Institutional Repository - Research Portal
Dépôt Institutionnel - Portail de la Recherche
researchportal.unamur.be
University of Namur

THESIS / THÈSE

MASTER DE SPÉCIALISATION EN INFORMATIQUE ET INNOVATION

De l'ERP à la BI: Complémentarité et Intégration

Thiry, Cédric

Award date:
2016

Awarding institution:
Universite de Namur

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 19. Apr. 2024

Business Analysis et Gouvernance IT

PROJET D'ANALYSE

DE L'ERP À LA BI : COMPLÉMENTARITÉ ET INTÉGRATION

Cédric THIRY

Mémoire présenté en vue de

l'obtention du titre de

Master de spécialisation en

Informatique et Innovation

Mémoire encadré par

Professeur Naji HABRA et

Bertrand VERLAINE

ANNÉE ACADÉMIQUE 2015 - 2016

Faculté d'informatique

Faculté des Sciences économiques, sociales et de gestion

Département des Sciences de gestion

Université de Namur

TABLE DES MATIÈRES

INTRODUCTION **3**

CHAPITRE 1 - L'entreprise : Agilux **4**

1.1 Présentation de l'entreprise 4

1.2 Stratégie commerciale 5

1.3 Partenariats et certifications 6

1.4 Méthodes de travail 6

1.5 *Business Model* 7

CHAPITRE 2 - Le progiciel : Mercator **8**

2.1 Présentation du logiciel 8

2.2 Fonctionnalités de base 8

2.3 Philosophie et principes 10

2.4 Structure des données 11

2.5 Technologies 14

2.6 Module de *reporting* 15

CHAPITRE 3 - Le projet : *Business Intelligence* **20**

3.1 Origine du projet 21

3.2 Étude de faisabilité 21

3.3 Collecte des exigences 25

3.4 Analyse des exigences 28

3.5 Triage des exigences 31

3.6 Spécification des exigences 32

CHAPITRE 4 - La solution : Microsoft Power BI **39**

4.1 Étude de marché 39

4.2 Focus sur Microsoft Power BI 43

4.3 Perspectives 46

CONCLUSION **47**

RÉFÉRENCES BIBLIOGRAPHIQUES **47**

INTRODUCTION

Ce projet d'analyse s'inscrit dans le cadre conjoint du Master de spécialisation en Informatique et Innovation dispensé par l'UNamur et des activités de l'entreprise de services informatiques Agilux. Le point de départ de la réflexion se situe au niveau du logiciel ERP Mercator qui, en standard, dispose d'un module de *reporting* dont les limitations seront énoncées en cours d'analyse. Sur base de ce constat, un client d'Agilux s'est posé la question de savoir s'il existait une alternative plus efficace pour analyser les données issues de l'ERP. La *Business Intelligence* semble être la réponse à la question qui a initié ce projet d'analyse visant un double objectif. D'une part, il a pour vocation initiale de répondre à la demande de mise en place d'une solution de *Business Intelligence*. D'autre part, il s'inscrit dans les premières démarches d'Agilux sur le marché de la *Business Intelligence*.

D'un point de vue chronologique, l'analyse se situe avant l'acceptation du projet par le client. Il s'agit donc de prémices articulés en quatre chapitres. Le premier vise à présenter l'entreprise Agilux en termes de stratégie commerciale, de partenariats, de méthodes de travail et de *Business Model*. Le deuxième chapitre décrit de façon pratique, technique, schématique et détaillée le logiciel ERP Mercator et de son module de *reporting* pour lequel les faiblesses sont mises en évidence. Le troisième chapitre reprend la structure des phases de l'ingénierie des exigences dans le but de bien cerner les besoins du métier et d'y répondre au mieux grâce à la *Business Intelligence*. Le quatrième et dernier chapitre présente les solutions qui découlent des exigences collectées, tout en mettant l'accent sur celle qui tient le haut pavé aux yeux d'Agilux.

Tout en long de l'analyse, les besoins du client sont mis en parallèle avec les solutions mises en œuvre par Agilux pour y répondre. Au final, le résultat escompté est d'aboutir à une solution *win-win* dans laquelle le client disposerait d'un outil de *Business Intelligence* adapté à ses desiderata et Agilux aurait saisi l'opportunité de se spécialiser dans ce domaine. En effet, Agilux estime que l'intégration ERP et la BI pourraient être des disciplines complémentaires dans le sens où la première constitue la principale source de données pour la seconde.

CHAPITRE 1 - L'entreprise : Agilux

Ce premier chapitre a pour objectif de décrire l'environnement de travail dans lequel le projet d'analyse est mené. Tout d'abord, les activités et la structure de l'entreprise Agilux sont présentées. Ensuite, les aspects stratégiques ainsi que les partenariats sont décrits afin de démontrer la dynamique dans laquelle s'inscrit Agilux. Les méthodes de travail sont, quant à elles, brièvement expliquées. Au final, tous ces éléments sont mis en commun afin de construire le Business Model de l'entreprise.

1.1 Présentation de l'entreprise

Agilux est une entreprise de services informatiques initialement fondée en 2008 à Tenneville, en Belgique, sous la forme d'une SPRL portant le nom de SI Genin. Celle-ci est née suite au rachat d'un portefeuille de clients appartenant à une entreprise mettant fin à ses activités. En octobre 2014, l'entreprise délocalise son siège social à Pommerloch, au nord-ouest du Grand-Duché de Luxembourg, et prend la forme d'une SARL répondant au nom d'Agilux. Les activités d'Agilux gravitent autour de trois axes interdépendants, à savoir « ERP - WEB - APP ». En effet, les trois spécialisations de l'entreprise sont *(i)* l'intégration et le paramétrage d'un progiciel de gestion intégré (ERP) nommé Mercator[1] et édité par Ineo ; *(ii)* la conception de sites (WEB) e-commerce basés sur le moteur nopCommerce[2] ; *(iii)* le développement d'applications mobiles (APP) sous les environnements Windows, Androïd et iOS grâce l'outil Xamarin[3].

Actuellement, Agilux emploie six personnes, à savoir le gérant qui prend en charge les tâches technico-commerciales, deux développeurs spécialisés dans l'intégration de Mercator, un développeur spécialisé en nopCommerce, un développeur spécialisé en Xamarin et moi-même qui occupe le poste de *Business Analyst*. Cette position consiste à identifier les besoins du client, à les retranscrire au sein d'une analyse, à faire le lien entre les aspects techniques et les besoins métier tout en suivant l'évolution de chaque projet en cours. De plus, mes tâches m'amènent régulièrement à effectuer des développements spécifiques au sein de Mercator, notamment au niveau des modules de comptabilité.

En prenant du recul, la structure de l'entreprise se rapproche fortement de la configuration entrepreneuriale proposée par MINTZBERG (1982). En effet, la division du travail est imprécise et chaque membre intervient au niveau de plusieurs disciplines au sein de différents projets. Par exemple, il est fréquent qu'une même personne s'occupe de l'analyse, implémente les *requirements* et gère le projet en parallèle. De plus, le pouvoir est entièrement détenu par le gérant qui propose des solutions pour subvenir aux besoins des clients. Agilux évolue donc dans un environnement dynamique.

1. Le site web officiel du logiciel Mercator est accessible via le lien ci-après : `www.mercator.eu`
2. Le site web officiel du moteur nopCommerce est accessible via le lien ci-après : `www.nopcommerce.com`
3. Le site web officel de Xamarin est accessible via le lien ci-après : `www.xamarin.com`

Pour aller plus loin, en se basant sur les styles organisationnels mis au point par HANDY (1976) et synthétisés par MAQUET (2015, p. 49), Agilux allie une culture de club et, dans une moindre mesure, une culture de tâche. De fait, le gérant se situe au centre de l'organisation, prend toutes les décisions et délègue les missions aux employés. Ces missions peuvent être assimilées à des tâches qui sont confiées à la personne la plus apte à la résoudre. Ce profil présente évidemment des inconvénients car les décisions prises unilatéralement par le gérant ne bénéficient pas toujours du recul nécessaire, la pression est parfois forte et les employés ne sont pas assez nombreux pour acquérir toute l'expertise nécessaire à la réalisation des tâches.

1.2 Stratégie commerciale

Depuis son implantation au Grand-Duché de Luxembourg en 2014, Agilux a traité avec environ quatre-vingt clients. D'un point de vue géographique, les clients se concentrent principalement au Grand-Duché de Luxembourg et en Belgique, majoritairement en province de Luxembourg et au sud de la province de Namur. Enfin, une minorité de clients sont localisés dans les autres provinces wallonnes.

En termes de marché, Agilux fournit ses services aux (très) petites et moyennes entreprises, et ce quel que soit le secteur d'activité. En effet, les entreprises qui font appel aux services sont actives dans des secteurs aussi divers que variés, comme par exemple la distribution de boissons, la quincaillerie, l'alimentaire, la vente au détail, la mode, l'agriculture, l'enseignement, etc.

La stratégie commerciale de base consiste donc en l'intégration du logiciel ERP Mercator. Cette activité génère des prestations de services rémunérées qui sont destinées à mettre en place et à personnaliser le comportement du progiciel de gestion intégré, et ce dans l'optique de répondre aux spécificités de l'activité du demandeur. De plus, suite à la vente de licences et aux prestations réalisées, Agilux perçoit un revenu récurrent sous forme de contrats de services qui correspond à un pourcentage du chiffre d'affaires généré par le client. Ensuite, en fonction des besoins du demandeur, Agilux propose des solutions avancées qui permettent à celui-ci de développer son activité, soit en optimisant ses processus internes à l'aide d'applications dédiées, soit en améliorant les flux avec les partenaires commerciaux via des portails d'échanges B2B, soit en élargissant l'horizon commercial par le biais d'un site e-commerce, d'applications mobiles, etc. Cet accompagnement du client est également source de prestations facturées.

1.3 Partenariats et certifications

Dans le but de proposer une gamme de services complète et professionnelle, Agilux a développé plusieurs partenariats et est engagé dans des programmes de certification.

Outre le partenariat avec Ineo évoqué dans la section 1.2 (page 5), Agilux est depuis 2015 *Silver Partner* avec la société qui a donné naissance au moteur e-commerce nopCommerce, à savoir Nop Solutions Ltd[4]. Deux développeurs sont actuellement certifiés par Nop Solutions Ltd. De plus, afin de soigner l'aspect graphique des sites, Agilux exploite les thèmes et plugins visuels proposés par Nop Templates[5]. Au niveau des applications mobiles, des partenariats ont été noués avec Xamarin et Telerik[6]. Enfin, en ce qui concerne les hébergements, Agilux travaille en collaboration avec Cloud Power[7] et SmarterASP[8].

1.4 Méthodes de travail

Comme son nom l'indique, Agilux applique des pratiques de réalisation de projet qui s'apparentent aux méthodes Agiles. D'une part, lorsqu'il s'agit de subvenir aux besoins de petits clients, certains principes Agile énoncés par AGILE ALLIANCE (2016) et résumés par KOLP (2016, p. 25) sont mis en pratique, à savoir le fait de travailler main dans la main avec le demandeur, la communication directe et l'acceptation des *Change Requests*. D'autre part, c'est seulement dans le cadre de la mise en place de modules plus conséquents que des itérations sont mises en place, et ce à un rythme soutenu d'une livraison toutes les deux semaines. Dans les deux cas, l'objectif est de satisfaire le client en prenant soin de livrer des solutions simples et robustes.

4. La liste des partenaires de Nop Solutions Ltd est accessible via le lien ci-après : `www.nopcommerce.com/solutionpartners.aspx`
5. Le site web officiel de Nop Templates est accessible via le lien ci-après `www.nop-templates.com`
6. Le site web officiel de Telerik est accessible via le lien ci-après : `www.telerik.com`
7. Le site web officiel de Cloud Power est accessible via le lien ci-après : `www.cloud-power.be`
8. Le site web officiel de l'hébergeur SmarterASP est accessible via le lien ci-après : `www.smarterasp.net`

1.5 *Business Model*

Afin de mettre en lumière les facteurs clés d'Agilux, le modèle utilisé est le *Business Model Canvas* fourni par STRATEGIZER (2014). La figure 1.1 (page 7) illustre le *Business Model* actuel de l'entreprise. Quatre couleurs sont employées dans le but de distinguer les éléments communs (jaune), le segment ERP (orange), le segment WEB (bleu) et le segment APP (vert).

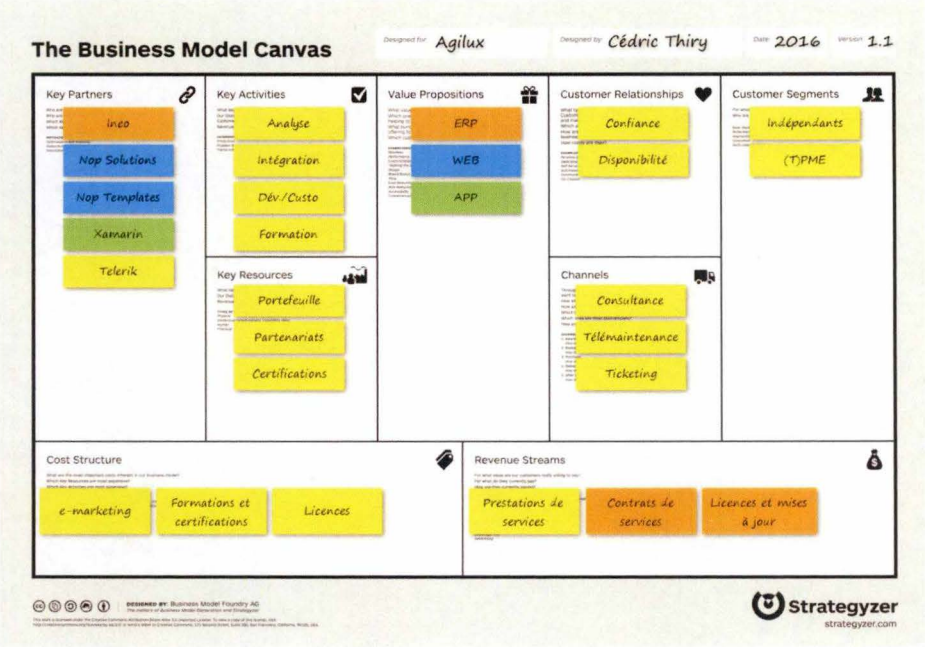

FIGURE 1.1 – *Business Model* d'Agilux

La proposition de valeur d'Agilux réside en la prestation de services visant à fournir un logiciel ERP, des sites web et des applications spécifiques. Les clients sont des (très) petites et moyennes entreprises avec qui Agilux entretient des relations basées sur la confiance et la disponibilité dans le but de subvenir aux besoins. Pour ce faire, Agilux propose des journées de consultance sur site, agit par télémaintenance et assure le support par le biais d'un système de ticketing. Toutes ces activités sont fournies dans le cadre de contrats de services et de prestations de services, auxquels s'ajoutent des revenus perçus par la vente de licences et de contrats de mises à jour. En effet, le logiciel ERP Mercator commercialisé par Ineo nécessite l'achat de licences et d'un contrat de mises à jour annuel dont le coût est proportionnel à la valeur des licences. Agilux perçoit une commission sur ces ventes. Cependant, les principaux revenus proviennent des activités telles que l'analyse, l'intégration, la formation et les développements sur-mesure réalisés dans Mercator et/ou dans le cadre de la mise en place de sites web, d'applications, etc. Ces développements particuliers sont respectivement mis en place à l'aide d'outils proposés par les différents partenaires. Ces derniers, par le biais de certifications, constituent les ressources clés d'Agilux. En effet, Agilux n'hésite pas à investir dans des formations et dans des technologies de pointe afin de proposer des services de qualité. Enfin, en plus du bouche à oreille, des campagnes d'e-marketing permettent de toucher de nouveaux clients qui viennent grossir le portefeuille actuel.

CHAPITRE 2 - Le progiciel : Mercator

Dans le but de poser les bases nécessaires au projet d'analyse, il est nécessaire de présenter l'origine, le fonctionnement (pratique et technique) et la philosophie de Mercator qui n'est autre que le logiciel ERP qui génère la grande majorité des données exploitables par la *Business Intelligence*. De plus, la structure technique est décrite sur base de deux approches, à savoir le schéma entité-association et le schéma de la base de données. Enfin, ce chapitre se referme par la présentation et la critique du module de *reporting* de Mercator.

2.1 Présentation du logiciel

Mercator est un progiciel de gestion intégré plus couramment abrévié ERP sur base de la dénomination anglaise *Enterprise Resources Planning software*. Le logiciel est édité par la société Ineo depuis 1997. Celui-ci est né sur base du constat qu'une solution standard ne permettait pas de répondre parfaitement aux besoins des entreprises et que le développement d'un logiciel sur-mesure était excessivement onéreux et chronophage. Par conséquent, Ineo a mis au point une base de travail qui permet un paramétrage extrêmement poussé, et ce via un grand nombre d'options et des possibilités de customisation dans les différents éditeurs de scripts.

En termes de cible, Mercator s'adresse aux (très) petites et moyennes entreprises par le biais d'un réseau de revendeurs. Actuellement, Ineo travaille avec plusieurs dizaines de revendeurs tels qu'Agilux et se targue de toucher 20 000 utilisateurs[9]. Le logiciel est principalement utilisé en Belgique mais s'adapte également aux législations en vigueur au Grand-Duché de Luxembourg et en France. La figure 2.1 (page 9) donne un aperçu visuel du logiciel.

2.2 Fonctionnalités de base

En termes de fonctionnalités, Mercator existe en plusieurs versions présentées de façon sommaire dans le tableau 2.1 (page 8).

Version	Description
Full Edition	La version la plus complète qui possède tous les modules de base
Sales Only Edition	Une version dans laquelle seules les ventes sont possibles
Light Edition	Une version dans laquelle seules les ventes sont possibles et pour laquelle aucun module optionnel n'est disponible
First Business Edition	Une version dans laquelle seules les ventes sont possibles mais sans gestion de stock
Facturation Décentralisée	Une version qui est prévue pour être utilisée à distance et qui se synchronise avec un Mercator Full Edition qui fait office de centrale
Caisse Décentralisée	Une version qui est prévue pour être utilisée en tant que caisse dans différents points de vente et qui se synchronise avec la centrale
Comptabilité Autonome	Une version dans laquelle seule la comptabilité est disponible

TABLEAU 2.1 – Versions de Mercator

9. Ce chiffre est issu de la vidéo promotionnelle accessible via le lien ci-après : `https://youtu.be/4WXZ8URMH1M?t=34s`.

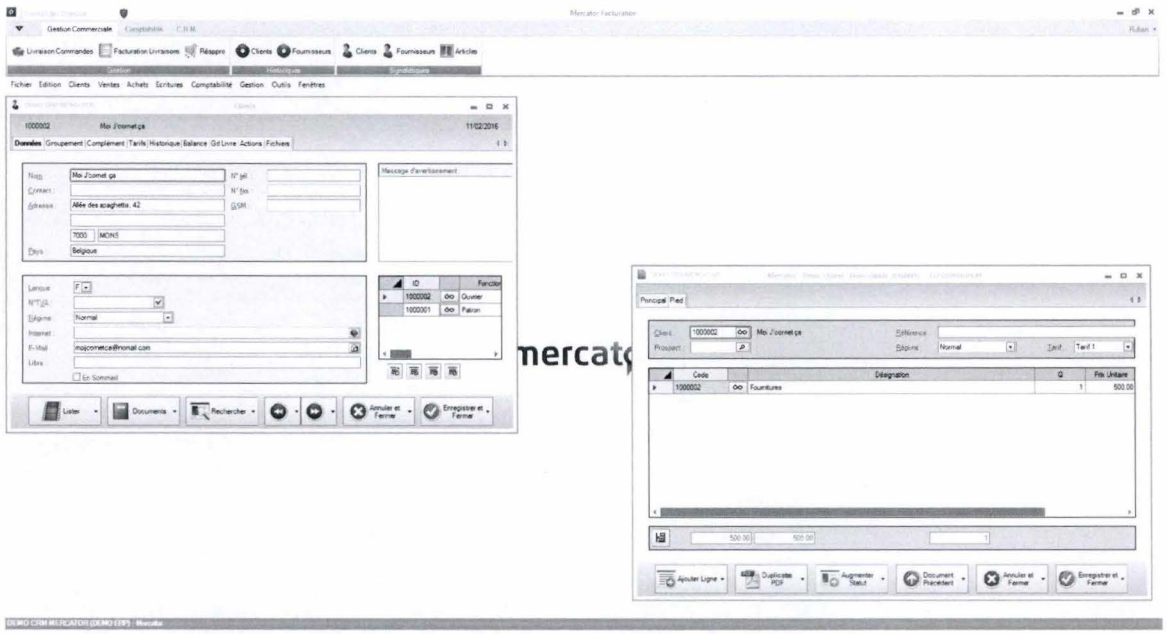

FIGURE 2.1 – Capture d'écran de Mercator (fiche client et devis)

Par ailleurs, Mercator dispose de plusieurs modules optionnels qui permettent d'enrichir ses fonctionnalités de base. Les principaux sont brièvement décrits le tableau 2.2 (page 9).

Module	Description
Lots	Permet de gérer les numéros de lot des articles achetés et vendus
PDF	Permet d'envoyer les documents par e-mail et de générer des duplicatas qui sont historisés
Production	Permet de combiner plusieurs articles considérés comme des matières premières pour former un produit fini tout en gérant les stocks en conséquence
Projets	Permet de créer des projets qui peuvent être budgétisés et auxquels des ventes et des achats se rattachent
Intrastat	Permet d'effectuer des déclarations Intrastat
CRM	Permet de gérer la relation client
Comptabilité	Permet de transformer les documents de vente et d'achat en écritures desquelles découlent la gestion comptable
Comptabilité Analytique	Ajoute une fiche signalétique permettant de regrouper les flux d'une comptabilité analytique
Liaison Bancaire	Permet la gestion des fichiers utilisés lors des transactions avec les banques (extraits de compte codifiés, virements, domiciliations, etc.)
Immobilisés	Ajoute une fiche signalétique permettant de gérer l'acquisition d'immobilisés ainsi que leurs amortissements

TABLEAU 2.2 – Modules optionnels de Mercator

En raison de ses nombreuses facettes, Mercator génère donc un flux important de données. La section suivante présente comment ces informations sont structurées.

2.3 Philosophie et principes

Dans sa conception de la gestion commerciale, Mercator repose sur trois fiches signalétiques principales qui sont les articles, les clients et les fournisseurs. Ces trois axes sont inter-connectés par le biais de séquences de vente ou d'achat. La grande partie des flux de gestion commerciale sont donc générés à partir de ces trois fiches.

D'un point de vue pratique, l'utilisateur gère une liste d'articles pour lesquels il renseigne la désignation, la description, la marque, les (sous-)catégories, la structure (rayon, famille, sous-famille), les prix, les conditionnements, les remises ainsi qu'un nombre illimité de champs personnalisables.

Ces articles peuvent entrer dans un processus d'achat auprès d'un fournisseur qui, dans sa fiche signalétique, comporte des informations telles que le nom, l'adresse, les personnes de contact, les adresses e-mail, les numéros de téléphone, le compte bancaire, etc. Ce flux d'achat compte quatre étapes successives, à savoir la préparation de commande, la commande, la livraison et la facture.

A l'instar des achats, les séquences de vente à un client comptent également quatre étapes qui sont le devis, la commande, la livraison et la facture. De même, la fiche signalétique des clients affiche des données telles que le nom, l'adresse, les personnes de contact, les adresses e-mail, les numéros de téléphone, etc.

En fin de flux de gestion commerciale, la comptabilité est initiée par le biais de la centralisation, c'est-à-dire que les documents de gestion commerciale sont figés et transformés en écritures comptables. Celles-ci permettent d'afficher le plan comptable, de générer le bilan, de gérer les paiements des fournisseurs, les domiciliations des clients, les encours et les rappels, les déclarations de TVA, etc.

Tous ces éléments sont repris dans le schéma entité-association présenté sur la figure 2.2 (page 11). Quelques remarques sont formulées afin de préciser certains points :

(i). En standard, les documents de vente et d'achat possèdent les mêmes attributs, à la différence près que les premiers sont associés aux clients alors que les seconds sont associés aux fournisseurs. L'entité « Pied Document » est présentée sous forme d'une partition.

(ii). Le pied d'un document ne contient pas obligatoirement des lignes. Par conséquent, la cardinalité de l'association « Composer » est 0-N.

(iii). Une ligne d'un document ne contient pas nécessairement un article car une ligne peut éventuellement être utilisée pour ajouter des commentaires. L'association « Ajouter » est donc 0-1.

(iv). Les cardinalités 0-N qui s'appliquent sur les associations « Attribuer » et « Appliquer » s'expliquent par le fait que des conditions trop fortes soient appliquées sur les remises créées et qu'aucun client ni article ne bénéficie de la promotion.

(v). La cardinalité 0-N définie au niveau de l'association « Stocker » est due au fait que Mercator est capable de gérer des articles matérialisant des services. Ces articles ne sont pas liés à un magasin étant donné qu'ils ne disposent pas de stock physique.

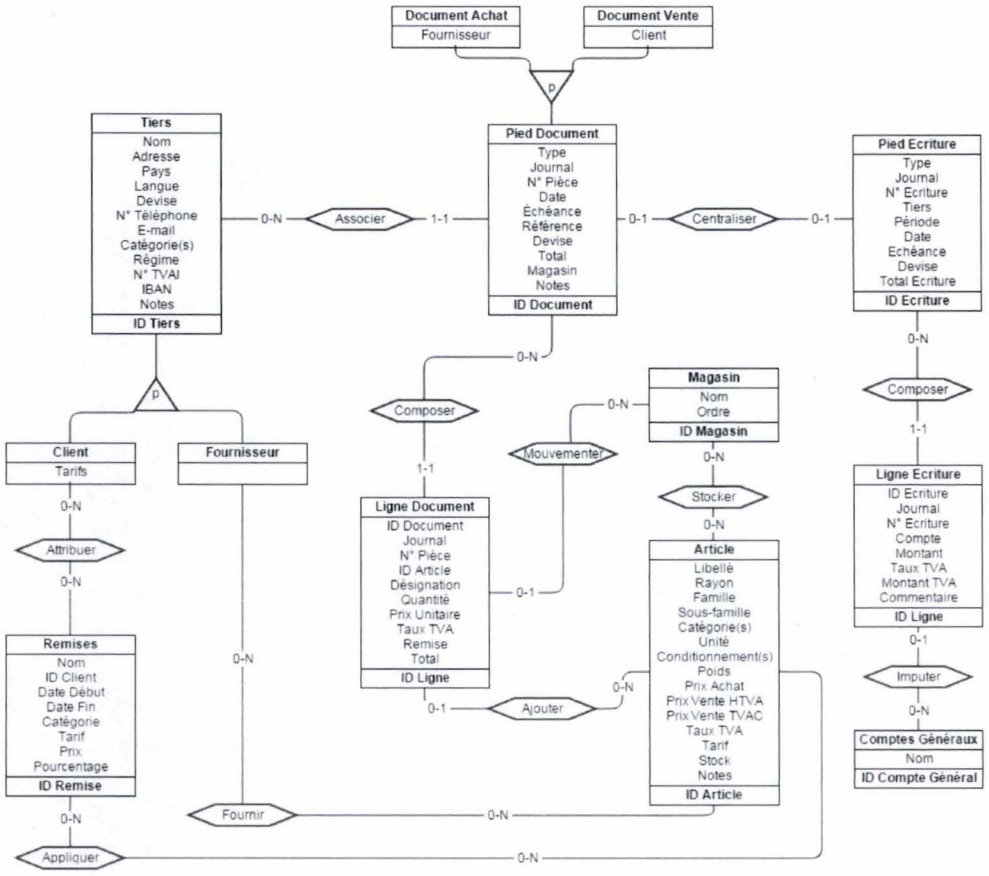

FIGURE 2.2 – Entité-Association dans Mercator

2.4 Structure des données

La base de données se présente au format Microsoft SQL Server et est gérée par le biais du système de gestion de base de données (ou DBMS) Microsoft SQL Server Management Studio (SSMS). Afin de présenter la structure de cette base de données, la figure 2.3 (page 12) représente le schéma des tables principales [10] de Mercator ainsi que leurs relations.

D'une part, la partie gauche la figure 2.3 (page 12) schématise la structure des achats. D'autre part, la partie droite décrit la structure des ventes. La partie centrale supérieure, quant à elle, représente les articles. Enfin, la partie centrale inférieure regroupe les tables inhérentes à la comptabilité.

10. Afin de ne pas surcharger le schéma structurel, seules les tables ayant un intérêt pour le projet BI sont représentées.

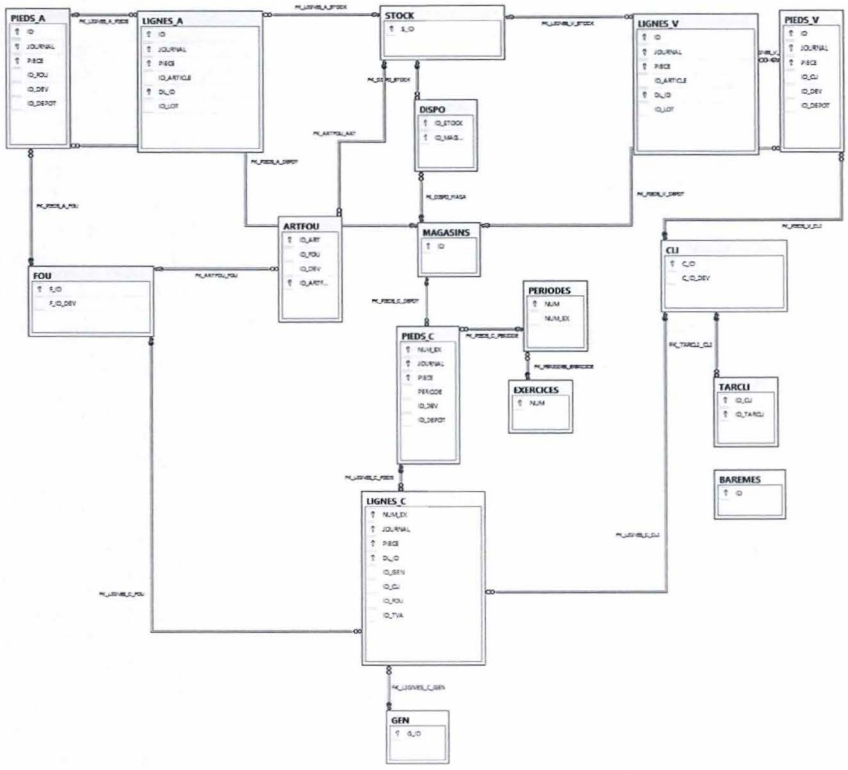

FIGURE 2.3 – Schéma de la base de données de Mercator

2.4.1 Articles

La table qui regroupe les données relatives aux articles se nomme « STOCK » et ses colonnes sont identifiables grâce à leur nom qui commence systématiquement par « S_ ». Cette table possède comme clé primaire la colonne « S_ID » qui est la référence unique de l'article composée de dix caractères alphanumériques.

En vue de gérer l'état des stocks dans les différents magasins ou dépôts, la table « STOCK » est en relation 1 : N avec la table « DISPO ». Par ailleurs, la table « MAGASINS » qui contient les données inhérentes aux différents dépôts est également en relation 1 : N avec la table « DISPO ». En d'autres termes, cela signifie que, dans la table « DISPO », chaque article possède une entrée pour chaque dépôt.

2.4.2 Catégories et hiérarchies

Dans un souci de clarté, la figure 2.3 (page 12) omet volontairement les tables permettant de catégoriser les fiches signalétiques. En effet, les articles, les clients et les fournisseurs peuvent être ordonnés selon des catégories. Celles-ci se présentent sous la forme de listes déroulantes dans lesquelles l'utilisateur peut sélectionner la classification désirée. Par exemple, pour les articles, une catégorie pourrait être nommée « Marque » avec comme choix possibles Coca-Cola, Pepsi, Spa, Lipton, etc.

Par ailleurs, les articles disposent également d'une structure hiérarchique composée de rayons, de familles et de sous-familles. Cette structure en arbre, couramment abrégée « RFS », permet par exemple de classer les articles dans une sous-famille qui dépend d'une famille et qui elle-même dépend d'un rayon. Le tableau 2.3 (page 13) présente un exemple de structure incomplète et fictive qui pourrait être mise en place dans le cas d'un grossiste en boissons.

Rayons	Familles	Sous-familles
Boissons non-alcoolisées	Eaux	Gazeuses
		Non-gazeuses
	Sodas	Limonades
		Colas
Boissons alcoolisées	Alcools	Whiskys
		Rhums
	Bières	Brunes
		Blondes
		Ambrées
		Trappistes
	Vins	Blancs
		Rouges
		Rosés

TABLEAU 2.3 – Strucure RFS

2.4.3 Achats

Du côté des achats, la structure compte quatre tables principales. La première, « PIEDS_A », regroupe toutes les informations uniques d'un document d'achat, comme par exemple le numéro d'identification du fournisseur, celui du dépôt, la date, la devise, la référence, etc. L'unicité de chaque entrée dans cette table est garantie par le trio « ID », « JOURNAL » et « PIECE ». Ces trois clés permettent d'établir la relation 1 : N avec la table « LIGNES_A » qui correspond aux articles et commentaires ajoutés dans le document. De plus, l'unicité d'une ligne est vérifiée par la colonne « DL_ID ».

Chaque document d'achat est lié à un seul et unique fournisseur de la table « FOU » via une relation 1 : N. Étant donné qu'un fournisseur peut être lié à plusieurs achats, le champ « F_ID » peut se retrouver plusieurs fois dans « PIEDS_A ».

Enfin, la dernière table, « ARTFOU » permet d'établir le lien entre les fournisseurs et les articles. En effet, la fiche signalétique article permet de renseigner tous les fournisseurs chez lesquels sont habituellement acheté l'article. Ainsi, il est possible de renseigner quel fournisseur est le principal, quels sont respectivement les prix d'achat, les remises, les références de l'article et les conditionnements auprès des différents fournisseurs.

2.4.4 Ventes

Par analogie avec la structure des achats, le schéma correspondant aux ventes est en tout point identique, à la différence des noms des différentes entités. En effet, il s'agit maintenant des tables « PIEDS_V », « LIGNES_V » et « CLI » qui sont la symétrie respective des tables « PIEDS_A », « LIGNES_A » et « FOU ».

Cependant, deux différences notables sont à relever. La première est qu'il n'y a pas de table de liaison directe entre les clients et les articles. La table « TARCLI » permet en réalité de recenser les éventuelles conditions de vente avantageuses liées à un client spécifique. Ces conditions qui prennent en considération, entre autres, les articles ne sont pas matérialisées par des relations dans la base de données. Celles-ci sont directement gérées dans une classe de la solution de Mercator, et ce afin de pouvoir gérer des cas plus complexes dans lesquels des ordres de priorité sont définis. La second est la présence de la table « BAREMES » qui donne la possibilité à l'utilisateur de programmer des promotions temporelles valables pour tous les clients. Pour les mêmes raisons que « TARCLI », les relations sont définies au niveau du code afin de prendre en compte des combinaisons avancées.

2.4.5 Comptabilité

La structure inhérente à la comptabilité est alimentée par un processus de centralisation qui transforme les documents de gestion commerciale en écritures comptables. La structure est analogue à celle des achats et des ventes, à savoir que les écritures comportent des entrées uniques dans « PIEDS_C » et des entrées répétées dans « LIGNES_C ». Cette dernière table est liée à celle des comptes généraux, « GEN », et aux signalétiques client et fournisseur. De plus, les notions d'exercices et de périodes comptables sont définies dans les tables « EXERCICES » et « PERIODES ».

2.5 Technologies

Depuis 2012, Mercator repose sur une solution basée sur le langage de programmation développé par Microsoft, le C#. Le framework .NET requis est la version 4.0. Étant donné que la solution est semi-ouverte, les développements particuliers sont réalisés dans des scripts appelés « Customizers » qui utilisent également le langage C#. Les intégrateurs n'ont donc pas accès au code source du logiciel.

Le stockage des données est réalisé par l'intermédiaire d'une base de données de type Microsoft SQL Server administrée grâce au DBMS Microsoft SQL Server Management Studio.

Enfin, la construction des rapports au format PDF est rendue possible grâce à l'intégration de l'éditeur FastReport [11] qui, en plus de ses composants propres, s'appuie également sur le langage C#.

11. Le site web officiel de l'éditeur FastReport est accessible via le lien ci-après : `www.fast-report.com`

2.6 Module de *reporting*

Cette section a pour objectif de décrire le module de *reporting* proposé par Mercator. Quelques exemples d'utilisation et de construction de rapports sont tout d'abord présentés. Ensuite, une analyse de l'outil est réalisée afin de pointer les faiblesses de l'outil. Enfin, ces problèmes sont structurés sur base de la méthodologie proposée par ISHIKAWA (1968).

2.6.1 Présentation

En standard, Mercator dispose d'un module de génération de rapports. Celui-ci permet de produire des rapports structurés (Excel, CSV ou XML) ou non (PDF) propres à chaque signalétique. Les figures 2.4 (page 15) et 2.5 (page 16) illustrent chronologiquement la procédure à suivre pour qu'un utilisateur puisse générer un rapport, à savoir :

(i). Ouvrir une fiche signalétique et, éventuellement, appliquer un filtre ;

(ii). Cliquer sur le bouton « Documents » ;

(iii). Sélectionner le rapport ;

(iv). Cliquer sur « OK » ;

(v). Choisir les paramètres [12] (dates, périodes, etc.) ;

(vi). Imprimer et/ou sauvegarder le rapport.

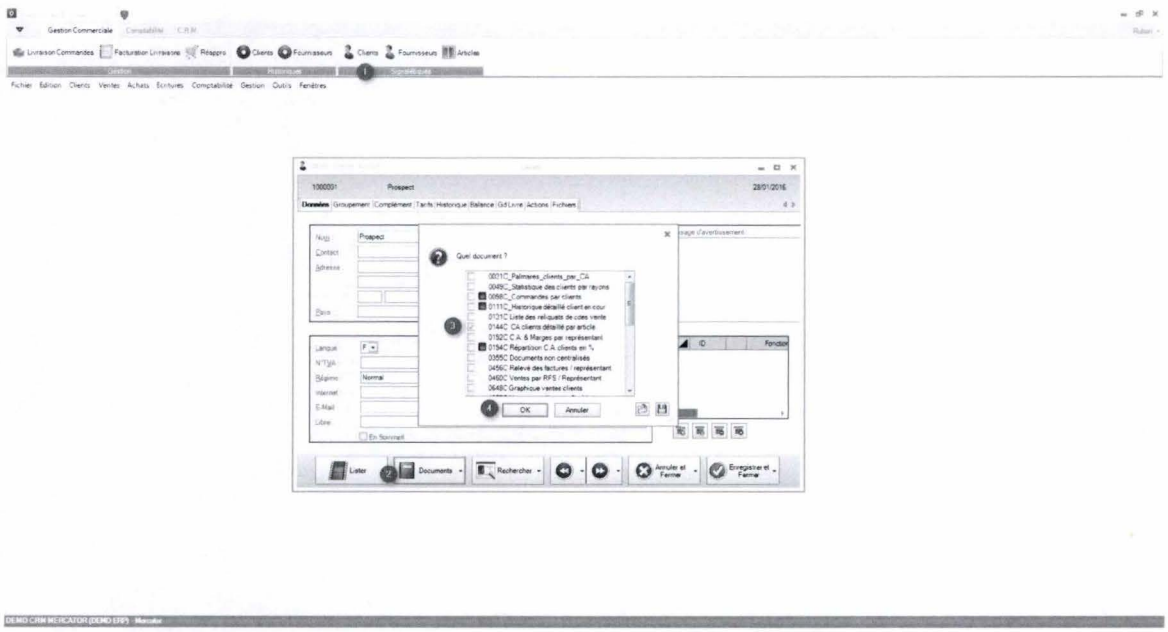

FIGURE 2.4 – Génération d'un rapport dans Mercator

12. En raison de la diversité des paramètres, ceux-ci ne sont pas illustrés sur les figures 2.4 (page 15) et 2.5 (page 16). Néanmoins, les rapports standards de Mercator indiquent dans leur en-tête quels paramètres ont été sélectionnés avant la génération.

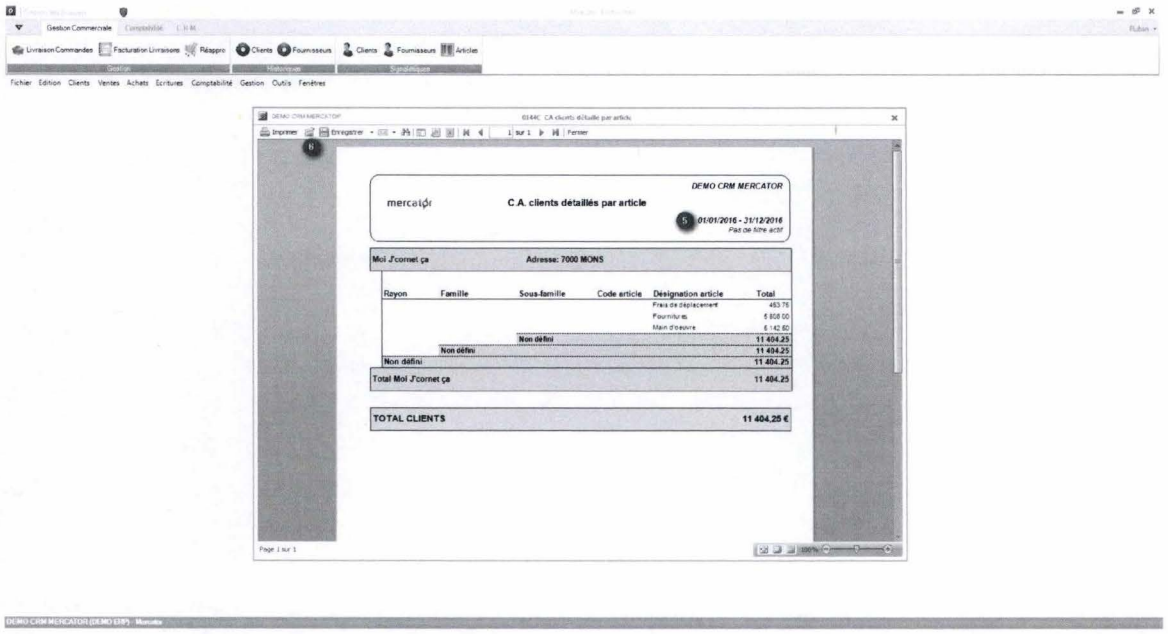

FIGURE 2.5 – Prévisualisation d'un rapport dans Mercator

2.6.2 Exemples

Par exemple, le rapport présenté sur les figures 2.4 (page 15) et 2.5 (page 16) s'intitule « C.A. clients détaillés par article ». Il permet de présenter, pour une période préalablement choisie, le chiffre d'affaires des clients avec un niveau de précision allant jusqu'à l'article. Il affiche également des sous-totaux au niveau des RFS.

Au niveau des articles, un rapport pourrait par exemple présenter la valorisation de l'inventaire sur base du dernier prix d'achat de chaque article, pour autant que les articles puissent être valorisés. La figure 2.6 (page 16) illustre cet « Inventaire global ».

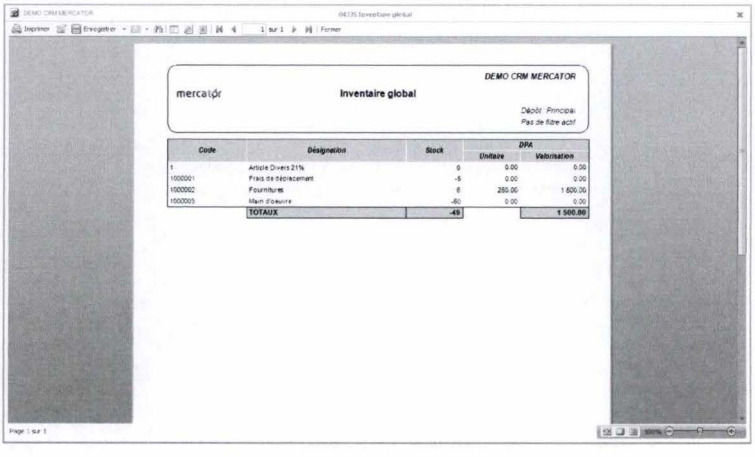

FIGURE 2.6 – Rapport « Inventaire global »

2.6.3 Construction

Au niveau technique, ces rapports sont construits sur base d'une ou plusieurs requêtes SQL qui acceptent différents paramètres afin d'élaborer des choix avant la génération (dates, périodes, affichage détaillé ou condensé, etc.). La figure 2.7 (page 17) affiche les écrans qui permettent aux intégrateurs de modifier et/ou de créer des rapports. Ces outils disposent, entre autres, des possibilités suivantes :

(i). Titre du rapport ;

(ii). Type du rapport, « Document » (PDF) ou « Fichier » (Excel ou XML) ;

(iii). Module sur lequel s'applique le rapport (Articles, Clients, Fournisseurs, etc.) ;

(iv). Editeur de requête SQL ;

(v). Bouton pour modifier le modèle d'impression (uniquement dans le cas d'un PDF) ;

(vi). Données résultant de la ou des requêtes écrites dans l'éditeur ;

(vii). Editeur de modèle dans lequel différents objets sont positionnés (bandes, données, images, graphiques, etc.).

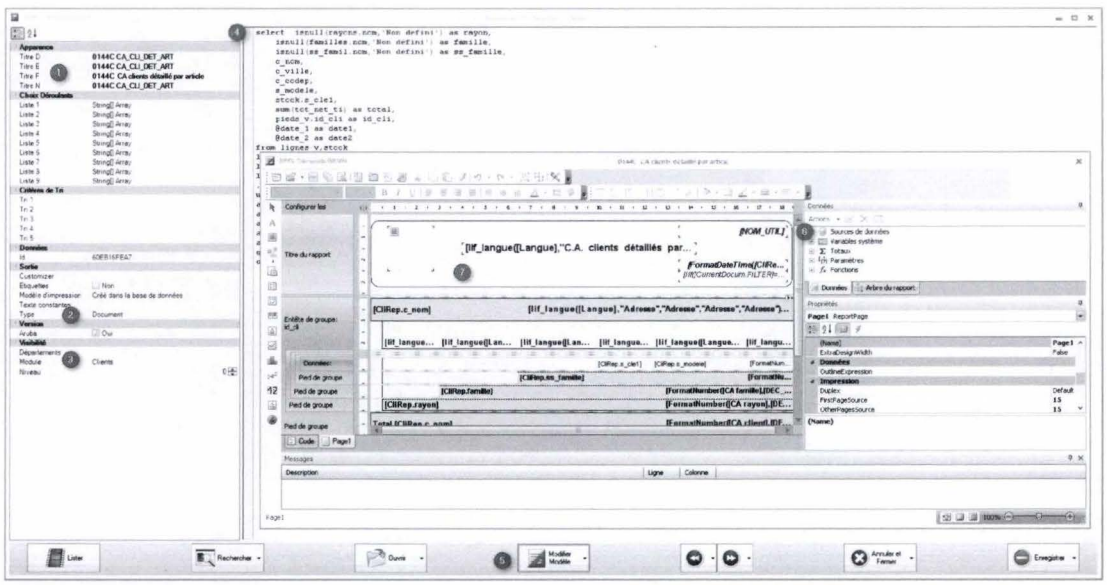

FIGURE 2.7 – Construction d'un rapport dans Mercator

2.6.4 *Root-Cause Analysis*

Bien que ce module de *reporting* soit capable de produire des rapports de grande qualité, les utilisateurs quotidien de Mercator se plaignent régulièrement de son utilisation. Afin d'identifier les problèmes rencontrés, le diagramme en arrêtes de poisson d'ISHIKAWA (1968) présenté par BURNAY (2016, p. 151) est mis en application. En effet, cette méthode part du principe qu'il est nécessaire d'identifier les causes afin de traiter un problème. Pour ce faire, les causes sont décomposées en six groupes, à savoir *Equipment*, *Process*, *People*, *Materials*, *Environment* et *Management*. La figure 2.8 (page 18) présente la décomposition du problème sur base d'un canevas proposé par l'AMERICAN SOCIETY FOR QUALITY (2008).

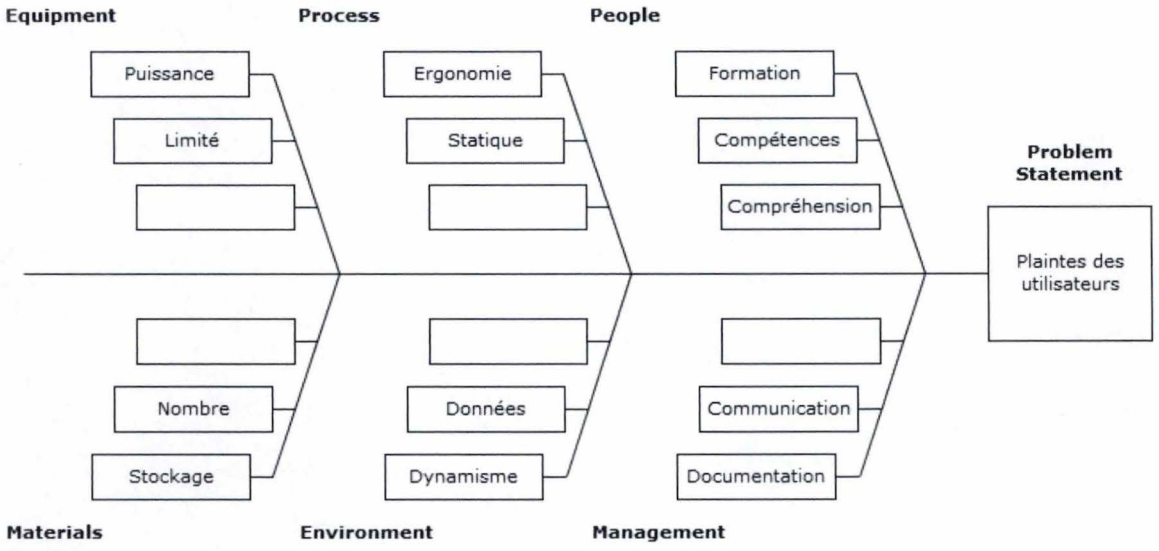

FIGURE 2.8 – Diagramme d'ISHIKAWA

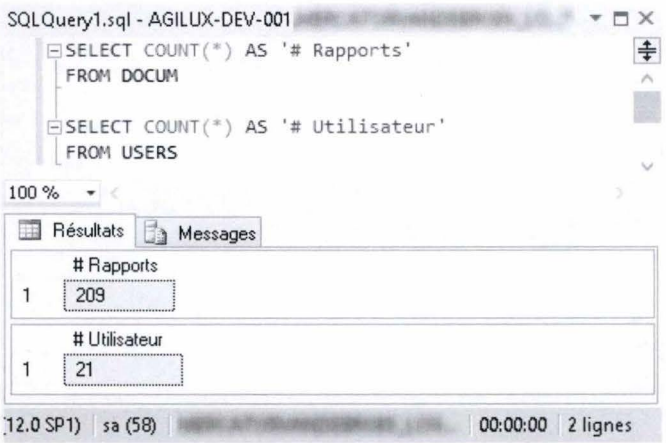

FIGURE 2.9 – Nombre de rapports dans le Mercator d'un client

Equipment : Pour fonctionner, ce module nécessite une certaine puissance afin que la génération des rapports ne prenne pas trop de temps. Si la puissance fournie par le matériel n'est pas suffisante, la génération peut durer plusieurs minutes. De plus, l'éditeur de rapport impose certaines contraintes techniques au niveau de la mise en page (format A4), de la présentation des données, etc. Par exemple, il est parfois compliqué de présenter des tableaux affichant de nombreuses données.

Process : Comme expliqué dans la section 2.6.1 (page 15), les étapes permettant de générer un rapport sont relativement nombreuses. En effet, il faut compter une quinzaine de clics avec la souris pour atteindre les données. Par ailleurs, après génération, les données présentées sont statiques, c'est-à-dire qu'il faut procéder à plusieurs générations par jour si l'utilisateur désire suivre des indicateurs en temps réel.

People : De manière générale, les utilisateurs n'ont pas suivi de formation pour maîtriser l'outil qui leur est mis à disposition. Par exemple, ils sont nombreux à ignorer que les rapports peuvent être configurés de telle sorte à prendre en compte les filtres appliqués sur les fiches signalétiques. De plus, certains utilisateurs ne comprennent pas la finalité, la structure et/ou les données des rapports qu'ils utilisent. Enfin, la configuration d'un rapport nécessite des compétences en SQL, C#, FastReport et *Business Intelligence*. Par conséquent, les utilisateurs ne sont pas autonomes lorsqu'il s'agit d'effectuer ne serait-ce que des petites modifications sur les rapports existants.

Materials : Dans le but de conserver un historique des rapports générés, les utilisateurs peuvent les imprimer et/ou les enregistrer sur disque dur. Au fil du temps, un problème de stockage pourrait survenir, qu'il soit physique (armoires) ou virtuel (disques durs). Ce problème est d'autant plus probable si le nombre de rapports différents utilisés est important.

Environment : Au sein des (très) petites entreprises, l'environnement de travail est changeant. De ce fait, les besoins évoluent régulièrement et les rapports utilisés hier ne sont pas spécialement ceux qui seront utilisés demain. De surcroît, la qualité des données saisies dans le logiciel déterminent directement la pertinence des rapports. Or, ces données ne sont pas toujours disponibles et consistantes.

Management : Habituellement, les demandes d'ajout de rapports sont initiées par les décideurs. Cependant, ceux-ci ne se concertent pas toujours avec les utilisateurs finaux. Par conséquent, les rapports mis à disposition ne sont pas pleinement utilisés car leur utilité ne correspond pas aux besoins, leur existence n'est pas connue et/ou leur utilisation n'est pas clairement documentée.

Le diagramme d'ISHIKAWA permet donc de mettre en évidence différentes causes sur lesquelles il est possible d'agir pour réduire les plaintes émises par les utilisateurs. Le chapitre 3 (page 20) propose donc l'utilisation de la *Business Intelligence* qui pourrait, du moins en partie, résorber ces problèmes.

CHAPITRE 3 - Le projet : *Business Intelligence*

Compte-tenu des éléments présentés dans le chapitre 2 (page 8), le logiciel Mercator, de par ses fonctionnalités et sa structure, permet de développer une gestion commerciale et comptable efficace. Les flux de vente et d'achat ont la particularité d'être générateurs de données qui sont ensuite figées en comptabilité et qui peuvent ensuite être exploitées par le module de *reporting*. Cependant, la section 2.6.4 (page 17) a mis en évidence un certain nombre de problèmes relatifs à l'analyse des données.

Afin de traiter ces problèmes, du moins en partie, le projet d'analyse étudie étape par étape la possibilité d'intégrer un outil de *Business Intelligence*. Le processus d'analyse est calqué sur les phases de l'ingénierie des exigences synthétisées par BURNAY (2016, p. 23) sur base de l'approche proposée par HICKEY et DAVIS (2004, p. 67). Ces phases sont illustrées sur la figure 3.1 (page 20) et, comme le précise VAN LAMSWEERDE (2009, p. 34), elles peuvent prendre part à un processus itératif. Les sections qui composent ce chapitre constituent la première itération, ce qui implique que certaines exigences pourront être révisées et d'autres pourront être ajoutées en cours de projet.

Ce chapitre vise donc à présenter l'origine du projet BI induit par le client Dyonysos [13], à établir une étude de faisabilité et, en supposant que les décideurs de l'entreprise valident le projet, à effectuer une première collecte des exigences par introspection. Enfin, ces exigences sont analysées et triées en vue d'effectuer les bons choix au niveau des différentes approches à adopter.

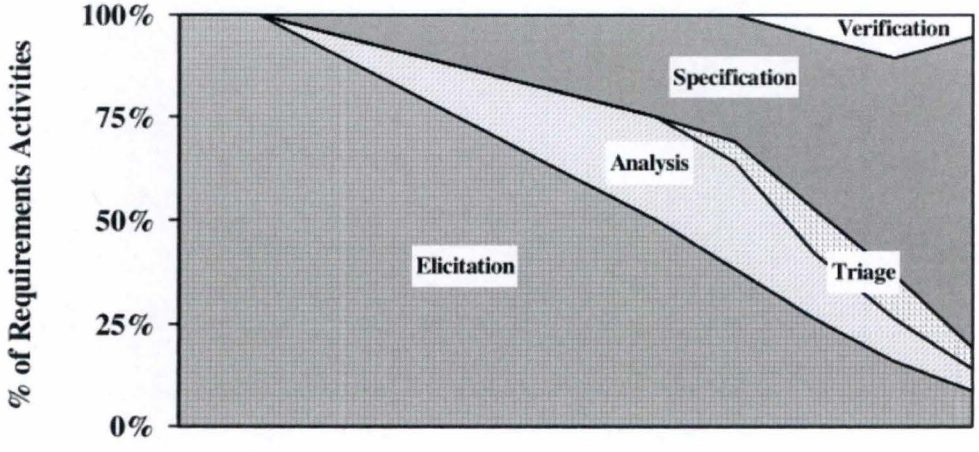

FIGURE 3.1 – Phases de l'ingénierie des exigences

13. Pour des raisons de confidentialité, il s'agit d'un nom d'emprunt.

3.1 Origine du projet

L'idée d'utiliser une approche de BI pour exploiter, entre autres, les données de Mercator a germé suite aux nombreuses demandes de *reporting* émises par un client. Actif dans le marché de la distribution de vins et spiritueux, Dyonysos dispose de plus de 200 rapports, comme l'indique la figure 2.9 (page 18). Comparativement au nombre d'utilisateurs qui s'élève à vingt-et-un, ce nombre de rapports est extrêmement élevé. Par conséquent, cet important ratio renforce les problèmes rencontrés au niveau des groupes *Process*, *People* et *Management* présentés dans la section 2.6.4 (page 17) consacrée à la *Root-Cause Analysis*.

Process : Les utilisateurs de Dyonysos génèrent plusieurs fois par jour les mêmes rapports, et ce afin de surveiller la tendance des certains indicateurs essentiels pour leur activité. En effet, les employés chargés de la communication apprécient analyser en temps réel les retombées des promotions. Par exemple, ils aiment connaître le nombre de commandes qui ont été générées à partir de leur site web e-commerce connecté à Mercator et voir le pourcentage d'utilisation des codes promotionnels. Le fait de devoir produire un rapport et, éventuellement, effectuer quelques manipulations dans un tableur pour prendre connaissance de ces chiffres est rébarbatif.

People : En raison des multiples changements opérés quotidiennement dans Mercator afin de répondre aux évolutions du métier, de nombreux rapports sont utilisés alors qu'ils ne prennent pas correctement en compte de nouveaux aspects de l'activité de Dyonysos. Par exemple, la législation au niveau du recyclage des emballages (Fost+) a récemment évolué et certains rapports ne prennent pas en compte le montant de cette taxe.

Management : Seuls quelques rapports ont correctement été documentés. Par conséquent, les personnes qui cherchent à utiliser un rapport non documenté sont pas en mesure de trouver de l'information au sujet de celui-ci. Le seul moyen est de se référer au(x) collègue(s) qui utilise(nt) ledit rapport, à condition que la personne qui l'utilise soit disponible et qu'elle connaisse de mémoire son fonctionnement.

3.2 Étude de faisabilité

L'étude de faisabilité a pour principale ambition de présenter les objectifs du projet. Elle reprend également quelques arguments qui plaident en faveur des bénéfices potentiels du projet tout délilimitant le périmètre et en énumérant les différentes contraintes. Afin que les parties prenantes puissent dialoguer en utilisant des termes communs et compris, quelques définitions de concepts propres à la *Business Intelligence* sont présentées. Au final, cette brève étude reprend les « grandes lignes » du projet.

3.2.1 Objectifs

Mise en place d'une solution de *Business Intelligence* permettant l'exploitation des données générées par le logiciel ERP Mercator. Les objectifs de ce projet ont été classés sur base des critères de qualité définis par la norme ISO/IEC 9126-1 (2001) explicitée par KOLP (2016, p. 257-259).

Functionality : L'outil répond aux besoins d'analyse de données.

Reliability : Les données fournies par l'outil sont fiables.

Usability : Les utilisateurs sont en mesure d'exploiter rapidement l'outil.

Efficiency : L'outil permet de restituer les informations de façon performante.

Maintainability : L'outil est capable de s'adapter aux évolutions du métier.

Portability : L'outil est accessible depuis différents supports.

Afin de vérifier si ces objectifs sont remplis, DORAN (1981, p. 35-36) propose d'utiliser des objectifs SMART, littéralement **s**pécifiques, **m**esurables, **a**cceptables, **r**éalistes et **t**emporellement définis. En pratique, la *functionality* est mesurée par le nombre de *requirements* qui sont satisfaits ; la *reliability* est vérifiée par des méthodes de testing ; la *usabilily* est estimée par le temps nécessaire à l'utilisateur pour obtenir les informations souhaitées ; l'*efficiency* est calculé par le biais du temps de génération des informations, la *maintainability* est jaugée par le pourcentage d'évolutions qui peuvent être satisfaites ; la *portabilty* dépend du nombre de moyens par lesquels les informations sont accessibles.

3.2.2 Valeur ajoutée

Dans son analyse, RANJAN (2009, p. 64) indique que la *Business Intelligence* présente des avantages en termes de :

- ✓ Élimination de la conjecture au sein de l'entreprise ;
- ✓ Amélioration de la communication entre les différents départements ;
- ✓ Réponse au changement, qu'il soit financier, commercial ou logistique.

De manière générale, la *Business Intelligence* permet d'augmenter le niveau de performance de l'entreprise grâce à une meilleure prise de décision. Plus concrètement, en fonction des choix au niveau de l'implémentation d'un tel outil, l'entreprise pourrait améliorer ses ventes grâce à un meilleur ciblage, augmenter son chiffre d'affaires en structurant sa stratégie de vente, donner une vision claire des revenus futurs, évaluer si les objectifs sont atteints, identifier ses forces et ses faiblesses, etc. De plus, la *Business Intelligence* permet la mise en place de toute une batterie d'indicateurs permettant d'assurer le pilotage de l'entreprise.

3.2.3 Périmètre

Dans un premier temps, seules les données générées par le logiciel ERP Mercator seront prises en compte. Dans un second temps, les données propres aux logiciels de gestion d'entrepôt ou *Warehouse Manangement System* (WMS) pourraient être intégrées. Étant donné que la mise en place et la maintenance de ce logiciel, nommé BIVTrade, est gérée par le prestataire BIVTeam [14], une analyse spécifique et conjointe devra être réalisée, et ce afin d'identifier quelles données mériteraient d'être traitées. Actuellement, les interactions entre Mercator et BIVTrade se limitent à l'échange des préparations des commandes client et des livraisons fournisseur.

3.2.4 Parties prenantes

La figure 3.2 (page 23) basée sur le canevas proposé par THOMPSON (2002) illustre l'*Interest Map* des différentes parties prenantes dans réalisation de ce projet. Tout d'abord, les bénéficiaires et les décideurs de la mise en place du projet sont les dirigeants. De plus, le département financier, même si son pouvoir est moindre, pourrait voir un intérêt dans ce projet. De manière générale, tous les utilisateurs pourraient avoir accès à l'outil, en fonction de leurs droits et besoins. À l'inverse, le gestionnaire de l'infrastructure informatique, les magasiniers et les clients, cités par ordre de pouvoir décroissant, n'ont pas vraiment d'intérêt à faire valoir dans ce projet. Par contre, si ce projet se concrétise et si des données générées par BIVTrade peuvent enrichir les analyses, alors BIVTeam pourrait être plus impliqué durant la réalisation du projet.

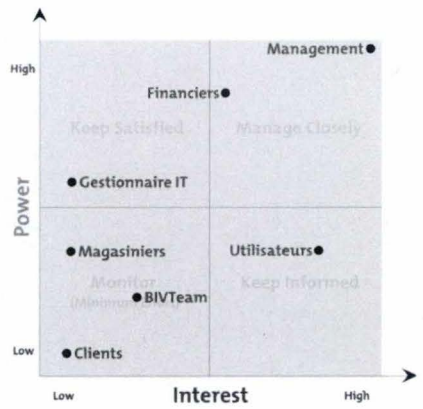

FIGURE 3.2 – Interest Map

14. Le site web officiel de BIVTeam est accessible via le lien ci-après : `http://www.biv-team.com/`

3.2.5 Contraintes

Quatre contraintes principales ont été identifiées. Les deux premières, à savoir le budget qui est alloué au projet et les délais, doivent être définis en collaboration avec Dyonysos. Les deux dernières, respectivement l'allocation des ressources et la technologie, sont du ressort d'Agilux. Au niveau de la technologie, les logiciels actuellement employés par Dyonysos fonctionnent sur base de l'environnement Microsoft .NET avec l'appui de bases de données de type Microsoft SQL Server, le tout étant installé sur des serveurs équipés de Windows Server 2008 et 2012. Dans un souci de compatibilité et d'expérience, les solutions de *Business Intelligence* supportant ces technologies seront privilégiées.

3.2.6 Définitions

Dans le but de disposer d'un vocabulaire commun et d'abréviations communément établies, plusieurs concepts propres au projet sont brièvement définis.

Business Intelligence (BI) : NEGASH (2004, p. 177) indique que la *Business Intelligence combine les données opérationnelles et les outils analytiques afin de présenter une information précise et valorisable pour les décideurs. L'objectif est d'améliorer la disponibilité et la qualité de l'information inhérente aux processus décisionnels. Elle est utilisée à des fins d'analyse des capacités d'une entreprise, de l'état de l'art, des tendances des marchés, des technologies et de l'environnement dans lequel une entreprise évolue ainsi que des initiatives des concurrents et de leurs conséquences.* Il s'agit donc d'une discipline hautement stratégique pour toute entreprise qui se soucie de la gestion du changement. L'exploitation des données collectées dans le *Data Warehouse* de l'entreprise est la base du processus d'analyse. Celui-ci se poursuit par la mise en place des méthodes de transformation et/ou de visualisation des données. Cette étape nécessite l'utilisation de logiciels spécifiques. Enfin, les données retravaillées peuvent alors être interprétées par les décideurs en vue de prendre les décisions stratégiques.

Dashboard & Scorecards : ECKERSON (2006, p. 6) les définit comme étant des systèmes de gestion de la performance, construits dans le cadre d'une structure de *Business Intelligence* et d'intégration des données et qui donnent la possibilité aux entreprises de mesurer, surveiller et de gérer leur activité en se basant sur des indicateurs financiers et non-financiers.

Pyramide DIKW : ROWLEY (2007, p. 164) indique que la pyramide DIKW représente les relations structurelles et/ou fonctionnelles supposées entre les données (**d**ata), les informations (**i**nformation) , les connaissances (**k**nowledge) et la sagesse (**w**isdom). La figure 3.3 (page 25) illustre la hiérarchie de ces quatre notions.

Data Warehouse (DW) : KIMBALL et ROSS (2002, p. 2-4) définissent le *Data Warehouse* comme étant *le lieu de stockage des informations récoltées durant les systèmes opérationnels d'encodage* (inscriptions, commandes, logs, etc.). Ils insistent sur l'importance d'une telle structure, qu'elle soit physique (*data center*) et/ou virtuelle (bases de données dans le *cloud*). Les principaux objectifs sont *(i)* de faciliter l'accès à une information pertinente, *(ii)* de protéger l'information et *(iii)* d'aider à la prise de décision suite à l'exploitation du contenu.

Data Mart (DM) : PEDERSEN (2013, p. 2) définit le *Data Mart* comme étant un sous-ensemble du *Data Warehouse* qui est dédié à un groupe d'utilisateurs spécifique.

Extract-Transform-Load (ETL) : PEDERSEN (2013, p. 2) décrit les trois étapes du processus ETL, respectivement l'extraction qui consiste à récupérer les données dans les systèmes opérationnels, la transformation qui convertit les données afin qu'elles correspondent au format du *Data Warehouse* et le chargement qui permet d'insérer les données transformées dans le *Data Warehouse*.

OnLine Analytical Processing (OLAP) : RANJAN (2009, p. 61) indique que le processus OLAP permet de fournir des vues multidimensionnelles et résumées des données opérationnelles, et ce afin qu'elles puissent être utilisées pour le *reporting*, l'analyse, la modélisation et la planification. Grâce aux actions de *slice & dice*, les utilisateurs sont ainsi capables d'explorer les données.

Indicateur clé de résultat (KRI) : LINDEN (2015) donne une définition sur base de la proposition de PARMENTER (2015) qui indique que les KRI sont le résultat des actions permettant de donner une tendance sans pour autant informer quant à la façon d'améliorer ces résultats.

Indicateur clé de performance (KPI) : LINDEN (2015) reprend à nouveau la définition de PARMENTER (2015) qui décrit un KPI comme étant un ensemble de mesures inhérentes aux aspects organisationnels dont la performance est la plus cruciale pour la pérennité actuelle et future de l'entreprise.

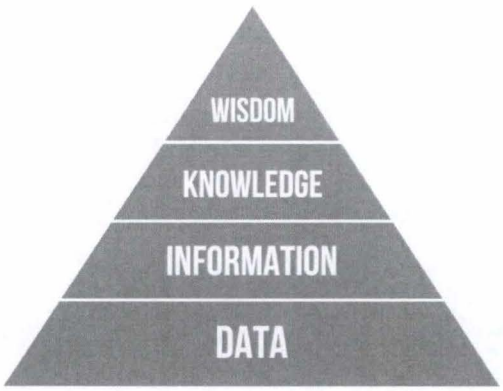

FIGURE 3.3 – Pyramide DIKW

3.3 Collecte des exigences

En raison de la situation pré-projet et en dépit des mises en gardes formulées par NUSEIBEH et EASTERBROOK (2000, p. 38), la méthode utilisée pour collecter les premières exigences est l'instrospection. En effet, BURNAY (2016, p. 64) recommande d'employer cette méthode comme point de départ pour mieux cibler les besoins. Les exigences collectées émanent donc exclusivement d'Agilux qui, grâce à sa longue collaboration avec Dyonysos, est capable d'anticiper bon nombre de besoins. Ces exigences sont structurées en trois parties, à savoir les exigences fonctionnelles, les exigences non-fonctionnelles et les exigences globales.

3.3.1 Exigences fonctionnelles

F.1 : Les utilisateurs occupant des postes de management au sein de Dyonysos doivent avoir accès à l'outil.

F.2 : L'outil doit permettre la génération de graphiques basés sur différents jeux de données, à savoir les ventes, les achats, les articles, les clients, les fournisseurs et les données financières.

F.3 : Les utilisateurs doivent être en mesure de traiter ces données afin de réaliser des analyses.

F.4 : En plus d'apparaître sur des graphiques, ces données doivent pouvoir être exportées aux formats Excel et PDF, et ce sous forme de tableaux et de graphiques.

F.5 : L'outil doit être muni d'un tableau de bord affichant différents KRI et KPI pré-établis.

F.6 : Les graphiques pré-établis doivent être présentés sous formes de séries temporelles ou de diagrammes circulaires.

F.7 : Les graphiques présentés sous formes de séries temporelles doivent être en mesure d'accepter plusieurs séries de données en vue de réaliser des comparaisons.

F.8 : Les données doivent pouvoir être présentées sous forme nominale ou sous forme de taux de croissance.

F.9 : Le taux de croissance doit pouvoir être calculé en *year to date* avec différentes granularités ou pour une période donnée par rapport à cette même période lors d'une année précédente.
Exemple : La croissance du chiffre d'affaires doit pouvoir être présentée de façon mensuelle depuis de le début de l'année en cours, c'est-à-dire que la série temporelle affiche le taux de croissance par rapport au mois précédent.

F10 : Les graphiques pré-établis doivent disposer de filtres permettant à l'utilisateur de choisir l'unité (pièce, litre, poids ou prix en euros), l'horizon, la granularité (jour, semaine, mois ou année), le représentant, le journal de vente ou d'achat, le dépôt, les RFS, les pays et les (catégories) clients ou fournisseurs.
Exemple : L'utilisateur peut choisir d'afficher une série temporelle sur un horizon de deux ans, groupée par mois, en euros et pour tous les clients.

F.11 : Les filtres doivent accepter une, plusieurs ou toutes les valeurs pour une même spécification.
Exemple : L'utilisateur doit pouvoir sélectionner un, plusieurs ou tous les journaux de vente à prendre en compte dans la présentation des données.

F.12 : L'utilisateur doit être en mesure d'ajouter des filtres personnalisés sur base des attributs des données disponibles.
Exemple : Pour un client, l'attribut « Ville » n'est pas disponible par défaut mais l'utilisateur doit pouvoir l'ajouter et appliquer le filtre. Ainsi, il sera en mesure d'analyser les clients dont la ville est Namur.

F.13 : Les unités à la pièce doivent pouvoir être converties en bouteilles, caisses ou palettes.

F.14 : Les composants du prix doivent pouvoir être sélectionnés, à savoir taxes comprises (TVA, accises, Fost+ et éco-boni) ou non, avec ou sans remises et en prenant en compte les tarifs et promotions.

F.15 : L'utilisateur doit être en mesure d'effectuer des zooms sur les éléments graphiques.
Exemple : Cliquer sur un client dans un diagramme circulaire afin d'appliquer un filtre sur ce client ou encore effectuer un cliquer-déplacer sur une série temporelle pour afficher un horizon plus restreint.

F.16 : L'utilisateur doit être en mesure de sauvegarder ses préférences afin de disposer d'un tableau de bord et de graphiques personnalisés.

F.17 : Les préférences des utilisateurs doivent être restituées lors de chaque connexion.

F.18 : Les préférences doivent pouvoir être rétablies afin de revenir à l'affichage par défaut.

F.19 : Tous les graphiques créés doivent être historisés afin que l'utilisateur puisse réutiliser les graphiques qui ont été construits par le passé.

F.20 : Les utilisateurs doivent être en mesure de partager avec d'autres utilisateurs les graphiques qu'ils ont conçus.

3.3.2 Exigences non-fonctionnelles

N.1 : La construction des graphiques doit être simple, intuitive et visuelle (*What You See Is What You Get*).

N.2 : Les utilisateurs ne doivent pas utiliser de code pour construire les graphiques.

N.3 : Les intégrateurs doivent pouvoir effectuer du paramétrage.
Exemple : Ajouter un attribut en tant que filtre par défaut.

N.4 : Les éléments visuels doivent utiliser le maximum d'espace visible à l'écran.

N.5 : Les éléments visuels doivent s'adapter aux différentes résolutions d'écran.

N.6 : Les informations doivent pouvoir être consultées depuis différents supports (ordinateur, tablette ou smartphone). Le design doit donc être *responsive*.

N.7 : Les informations doivent être accessibles depuis n'importe quel emplacement.

N.8 : L'outil doit faire preuve d'ergonomie. La localisation des différents composants doivent être placés de telle sorte à ce qu'ils soient accessibles et doivent être présentés dans un ordre logique.
Exemple : Dans la liste déroulante, la granularité doit être présentée dans l'ordre jour, semaine, mois et année.

N.9 : L'outil doit être doté d'effets visuels qui rendent son utilisation agréable.
Exemple : Lors du chargement des données, une barre de progression doit être affichée afin de donner une estimation du temps de chargement.

N.10 : Le chargement des données et des composants visuels ne doit pas être trop long.

N.11 : L'interaction avec les éléments visuels doit être réactive lorsque des actions jugées basiques sont réalisées. Toutes les actions ne doivent pas être accompagnées d'un temps de chargement.
Exemple : Le fait de zoomer sur une zone d'une série temporelle doit être instantané car les données sont censées être déjà chargées.

N.12 : La disponibilité des informations doit être maximale.

N.13 : L'accès à l'outil doit être sécurisé. Un utilisateur ne disposant pas des droits d'accès ou une personne malveillante ne doivent pas y avoir accès.

N.14 : Les données stockées au sein de l'outil doivent être sécurisées, et ce en raison de leur caractère confidentiel.

3.3.3 Exigences globales

G.1 : L'outil doit pouvoir exploiter des données stockées dans une base de données Microsoft SQL Server 2012.

G.2 : L'outil doit, de préférence, utiliser les technologies fournies par Microsoft, à savoir Microsoft .NET, C#, Microsoft SQL Server.

G.3 : L'outil doit utiliser des technologies modernes, adaptées, mises à jour et durables.
Exemple : Des langages de programmation désuets tels que le Microsoft Visual Basic, WinDev, Cobol, etc. ne sont pas tolérés.

G.4 : L'outil doit être opérationnel en 2017.

G.5 : Les utilisateurs ayant accès à l'outil devront être formés à son utilisation.

G.6 : Après mise en production, le support devra être assuré.

G.7 : La mise en place de l'outil ne doit pas dépasser le budget alloué au projet.

3.3.4 Non-exhaustivité des exigences

En raison du fait qu'il s'agit d'une pré-collecte réalisée unilatéralement par instrospection, d'autres méthodes de découverte des exigences seront mises en œuvre afin de comprendre parfaitement les besoins du Dyonysos. En effet, après acceptation du projet, des entretiens, des réunions et des groupes de travail seront organisés dans le but de valider et de compléter la pré-collecte. Cette approche est nécessaire car, comme le mentionnent NUSEIBEH et EASTERBROOK (2000, p. 38), malgré les croyances des experts, l'introspection ne permet pas de couvrir tous les besoins. Par conséquent, de nouvelles itérations au niveau du processus d'élicitation devront être réalisées.

3.4 Analyse des exigences

Dans le but de préciser certains points jugés cruciaux cités dans la section 3.3 (page 25), une analyse plus poussée est menée. Celle-ci concerne les KRI et KPI, l'architecture choisie pour le *Data Warehouse* et l'implémentation OLAP.

3.4.1 Définition des KRI et KPI

L'analyse concerne principalement l'exigence F.5 qui stipule que « L'outil doit être muni d'un tableau de bord affichant différents KRI et KPI pré-établis ». Sur ce point, il est évident que les KRI et les KPI doivent être définis. Tout d'abord, au niveau du nombre, la plupart des auteurs s'accordent à dire qu'il n'est pas nécessaire de disposer d'une kyrielle d'indicateurs. PARMENTER (2015) recommande la règle « 10-80-10 », ce qui signifie que 10 KRI, 80 indicateurs de résultat/performance et 10 KPI sont suffisants. De plus, KAPLAN et NORTON (2008) suggèrent de ne pas utiliser plus de 20 KPI.

D'une part, niveau des KRI, Agilux suggère d'afficher sur le tableau de bord les indicateurs mentionnés ci-après :

(i). Le chiffre d'affaires présenté sous forme d'une série temporelle ;

(ii). Le nombre de documents de vente émis présentés sous forme d'une série temporelle ;

(iii). Le nombre de documents d'achats émis présentés sous forme d'une série temporelle ;

(iv). La répartition des ventes par représentant présentée sous forme d'un diagramme circulaire ;

(v). La répartition des ventes par région présentée sous forme d'un diagramme circulaire ;

(vi). La répartition des ventes par catégorie client présentée sous forme d'un diagramme circulaire ;

(vii). Le top 10 des clients ;

(viii). Le top 10 des fournisseurs.

D'autre part, au niveau des KPI, Agilux propose d'intégrer au *dashboard* les indicateurs décrits ci-dessous :

(i). Le taux de croissance du chiffre d'affaires présenté sous forme d'une série temporelle ;

(ii). Le taux de conversion des devis en commandes présenté sous forme d'une série temporelle ;

(iii). Le taux de conversion des propects en clients présenté sous forme d'une série temporelle ;

(iv). Le taux de croissance du nombre de propects en clients présenté sous forme d'une série temporelle ;

(v). La répartition des ventes d'articles en promotion ou non sous forme d'un diagramme circulaire.

Ces propositions pourront être adaptées en fonction des besoins et complétées par des indicateurs de résultat et de performance. De plus, comme le précise ECKERSON (2009, p. 32), le processus de création des différents indicateurs n'est pas régi par une science exacte. En effet, même la meilleure des analyses ne pourrait pas accoucher d'indicateurs parfaits.

3.4.2 Architecture du *Data Warehouse*

Dans le domaine de la *Business Intelligence*, deux grandes théories d'architecture du *Data Warehouse* tiennent le haut du pavé. D'une part, INMON (2005) propose une approche *top-down* dans laquelle le *Data Warehouse* est composé de plusieurs *Data Marts* représentant chacun un département de l'entreprise. D'autre part, KIMBALL et ROSS (2002) préconisent plutôt une approche *bottom-up* basée sur la modélisation dimensionnelle. HAZIYEV (2013, p. 20) schématise les différences entre les deux théories sur la figure 3.4 (page 30).

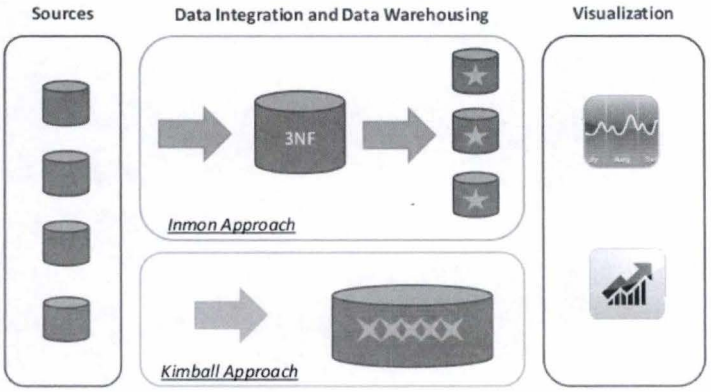

FIGURE 3.4 – Différences entre INMON et KIMBALL

Bien entendu, chaque approche possède ses propres avantages mais aussi quelques inconvénients. LINDEN (2015) les résume dans le tableau 3.1 (page 30).

	Avantages	**Inconvénients**
INMON	+ Intégration complète de l'entreprise	– Assez complexe – Mise en place lente – Peu orienté utilisateur final
KIMBALL	+ Simple + Mise en place rapide + Orienté utilisateur final	– Peu évolutif

TABLEAU 3.1 – Avantages et inconvénients d'INMON et KIMBALL

En raison de la taille de l'entreprise Dyonysos qui ne compte que quatorze utilisateurs, ce qui la situe dans la moyenne haute des clients actuels d'Agilux, l'approche « par département » proposée par INMON (2005) semble être trop complexe à mettre en place. La philosophie popularisée par KIMBALL et ROSS (2002) est donc privilégiée.

3.4.3 Implémentation OLAP

Dans l'optique de faciliter la mise en place de l'outil, la méthode *Relational* OLAP (ROLAP) est préférée car la principale source de données est la base de données relationnelle de Mercator présentée à la section 2.4 (page 11). Dans cette configuration, deux schémas peuvent être mis en place, à savoir le schéma en étoile et le schéma flocon de neige. Toujours dans un souci de facilité, le schéma en étoile est privilégié. Comme le suggère la schéma entité-association présenté à la section 2.3 (page 10), les tables de faits pourraient être les ventes et les achats sur lesquelles viendraient se greffer les dimensions de tiers, d'articles, de magasins, de date, de promotions, etc. De nouveau, le choix des dimensions devra être défini lors des prochaines itérations de collecte des exigences.

3.5 Triage des exigences

Dans le but de trier et de prioritiser les exigences citées à la section 3.3 (page 25), la méthode MoSCoW est utilisée. BURNAY (2016, p. 176) indique que cette technique a pour objectif de classer les exigences selon quatre grandes catégories, à savoir ce qui est vital (*Must Have*), ce qui est essentiel (*Should Have*), ce qui est intéressant (*Could Have*) et ce qui n'est pas pris en compte pour le moment (*Won't Have*). Dans un souci de clarté, le tri est présenté dans trois tableaux qui distinguent les exigences fonctionnelles, non-fonctionnelles et générales, respectivement 3.2 (page 31), respectivement 3.3 (page 32) et respectivement 3.4 (page 32). Ces tableaux affichent chacun le numéro de l'exigence et les quatre colonnes de classification.

#	*M*	*S*	*C*	*W*
F.1	√			
F.2	√			
F.3	√			
F.4		√		
F.5	√			
F.6	√			
F.7		√		
F.8		√		
F.9		√		
F.10	√			
F.11	√			
F.12		√		
F.13			√	
F.14			√	
F.15		√		
F.16			√	
F.17			√	
F.18			√	
F.19			√	
F.20			√	

TABLEAU 3.2 – Triage des exigences fonctionnelles selon MoSCoW

#	*M*	*S*	*C*	*W*
N.1	√			
N.2	√			
N.3	√			
N.4		√		
N.5		√		
N.6		√		
N.7		√		
N.8		√		
N.9			√	
N.10	√			
N.11	√			
N.12	√			
N.13	√			
N.14	√			

TABLEAU 3.3 – Triage des exigences non-fonctionnelles selon MoSCoW

#	*M*	*S*	*C*	*W*
G.1	√			
G.2		√		
G.3	√			
G.4	√			
G.5	√			
G.6	√			
G.7	√			

TABLEAU 3.4 – Triage des exigences générales selon MoSCoW

En conclusion, les exigences fonctionnelles qui constituent les fonctionnalités de base de l'outil sont jugées vitales. Elles représentent un tiers des exigences fonctionnelles collectées. Les deux autres tiers regroupent les fonctionnalités plus avancées. Même constat pour les exigences non-fonctionnelles, celles concernant la facilité d'accès, la réactivité et la disponibilité de l'outil sont prioritaires tandis que celles inhérentes à l'apparence de l'outil sont relayées au second plan. Par contre, toutes les exigences générales, à l'exception de la technologie utilisée, sont primordiales. En raison de la méthode utilisée, à savoir l'introspection, aucune exigence n'a été classée dans la case *Won't Have*. Sans doute que les futures collectes d'exigences mèneront à classer certaines demandes dans cette case, et ce en raison de contraintes budgétaires, temporelles, humaines et/ou techniques.

3.6 Spécification des exigences

En vue de documenter les exigences, un schéma *Use Case* et un dessin d'écran ont été réalisés. Ceux-ci ont pour vocation de respectivement présenter les différents cas d'utilisation de l'outil et de donner une idée quant à la disposition visuelle des différents composants, comme les menus, les boutons, les filtres, les graphiques, les widgets, les tableaux, etc.

3.6.1 Schéma Use Case

Dans un premier temps, la figure 3.5 (page 33) illustre le schéma *Use Case*. Dans un second temps, ce schéma est traduit sous forme d'une expression narrative.

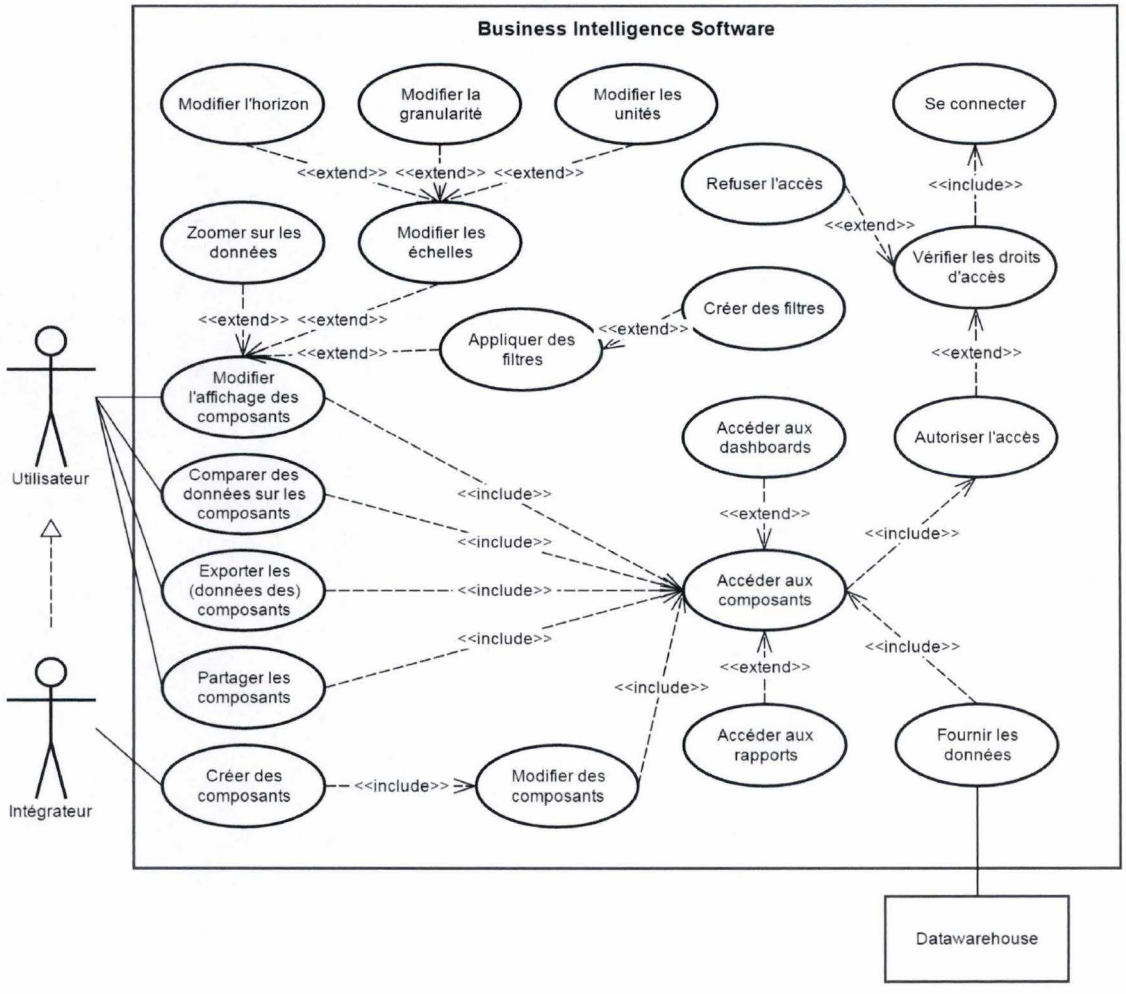

FIGURE 3.5 – Schéma Use Case

Se connecter

Pré-condition(s)	L'outil dispose d'une page de connexion
Post-condition(s)	Le système est en mesure de récupérer les données saisies par l'utilisateur
Acteur	**Système**
1. L'utilisateur accède à la page de connexion de l'outil	
2. L'utilisateur saisit son identifiant et son mot de passe	
	3. Le système récupère la combinaison identifiant

Vérifier les droits d'accès

Pré-condition(s)	L'utilisateur a saisi son identifiant et son mot de passe
Post-condition(s)	L'utilisateur accède ou non à l'outil
Acteur	**Système**
	1. Le système vérifie que la combinaison identifiant/mot de passe correspond à un compte actif

Refuser l'accès

Pré-condition(s)	L'utilisateur a saisi son identifiant et son mot de passe
Post-condition(s)	La combinaison identifiant/mot de passe ne correspond à aucun compte actif
Acteur	**Système**
	1. Le système indique à l'utilisateur que la combinaison identifiant/mot de passe n'est pas valide
2. L'utilisateur est redirigé vers la page de connexion	

Autoriser l'accès

Pré-condition(s)	L'utilisateur a saisi son identifiant et son mot de passe
Post-condition(s)	La combinaison identifiant/mot de passe correspond à un compte actif
Acteur	**Système**
	1. Le système octroie l'accès à l'utilisateur
	2. Le système vérifie à quelles fonctionnalités l'utilisateur a accès
3. L'utilisateur est redirigé vers la page d'accueil de l'outil	

Accéder aux composants (tableaux de bord ou rapports)

Pré-condition(s)	L'utilisateur a le droit d'accéder au composant
Post-condition(s)	L'utilisateur accède au composant
Acteur	**Système**
1. L'utilisateur clique sur le composant qu'il souhaite afficher	
	2. Le système redirige l'utilisateur vers le composant
3. L'utilisateur accède au composant	

Modifier des composants

Pré-condition(s)	L'utilisateur a accédé au composant
Post-condition(s)	Le composant est modifié
Acteur	**Système**
1. L'utilisateur effectue des actions qui modifient le fonctionnement du composant	
	2. Le système prend en compte les actions de l'utilisateur pour modifier l'affichage en conséquence

Créer des composants

Pré-condition(s)	L'utilisateur a accédé au module de création de composant
Post-condition(s)	Le composant est créé
Acteur	**Système**
1. L'utilisateur effectue des actions qui permettent de créer le composant	
	2. Le système prend en compte les actions de l'utilisateur pour créer le composant

Partager les composants

Pré-condition(s)	L'utilisateur a accédé au composant
Post-condition(s)	Le composant est partagé avec un ou plusieurs autres utilisateurs
Acteur	**Système**
1. L'utilisateur clique sur le bouton de partage du composant 2. L'utilisateur sélectionne la méthode de partage (envoi par e-mail, accès via l'outil, etc.)	
	3. En fonction de la méthode de partage sélectionnée, le système notifie le ou les utilisateurs avec lesquels le composant est partagé 4. Le système octroie l'accès aux composants partagés

Exporter les (données des) composants

Pré-condition(s)	L'utilisateur a accédé au composant
Post-condition(s)	Le composant est exporté sous forme de fichier
Acteur	**Système**
1. L'utilisateur clique sur le bouton d'export du composant 2. L'utilisateur sélectionne le format de l'export (Excel ou PDF) 2. L'utilisateur sélectionne le contenu du fichier à exporter (graphiques et/ou données)	
	4. Le système génère le fichier sur base des choix effectués par l'utilisateur 5. Le système propose d'enregistrer le fichier
6. L'utilisateur sélectionne un emplacement dans lequel il sauvegarde le fichier généré	

Comparer des données sur les composants

Pré-condition(s)	L'utilisateur a accédé au composant et le composant autorise la comparaison de données
Post-condition(s)	Le composant affiche une ou plusieurs séries de données supplémentaires
Acteur	**Système**
1. L'utilisateur clique sur le bouton d'ajout de séries supplémentaires	
	2. Le système propose une liste de séries de données compatibles
3. L'utilisateur sélectionne la ou les séries de données souhaitées	
	4. Le système ajoute la ou les séries sélectionnées sur le composant

Modifier l'affichage des composants

Pré-condition(s)	L'utilisateur a accédé au composant
Post-condition(s)	L'affichage du composant est modifié
Acteur	**Système**
1. L'utilisateur applique une modification sur l'affichage du composant	
	2. Le système adapte l'affichage des données du composant en fonction du choix de l'utilisateur

Zoomer sur les données

Pré-condition(s)	L'utilisateur a appliqué une modification sur l'affichage du composant et le composant permet le zoom
Post-condition(s)	Le zoom est appliqué sur les données
Acteur	**Système**
1. L'utilisateur applique le zoom sur les données, soit par clic-glissé (séries temporelles), soit par clic sur une zone (diagramme circulaire, bâtonnets, etc.)	
	2. Le système adapte l'affichage des données des composants de telle sorte à ce que la zone sélectionnée soit prise en compte dans le zoom

Modifier les échelles (horizon, granularité et unités)

Pré-condition(s)	L'utilisateur a appliqué une modification sur l'affichage du composant et le composant permet une modification de l'échelle choisie
Post-condition(s)	L'échelle des données est modifiée
Acteur	**Système**
1. L'utilisateur modifie l'échelle des données	
	2. Le système adapte l'échelle des données en fonction du choix de l'utilisateur

Appliquer des filtres

Pré-condition(s)	L'utilisateur a appliqué une modification sur l'affichage du composant et le composant permet l'utilisation de filtres
Post-condition(s)	Les données du composant sont filtrées
Acteur	**Système**
1. L'utilisateur applique un filtre sur les données	2. Le système adapte les données de telle sorte à ne prendre en compte que les valeurs sélectionnées dans le filtre

Créer des filtres

Pré-condition(s)	Le composant permet la création de filtres
Post-condition(s)	Un nouveau filtre est créé
Acteur	**Système**
1. L'utilisateur clique sur le bouton de création de filtre 2. L'utilisateur sélectionne une possibilité de filtrage	3. Le système crée le filtre sur base du choix effectué par l'utilisateur

Fournir les données

Pré-condition(s)	La connexion avec le *Data Warehouse* est active
Post-condition(s)	Les données du *Data Warehouse* sont transmises dans l'outil
Acteur	**Système**
	1. Le système sollicite le *Data Warehouse* pour obtenir les données à afficher
2. Le *Data Warehouse* renvoie les données sur base des conditions demandées par le système	

3.6.2 Dessin d'écran

Dans le but de donner un premier aperçu de l'interface aux utilisateurs potentiels, un dessin d'écran d'un tableau de bord a été réalisé. La figure 3.6 (page 38) présente donc un premier jet des différents composants dont les cas d'utilisation sont décrits à la section 3.6.1 (page 33). Il s'agit évidemment d'une ébauche qui pourra être révisée en fonction des besoins des utilisateurs et des possibilités de l'outil. Par exemple, il se peut que les tableaux de bord et rapports soient disposés dans un menu, que certaines fonctionnalités comme l'ajout de filtres ou la comparaison de données ne soient pas réalisables, que le menu soit présenté différemment, que les boutons soient disposés différemment, etc.

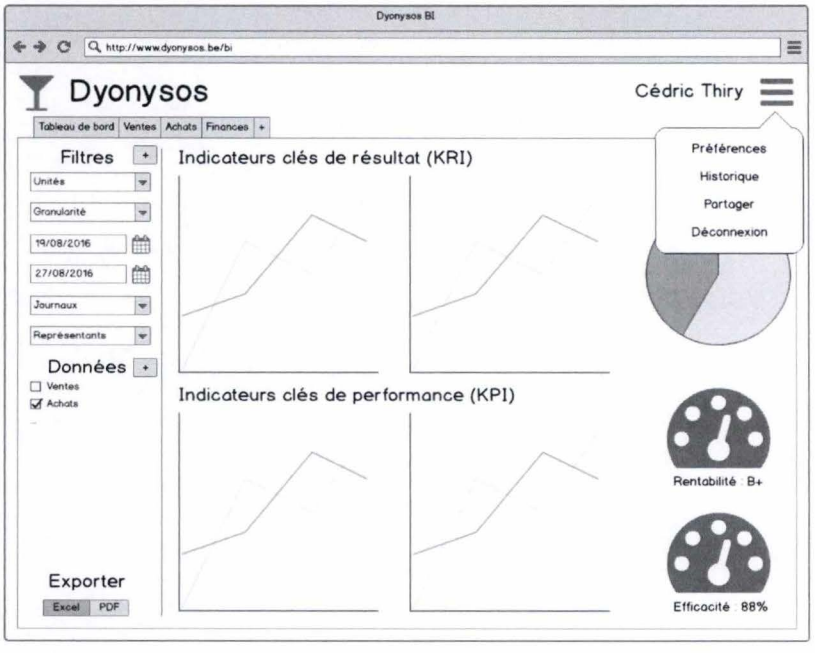

FIGURE 3.6 – Dessin d'écran du *dashboard*

CHAPITRE 4 - La solution : Microsoft Power BI

En vue de réaliser le projet présenté tout au long du chapitre 3 (page 20), certaines solutions de *Business Intelligence* disponibles sur le marché vont désormais être étudiées et testées. Pour des raisons de proximité, deux acteurs locaux ont tout d'abord été contactés. Ensuite, d'autres solutions proposées par des géants du secteur sont présentées, plus particulièrement le logiciel Power BI développé par Microsoft. Enfin, ce chapitre est conclu avec une analyse des impacts inhérents à l'adoption d'une solution de *Business Intelligence* dans le cadre des activités d'Agilux.

4.1 Étude de marché

L'étude de marché présente brièvement trois acteurs de la *Business Intelligence* avec qui Agilux est entré en contact. Dans un souci d'exhaustivité, le marché dans son ensemble est brièvement décrit.

4.1.1 EMAsphère

EMAsphère [15] est une *startup* qui propose une solution de tableau de bord de gestion en ligne pour les PME. Afin d'obtenir de plus amples informations, une réunion s'est tenue le 27 juillet 2016 dans les locaux de l'entreprise, à Marche-en-Famenne. Monsieur Tesla [16], responsable commercial, a présenté la solution dont les caractéristiques sont les suivantes :

- ✓ Interface graphique simple, intuitive et visuelle ;
- ✓ Accessible partout, tout le temps et sur tous les périphériques ;
- ✓ Les données comptables compréhensibles et au service de la gestion ;
- ✓ Communication optimisée pour l'équipe dirigeante et les collaborateurs ;
- ✓ Un outil intégré à l'environnement logiciel de l'entreprise et opérationnel en quelques jours ;
- ✓ Un outil orienté décisions/actions avec un système d'alertes et de notifications pour une gestion pro-active ;
- ✓ Aucune compétence, aucun coût d'infrastructure ou de maintenance informatique à un prix adapté à la PME.

Suite à cette réunion, Agilux a listé les avantages et inconvénients de la solution. Cette analyse est matérialisée par le tableau 4.1 (page 40).

15. Le site web officiel d'EMAshpère est accessible via le lien ci-après : `www.emasphere.com`
16. Pour des raisons de confidentialité, il s'agit d'un nom d'emprunt.

Avantages	Inconvénients
+ Présentation des données + Clarté de l'interface + Réactivité du logiciel + Accessible en ligne + *Responsive* + Proximité de l'éditeur + Paramétrage possible via l'interface	– Limité aux données comptables – Limité aux logiciels de comptabilité standards (Bob50, Winbooks et Popsy) – Aucun contrôle sur le processus ETL – Nécessité de développer un connecteur standard pour Mercator) – Tarification compliquée (basée sur le chiffre d'affaires du client) – Peu de possibilités de personnalisation – Technologie non-maîtrisée au sein d'Agilux

TABLEAU 4.1 – Avantages et inconvénients d'EMAsphère

En conclusion, en dépit des qualités et du potentiel de la solution, Agilux ne compte pas poursuivre les démarches car deux inconvénients bloquants ont été identifiés. Le premier est la limitation aux données comptables. Or, seule une faible proportion des clients d'Agilux utilise le module de comptabilité de Mercator. De plus, la plupart des KRI et KPI mentionnés au chapitre 3.4 (page 28) se basent sur des données de gestion commerciale. Le second est l'impossibilité d'adapter le processus ETL. Cette restriction signifie que, en cas de développement d'un connecteur standard pour la gestion commerciale de Mercator, les champs personnalisés propres à chaque client ne pourraient pas être pris en compte dans le tableau de gestion.

4.1.2 Analyza

Analyza[17] est une solution de *Business Intelligence* éditée par PIT Business[18]. Monsieur Veber[19], le fondateur de la *startup* composée de six personnes, a présenté son produit le 27 juillet 2016 dans les locaux d'Agilux. Analyza a été décrit comme étant une plateforme :

- ✓ Rapide, Analyza affiche les indicateurs demandés en quelques secondes à peine ;
- ✓ Simple, pas besoin de connaissances informatiques pour utiliser Analyza ;
- ✓ Jolie, avec une visualisation des plus efficaces ;
- ✓ Flexible, adaptable en fonction des besoins ;
- ✓ Stable, nécessite peu de maintenance ;
- ✓ Performante, traitement de centaines de milliers de lignes de données sans impacter la performance ;
- ✓ Sécurisée, par le biais de protocoles HTTPS et SSL ;
- ✓ Transparente, traduction automatique des requêtes utilisateur ;
- ✓ Abordable, positionnement tarifaire en-dessous des acteurs principaux du marché.

Suite à cette présentation, Agilux a pesé le pour et le contre. Cette analyse est résumée dans le tableau 4.2 (page 41).

17. Le site web officiel d'Analyza est accessible via le lien ci-après : `http://www.analyza-dashboard.com/`
18. Le site web officiel de PIT Business est accessible via le lien ci-après : `http://www.pit-business.com/`
19. Pour des raisons de confidentialité, il s'agit d'un nom d'emprunt.

Avantages	Inconvénients
+ Présentation des données	
+ Clarté de l'interface	
+ Réactivité du logiciel	
+ Accessible en ligne	– Maintenu mais plus développé
+ *Responsive*	– Édité par une petite équipe
+ Proximité de l'éditeur	– Peu de contrôle sur le processus ETL
+ Flexibilité en termes de personnalisation et de paramétrage	
+ Tarification flexible	

TABLEAU 4.2 – Avantages et inconvénients d'Analyza

Au final, Monsieur Veber a eu l'honnêteté de préciser qu'Analyza n'était plus mis en avant car, faute de ressources humaines et financières, la solution n'est plus développée. Par conséquent, PIT Business se positionne plutôt comme étant un prestataire de services spécialisé en *Business Intelligence*, et ce quel que soit l'outil utilisé. C'est pour cette raison qu'Agilux ne prend pas le risque d'investir dans la solution Analyza. Par contre, dans le cadre de projets BI conséquents, un partenariat avec PIT Business reste envisageable.

4.1.3 Telerik

Agilux n'exclut pas la possibilité la conception sur-mesure d'outils de *Business Intelligence*. Pour ce faire, Agilux a acquis les droits d'utilisation des composants visuels et des outils de développement fournis par Telerik[20]. En effet, Telerik propose des outils dédiés à la conception de tableau de bord et de rapports. Après analyse, Agilux a mis évidence les aspects positifs et négatifs de cette approche dans le tableau 4.3 (page 41) :

Avantages	Inconvénients
	– Prix
+ Possibilités de personnalisation extrêmement poussées	– Nécessité de combiner les composants Reporting et la plateforme Report Server
+ Accessible en ligne	– Nécessité d'acquérir des compétences pour utiliser les outils à disposition
+ Richesse des composants visuels	– Difficile à maintenir

TABLEAU 4.3 – Avantages et inconvénients de Telerik

En définitive, Telerik se positionne comme un outil réservé pour des projets dont les besoins ne rentrent pas dans le cadre des standards proposés par les grands acteurs du marché. En tant qu'utilisateur de la technologie proposée par Telerik, Agilux n'exclut pas l'utilisation des composants *Reporting*. Mais, à ce stade, Agilux préfère opter pour un outil nécessitant moins d'investissement financier et intellectuel.

20. Le site web officiel de Telerik est accessible via le lien ci-après : `http://www.telerik.com/`

4.1.4 Autres solutions

Le marché regorge d'outils de *Business Intelligence* plus ou moins élaborés et/ou spécialisés. Sur ce point, PARENTEAU et al. (2016), dans leur « Magic Quadrant » illustré sur la figure 4.1 (page 42), recensent les différents acteurs. Cette méthodologie se base sur deux axes, *completeness of vision* et *ability to execute*, qui définissent quatre zones dans lesquelles sont placés sont les acteurs, à savoir les acteurs de niche, les visionnaires, les challengers et les leaders. PARENTEAU et al. (2016) identifient trois leaders qui sont Microsoft, Qlik et Tableau. Parmi les principaux visionnaires, l'étude cite, entre autres, SAS, MicroStrategy et SAP.

FIGURE 4.1 – Gartner BI Magic Quadrant 2016

Au vu des affinités avec les technologies diffusées par Microsoft et présentées à la section 2.5 (page 14), Agilux s'oriente instinctivement vers les solutions proposées par la firme de Redmond. Si ce choix s'avère être un échec, Agilux n'exclut pas d'explorer les pistes suggérées par Qlik, Tableau, MicroStrategy, etc.

4.2 Focus sur Microsoft Power BI

Power BI[21] est une solution de *Business Intelligence* développée par Microsoft. Cette solution est, comme l'explique RODMAN (2015), le fruit des élucubrations de Microsoft dans le domaine de la BI qui a successivement publié les solutions Gemini, Power Pivot et Power Query. Power BI est donc l'héritier qui a repositionné Microsoft dans la course, comme le prouve l'évolution de positionnement entre les « Magic Quadrant » de 2015, figure 4.2 (page 43), et 2016, figure 4.1 (page 42).

FIGURE 4.2 – Gartner BI Magic Quadrant 2015

Premièrement, en termes de fonctionnalités, Power BI propose de :

- ✓ Réunir toutes les données pertinentes sur un même tableau de bord ;
- ✓ Créer des rapports interactifs ;
- ✓ Globaliser les données pour fournir de la cohérence au sein des entreprises ;
- ✓ Partager facilement les analyses réalisées.

21. Le site officiel de Microsoft Power BI est accessible via le lien ci-après : `https://powerbi.microsoft.com/fr-fr/`

Deuxièmement, au niveau de l'interface illustrée sur la figure 4.3 (page 44), l'outil présente les éléments suivants :

(i). Le menu principal qui affiche les tableaux de bord, les rapports et les jeux de données ;

(ii). La barre de menu qui permet d'interagir avec le composant en cours ;

(iii). Les composants visuels qu'il est possible de déplacer et de positionner sur un tableau de bord choisi ;

(iv). Les différents onglets du composant en cours ;

(v). Un menu permettant d'agir sur les visualisations ;

(vi). Un menu permettant d'agir sur les champs pris en compte ;

(vii). Un bouton permettant d'accéder au menu utilisateur ;

(viii). Un bouton permettant d'obtenir des données.

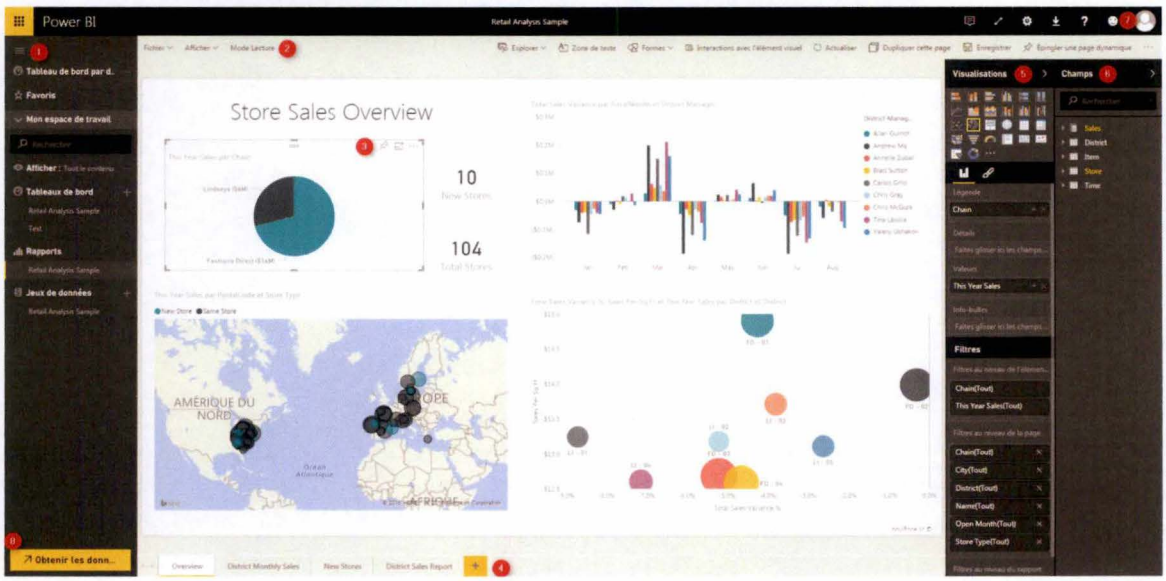

FIGURE 4.3 – Interface de Microsoft Power BI

Troisièmement, d'un point de vue technique, Power BI permet d'acquérir des données émanant de fichiers Excel ou de bases de données de type Microsoft SQL Server ou Azure. Ce dernier point, implique que le *Data Warehouse* doit nécessairement être construit dans ce format. Ensuite, sur base de ces données, l'utilitaire Power BI Desktop permet de créer les rapports et tableaux de bord pour ensuite les publier. Enfin, les utilisateurs peuvent accéder à ces composants via l'interface web, via l'application mobile (Power BI Mobile) ou via une application tierce qui intègre lesdits composants (Power BI Embedded). Ce processus est résumé sur la figure 4.4 (page 45).

Après plusieurs tests, Agilux a dressé la liste des avantages et inconvénients de la solution de Microsoft. Ceux-ci sont divulgués dans le tableau 4.4 (page 45).

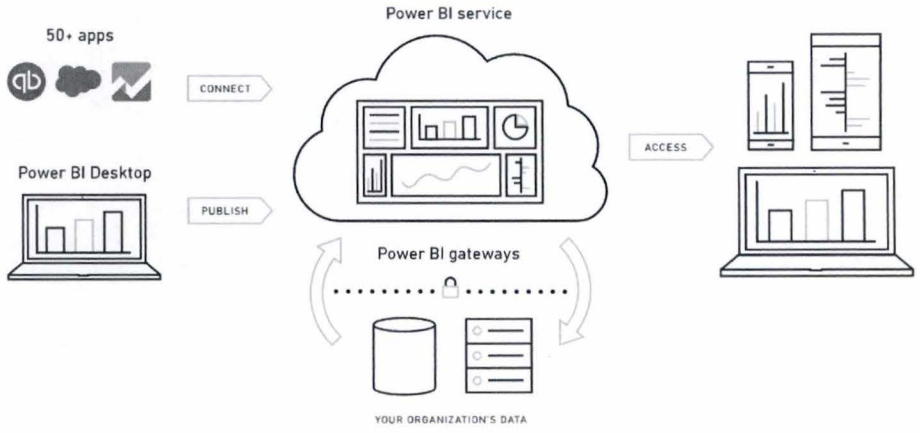

FIGURE 4.4 – Fonctionnement technique de Power BI

Avantages	Inconvénients
+ Présentation des données	
+ Clarté de l'interface	
+ Réactivité du logiciel	
+ Accessible en ligne	
+ Possibilité de partager les analyses	
+ Disponibilité d'une application mobile	– Peu de visibilité sur les conditions de partenariat
+ Simplicité du processus de création des composants	– Version gratuite extrêmement bridée
+ Nombreuses possibilités de paramétrage	– Interface web peu *responsive*
+ Contrôle total sur le processus ETL	– Manque de maturité de Microsoft sur le marché de la BI
+ Compatible avec Microsoft SQL Server, technologie maîtrisée au sein d'Agilux	
+ Fiabilité de Microsoft	
+ Prix abordable	

TABLEAU 4.4 – Avantages et inconvénients de Microsoft Power BI

En conclusion, Power BI est l'outil qui a été choisi par Agilux pour effectuer les premiers pas dans le monde de la *Business Intelligence*. En effet, cette solution ne souffre, à priori, d'aucun inconvénient majeur qui pourrait entraver la réussite du projet. De plus, son utilisation semble être simple, tant pour l'utilisateur que pour l'intégrateur. Les technologies de Microsoft étant bien connues par Agilux, l'investissement temporel et humain nécessaire à la maîtrise de l'outil devrait être relativement faible. Plus spécifiquement, la solution répond à la plupart des exigences non-fonctionnelles et générales présentées aux sections 3.3.2 (page 27) et 3.3.3 (page 28). Par exemple, elle est compatible avec des technologies modernes telles SQL Server, elle est extrêmement réactive, les effets visuels sont agréables, elle est disponible sur différents supports, etc.

4.3 Perspectives

La réalisation d'un premier projet dans le domaine de la *Business Intelligence* engendre deux conséquences au sein d'Agilux. D'une part, un ou plusieurs membres de l'équipe décrite à la section 1.1 (page 4) devront prendre part à des formations afin de maîtriser les concepts de la *Business Intelligence* ainsi que les différents outils utilisés (Microsoft Power BI, bases de données tierces, etc.). Dans un premier temps, Agilux privilégie l'auto-formation sur base de la formation guidée [22] proposée par Microsoft. Si cette formation ne s'avère pas suffisante, Agilux est disposé à prendre part à des enseignements plus poussés, pour autant que Microsoft ou un acteur spécialisé en dispensent.

D'autre part, ces nouvelles compétences et expériences qui seront acquises dans le domaine de la *Business Intelligence* impactent légèrement le *Business Model* présenté à la section 1.5 (page 7). En effet, Agilux pourra mettre en avant une nouvelle proposition de valeur et entamera les démarches pour devenir un partenaire [23] de Microsoft. Par conséquent, cette nouvelle activité permettra de proposer de nouvelles solutions aux clients actuels, d'attaquer un nouveau marché et d'acquérir de nouveaux de clients. Cette diversification pourrait donc être génératrice de prestations de services. La figure 4.5 (page 46) représente en gris ce nouveau segment.

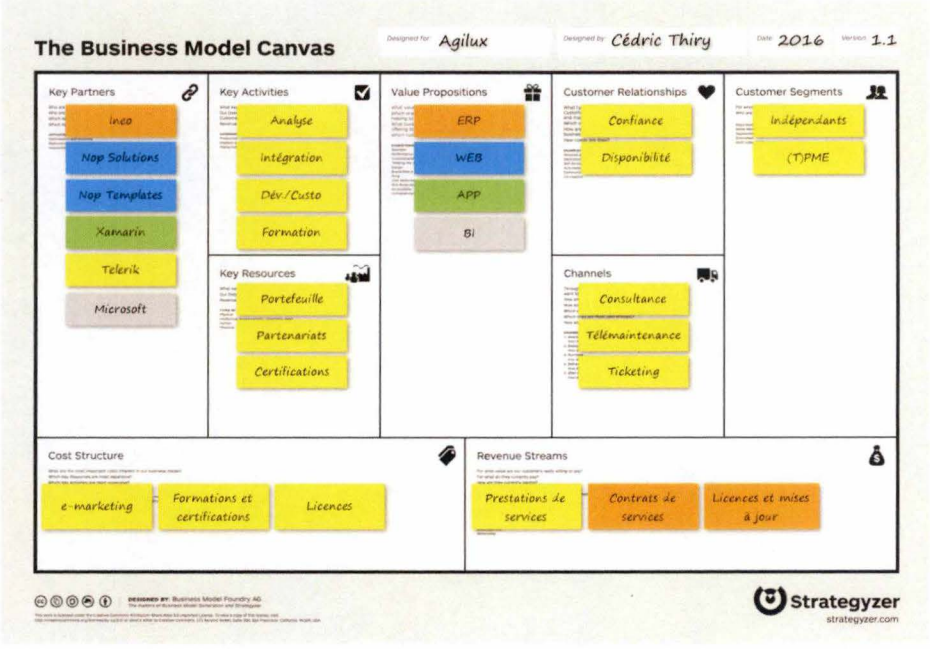

FIGURE 4.5 – Nouveau *Business Model* d'Agilux

22. La formation guidée de l'outil Microsoft Power BI est accessible via le lien ci-après : `https://powerbi.microsoft.com/fr-fr/guided-learning/`

23. La procédure pour devenir partenaire Microsoft Power BI est accessible via le lien ci-après : `https://powerbi.microsoft.com/fr-fr/become-a-partner/`

CONCLUSION

Au terme de cette pré-étude non-exhaustive, plusieurs éléments de réflexion ont été mis en évidence. Tout d'abord, le diagramme en arrêtes de poisson a permis de confirmer l'hypothèse de base selon laquelle le module de *reporting* de Mercator souffrait de certains problèmes, notamment au niveau du processus de génération, de la configuration des rapports réservée aux utilisateurs avertis et de leurs fondements méconnus des utilisateurs. Sur base de ce constat, le projet d'analyse plaide pour la mise en place d'une solution de *Business Intelligence* au sein d'une entreprise grossiste en vins et spiritueux, pour autant que l'étude de faisabilité permette d'obtenir l'accord pour sa réalisation. Celle-ci met l'accent sur les bénéfices d'un tel outil tout en mentionnant les différentes contraintes qui s'exercent sur le projet, à savoir le budget, le temps, les ressources humaines et la technologie.

À ce stade, la collecte des exigences est limitée à la vision d'Agilux. Des méthodes d'élicitation autres que l'introspection devront être mises en place lors des prochaines itérations visant à découvrir de nouvelles exigences formulées par le client. Celles-ci devront impérativement être analysées, triées et documentées. Dans l'état actuel, plusieurs approches techniques ont déjà été abordées, voire privilégiées, au niveau de l'architecture de *Data Warehouse* et de l'implémentation OLAP. La ligne de conduite consiste à employer des processus et des méthodologies simples, adaptées aux besoins et la taille du client.

Diverses solutions de *Business Intelligence* ont donc été analysées. Au terme de l'étude de marché, l'outil développé par Microsoft, Power BI, a été retenu sur base des nombreux avantages qu'il présente par rapport aux exigences, aux contraintes techniques et au *Business Model* d'Agilux. En cas d'acceptation du projet, Agilux investira du temps sous forme d'auto-formations afin de maîtriser de l'outil et de l'intégrer de façon optimale pour le client. Cependant, même si l'outil répond parfaitement aux besoins non-fonctionnels, la mise en place d'une batterie d'indicateurs reste un exercice compliqué pour lequel il n'existe aucune solution miracle. Le processus de définition des indicateurs devra être minutieusement réalisé en collaboration avec le client.

Au final, la réussite d'un tel projet aurait pour conséquences de donner l'opportunité au client d'améliorer sa capacité décisionnelle et de modifier le *Business Model* d'Agilux qui ajouterait une nouvelle corde à son arc, c'est-à-dire une expertise BI complémentaire avec le segment principal qu'est l'ERP.

RÉFÉRENCES BIBLIOGRAPHIQUES

AGILE ALLIANCE (2016), « Guide to Agile Practices », consulté en ligne le 25 mai 2016 : www.agilealliance.org/agile101/guide-to-agile.

AMERICAN SOCIETY FOR QUALITY (2008), « Fishbone (Ishikawa) Diagram », consulté en ligne le 12 juin 2016 : http://asq.org/learn-about-quality/cause-analysis-tools/overview/fishbone.html.

BURNAY, C. (2016), « IBAGM321 - Ingénierie des exigences », Slides.

DORAN, G. T. (1981), « There's a S.M.A.R.T. way to write management's goals and objectives », *Management Review* .

ECKERSON, W. W. (2006), « Deploying Dashboards and Scorecards », *The Data Warehouse Institute*.

ECKERSON, W. W. (2009), « Performance Management Strategies : How to Create and Deploy Effective Metrics », Technical Report Q1, IBM.

HANDY, C.B. (1976), *Understanding organizations*, Penguin.

HAZIYEV, S. (2013), « SoftServe BI-BigData Workshop in Utah », consulté en ligne le 13 août 2016 : http://www.slideshare.net/SerhiySergeHaziyev/softserve-bibigdata-workshop-in-utah.

HICKEY, A. M. et A. M. DAVIS (2004), « A Unified Model of Requirements Elicitation », *Journal of Management Information Systems*.

INMON, W. H. (2005), *Building the Data Warehouse*, Wiley.

ISHIKAWA, K. (1968), *Guide to Quality Control*, JUSE.

ISO/IEC 9126-1 (2001), « Software engineering - Product quality - Part 1 : Quality model », *ISO.org*, consulté en ligne le 19 mai 2016 : http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=22749.

KAPLAN, R. S. et D. P. NORTON (2008), *The Execution Premium : Linking Strategy to Operations for Competitive Advantage* , Harvard Business Review Press.

KIMBALL, R. et M. ROSS (2002), *The Data Warehouse Toolkit : The Complete Guide to Dimensional Modeling*, John Wiley & Sons Inc.

KOLP, M. (2016), « IBAGM323 - Software Project Management », Slides.

LINDEN, I. (2015), « IBAGM332 - Business Intelligence », Slides.

MAQUET, B. (2015), « IBAGM313 - Négociation et analyses des conflits », Syllabus v3.4.

MINTZBERG, H. (1982), « Structure et dynamique des organisations », *Paris : Les éditions d'organisation*.

NEGASH, S. (2004), « Business Intelligence », *communications of the Association for Information Systems*, 13, 177–195.

NUSEIBEH, E. et S. EASTERBROOK (2000), « Requirements Engineering : A Roadmap », *Proceedings of the Conference on The Future of Software Engineering*, 35–46.

PARENTEAU, J., R. L. SALLAM, C. HOWSON, J. TAPADINHAS, K. SCHLEGEL, et T. W. OESTREICH (2016), « Magic Quadrant for Business Intelligence and Analytics Platforms », *Gartner*, consulté en ligne le 27 juillet 2016 : `https://www.gartner.com/doc/reprints?id=1-2XXET8P&ct=160204`.

PARMENTER, D. (2015), *Key Performance Indicators : Developing, Implementing, and Using Winning KPIs*, Wiley.

PEDERSEN, T. B. (2013), « Managing Complex Multidimensional Data », *Lecture Notes in Business Information Processing*.

RANJAN, J. (2009), « Business Intelligence : concepts, components, techniques and benefits », *Journal of Theoretical and Applied Information Technology*.

RODMAN, T. (2015), « What is Power BI ? », consulté en ligne le 13 août 2016 : `http://www.timrodman.com/what-is-power-bi/`.

ROWLEY, J. (2007), « The wisdom hierarchy : representations of the DIKW hierarchy », *Journal of Information and Communication Science*.

STRATEGIZER (2014), « The Business Model Canvas », consulté en ligne le 2 janvier 2015 : `www.businessmodelgeneration.com/canvas/bmc`.

THOMPSON, R. (2002), « Stakeholder Analysis », consulté en ligne le 12 juin 2016 : `https://www.mindtools.com/pages/article/newPPM_07.htm`.

VAN LAMSWEERDE, A. (2009), *Requirements Engineering : From System Goals to UML Modles to Software Specifications*, Wiley.