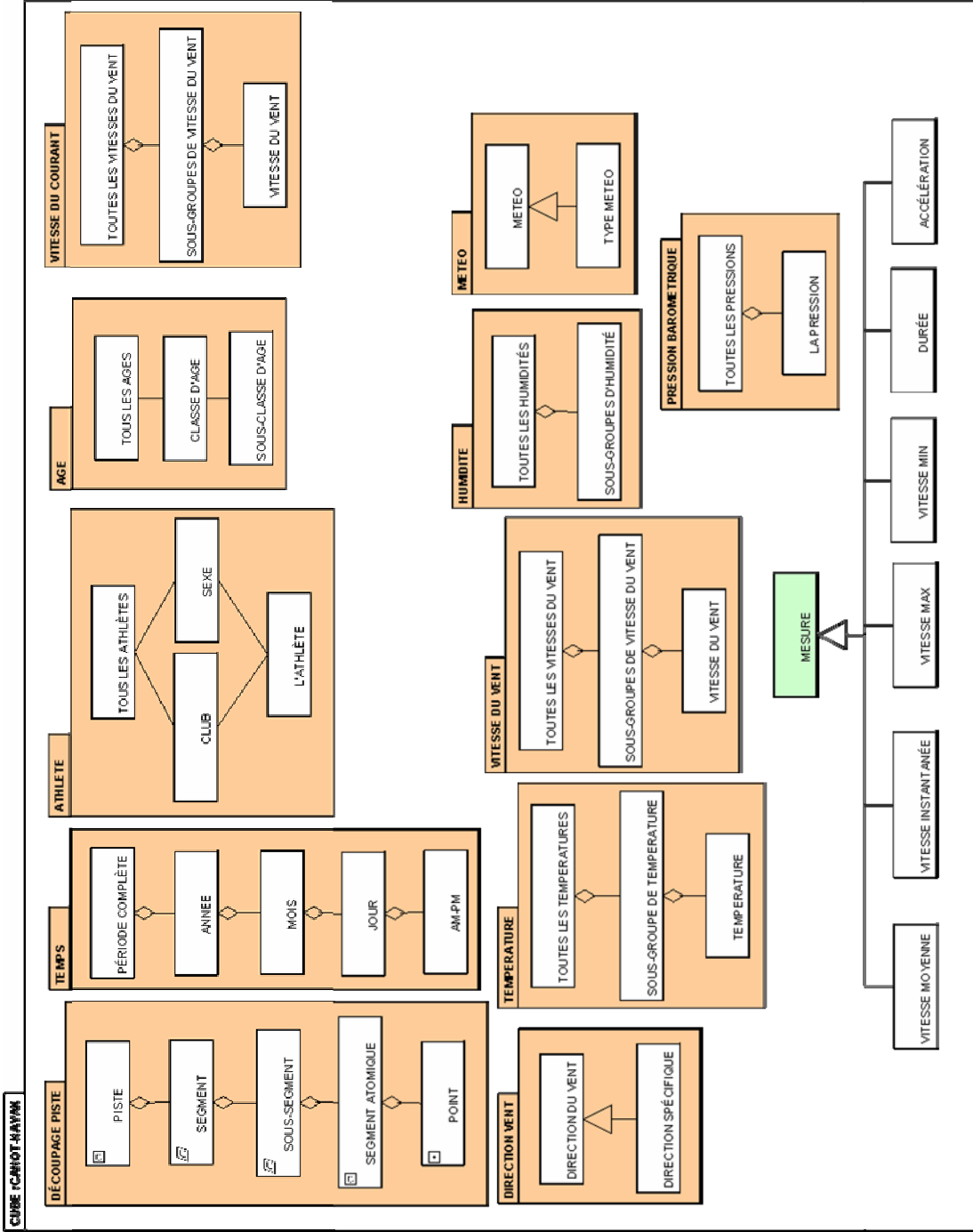


Figure 0-5 : Dimensions et mesures pour le canoë-kayak

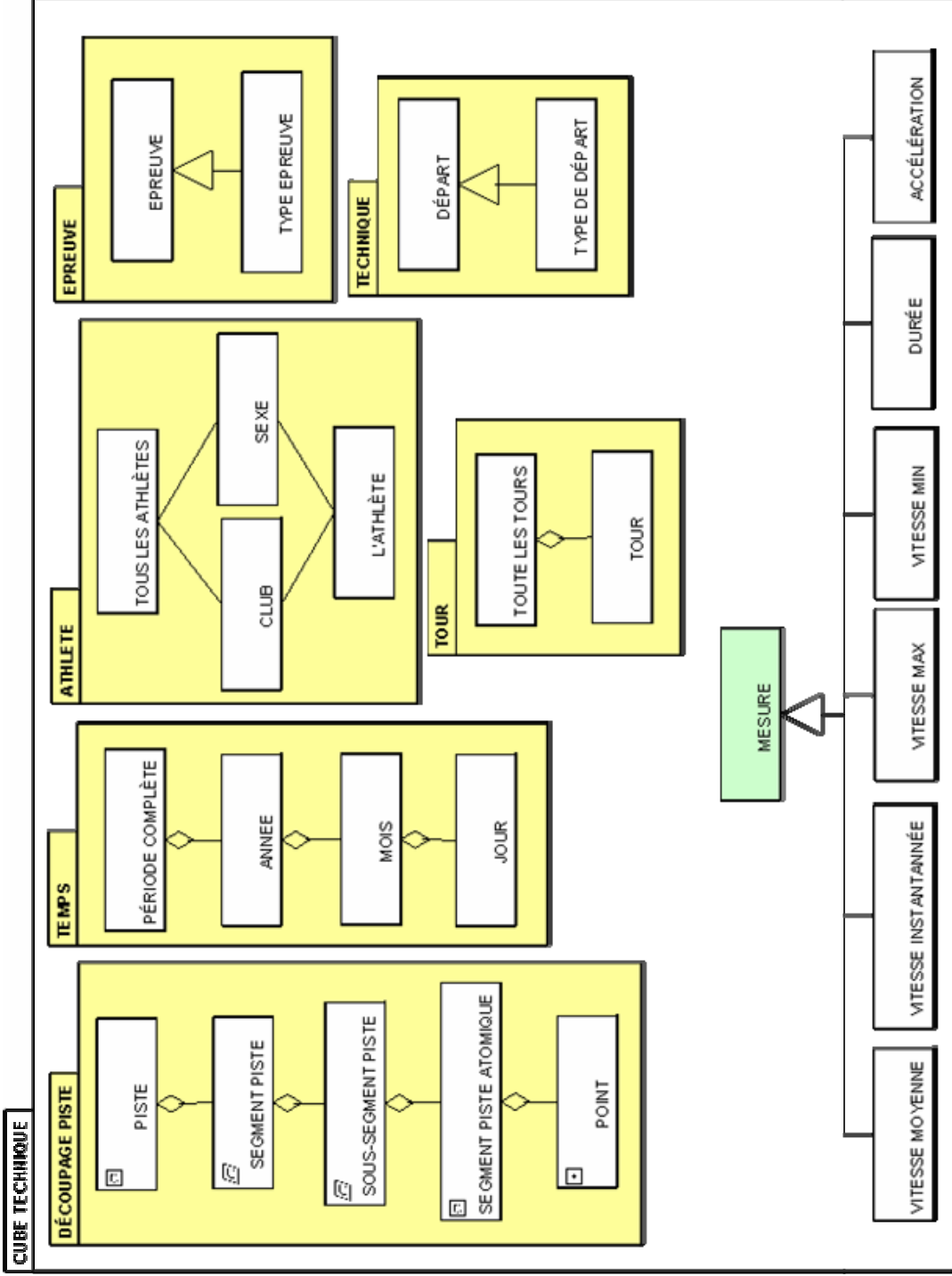


Figure 0-6 : Dimensions et mesures du cube TECHNIQUE

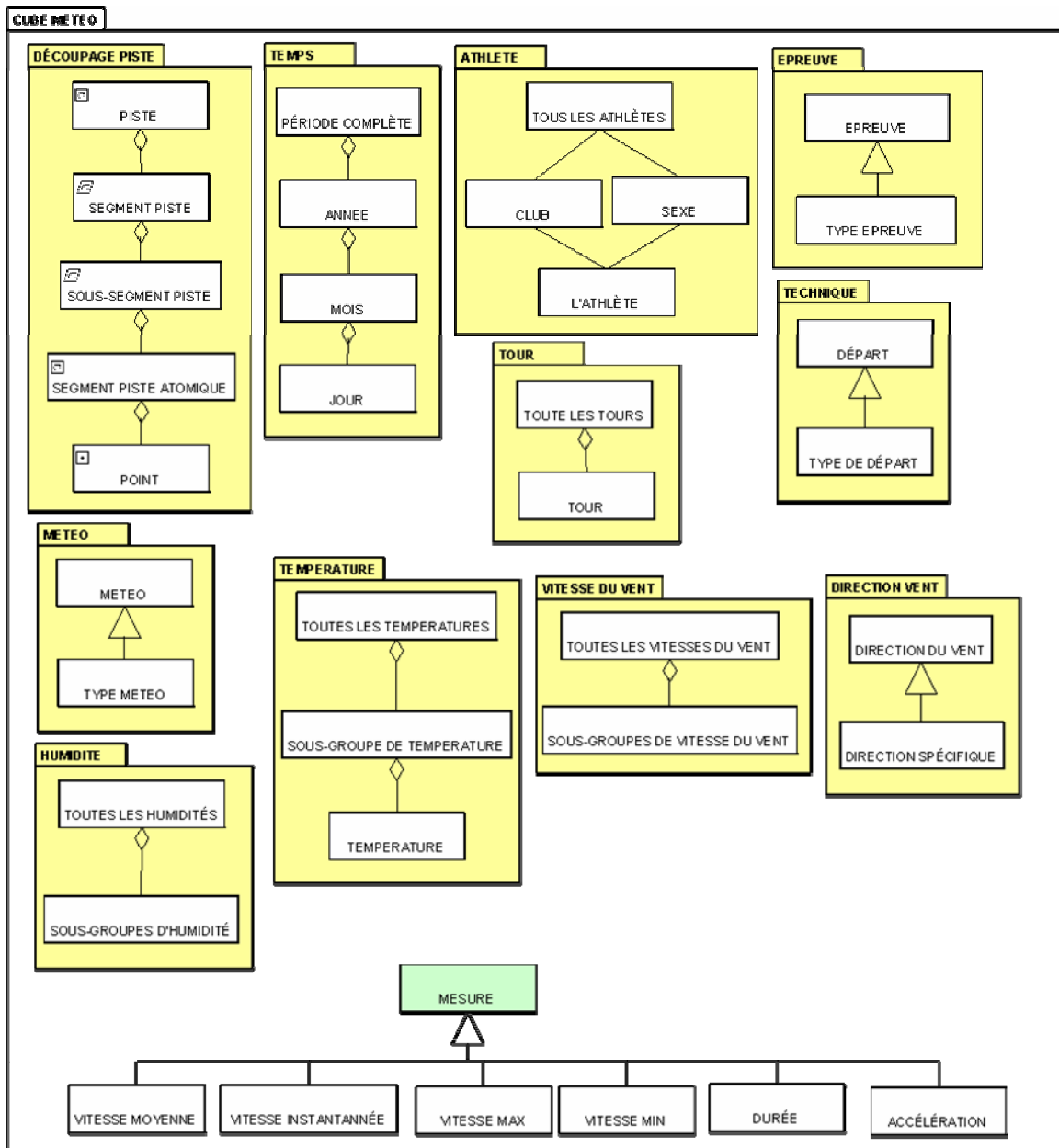


Figure 0-7 : Dimensions et mesures du cube MÉTÉO

ANNEXE C - Information implicite et explicite sur un SOLAP

Information implicite et explicite sur un SOLAP

Le SOLAP, grâce à son volet spatial, donne la possibilité de choisir entre conserver une information spatiale implicite ou la rendre explicite. Globalement, une information *implicite* est une information qui n'est pas formellement énoncée, mais qui peut tout de même être déduite. Alors qu'une information *explicite* est énoncée clairement.

Dans certaines applications, il peut être suffisant de laisser l'information spatiale implicite et compter seulement sur la visualisation cartographique pour effectuer des analyses, comparer et identifier des corrélations spatiales. La simple référence spatiale des données apporte même fréquemment une information supplémentaire. En effet, le nombre de cas de cancer présent dans trois régions administratives connexes peut sembler normal avec un affichage tabulaire, alors que visuellement et grâce à une carte on peut découvrir que tous les cancers sont concentrés à la frontière de ces trois régions administratives. Cet exemple est présenté sur le Tableau 0-1. Dans les trois régions, le taux de cancer correspond à la moyenne : quatre cancers ont été détectés dans la région #1 et trois dans les régions #2 et #3.

Régions administratives	Nombre de cancers
Région 1	4
Région 2	3
Région 3	3

Tableau 0-1 : Nombre de cancers pour les régions

Cependant, sur la carte de la Figure 0-8, la position de ces cancers indique une zone élevée de cancer près de la frontière de ces trois régions.

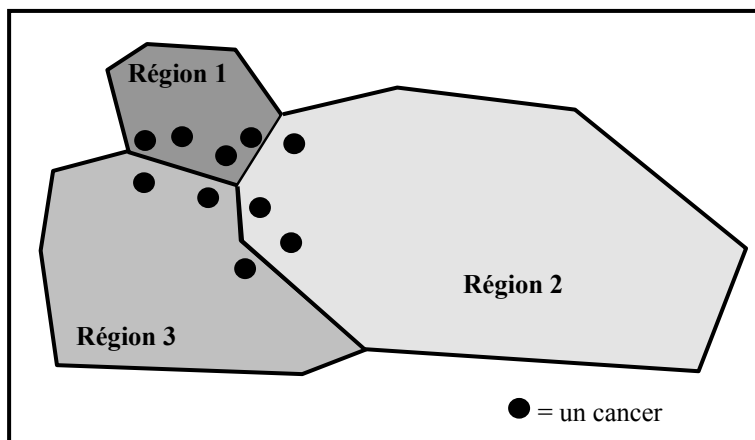


Figure 0-8 : Carte affichant le nombre de cancers dans les régions

L'information spatiale, même implicite, est importante et permet de découvrir des phénomènes qui n'auraient pu être détectés autrement.

Si un niveau plus détaillé est présent dans la dimension spatiale (information spatiale explicite), la zone suspecte est alors identifiée directement avec l'affichage tabulaire. En effet, un niveau hiérarchique supplémentaire permet d'obtenir des valeurs quantitatives à l'intérieur d'un tableau (Tableau 0-2).

Régions administratives	Nombre de cancers
Sous-région 1A	4
Sous-région 1B	0
Sous-région 2A	3
Sous-région 2B	0
Sous-région 3A	3
Sous-région 3B	0
Sous-région 3C	0

Tableau 0-2 : Nombre de cancers pour les sous-régions.

ANNEXE D - Diagrammes SOLAP-SPORT

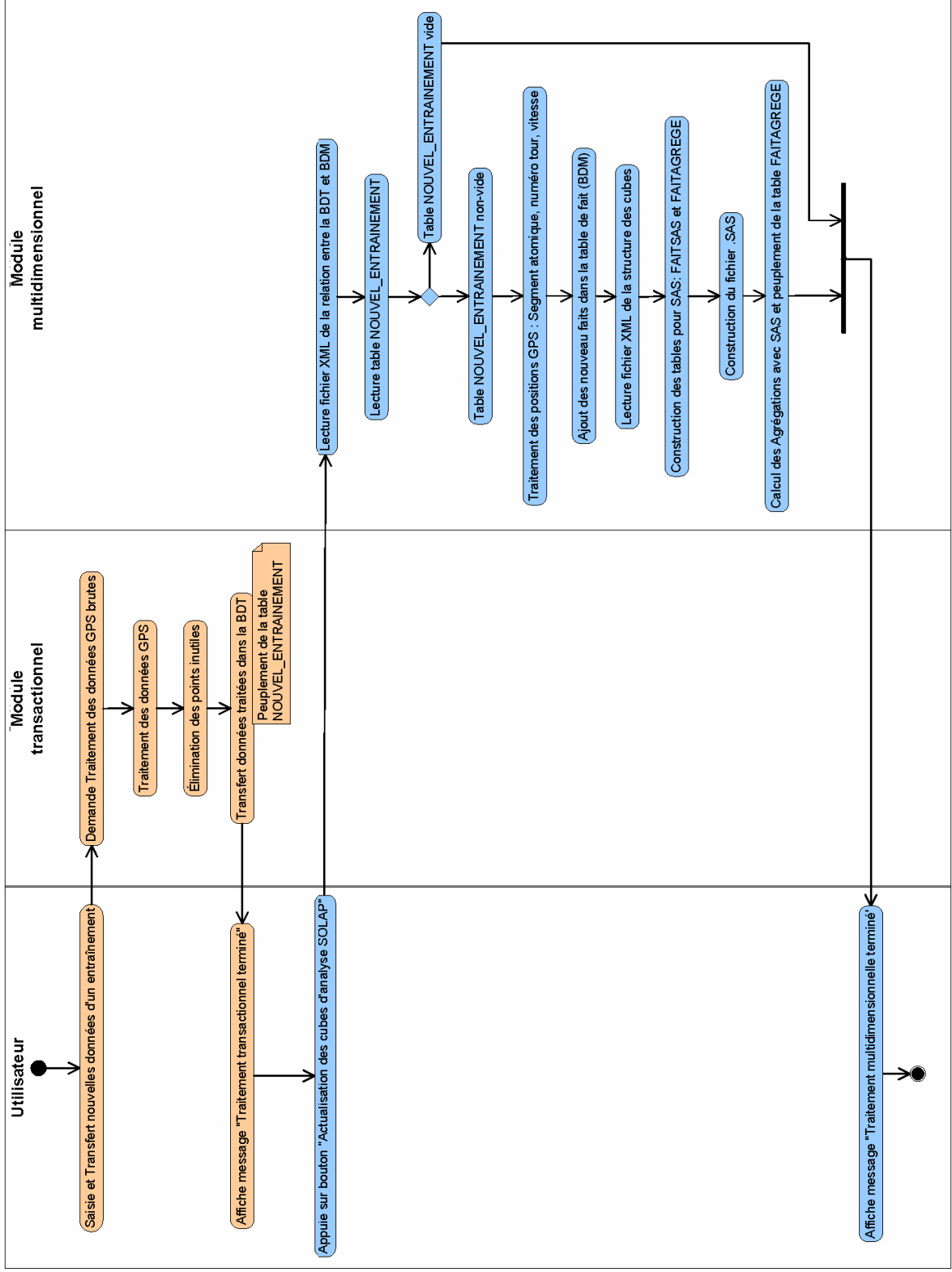


Figure 0-9 : Diagramme d'activités SOLAP-SPORT

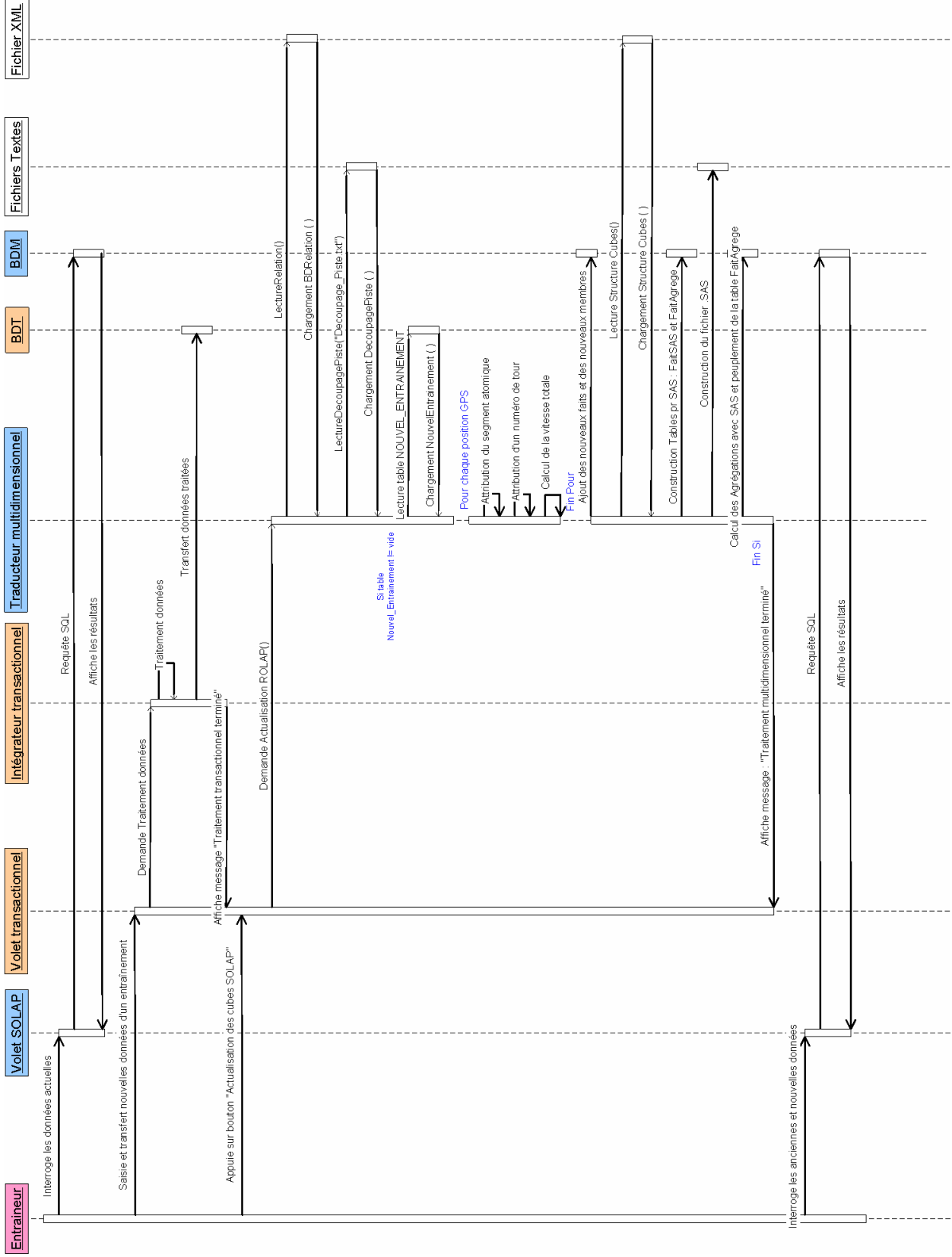


Figure 0-10 : Diagramme de séquence SOLAP-SPORT

ANNEXE E - Format des fichiers XML dans le SOLAP-SPORT

RelationTransactMultidim.dtd

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!ELEMENT BD (Source, Destination) >
<!ELEMENT Source (Tables) >
<!ATTLIST Source
    type CDATA #REQUIRED
    nom CDATA #REQUIRED
    odbc CDATA #REQUIRED
    tableNouvelEntrainement CDATA #REQUIRED
    user CDATA #IMPLIED
    pwd CDATA #IMPLIED
>
<!ELEMENT Tables (Table+) >
<!ELEMENT Table (Champs) >
<!ATTLIST Table
    nom CDATA #REQUIRED
>
<!ELEMENT Champs (Champ+) >
<!ELEMENT Champ (#PCDATA)>
<!ATTLIST Champ
    nom CDATA #REQUIRED
    dataType CDATA #REQUIRED
>
<!ELEMENT Destination (TablesDim, TablesFait) >
<!ATTLIST Destination
    type CDATA #REQUIRED
    nom CDATA #REQUIRED
    odbc CDATA #IMPLIED
    user CDATA #REQUIRED
    pwd CDATA #REQUIRED
    hote CDATA #REQUIRED
    port CDATA #REQUIRED
    instance CDATA #REQUIRED
>
<!ELEMENT TablesDim (Table+) >
<!ELEMENT Table (Champs) >
<!ATTLIST Table
    nom CDATA #REQUIRED
    changing CDATA #REQUIRED
    sourceChampsNouvelEntrainement CDATA #REQUIRED
    temps CDATA #REQUIRED
>
<!ELEMENT Champs (Champ+) >
<!ELEMENT Champ (#PCDATA)>
<!ATTLIST Champ
    nom CDATA #REQUIRED
    sourceTable CDATA #REQUIRED
    sourceChamps CDATA #REQUIRED
>
<!ELEMENT TablesFait (Table+) >
<!ELEMENT Table (Champs) >
<!ATTLIST Table
    nom CDATA #REQUIRED
>
<!ELEMENT Champs (Champ+) >
<!ELEMENT Champ (#PCDATA)>

```



```
<!ATTLIST Champ
  nom CDATA #REQUIRED
  sourceTable CDATA #IMPLIED
  sourceChamps CDATA #IMPLIED
>
```

RelationTransactMultidim.xml

```

- <BD>
- <Source tableNouvelEntrainement="Nouvel_Entrainement">
- <Tables>
- <Table nom="Athlete">
- <Champs>
  <Champ nom="id" dataType="AutoNumber"/>
  <Champ nom="nom" dataType="Text"/>
  <Champ nom="date_naissance" dataType="Text"/>
  <Champ nom="sexe" dataType="Text"/>
  <Champ nom="Niveau" dataType="Text"/>
  <Champ nom="Club" dataType="Text"/>
</Champs>
</Table>
+ <Table nom="Conditions"></Table>
+ <Table nom="Nouvel_Entrainement"></Table>
</Tables>
</Source>
- <Destination>
+ <TablesDim></TablesDim>
- <TablesFait>
- <Table nom="Fait_Technique">
- <Champs>
  <Champ nom="Vitesse"/>
  <Champ nom="Tour" sourceTable="" sourceChamps=""/>
  <Champ nom="Segment_atomique" sourceTable="" sourceChamps=""/>
  <Champ nom="Athlete" sourceTable="Athlete" sourceChamps="id"/>
  <Champ nom="Jour_Course" sourceTable="Nouvel_Entrainement" sourceChamps="Temps"/>
  <Champ nom="Epreuve" sourceTable="Nouvel_Entrainement" sourceChamps="Epreuve"/>
  <Champ nom="Duree"/>
  <Champ nom="Depart" sourceTable="Nouvel_Entrainement" sourceChamps="Depart"/>
  <Champ nom="Point" sourceTable="" sourceChamps=""/>
  <Champ nom="Vitesse_inst"/>
</Champs>
</Table>
+ <Table nom="Fait_Meteo"></Table>
+ <Table nom="Fait_Entrainement"></Table>
</TablesFait>
</Destination>
</BD>

```

Cubes.dtd

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!ELEMENT BDMultidim (Cubes) >
<!ATTLIST BDMultidim
    type CDATA #REQUIRED
    nom CDATA #REQUIRED
    odbc CDATA #IMPLIED
    user CDATA #REQUIRED
    pwd CDATA #REQUIRED
    host CDATA #REQUIRED
    port CDATA #REQUIRED
    instance CDATA #REQUIRED
>
<!ELEMENT Cubes (Cube+)>
<!ELEMENT Cube (TableFait, Dimensions, Mesures, TableFaisSAS, TableFaitAgrege)>
<!ATTLIST Cube
    nom CDATA #REQUIRED
>
<!ELEMENT TableFait (Champs)>
<!ATTLIST TableFait
    nom CDATA #REQUIRED
>
<!ELEMENT Champs (Champ+)>
<!ELEMENT Champ (#PCDATA)>
<!ATTLIST Champ
    nom CDATA #REQUIRED
>
<!ELEMENT Dimensions (Dimension+)>
<!ELEMENT Dimension (Niveaux)>
<!ATTLIST Dimension
    nom CDATA #REQUIRED
    champFait CDATA #REQUIRED
>
<!ELEMENT Niveaux (Niveau+)>
<!ELEMENT Niveau (#PCDATA)>
<!ATTLIST Niveau
    id CDATA #REQUIRED
    nom CDATA #REQUIRED
>
<!ELEMENT Mesures (Mesure+)>
<!ELEMENT Mesure (#PCDATA)>
<!ATTLIST Mesure
    nom CDATA #REQUIRED
    type CDATA #REQUIRED
    sourceMesure CDATA #REQUIRED
>
<!ELEMENT TableFaisSAS (Champs)>
<!ATTLIST TableFaisSAS
    nom CDATA #REQUIRED
>
<!ELEMENT Champs (Champ+)>
<!ELEMENT Champ (#PCDATA)>
<!ATTLIST Champ
    nom CDATA #REQUIRED
    racine CDATA #REQUIRED
    sourceTable CDATA #REQUIRED
    sourceChamps CDATA #REQUIRED
    joinChamps CDATA #REQUIRED
    joinChampsSource CDATA #REQUIRED
    dataType CDATA #REQUIRED
>
<!ELEMENT TableFaitAgrege (Champs)>
<!ATTLIST TableFaitAgrege
    nom CDATA #REQUIRED
    pathSAS CDATA #REQUIRED
>
<!ELEMENT Champs (Champ+)>
<!ELEMENT Champ (#PCDATA)>
<!ATTLIST Champ

```

```

nom CDATA #REQUIRED
dataType CDATA #REQUIRED
>

```

Cubes.xml

```

- <BDMultidim>
- <Cubes>
- <Cube nom="Fait_Technique">
- <TableFait nom="Fait_Technique">
- <Champs>
  <Champ nom="Tour"/>
  <Champ nom="Segment_atomique"/>
  <Champ nom="Athlete"/>
  <Champ nom="Jour_Course"/>
  <Champ nom="Epreuve"/>
  <Champ nom="Depart"/>
  <Champ nom="Vitesse"/>
  <Champ nom="Duree"/>
  <Champ nom="Vitesse_max"/>
  <Champ nom="Vitesse_min"/>
  <Champ nom="Point"/>
  <Champ nom="Vitesse_inst"/>
</Champs>
</TableFait>
- <Dimensions>
- <Dimension nom="Tour" champFait="TOUR">
- <Niveaux>
  <Niveau id="1" nom="TOUR"/>
  <Niveau id="2" nom="TOUS_TOURS"/>
</Niveaux>
</Dimension>
+ <Dimension nom="DECOUPAGE_PISTE" champFait="SEGMENT_ATOMIQUE"></Dimension>
+ <Dimension nom="ATHLETE" champFait="ATHLETE"></Dimension>
+ <Dimension nom="TEMPS" champFait="JOUR_COURSE"></Dimension>
+ <Dimension nom="EPREUVE" champFait="EPREUVE"></Dimension>
+ <Dimension nom="DEPART" champFait="DEPART"></Dimension>
</Dimensions>
- <Mesures>
  <Measure nom="VITESSE_MOYENNE" type="MEAN" sourceMeasure="VITESSE"/>
  <Measure nom="SUM_DUREE" type="SUM" sourceMeasure="DUREE"/>
  <Measure nom="VITESSE_MAX" type="MAX" sourceMeasure="VITESSE"/>
  <Measure nom="VITESSE_MIN" type="MIN" sourceMeasure="VITESSE"/>
  <Measure nom="VITESSE_INSTANTANEE" type="MEAN" sourceMeasure="VITESSE_INST"/>
</Mesures>
+ <TableFaitSAS nom="Fait_Technique_sas"></TableFaitSAS>
+ <TableFaitAgregé nom="Fait_Agregé_Tech"></TableFaitAgregé>
</Cube>
</Cubes>
</BDMultidim>

```

ANNEXE F - Diagrammes SOLAP-SPORT après optimisation

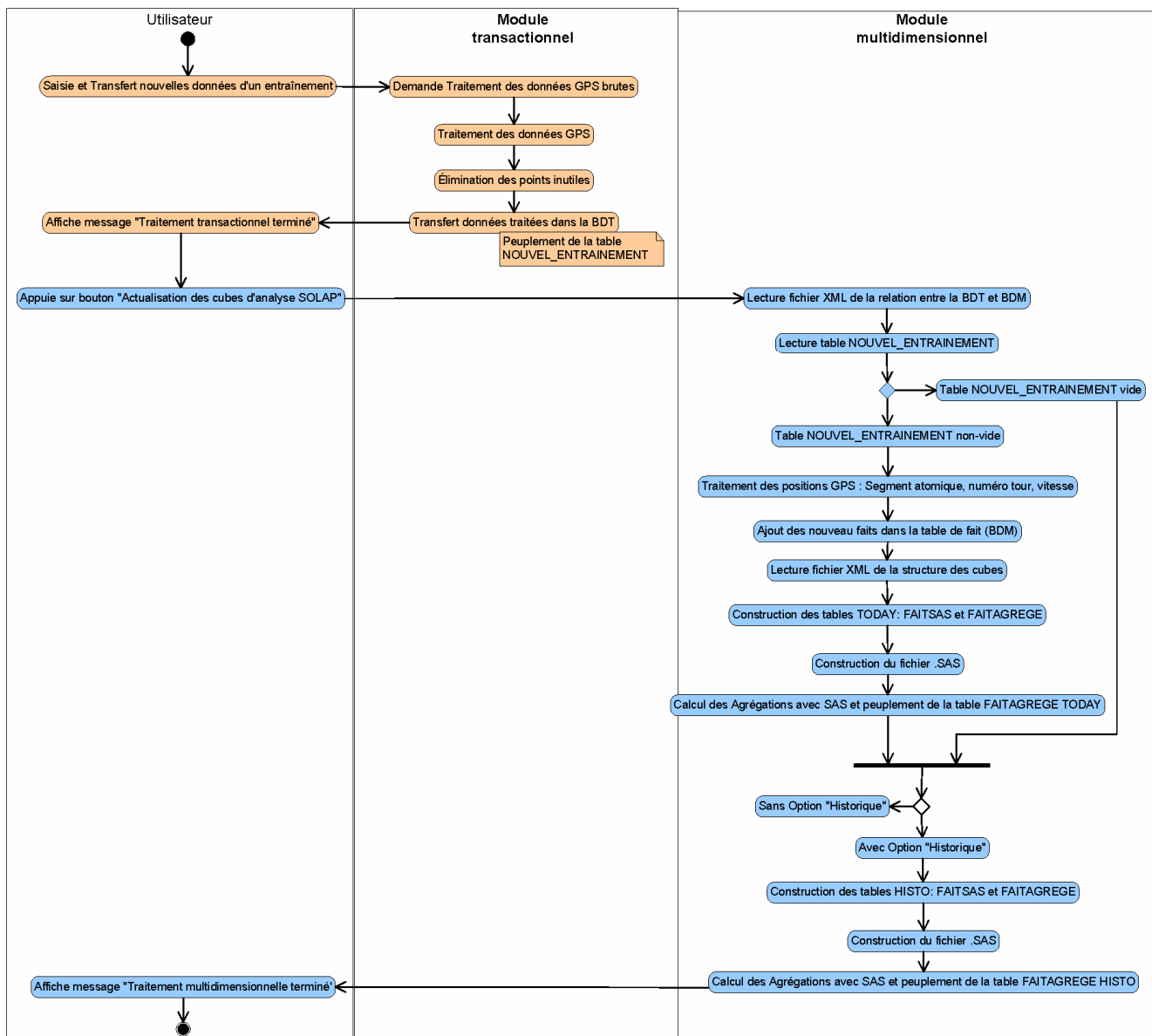


Figure 0-11 : Diagramme d'activités SOLAP-SPORT après optimisation du prototype

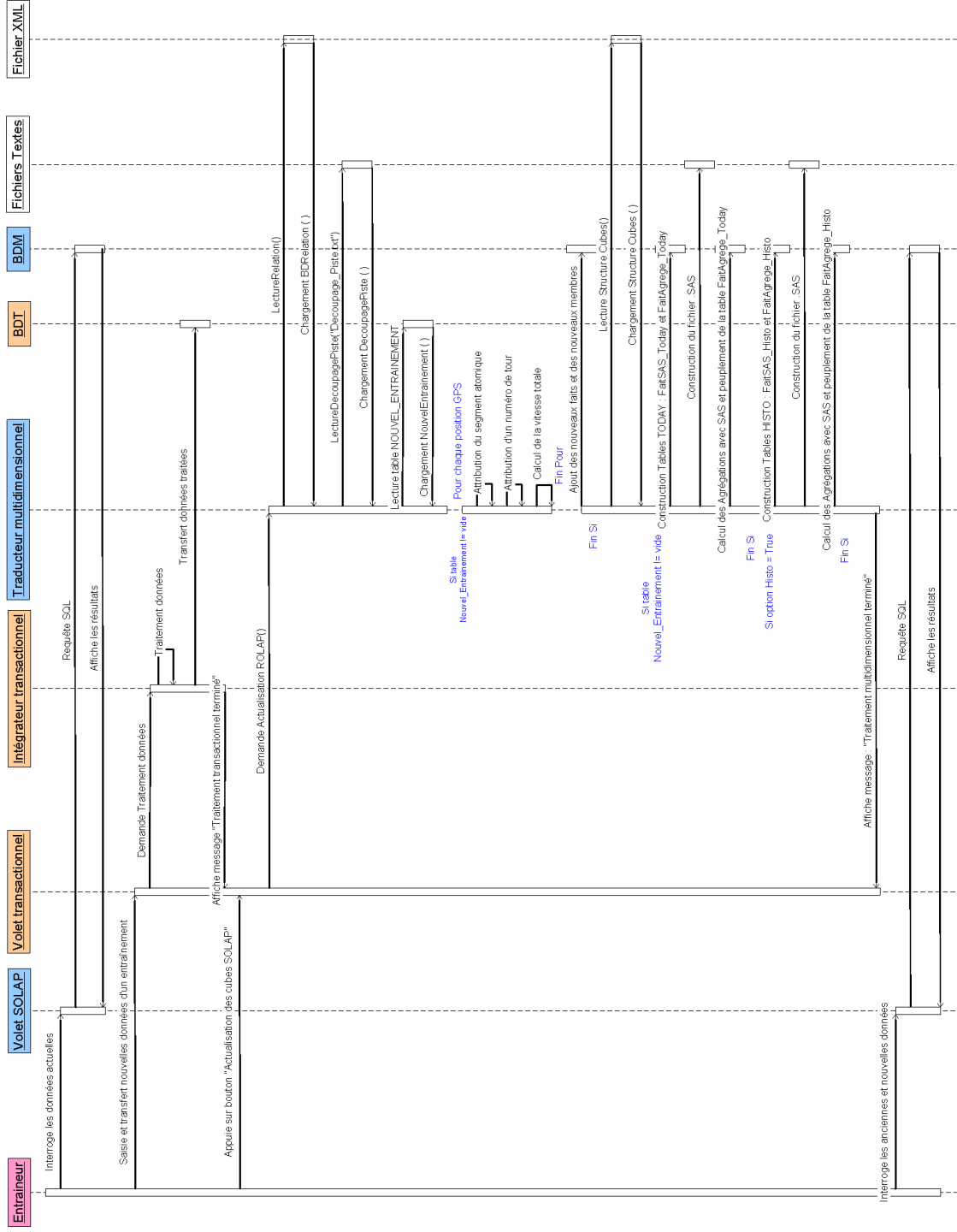


Figure 0-12 : Diagramme d'activités SOLAP-SPORT après optimisation du prototype

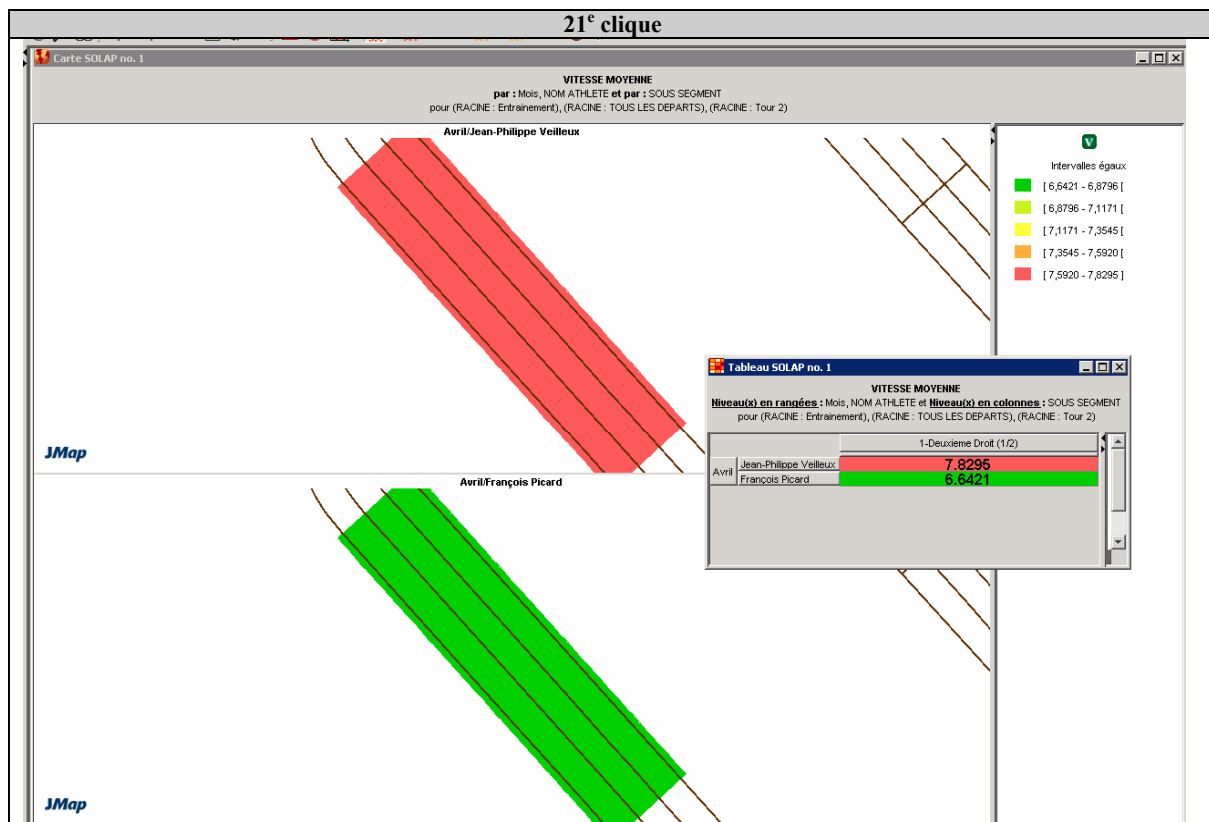
ANNEXE G - Comparaison entre une analyse réalisée sur un SIG et sur un SOLAP

Un entraîneur désire comparer les performances de deux athlètes, *François* et *Jean-Philippe*, lors des entraînements qui ont eu lieu au mois d'*Avril* de l'année 2004. Il désire comparer plus particulièrement la vitesse moyenne atteinte dans la *première moitié du 2^e droit* (Niveau *Sous-segment*), mais seulement pour le deuxième tour de piste.

(1) Étapes requises sur un SOLAP.

1. Ouvrir le cube *Patin_de_vitesse_Technique_Histo*
2. Sélectionner :
 - le membre Vitesse moyenne sous le bouton **Mesures**
 - le membre 2^{ème} Droit (1/2) sous le bouton **Découpage Piste**
 - le membre Entraînement sous le bouton **Épreuve**
 - le membre Tour#2 sous le bouton **Tour de Piste**
 - les membres François et Jean-Philippe sous le bouton **Athlète**
 - et le membre Avril sous le bouton **Temps**

3 ^e clique	7 ^e clique	9 ^e clique
<p>Mesures</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> VITESSE MOYENNE <input type="checkbox"/> DUREE <input type="checkbox"/> VITESSE MAX <input type="checkbox"/> VITESSE MIN <input type="checkbox"/> VITESSE INSTANTANEE <p>Decoupage Piste Point</p> <p>Epreuve</p> <p>Depart</p> <p>Tour de piste</p> <p>Athlete (H)</p> <p>Temps Sport</p>	<p>Mesures</p> <p>Decoupage Piste Point</p> <ul style="list-style-type: none"> Decoupage Piste Point <ul style="list-style-type: none"> 1-Deuxieme Droit (1/2) <p>Epreuve</p> <p>Depart</p> <p>Tour de piste</p> <p>Athlete (H)</p> <p>Temps Sport</p>	<p>Mesures</p> <p>Decoupage Piste Point</p> <p>Epreuve</p> <ul style="list-style-type: none"> Entraînement <p>Depart</p> <p>Tour de piste</p> <p>Athlete (H)</p> <p>Temps Sport</p>
11 ^e clique	15 ^e et 16 ^e clique	19 ^e clique
<p>Mesures</p> <p>Decoupage Piste Point</p> <p>Epreuve</p> <p>Depart</p> <p>Tour de piste</p> <p>Athlete (H)</p> <p>Temps Sport</p>	<p>Mesures</p> <p>Decoupage Piste Point</p> <p>Epreuve</p> <p>Depart</p> <p>Tour de piste</p> <p>Athlete (H)</p> <p>Temps Sport</p>	<p>Mesures</p> <p>Decoupage Piste Point</p> <p>Epreuve</p> <p>Depart</p> <p>Tour de piste</p> <p>Athlete (H)</p> <p>Temps Sport</p>



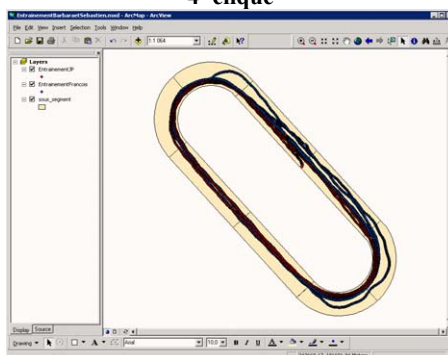
Environ 20 clics, mais une seule opération : **Sélection de membres...**

(2) Étapes requises sur un SIG

Voici les étapes décrivant une façon d'obtenir le même résultat avec le logiciel *ArcGIS* de la compagnie *ESRI*.

1. Ajouter le thème contenant tous les entraînements réalisés par *Jean-Philippe*
2. Ajouter le thème contenant tous les entraînements réalisés par *François*
3. Ajouter le thème Sous-segments

4^e clique

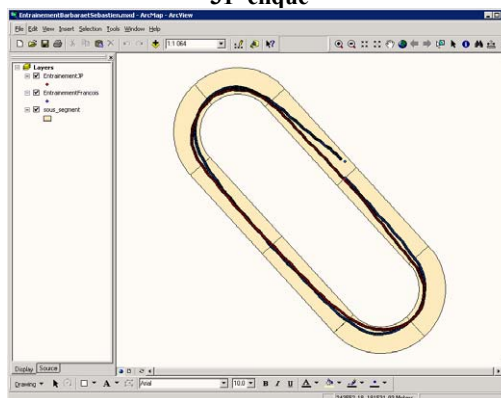


4. Conserver seulement les entraînements du mois d'*Avril 2004* et les tours de piste #2 pour chaque athlète.

Opérations : Bouton de droit sur le thème **EntraînementJP** – Properties... – Definition Query – Query Builder... - "Tour" =2 And "Mois"= "Avril" And "Annee"= "2004" .

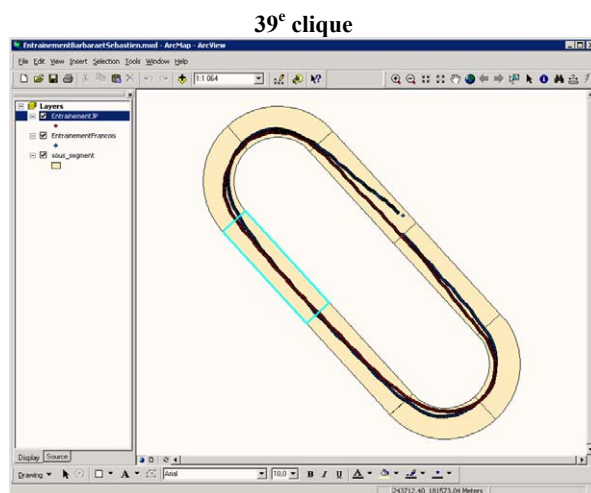
Les mêmes opérations doivent être effectuées, mais cette fois pour le thème **EntraînementFrançois**

31^e clique

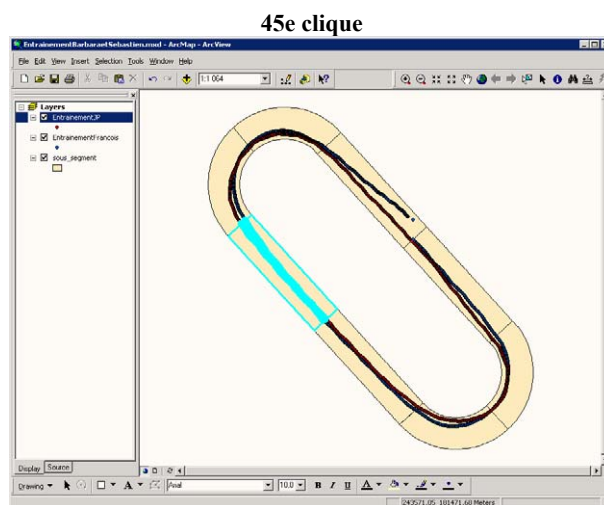


5. Sélectionner la première moitié du 2^e droit :

Opérations : Selection – Select By Attributes... – Layer : *sous_segment* – Fields : *nom_enfant* – bouton = - 2^e droit (1/2) – Apply – Close.

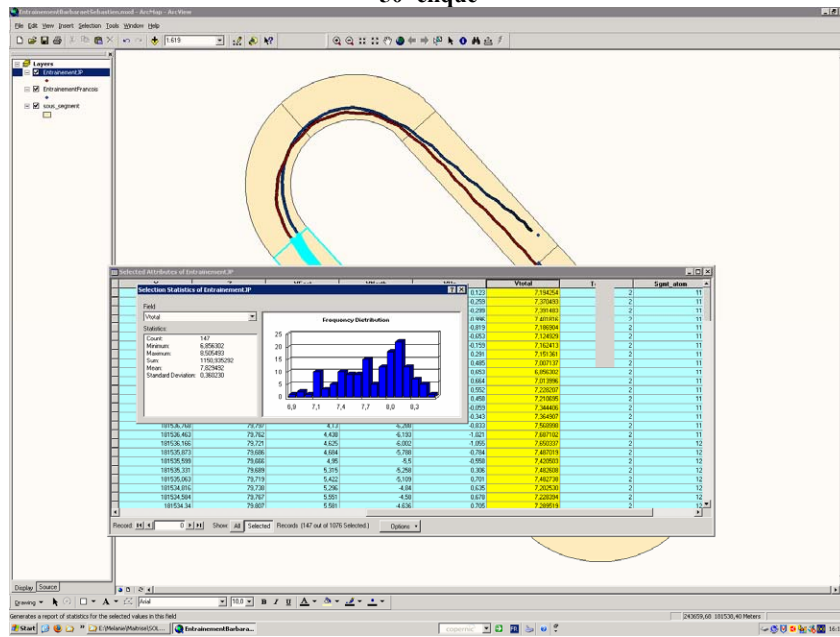


6. Sélectionner les positions instantanées présentes à l'intérieur ce sous-segment :
Opérations : Sélection – Select By Location...

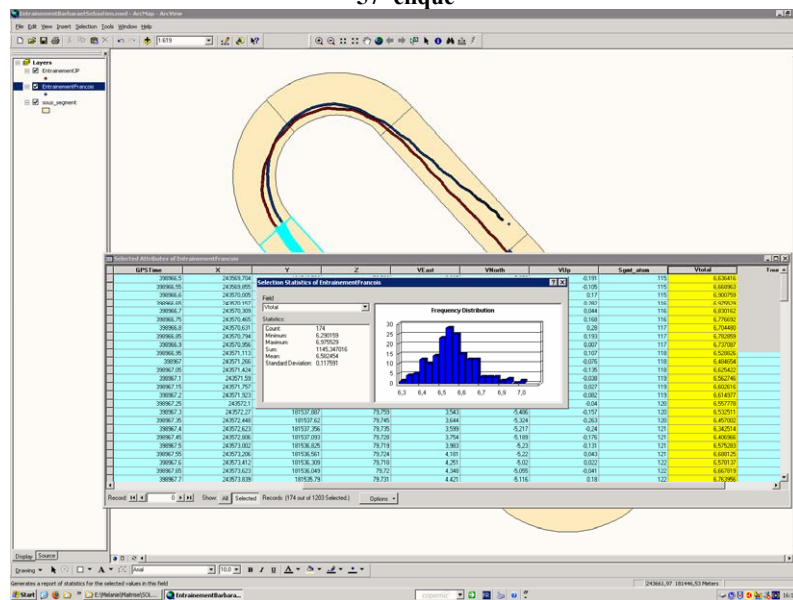


7. Calcul la moyenne des vitesses instantanées pour chaque entraînement à partir de la table des attributs.
Opérations : Bouton de droit – Open Attributes table ... - Selected – Sélectionne la colonne VTotale – Bouton de droit – Statistics..

50° cliques



57° cliques



Près de 60 clics et plusieurs opérations...