Smals

Améliorer la gestion de données partagées

Master Data Management
Mise en place d'un référentiel de données

Clients & Services

Section Recherches

Date : Décembre 2009
Deliverable : 2009/TRIM4/01
Statut : Final
Auteur : Jean-Christophe Trigaux

Koninklijke Prinsstraat 102
1050 Brussel

Rue du Prince Royal 102
1050 Bruxelles

Tel : 02/787.57.11
Fax : 02/511.12.42

Management Summary

Quelle que soit sa sophistication, un système informatique ne peut fournir une aide efficace que s'il traite et partage des données cohérentes et de bonne qualité. L'apparition de données hétérogènes entraîne notamment : (1) des dysfonctionnements opérationnels dans des processus métier critiques, (2) des choix stratégiques fondés sur des données potentiellement incohérentes et (3) la mobilisation d'importantes ressources afin de resynchroniser les données entre différents services, voire différentes organisations. Cette hétérogénéité est principalement due au cloisonnement des données au sein des différentes applications existantes qui demeurent difficilement interopérables.

Au final, les données sont généralement peu valorisables car dupliquées dans plusieurs silos fonctionnels, chacun exploitant sa propre base de données avec ses propres structures de données, sa propre interprétation de leur contenu et ses propres règles métier.

L'enjeu du Master Data Management (MDM) est de faciliter la gestion des données de référence transversalement à différentes applications en mettant en place une organisation de circonstance supportée par un référentiel de données. La mise en place d'un tel référentiel permettrait de se réapproprier ses données métier, de les enrichir et d'assurer leur pérennité, indépendamment des processus qui les manipulent. D'un point de vue opérationnel, l'approche MDM permet de mutualiser les efforts et d'assurer la synchronisation, le partage et la qualité des données à travers plusieurs silos d'informations en quasi temps réel.

L'approche MDM est directement liée au partage des données entre acteurs de la sécurité sociale et/ou des soins de santé. Aussi bien la BCSS que eHealth ont été des précurseurs dans ce domaine. Chaque jour, de nombreuses banques de données gèrent et partagent des données ; quelques exemples révélateurs sont : la signalétique des citoyens, la signalétique des travailleurs, la carrière des employés, l'identification des entreprises, les vaccinations, les dons d'organes, la description des médicaments, etc.

Clairement, l'approche MDM est au cœur du eGovernment et de eHealth. L'appropriation des concepts MDM par les institutions gouvernementales représente une opportunité pour faciliter la collaboration entre les consommateurs et les fournisseurs de données, faciliter la mise en place d'une approche SOA, améliorer les services associés aux banques de données et enfin mutualiser les efforts en terme de synchronisation des données, d'amélioration de leur qualité et de gestion des anomalies.

En revanche, les outils MDM doivent être manipulés avec précaution. Même si la complémentarité des technologies peut devenir un avantage indéniable, leur intégration est encore difficile au sein d'un seul outil. De plus, le support lié à la gouvernance des données est pour l'instant la pierre angulaire qui manque crucialement aux outils MDM. Enfin, l'intégration de ces outils avec les applications existantes peut s'avérer délicate.

Table des matières

Liste des Figures

Liste des Tables

But et structure du document

But du document

Comme vous allez le découvrir tout au long de ce document, l'approche du Master Data Management (MDM) est très large et couvre de nombreuses dimensions. La contribution principale de ce document est de donner un aperçu global de la problématique et des différentes solutions envisageables pour la résoudre. Nous ne pouvons présenter en détail chaque dimension car elles mériteraient à elles seules une étude complète.

L'objectif est essentiellement de familiariser le lecteur avec les méthodes et techniques permettant de mieux gérer et partager ses données métier. Cette problématique est au cœur de l'approche MDM dont nous présenterons les principes fondamentaux ainsi que les aspects pratiques et organisationnels liés à la mise en place d'un référentiel de données. Nous illustrerons la présente étude à l'aide d'un cas pratique basé sur l'échange de dossiers électroniques médicaux minimaux entre prestataires de soins de santé belges.

Nous espérons que ce document pourra servir de cadre de réflexion pour les acteurs impliqués dans la gestion et l'échange de données entre différentes applications et systèmes. Nous espérons qu'il donnera un éclairage nouveau sur l'interopérabilité des données et sur les différents aspects méthodologiques et techniques qui favorisent la bonne gouvernance de nos données.

Structure du document

Dès lors, après avoir introduit l'approche MDM ainsi que le cas pratique qui nous servira de fil conducteur tout au long de l'étude (Chapitre 1), nous présenterons les principaux concepts de l'approche MDM (Chapitre 2). Dans ce chapitre, nous aborderons plus particulièrement les trois piliers fondamentaux du MDM (Data Governance, Data Quality, Data Integration) et nous analyserons les différents avantages et inconvénients des architectures MDM proposées dans la littérature.

Ensuite, nous décrirons les différentes étapes à suivre pour mettre en place un référentiel de données (Chapitre 3) et les illustrerons à l'aide de notre cas pratique. Au terme de ce chapitre, nous présenterons un prototype basé sur l'outil open-source Mural simulant un référentiel de données permettant d'échanger des dossiers électroniques médicaux minimaux entre prestataires de soins.

Dans le chapitre suivant, nous présenterons les principales fonctionnalités logicielles permettant de faciliter la gestion et la maintenance d'un référentiel de données. Nous analyserons également le marché des outils MDM en constante évolution (Chapitre 4).

Un dernier chapitre examinera les liens entre les approches MDM et SOA et pourquoi le MDM pourrait jouer un rôle de facilitateur pour le SOA (Chapitre 5).

1. MDM : Introduction

Ce chapitre a pour objectif d'introduire l'approche du Master Data Management (MDM). Dans un premier temps, nous décrirons ce qu'est l'approche MDM (Section 1.1) et en quoi l'approche MDM peut être utile (Section 1.2). Dans un second temps, nous aborderons différentes situations dans lesquelles l'approche MDM a déjà fait ses preuves d'une manière générale (Section 1.3) mais également dans le contexte de la sécurité sociale et des soins de santé en Belgique (Section 1.4). Au terme de ce chapitre, nous décrirons un cas pratique (Section 1.4.4) dans le domaine des soins de santé belge qui nous servira d'illustration tout au long des chapitres suivants.

1.1. Qu'est ce que l'approche MDM ?

Quelle que soit sa sophistication, un système d'information ne fournit une aide efficace que s'il peut proposer et traiter des données pertinentes et de qualité. Il est donc primordial de pouvoir (1) mesurer cette qualité, (2) l'améliorer de manière continue et (3) la piloter suivant nos besoins et usages (fitness for use). Pourtant, on ne peut mesurer ce qu'on ne contrôle pas et on ne peut contrôler ce qu'on ne connaît pas ou plus.

Ces impératifs apparaissent encore plus clairement lorsque plusieurs applications s'échangent des données. D'un côté, la signification exacte et même parfois le contenu des données peuvent ne plus être maîtrisés. D'un autre côté, une fois que les données sont échangées, il devient délicat d'établir qui en est responsable et comment elles peuvent être contrôlées.

L'échange contrôlé de données entre applications est un problème complexe. En effet, force est de constater que les applications constituant les systèmes d'information aussi bien des grandes entreprises que des administrations publiques sont fortement hétérogènes. Cette hétérogénéité engendre une perte de la maîtrise des données qui, dans la majorité des cas, restent cloisonnées au sein des différentes applications existantes difficilement interopérables. Au final, les données sont souvent dupliquées dans plusieurs silos fonctionnels, chacun exploitant sa propre base de données avec ses propres structures de données, sa propre interprétation de leur contenu et ses propres règles métier.

L'apparition de données hétérogènes entraîne notamment : (1) des dysfonctionnements opérationnels dans des processus métier critiques, (2) des choix stratégiques fondés sur des données potentiellement incohérentes et (3) la mobilisation d'importantes ressources afin de resynchroniser les données entre différents services, voire différentes organisations.

L'**enjeu du MDM** est de pouvoir mettre en place un référentiel de données ainsi qu'une organisation adaptée qui permettront de gérer les données transversalement aux différents projets et applications. L'objectif de l'approche MDM est de mutualiser les efforts et d'assurer la synchronisation, le partage et le contrôle des données à travers les différents silos en quasi temps réel.

Un **référentiel de données** consiste essentiellement en une application qui supervise la gestion d'une banque de données alimentée par plusieurs fournisseurs et consultable par des différents consommateurs. Ce référentiel se focalise sur les données à haute valeur ajoutée dont la qualité et l'accessibilité sont cruciales pour les partenaires métier. Ces données sont aussi appelées **données de référence** ou **master data**. L'objectif du référentiel est d'intégrer et d'uniformiser les différentes données reçues ou collectées pour ensuite les rendre facilement accessibles. Cette intégration peut-être réalisée de manière *logique* ou *physique* :

- Soit le référentiel de données joue un rôle d'**annuaire de données** permettant aux consommateurs d'identifier à quel(s) fournisseurs de données ils doivent s'adresser (*Intégration Logique*). Cet annuaire suit le même principe qu'un annuaire téléphonique ; il facilite les échanges de données sans s'occuper de leur contenu.

- Soit le référentiel de données joue un rôle de **consolidateur (harmonisateur) de données** et une base de données spécifiquement dédiée à la gestion des données de référence est créée afin de faciliter leur consolidation (*Intégration Physique*).

Depuis l'ouverture des systèmes d'information vers l'extérieur, la valorisation et le partage des données sont devenus des éléments primordiaux. Le cloisonnement des données constitue cependant un frein majeur à cette valorisation :

- Dans le meilleur des cas, les applications tentent de garantir la synchronisation des différentes données qu'elles partagent (produits, code pays, clients, employés, patients, etc.). Au fil du temps, cette synchronisation est malheureusement rarement conservée.

- Dans le pire des cas, les différents acteurs n'envisagent même plus la synchronisation de certaines de leurs données simplement en raison du fait qu'ils ne parlent pas le même langage. Les données qu'ils s'échangent ne peuvent être confrontées car ils ne les interprètent pas de la même manière. Dès lors, sans un référentiel commun pour partager leurs données, toute échange devient infructueux.

Les problèmes liés à la synchronisation et la qualité des données ne sont souvent identifiés que tardivement par les utilisateurs finaux, ce qui est évidemment la pire des situations. La qualité des données ne peut pas être garantie par le biais d'une opération curative et limitée dans le temps.

L'enjeu est de gérer les données transversalement aux applications réparties dans différents services et sur différents continents. Gérer transversalement des données signifie pouvoir les partager, contrôler leur synchronisation, identifier, en quasi temps réel, leurs problèmes de qualité et les corriger conformément à un processus bien défini. Cette gestion transversale des données amène à investir dans la gestion des données de référence, aussi appelée le **Master Data Management (MDM)**.

Le MDM est une approche récente (2005). Néanmoins, à l'échelle de l'histoire de l'informatique, son passé ne doit pas être oublié (voir Figure 1).

Durant les années 1980, les premiers travaux de recherche sur la gestion des données concernaient essentiellement leur stockage, la manière dont les données pouvaient être stockées sur disque et être rendues accessibles le plus rapidement possible.

À partir de 1995, des avancées particulièrement probantes ont été accomplies au niveau de l'analyse des données tant du point de vue de leur contenu que de leur signification. L'objectif était d'améliorer la connaissance qu'on avait sur ses données afin de les rendre à la fois mieux interprétables et plus exploitables.

Depuis 2005, au travers de l'approche MDM, on s'interroge non seulement sur la fiabilité du contenu des données mais également sur la fiabilité des échanges et sur la manière dont-elles sont partagées et réutilisées.

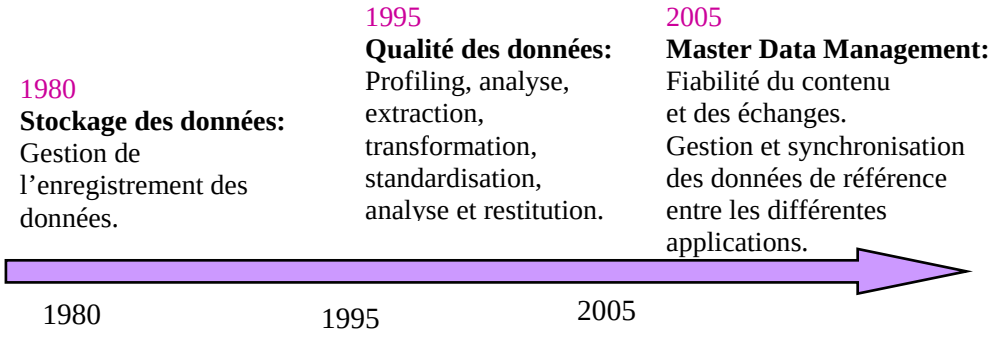


Figure 1: Historique du MDM

Le MDM n'est ni une technologie, ni un logiciel mais une méthode qui se focalise sur la rationalisation de la gestion des données partagées au sein d'une organisation ou entre plusieurs organisations. L'objectif du Master Data Management est de gérer de manière unifiée et transversale les données partagées. Malheureusement, elles sont souvent hétérogènes et dispersées dans plusieurs bases de données non synchronisées. Le MDM veut pallier cette problématique en :

- définissant un **référentiel** commun pour les données partagées,
- automatisant le **partage et la synchronisation** des données,
- déterminant les règles de **gouvernance** associées aux données : Qui peut y accéder ?, Qui peut les modifier ?, Qui peut faire du reporting d'anomalies ?, etc.
- garantissant la **qualité** des données dans le temps,
- favorisant l'**intégration** des données : soit physiquement dans une base de données commune, soit logiquement dans un annuaire de données déterminant les redirections vers les fournisseurs de données, soit en combinant une intégration physique et logique.

Évidemment, l'approche MDM n'est pas la solution miracle qui permettra de résoudre tous les problèmes liés aux données. C'est une approche conceptuelle dont la contribution majeure est de mettre ensemble diverses techniques et méthodes préexistantes. La mise en place d'une telle approche a des impacts technologiques mais surtout organisationnels. Gérer les données de manière

transversale à plusieurs services ou organisations nécessite forcément d'accorder les violons entre un grand nombre d'intervenants avec des objectifs stratégiques souvent différents, même au sein d'une même organisation.

De plus, l'intégration de l'ensemble des systèmes d'information d'une grande entreprise ou d'une grande administration constitue un défi majeur. Cette intégration risque à terme d'agréger des erreurs et d'amener les utilisateurs du système à perdre confiance dans les informations produites par celui-ci. C'est pourquoi des outils logiciels sont indispensables afin de supporter et de diminuer la complexité liée à la gestion des données.

En effet, bien que l'approche MDM soit avant tout une méthode, celle-ci devrait être supportée par des solutions MDM spécifiquement adaptées à cet effet. Au départ, les solutions MDM étaient spécifiques à certains domaines et à certains types de données. Essentiellement, deux catégories de solutions MDM dites « **verticales** » se sont distinguées :

1. *La gestion des catalogues produits* (Product Information Management (PIM)) notamment dans les domaines de la grande distribution et du manufacturing,

2. *L'intégration des données clients* (Customer Data Integration (CDI)) particulièrement pour l'administration de grosses bases de données transactionnelles (gestion des doublons, vérification et homogénéisation des adresses, etc.) dans des domaines tels que les banques ou les assurances.

À l'heure actuelle, les solutions MDM tendent de plus en plus vers des solutions « **horizontales** » et génériques qui prennent en compte tous types de données et qui couvrent l'ensemble de leur cycle de vie. Généralement, on distingue deux catégories de solutions MDM :

- **Le MDM analytique** où les solutions MDM se limitent principalement à faciliter les prises de décision sur des ensembles de données spécialement adaptées à ce type d'analyse.
- **Le MDM opérationnel** où les solutions MDM permettent de définir, créer et synchroniser les données de référence de qualité nécessaires au bon fonctionnement d'un système transactionnel et délivrées en quasi temps réel.

1.2. Pourquoi utiliser l'approche MDM ?

Au fil du temps, les organisations ont constaté qu'elles perdaient le contrôle de leurs données. L'origine de cette perte de contrôle s'explique au travers de différents facteurs :

- Le **volume** et la **complexité** des données ne cessent de croître. La quantité d'information à stocker est de plus en plus importante, le niveau de détails exigé pour décrire une donnée est de plus en plus fin et les structures de données deviennent de plus en plus complexes.

- Les données sont de plus en plus **interdépendantes**. En effet, les contraintes métier imposent souvent des dépendances fortes entre les données. À tout moment il faut pouvoir être capable de vérifier que ces contraintes sont respectées. L'augmentation du volume de données implique l'augmentation du nombre de ces contraintes, ce qui entraîne

une explosion de la complexité. Dès lors, lorsqu'une donnée est modifiée, il devient de plus en plus difficile d'identifier quels seront les impacts éventuels sur d'autres données et de vérifier que les contraintes métier sont toujours satisfaites.

- Dans un monde de plus en plus ouvert, où les données sont une ressource à part entière, elles doivent être **partagées.** Le problème est similaire à celui du jeu du téléphone arabe. Si j'énonce une phrase quelconque à la première personne en tête d'une file et que je lui demande de la répéter à son voisin et ainsi de suite, j'ai la quasi certitude, que, lorsque cette phrase arrivera à la fin, sa signification aura probablement changé du tout au tout. En effet, plus il existe d'intermédiaires, plus le contenu de la donnée risque d'être altéré ; soit parce que les intermédiaires n'utilisent pas le même langage, soit parce qu'ils estiment que certaines corrections peuvent être apportées ou que certaines données sont négligeables.

- Si aucune mesure n'est prise, la **qualité** des données manipulées a naturellement toujours tendance à se détériorer. Soit les données sont mal introduites, soit elles sont incomplètes, soit elles ne sont plus à jour, soit elles sont corrompues lors de mauvaises manipulations, etc.

- Certaines données sont **dispersées** et **dupliquées**. Les données peuvent être dupliquées pour des raisons d'efficacité ou de facilité d'accès. Cependant, il faut toujours s'assurer que ces données dupliquées restent synchronisées et qu'elles n'évoluent pas de manière anarchique. Malheureusement, cette synchronisation n'est souvent pas garantie, ce qui entraîne l'apparition de données **hétérogènes.** Si l'adresse d'un même citoyen est différente d'une application à l'autre, il faut en étudier la raison avec précaution. On doit être capable, d'une part, de retrouver et de rendre accessibles ses données où qu'elles se trouvent et, d'autre part, de détecter et de lever les divergences entre données hétérogènes. Les origines de ces divergences se retrouvent soit au niveau de leur contenu (valeur), soit dans la manière dont elles sont interprétées (définition) :

    - D'une part, une donnée peut avoir une **définition univoque mais des valeurs hétérogènes** ; soit parce qu'une faute de frappe a été malencontreusement introduite, soit parce que le rythme de mise à jour des données est différent. Par exemple, l'adresse du domicile légal d'un citoyen a une définition juridique univoque mais un même citoyen peut avoir trois adresses différentes selon les applications considérées. Dans la première application, il est domicilié « 176 avenue des alouettes », dans la seconde, « 176 avenue des allouettes » et dans la troisième, « 13 rue des mésanges ». Si la première adresse est considérée comme son véritable domicile légal, on peut présupposer qu'une erreur de frappe a été introduite dans la deuxième application, tandis que la troisième application n'a pas encore été avertie du changement d'adresse.

    - D'autre part, une même donnée peut avoir des **définitions différentes** suivant l'application ou le domaine métier considéré. Par exemple, la notion d'adresse peut être interprétée soit comme l'adresse effective d'un citoyen soit comme l'adresse de son domicile légal. Suivant la définition utilisée, la valeur de la donnée est correcte mais les applications ne l'interprètent pas de la même manière. Si ces différences

d'interprétation ne sont pas rendues explicites, les données deviendront difficilement exploitables pour les consommateurs de données. Dans ce cas de figure, même si les données transitent d'une application à l'autre, aucune des deux applications ne peut en tirer profit, que du contraire. En effet, ces données constituent du « bruit » qui risque de perturber le bon fonctionnement des applications.

L'apparition de données hétérogènes ou de mauvaise qualité engendrent notamment : (1) des dysfonctionnements opérationnels dans des processus métier critiques, (2) des choix stratégiques se basant sur des données potentiellement erronées et (3) la mobilisation d'importantes ressources afin de résoudre les problèmes liés aux données. Les difficultés à synchroniser et à (ré)homogénéiser ces données ont naturellement mis en lumière la nécessité de reprendre leur contrôle. Ces difficultés s'accentuent en fonction de l'éparpillement des données dans les applications et du nombre d'applications impliquées dans les échanges. La donnée est en quelque sorte la matière première de tout système d'information et doit clairement être au centre des préoccupations des entreprises et des institutions. Celles-ci vont devoir se doter d'architectures efficaces pour valoriser les données accumulées. À l'heure actuelle, les systèmes d'information sont de plus en plus ouverts et nécessitent de s'échanger des données valides mais aussi consolidées et cohérentes entre elles. L'objectif est d'offrir à l'ensemble des acteurs impliqués une vision unique et authentique des données offrant la possibilité :

- de garantir une meilleure réactivité des agents, notamment grâce à la facilité de mener une investigation,
- de répondre de manière plus rapide à des changements de réglementation,
- de restituer des données complètes et de qualité,
- d'échanger plus facilement les données entre applications hétérogènes sous un format standardisé,
- de fournir les données pertinentes rapidement et sous différentes formes afin d'améliorer la capacité décisionnelle du métier et d'augmenter le crédit accordé à ces données.

D'un point de vue économique, le principal avantage de l'approche MDM (et en particulier la mise en œuvre de référentiels centraux de données) est de réduire les coûts des services IT. L'éditeur de logiciel Siperian souligne comment l'approche MDM permet de réduire ses coûts IT [Shankar, 09] :

1) **Réduire les coûts d'interfaces applicatives** en (1) rationalisant les flux de données partagés par différents processus métier et en (2) réduisant le nombre d'interactions entre applications.

2) **Réduire les coûts des redondances de données** en (1) limitant les acquisitions dupliquées de données, réalisées par les différents départements d'une même organisation et en (2) constituant, grâce au référentiel central, un point unique d'acquisition, de stockage et de distribution de ces données pour l'ensemble des fournisseurs-consommateurs dans une ou plusieurs organisations.

3) **Réduire les coûts de nettoyage de données** en (1) centralisant les initiatives d'amélioration de qualité des données, en (2) évitant la

multiplication d'initiatives spécifiques à chaque application et en (3) permettant l'identification de doublons inter-applications.

4) **Réduire les coûts de traitement et de nettoyage de données externalisées** en (1) mutualisant les efforts de nettoyage - par exemple en utilisant un outil de gestion de qualité des données - et en (2) les mettant à disposition de tous au moyen d'un répertoire central partagé.

5) **Réduire les coûts de licence, de support et de matériel des systèmes redondants** en (1) réduisant le nombre d'entrepôts de données et en (2) rendant obsolètes ceux contenant des données dupliquées. À terme, cela permet de réduire le coût des licences, les coûts liés aux outils de développement pour personnaliser ces applicatifs, les coûts de maintenance liés aux plateformes physiques les hébergeant, etc.

6) **Réduire les coûts de développement et de maintenance** en utilisant une plateforme MDM configurable et évolutive offrant des services standardisés d'accès et de modification des données de référence.

7) **Réduire les coûts de livraison d'information** en mettant en œuvre un référentiel (1) délivrant des données de qualité, à jour et traçable, (2) évitant ainsi les aller-retour entre le métier et les services IT pour discuter de l'origine, de la pertinence ou de la fraîcheur des données.

1.3. Dans quelles situations l'approche et les solutions MDM ont-elles déjà fait leurs preuves ?

Les problématiques liées à l'approche MDM ne sont pas nouvelles et de nombreuses entreprises y sont confrontées tous le jours. Chaque fournisseur de solutions MDM met en avant ses principaux succès dans des entreprises à la fois privées et publiques. Nous attirons l'attention du lecteur sur le fait que ces « success stories » ne sont pas neutres et certainement trop superficielles pour comprendre comment les solutions proposées peuvent concrètement résoudre la problématique initiale. Néanmoins, elles donnent une idée du champ d'application de l'approche MDM.

Cette approche a fait ses premières armes dans le secteur bancaire pour lequel les données collectées sur les clients sont capitales. Aussi bien au niveau de la vente que du recouvrement des crédits, les données clients doivent être consolidées à la fois au sein des différents services d'une banque mais également lors de fusions ou d'acquisitions entre banques. En effet, ces fusions ont des impacts très importants sur les systèmes informatiques et sur les données qu'ils manipulent.

Le monde bancaire n'est néanmoins pas le seul à avoir profiter des bénéfices de l'approche MDM. Des domaines beaucoup plus proches de notre métier ont aussi appliqués avec succès les principes du MDM, par exemple :

- Les organismes publics français (Unedic-Assedic) en charge de la gestion de l'assurance chômage au niveau national et local.
- Certains réseaux régionaux d'hôpitaux désireux d'améliorer la qualité et l'intégrité des données médicales et administratives à destination des cliniciens et des patients.

1.3.1. Bank of Northern Ireland

Le cas : La Bank of Northern Ireland offre des services bancaires aux particuliers et aux entreprises et vend une grande variété de produits bancaires.

Le challenge : Les données des clients de cette banque étaient dispersées et dupliquées dans chaque application spécifiquement dédiée à un produit bancaire : emprunts hypothécaires, emprunts à la consommation, comptes courant et cartes de crédit [Datactics, 05].

La solution : La solution choisie par l'intégrateur de systèmes financiers Kainos fut Datactics [Datactics, 05].

Les résultats : Cette solution ainsi que le travail fournit par l'intégrateur et l'entreprise ont permis d'obtenir une « single view on customer », de réduire drastiquement le nombre de doublons, d'augmenter la complétude, l'intégrité et la qualité des données au travers des différents services de la banque. À l'époque, cette banque détenait des données bancaires sur 165000 clients [Datactics, 05].

1.3.2. Unedic

Le cas : L'unedic et les assedic sont les organismes publics français en charge de la gestion de l'assurance chômage, respectivement au niveau national et régional.

Le challenge : L'objectif était la création d'un système décisionnel afin de partager et communiquer des données, d'obtenir un rafraîchissement au quotidien des informations métier à des fins de production statistique et de pilotage, de faciliter le pilotage au niveau local et permettre la consolidation et le pilotage au niveau national, de faciliter la mission statistique (analyses, prospectives et études), de réduire les activités de collecte des données en institution afin de renforcer les activités d'analyse [Faucher, 07].

La solution : La solution MDM se base sur IBM InfoSphere DataStage pour la collecte des données et sur SAS Suite pour la diffusion des données stockées dans un entrepôt de données [Faucher, 07].

Les résultats : Depuis octobre 2006, c'est près de 100 millions d'évènements par mois qui sont chargés dans le DataWarehouse avec en 2007 un nombre total d'utilisateurs déclarés de 1353 dont 34,15% sont des producteurs de rapports [Faucher, 07].

1.3.3. Sussex Health Informatics Service

Le cas : Le Sussex Health Informatics Service offre des services informatiques aux différents centres hospitaliers du Sussex appartenant au réseau public des soins de santé anglais (NHS) dans le cadre du programme national d'informatisation (NPfIT).

Le challenge : Le challenge consistait à permettre aux différents départements des hôpitaux de la région du Sussex de produire des rapports à destination des patients et du ministère de la santé, d'améliorer la qualité et l'intégrité des données médicales et administratives et de donner la capacité aux cliniciens d'obtenir des informations médicales concernant des patients ayant subi des examens dans les hôpitaux membres du réseau [Davies, 09].

La solution : La solution MDM basée sur les produits Javacaps, Mural et BPEL de Sun a permis de mettre en place une architecture flexible de partage de données médicales basées sur le standard international HL7, d'automatiser certaines corrections de données et d'améliorer la qualité de celles-ci de manière continue [Davies, 09].

Les résultats : Par an, archivage et gestion des données médicales concernant 102000 consultations pour des patients hospitalisés, 376000 consultations pour des patients externes, 115000 urgences, 5700 naissances, 75000 opérations, 170000 tests sanguins, 600000 prescriptions, 210000 examens d'imagerie médicale (x-rays/scans) [Davies, 09].

1.3.4. Capital Health

Le cas : Capital Health Region est l'une des plus grandes régions de soins de santé au Canada qui regroupe différents hôpitaux universitaires.

Le challenge : Regrouper des silos de données sur les patients, résidant sur 14 systèmes disparates, afin de fournir une vue complète des patients par l'intermédiaire d'un nouveau système de dossiers de santé électroniques [Initiate, 03].

La solution : Intégration et correction des données dans le cas de données existantes, à l'aide du logiciel Initiate Identity Hub™ et de la technologie d'intégration QUOVADX™ via le plate-forme netCARE [Initiate, 03].

Les résultats : Cinq millions d'enregistrements médicaux représentant 1,8 million de patients, sont archivés. NetCARE fournit des informations sur les patients en temps réel, tous les jours, 24 heures sur 24 et traitent près de 13000 nouvelles transactions par jour. De plus, le nombre moyen de records dupliqués par mois a été réduit de plus de 95% [Initiate, 03].

1.4. Dans quelles situations l'approche MDM pourrait-elle s'appliquer dans le contexte de la sécurité sociale et des soins de santé belges ?

En réalité, c'est sans doute une des approches les plus récurrentes à tous projets d'échange de données entre acteurs de soins de santé et/ou acteurs sociaux. La BCSS et eHealth en sont deux exemples révélateurs. En effet, ces deux organisations ont notamment pour objectif de faciliter la mise en place de projets de type MDM en offrant une infrastructure d'échange de données et en jouant le rôle de facilitateur entre les différents acteurs impliqués.

Tout projet visant à constituer un référentiel de données est un projet MDM, surtout lorsque plusieurs sources authentiques collaborent pour alimenter ce référentiel. Comme dans tous projets, plus le nombre d'intervenants est important plus la conduite du projet est risquée. Plus le nombre de fournisseurs et de consommateurs de données sera grand, plus le référentiel sera difficile à constituer et à gérer. Outre le nombre d'intervenants métier, le nombre d'applications impliquées dans la création d'un référentiel est aussi un facteur déterminant de complexité.

- La **situation la plus simple** est l'utilisation d'une application pour saisir, stocker et consulter les données. Par exemple, la création d'un portail web permettant de consulter une base de données alimentée par différents intervenants via la même application et/ou les mêmes services web. Dans cette situation, l'approche MDM est sans doute trop lourde. Une centralisation forte des données n'est néanmoins pas toujours envisageable, surtout lorsque les données doivent être

récupérées à partir de plusieurs applications existantes et/ou qu'elles sont sensibles (ex : les données médicales ou sociales).

- La **situation la plus complexe** est de devoir faire collaborer plusieurs applications différentes à la fois au niveau de la saisie des données, de leur gestion, de leur stockage et de leur consultation. Toutes ces applications appartiennent généralement à des acteurs avec des intérêts divergents, confrontés à des besoins différents, utilisant des structures de données complexes et non uniformes, avec des rythmes de mise à jour des données différents et des interprétations spécifiques pour certaines données métier, etc.

De nombreux projets liés au MDM sont initiés par Smals et ses partenaires dans les domaines de la sécurité sociale et des soins de santé. L'approche MDM étant très étendue, certains projets se focalisent sur une facette bien précise. Voici une liste non exhaustive de différents projets au cœur desquels on retrouve certains aspects du MDM. Le projet SumEHR retiendra particulièrement notre attention, car il servira de fil conducteur tout au long de ce rapport d'étude.

1.4.1. Projets dans le domaine de la sécurité sociale

Projet	Membre	Description	Aspect MDM
Athena	ONSS-ONVA-SIGeDIS	Mise en place d'un concept de collaboration entre institutions de la sécurité sociale pour la gestion des données individuelles et de la gestion des anomalies : analyse à la mise en place d'une base de données, de contrôles sur ces données, d'un workflow de traitement des anomalies,...	Création d'un référentiel pour les données carrière avec gestion transversale des anomalies.
Argo	SIGeDIS ONP	SIGeDIS réalise le projet Argo qui a pour objectif de faire le reegineering des applications informatiques liées aux comptes individuels de pension.	Création d'un référentiel des données pour le premier pilier des pensions.
Capelo	SIGeDIS SdPSP	Capelo signifie Carrière publique électronique - Elektronische loopbaan overheid. L'initiative émane du Service des Pensions du Secteur public. Le but est de réaliser une banque de données des carrières du secteur public.	Création d'un référentiel des données pour les carrières.
Falco	ONSS	Système de Knowledge management pour la gestion des anomalies DmfA.	Documentation des règles de gouvernance pour gérer les anomalies.
Glossaire-Dmfa	Smals - Gestion de	Élaboration d'un dictionnaire de données avec workflow de	Définition du format et de la structure des

	l'information	validation afin de documenter les messages XML DmfA.	données de la DmfA en vue d'uniformiser l'échange des données au fil des versions.
Terminologie	Smals- Gestion de l'information	Harmonisation de la terminologie des pages statiques du portail de la sécurité sociale et des intitulés des formulaires électroniques.	Description de la signification des données en vue d'améliorer la qualité des données lors de leur saisie.

Table 1 : Projets MDM dans la sécurité sociale

1.4.2. Projets dans le domaine de la santé.

Projet	Membre	Description	Aspect(s) MDM
SAM	AFMP	Création d'une source authentique de données pour les médicaments des agences FAGG et BCFI.	Création d'un référentiel pour les données descriptives des médicaments résultant de l'intégration des données INAMI, FAGG, BCFI.
eCare	INAMI	eCare vise à fournir une solution technique afin que les professionnels de la santé puissent enregistrer une série de données anonymisées) afin d'assurer l'analyse du suivi clinique des patients, la documentation du processus de remboursement de l'INAMI ou l'amélioration de la politique d'assurance individuelle.	Création de plusieurs référentiels de données pour : - les rhumatologues, - les orthopédistes, - les cardiologues, - les pacemakers, - les endoprothèses, - les tuteurs coronaires.
Orgadon	SPF Santé publique	Mise en production d'une banque de données des donneurs d'organes qui peut être consultée par les coordinateurs de transplantation de divers hôpitaux belges.	Création d'un référentiel pour les dons d'organes.
Non déterminé	SPF Santé publique	L'objectif d'un tel projet serait de partager des dossiers médicaux minimaux entre prestataires de soins de santé en utilisant le format standard XML : SumEHR.	Échange de dossiers médicaux minimum sous format électronique entre prestataires de soins de santé

Table 2 : Projets MDM dans les soins de santé

1.4.3. Projets dans d'autres domaines

Projet	Membre	Description	Aspect MDM
ALINE	SPF intérieur	L'application ALINE (Alarm Information Exchange) permet l'enregistrement et la gestion des déclarations relatives aux systèmes d'alarme installés sur le territoire belge.	Croisement et échange de données entre la police, les centrales d'alarme et le registre national (principalement sur les localisations) pour améliorer l'efficacité des interventions suite à des intrusions.

Table 3 : Projets MDM dans d'autres domaines

1.4.4. Projet SumEHR

L'accès au soin de santé est un besoin fondamental. En Belgique, de nombreux professionnels de la santé répartis dans différents établissements prodiguent des soins. Les données concernant notre état de santé sont recueillies et conservées sur différents supports : papier, films, fichiers électroniques, etc. Les données nous concernant sont dispersées auprès de différents prestataires et institutions de soins. La qualité des soins de santé dépend aussi de la qualité des données nous concernant et de la capacité des prestataires et institutions de soins de santé à s'échanger ces données. Dans cette perspective, tout patient peut se poser les questions suivantes :

- Suis-je certain que le médecin que je consulte connaît tous mes antécédents médicaux ?
- Que se passera-t-il quand mon médecin traitant prendra sa retraite ?
- Où peut-on trouver un aperçu complet de mes antécédents médicaux ?
- Qui connaît les médicaments que je prends actuellement ?
- Que m'arrivera-t-il, si en cas d'urgence, je me trouve loin de la clinique ou de l'hôpital où je me rends habituellement ?

Le SumEHR (ou SUMmarized Electronic Health Record) correspond à la photographie sanitaire d'un patient. Il s'agit d'un instantané que le médecin traitant, en tant que gestionnaire du Dossier Médical Global (DMG), saisira lors de contacts privilégiés avec le patient. Il ne s'agit donc pas d'un dossier santé complet, mais bien d'une extraction à partir de celui-ci des éléments de soins utiles au suivi médical.

Le SumEHR constitue un document de liaison entre les Dossiers Médicaux Informatisés (DMI) des médecins généralistes, mais aussi avec ceux des médecins spécialistes hospitaliers ou non. Il pourrait être partagé selon plusieurs modes, par exemple :

- le simple transfert d'un fichier électronique à partir d'un logiciel médical vers un autre,
- sa conservation au sein d'un container, formant ainsi un dossier santé partagé établi au sein d'un réseau santé, etc.

L'échange de ces dossiers médicaux minimaux pourrait être facilité grâce à la plate-forme eHealth qui « est avant tout une institution publique, instituée par la loi, qui vise à promouvoir et à soutenir l'échange électronique et sécurisé de données entre tous les acteurs des soins de santé (médecins, hôpitaux, pharmaciens, patients, …) tout en respectant la protection de la vie privée et le secret médical »[1].

Lorsque les SumEHRs seront mis à la disposition de plusieurs prestataires de soins, il faudra établir des règles d'accès spécifiant qui peut accéder à quel dossier SumEHR et à quelles parties de celui-ci. Par exemple, une condition préalable est l'existence d'un lien thérapeutique fort entre le médecin et le patient. Cette règle peut paraître évidente, pourtant, définir la notion de « lien thérapeutique fort » est une question fondamentale et délicate dont les aspects juridiques ne doivent pas être négligés.

Les logiciels de médecine générale dédiés aux médecins généralistes et homologués par le SPF Santé Publique doivent pouvoir produire (exporter) et intégrer (importer) de manière standardisée des fichiers électroniques contenant les données de type SumEHR. D'un point de vue technique, le message qui transporte ces données SumEHR fait partie du standard belge KmEHR[2]. A l'initiative du SPF Santé Publique, ce standard décrit un ensemble de messages normalisant les principales transactions médicales (lettre de sortie, lettre de transfert, notes de liaison, protocoles médico-techniques, dossiers médicaux minimaux, etc.). Ces messages appartiennent à un langage commun, en l'occurrence KmEHR, qui détermine la structure des messages (syntaxe) et les règles régissant cette syntaxe (grammaire).

Parmi ces messages KmEHR, un message a été spécifiquement défini pour contenir les données SumEHR. Dès lors, comme tout message KmEHR, un message SumEHR est décomposé en un header et un folder. Chaque donnée incluse dans le header ou le folder possède une structure bien définie que nous ne détaillerons pas ici. Le détail complet de ces structures est disponible sur le site du CHU de Charleroi[3]. Voici un bref aperçu des structures nécessaires pour un message SumEHR :

1. Le **header** décrit les données obligatoires suivantes :

Data	Purpose
Standard	You must specify here the version of the Kmehr-Bis specification your message complies with.
Date	This is the date of creation of the message.
Time	This is the time of creation of the message.
Sender	The sender can contain a combination of healthcare party(s) to specify the sending organisation, medical specialty and/or physical person.
recipient(s)	The recipient can contain a combination of healthcare party(s) to specify the receiving organisation, medical specialty and/or physical person. It must contain at least one hcparty. By convention, the first recipient is the main addressee of the message. The following recipients are considered as cc (carbon

[1] https://www.ehealth.fgov.be/
[2] http://www.chu-charleroi.be/kmehr/htm/kmehr.htm
[3] http://www.chu-charleroi.be/kmehr/htm/spec-msg22.htm

	copy). Kmehr-Bis certified IT systems should use this information to route the message. Recipients are built in the same way as sender.

2. Le **folder** décrit les données obligatoires suivantes :

Data	Purpose
Patient	The folder must contain one patient element. In the future we could support other subjects of care (animal, emergency case, ...).
Folder	The folder must contain at least one **transaction**.

3. La **transaction** spécifique au SumEHR décrit les données suivantes :

Data	Purpose
Adr	to specify an *adverse drug reaction*
allergy	to specify an *allergy*
Socialrisk	to specify *social risk factors*
Risk	to specify *other risk factors*
Gmdmanager	to identify the *manager of the global medical record*, using a healthcare party element.
Contactperson	to specify a *contact person*
Ntbr	to specify if the patient is *not to be resuscitated*, using a boolean content. The patient should not be resuscitated if the value is 'true'.
Bloodtransfusionrefusal	to specify if the patient *refuses blood transfusion*, using a boolean content. The patient should not receive blood if the value is 'true'.
Vaccine	to describe each *vaccine administration*
medication	to describe the *actual medications* (one item per product)
healthcareelement	to describe the *actual problems* (one item per problem)
healthcareelement	to describe the patient's *relevant antecedents* (one item per antecedent)

2. MDM : Les concepts

Ce chapitre a un double objectif : (1) présenter les concepts fondamentaux de l'approche MDM (Section 2.1) et (2) décrire et analyser les différentes architectures MDM proposées dans la littérature (Section 2.2).

Dans un premier temps, nous décrirons les caractéristiques principales des données de référence (Section 2.1.1) et les principes sous-jacents à leur gestion (Section 2.1.2). Ensuite, nous introduirons les trois piliers fondamentaux de l'approche MDM : la Data Governance (Section 2.1.3), la Data Integration (Section 2.1.4) et la Data Quality (Section 2.1.5). Enfin, nous soulignerons l'originalité de l'approche MDM (Section 2.1.6).

Dans un second temps, nous présenterons et analyserons les différents avantages et inconvénients des trois architectures MDM : le répertoire virtuel (Section 2.2.1), la centralisation (Section 2.2.2) et la coopération (Section 2.2.3). Enfin, nous proposerons différents critères pour faciliter la sélection d'une architecture MDM (Section 2.2.4)

2.1. Les concepts fondamentaux

2.1.1. Qu'est ce qu'une donnée de référence ?

Assurer la synchronisation et l'intégration des données, soit en mode batch soit en temps réel, a un coût non négligeable et il est illusoire de vouloir appliquer ces principes à l'ensemble de ses données. Il faut se concentrer sur un sous-ensemble de celles-ci appelées **données de référence ou master data**.

Une donnée de référence est une information de base, fondamentale pour l'activité de l'entreprise, et partagée ou dupliquée dans plusieurs systèmes. Cette donnée métier doit être identifiable et reconnue comme telle partout dans l'organisation, quel que soit le service qui en est responsable, le système d'information, le serveur ou le logiciel qui l'héberge, la traite ou l'enregistre, la division ou la filiale qui la produit. Les données de référence décrivent généralement des objets métier tels que « client », « produit », « fournisseur », « adresse », « employé », etc. Traditionnellement, elles s'opposent aux données dites transactionnelles qui se réfèrent aux événements relatifs à ces objets métier. Elles possèdent un long cycle de vie et sont sujettes aux changements.

Néanmoins, délimiter l'ensemble des données à inclure dans une solution MDM n'est pas évident. Différents facteurs aussi bien économiques qu'organisationnels

entrent en jeu. Le premier facteur est cependant lié au flou entourant la notion même de « donnée ». Principalement, on distingue :

- **Les données structurées et non structurées.** Les données non structurées regroupent l'ensemble des données brutes se trouvant dans des e-mails, des procès-verbaux de réunion, des spécifications de produits, des articles, des rapports d'étude, etc. À l'inverse, les données structurées, conservées dans des bases de données relationnelles ou liées à une application métier, sont clairement identifiées et leur structure explicitée.
- **Les données transactionnelles et non transactionnelles.** Les données transactionnelles sont relatives aux ventes, livraisons, envois, réclamations ou tout autre évènement concernant un objet métier. À l'inverse, les données non transactionnelles concernent uniquement les objets métiers.
- **Les données hiérarchiques ou relationnelles** sont des données structurées et non transactionnelles décrivant les relations existant entre données. Ces relations expriment généralement des dépendances issues du monde réel, tels que la structure organisationnelle d'une compagnie ou des contraintes métier entre différents objets ou évènements métier.
- **Les méta-données[4].** Les méta-données sont des données sur des données. Elles déterminent par exemple la structure des données au travers de fichiers XML, des colonnes d'une table dans une base de données, etc.

Suivant ces distinctions, les **données de référence** peuvent être associées à des donnés structurées, non transactionnelles et non relationnelles qui décrivent les principaux objets métier. Parfois, les référentiels de données sont également utilisés pour échanger des méta-données (ex : glossaire). Le paradoxe est que ces méta-données deviennent alors des données pour le référentiel.

Habituellement, les données de référence concernent des personnes (citoyens, employés, consommateurs, patients, etc.), des objets (produits, immeubles, etc.), des lieux (succursales, bureaux, pays, etc.) ou des concepts (ventes, contrats, licences, fraudes, etc.).

Cependant, gérer des données de référence génère des coûts supplémentaires non négligeables. En conséquence, tous les objets métier ne doivent pas être pris en compte et doivent donc être sélectionnés avec précaution. Les principales caractéristiques d'une donnée de référence sont qu'elle est partagée et/ou échangée avec des tiers, qu'elle possède une haute valeur ajoutée et que sa (ses) source(s) authentique(s) est (sont) reconnue(s) (voir Section 3.1.1).

2.1.2. Qu'est ce que la gestion des données de référence ?

La gestion des données de référence ou le Master Data Management (MDM) n'est ni une technologie ni un logiciel, mais une démarche qui met en œuvre des procédures durables (la gouvernance des données). Cette gouvernance est assurée par une organisation de circonstance, composée d'individus aux tâches précises,

[4] Toutefois, cette notion de méta-données est toute relative car si on remonte d'un niveau « méta », une méta-donnée peut devenir une donnée.

et assistée par des outils dédiés en vue d'améliorer la qualité et le partage des données transversalement à l'organisation.
Il n'existe pas de définition de l'approche MDM qui soit communément acceptée par l'ensemble de la communauté MDM. Différentes définitions existent, en voici deux habituellement citées :

> ***Définitions***
>
> *"MDM is the authoritative, reliable foundation for data used across many applications & constituencies with the goal to provide a single view of the truth no matter where it lies". [MDM Institute, 2009]*
>
> *"Master Data Management (MDM) is a discipline in which the business and the IT organization work together to ensure the uniformity, accuracy, semantic persistence, stewardship and accountability of the enterprise's official, shared master data. Organizations apply MDM to eliminate endless, time-consuming debates about "whose data is right," which can lead to poor decision making and business performance". [Gartner]*

Évidemment, ces définitions prêtent à controverse. Par exemple, il est difficilement imaginable de fournir une vue unique de la vérité. La volonté serait plutôt d'expliciter une vue commune entre les différents acteurs qui pourra ensuite être contextualisée par ceux-ci suivant leur besoins spécifiques. De plus, les débats concernant la validité et/ou l'authenticité des données ne vont pas disparaître comme par enchantement mais seront rationalisés au sein de l'organisation gérant le référentiel de données.

À ce stade, il est important de distinguer l'approche MDM d'autres approches qui partagent les mêmes idées et principes. À première vue, tous ces acronymes semblent répondre à la même problématique. Néanmoins, cette multiplication de terminologies conduit souvent à de nombreuses confusions. Il est donc indispensable de déterminer ce qui est couvert par chacune de ces approches et s'intéresser à la véritable valeur ajoutée du MDM.

Le MDM est une approche définissant un ensemble de bonnes pratiques et de moyens facilitant la gestion à la fois opérationnelle et analytique des données de référence de manière générique et transversale aux applications. La solution préconisée est de centraliser les données dans un référentiel maître et indépendant des systèmes applicatifs afin d'en garantir sa pérennité et de mutualiser les efforts de contrôle et d'amélioration de la qualité. L'originalité de l'approche MDM est donc de regrouper un ensemble de démarches et d'outils préexistants afin de centraliser et de rationaliser la gestion et le partage des données critiques. L'approche MDM se base principalement sur trois approches fondamentales que sont la « **Data Governance** », la «**Data Quality** » et la « **Data Integration** ».

2.1.3. Data Governance

La Data Governance définit un ensemble de bonnes pratiques et de moyens facilitant la gestion des données. Plus les données sont partagées entre différents intervenants plus cette problématique devient cruciale. L'important est de gouverner ses données et d'éviter l'anarchie. Mais qu'entend-on par « gouverner ses données » ? Gouverner c'est essentiellement prévoir, négocier, soulever et résoudre des problèmes :

- **Prévoir**, c'est établir une stratégie de gestion des données déterminant les procédures afin de mettre à jour, partager, assurer la sécurité, contrôler l'accessibilité des données, préserver les droits de leur(s) propriétaire(s), etc.
- **Négocier**, c'est mettre les gens autour de la table afin de trouver des compromis permettant de gérer ces données tout en respectant les exigences des fournisseurs et consommateurs de données ainsi que les réglementations en vigueur.
- **Soulever des problèmes**, c'est être capable de détecter des anomalies relatives aux données et à leur utilisation.
- **Résoudre les problèmes**, c'est être capable de prendre les mesures nécessaires afin de corriger ces anomalies ou d'en diminuer au maximum les effets négatifs.

Définition

"Data Governance is the formal orchestration of people, process, and technology to enable an organization to leverage data as an enterprise asset". [MDM Institute, 2009]

La gouvernance des données consiste principalement à superviser l'exploitation des données et englobe les éléments qui permettent de gérer de manière optimale les dimensions qualité, disponibilité, sécurité et conformité réglementaire des données. Les principaux moyens préconisés sont :

- la création d'un comité de pilotage et de surveillance,
- l'identification des « propriétaires », des « fournisseurs » et des « consommateurs » des données,
- la définition des rôles et responsabilités,
- la description des données (glossaire métier, dictionnaires, méta-données,…),
- la définition des politiques et des processus de gestion des données,
- l'établissement des normes et des procédures pour l'utilisation des données,
- la mise en œuvre des vérifications et des contrôles.

La gouvernance de données est principalement une démarche organisationnelle fondée sur des consensus visant à améliorer l'utilisation et l'utilisabilité des données. Elle doit être systématique et rigoureuse essentiellement dans le cadre de la gestion des données de référence.
En effet, le caractère transverse des données de référence implique que leur périmètre d'usage s'étende potentiellement à un grand nombre d'applications hétérogènes. L'objectif est de maximiser la valorisation des données de référence dans le temps, notamment en définissant les politiques d'intégration des données (Data Integration) et d'amélioration continue de leur qualité (Data Quality).

2.1.4. Data Integration

L'intégration de données définit un ensemble de processus permettant de migrer, combiner et consolider des données provenant de différentes parties du système d'information [Bouzeghoub, 09]. L'intégration de données consiste

habituellement à extraire des données de différentes sources (bases de données, fichiers, applications, services web, emails, etc.), à leur appliquer des transformations (jointures, lookups, déduplication, calculs, etc.), et à envoyer les données résultantes vers les systèmes cibles. Suivant le niveau d'intégration demandé et le niveau de disponibilité des données, différentes technologies peuvent être utilisées (voir Figure 2).

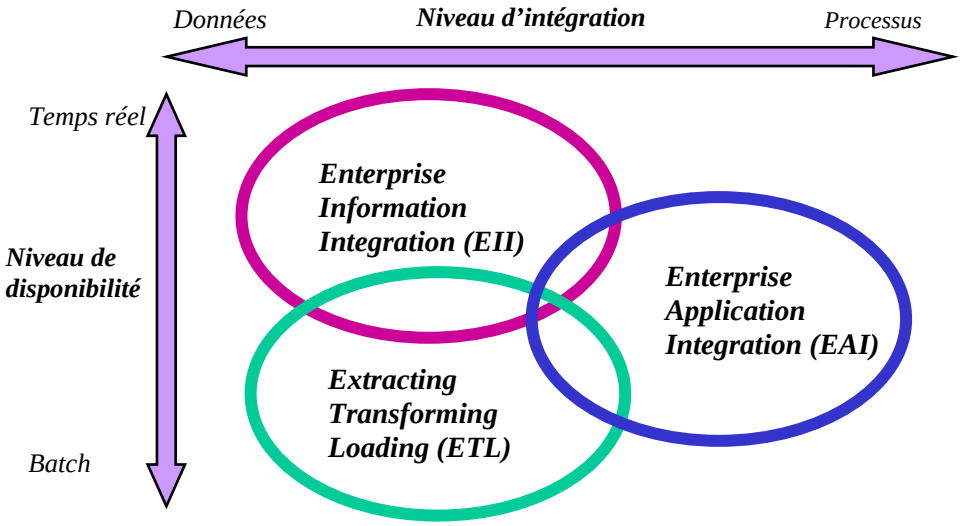


Figure 2 : Intégration de données : ETL, EII et EAI.

1. **Extraction Transformation Loading (ETL)**. Cette technologie permet d'effectuer des synchronisations massives de données d'une base de données vers une autre. Différentes techniques d'extraction, de réplication, de transformation et de conversion de données sont utilisées afin de (re)peupler différentes bases de données.

   Néanmoins, l'extraction des données avec des outils ETL est périodique (mode batch) et est difficilement utilisable du point de vue opérationnel. On l'utilise principalement à des fins décisionnelles pour alimenter des entrepôts de données (data warehouses). Cependant, un entrepôt de données est rarement considéré comme une base de données opérationnelle puisque les données sont rarement à jour.

   Toutes les données sont stockées, consolidées et historisées, ce qui réduit drastiquement l'efficacité et empêche une gestion en temps réel. La vérification de la qualité des données est souvent réalisée a posteriori, c'est-à-dire que les erreurs ne sont pas corrigées à la source mais ont le temps de se propager dans les systèmes.

   Les données contenues dans le data warehouse ne sont pas opérationnelles, elles ne sont utilisables que par des applications d'analyse de type Business Intelligence à des fins purement décisionnelles.

   Les *avantages* de cette technique sont la forte intégration des données, la grande disponibilité des indicateurs et agrégats et son adéquation avec des outils d'analyse. Les *désavantages* de cette technique sont les coûts souvent très élevés en matériel, logiciel, maintenance et service ainsi que le délai de rafraîchissement trop long.

2. **Enterprise Information Integration (EII).** Cette technologie permet d'interroger plusieurs sources de données afin d'obtenir une vue unifiée

et intégrée des données de l'entreprise. Le rôle du serveur EII est de faire la médiation entre les différentes sources de données en décomposant correctement les requêtes qu'il reçoit et en les redirigeant vers les différentes bases de données concernées pour ensuite rassembler les différents résultats renvoyés.

Les solutions EII permettent l'accès en quasi temps réel aux données, contrairement aux solutions ETL qui accèdent aux données de manière périodique. Cependant, dans une approche MDM il n'est pas seulement nécessaire de pouvoir décomposer et rediriger de manière intelligente des requêtes, il faut encore créer une vue intégrée de l'ensemble de ses données pour ensuite la partager, la maintenir à jour et améliorer sa qualité.

Les *avantages* de cette technique sont la grande disponibilité des données intégrées et leur fraîcheur. Les hypothèses fortes posées sur la disponibilité des données sources et sur leur qualité constituent le principal *désavantage* de l'EII. En effet, si les données sont de mauvaise qualité à la base ou si leur disponibilité n'est pas garantie, les techniques EII produiront des données dont les problèmes de qualité seront difficilement identifiables.

3. **Enterprise Application Integration (EAI)**. Cette technologie propose une architecture intergicielle (middleware) permettant à des applications hétérogènes de gérer l'échange et la conversion des données en quasi temps réel. Ces applications peuvent être développées indépendamment et peuvent utiliser des technologies différentes.

   Les outils EAI permettent de mettre en place une plateforme d'échange entre les différentes applications afin d'intégrer les processus métier et, dans une certaine mesure, les données. Les nouvelles plateformes d'échange EAI se basent sur des architectures SOA (Service Oriented Architecture) [Bell, 08] et l'utilisation d'ESB (Entreprise Service Bus) qui offrent la possibilité à des applications qui, au départ, n'ont pas été conçues pour travailler ensemble de communiquer par l'envoi de messages.

   Les messages échangés via l'ESB concernent aussi bien des appels à des services métier que des appels de services de consultation ou de modification des données. L'EAI apporte un cadre d'intégration à la fois souple et robuste. Néanmoins, une plateforme d'échange entre applications ne suffit pas, il faut encore définir comment et pourquoi les données vont être échangées et intégrées.

   Les *avantages* de cette technique sont la très grande fraîcheur des données et la réduction des problèmes de latence d'accès aux données grâce à la synchronisation et à la mise à jour des données en temps réel. Le principal *désavantage* de l'EAI est l'illusion qu'il donne de partager des données intégrées. En effet, l'EAI intègre des processus et non pas les données sous-jacentes. D'origine, aucun mécanisme de nettoyage, de réconciliation ou d'agrégation de données n'est prévu.

Une solution MDM combine généralement ces trois techniques (voir Figure 3). Les techniques *EII* sont utilisées comme **requêteur** pour retrouver les différentes données dispersées dans plusieurs bases de données et les intégrer de manière logique. Les techniques *ETL* sont utilisées comme **extracteur** pour récupérer les données dispersées et les intégrer de manière physique dans une nouvelle base de données qui constituera la base de données centrale de référence. Les techniques *EAI/ESB* sont utilisées comme **transporteur** pour échanger les données entre applications consommatrices et/ou fournisseuses de données.

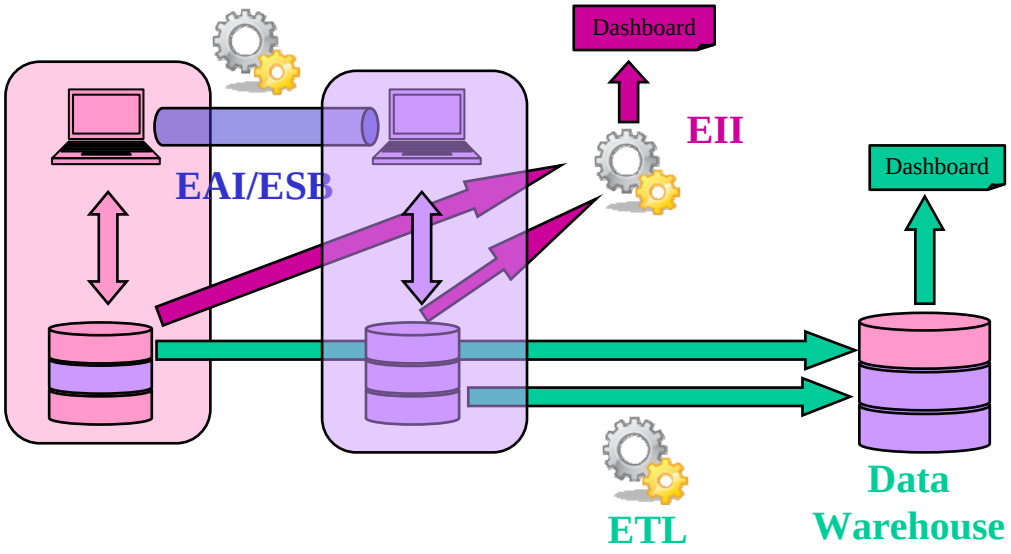


Figure 3 : MDM : Comment combiner ETL, EII et EAI ?

L'***intégration logique*** via un EII permet d'utiliser des données fraîches et complètes à des fins opérationnelles. En ce qui concerne l'***intégration physique***, deux cas de figure sont envisageables :

- Soit l'intégration physique se fait de manière **périodique** et implique la mise à jour complète de la base de données. Ce cas de figure s'inscrit généralement dans le cadre de la constitution d'un entrepôt de données. Cette technologie permet de collecter, de consolider et de stocker des données provenant d'autres bases de données afin d'établir des analyses de tendances sur des historiques de données. Des historiques de données sont en effet maintenus afin de pouvoir suivre l'évolution des données au cours du temps et d'en préserver la qualité. La collecte des données se fait principalement au moyen des technologies ETL.

- Soit l'intégration physique est réalisée de manière **continue** tout au long de la vie du système, pour chaque problème de synchronisation identifié. Des mécanismes sont mis en place afin d'assurer l'intégration et la synchronisation des données en quasi temps réel afin qu'elles puissent être utilisées à des fins opérationnelles (via une architecture SOA par exemple). L'évolution des données de référence est prise en compte dans leur contexte applicatif et à tout moment la synchronisation entre le référentiel et les systèmes sources qui y sont rattachés doit être garantie.

2.1.5. Data Quality

La Data Quality définit un ensemble de bonnes pratiques et de moyens en adéquation avec les usages (fitness for use) améliorant la qualité des données stockées dans une base de données ou dispersées dans plusieurs bases de données. Différentes techniques sont utilisées tel que le profiling, le monitoring, la standardisation et le matching des données.

Différentes études[5] sur les concepts, principes et outils liés à la Data Quality ont été menées au sein de la section Recherches de Smals. Ces publications sont consultables en ligne afin d'obtenir de plus amples informations sur cette approche.

Au-delà de la technologie, le contrôle et l'amélioration de la qualité des données (Best Practices[3]) sont des éléments primordiaux à toute approche MDM. Le taux d'anomalies augmente fortement en fonction du nombre de sources de données et du degré d'hétérogénéité de celles-ci. D'un côté, l'efficacité de l'échange des données dépend principalement de la confiance accordée par les consommateurs à ces données. D'un autre côté, la qualité de l'intégration des données dépend largement de la qualité des données de départ. L'intégration de données de mauvaise qualité ne peut générer que des données de mauvaise qualité pour lesquelles il sera plus complexe d'identifier l'origine du problème.

2.1.6. Quelle est l'originalité de l'approche MDM ?

Produire un annuaire de données de référence ou constituer une base de données rassemblant les données de référence ne suffit pas. Il faut pouvoir assurer la maintenance et garantir la qualité des données de référence sur le long terme. Les efforts investis dans la création de données de référence cohérentes et de qualité ne doivent pas être réduits à néant par un manque de bonnes pratiques favorisant la préservation de cette cohérence et de cette qualité.

Dans la majorité des cas, des changements importants doivent avoir lieu au niveau des processus métier et des outils adéquats doivent être mis en place. Néanmoins, les défis les plus importants sont bien souvent plus organisationnels que techniques.

L'exploitation des données de référence nécessite de les partager et de les tenir à jour de manière collaborative. Les données de référence passent du statut de données stockées dans de simples fichiers plats difficilement exploitables au statut de données réellement valorisables pour l'organisation. En conséquence, la gestion de ces données doit être rationalisée. Il est primordial de faciliter leur échange ainsi que de contrôler et d'améliorer leur qualité de manière continue.

Pour ce faire, l'approche MDM met en avant quatre principes fondamentaux :

1. Les données étant généralement dispersées et donc difficilement valorisables, il est indispensable de pouvoir **créer une vision unique** de ces données. Unique ne veut pas dire qu'elle est imposée à tous, mais que l'ensemble des acteurs a trouvé un consensus sur le contenu, le format et la sémantique des données échangées. C'est une vision commune et neutre qui a l'avantage d'être explicite et de permettre l'intégration des données. Chacun est libre de la contextualiser à sa guise suivant ses besoins métier.

   Par exemple, les données concernant le montant du salaire d'un employé sont stockées dans et utilisées par différentes applications chacune ayant son propre format et sa propre interprétation de la notion de salaire. Si maintenant on explicite que le salaire mensuel de base sera le format standard utilisé par le référentiel, celui-ci pourra :

   a. **Consolider** les données des différents fournisseurs concernant les salaires en

[5] Smals Research Publication : Data Quality - Best Practices (2006) and Data Quality Tools - Évaluer et améliorer la qualité des données (2007), consultables à l'adresse suivante: http://documentation.smals-mvm.be/.

i. les normalisant sous le format standard du référentiel.

ii. les comparant afin d'identifier et de résoudre les incohérences. Dès lors, on devrait pouvoir comparer les salaires mensuels de base avec les salaires nets mensuels, avec les salaires nets annuels comprenant les avantages sociaux (chèques repas, assurance, etc.).

b. **Partager** cette vue consolidée avec les différents consommateurs. Deux cas de figures sont alors envisageables :

i. Soit les consommateurs devront contextualiser eux-mêmes cette donnée, suivant leurs besoins. Cette contextualisation peut, par exemple, consister en une transformation du salaire mensuel de base en salaire net annuel.

ii. Soit le référentiel transformera lui-même la donnée avant de la communiquer au consommateur suivant le format et l'interprétation que le consommateur aura préalablement spécifiés.

2. Une fois que la vision consolidée est jugée exploitable il faut pouvoir la partager. La création de cette vue consolidée est une étape préalable qui peut se révéler très complexe, surtout dans un contexte multi-organisationnel. Elle requiert un travail d'harmonisation qui peut prendre de nombreuse années (ex : Harmonisation du droit social). Dans la Figure 4, nous illustrons comment le partage de données entre différentes applications via un référentiel de données peut être réalisé.

Ces applications (représentées au bas de la Figure 4), peuvent jouer deux rôles différents. Elles peuvent devenir fournisseuses et/ou consommatrices de données. D'un côté (gauche), nous représentons les applications fournisseuses de données qui alimentent le référentiel avec des données dites « authentiques ». De l'autre côté (droit), nous représentons les applications consommatrices de données qui consultent les données via le référentiel selon deux modes :

a. Le mode **pull** où les consommateurs doivent lancer une demande explicite de consultation pour obtenir une donnée.

b. Le mode **push** (Publish/Subscribe) où le référentiel prend l'initiative d'avertir les consommateurs à chaque fois qu'une donnée pour laquelle ils se sont inscrits a été modifiée. Ce mode est généralement préféré pour des données critiques pour lesquelles les consommateurs de données désirent être immédiatement avertis de tout changement. Néanmoins, ce mode peut rapidement atteindre ses limites si l'afflux de données échangées devient trop important.

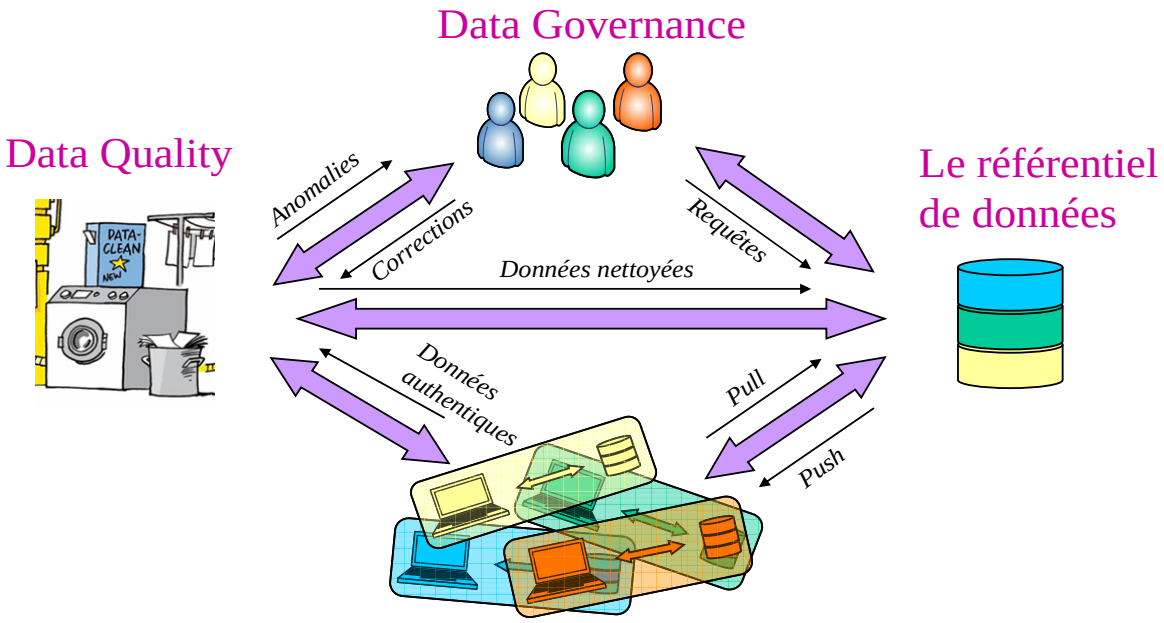


Figure 4 : MDM : Partage des données de référence

Dans le cadre du partage des données l'approche MDM met en avant les notions de **référentiel** de données, de **qualité** des données et de **gouvernance** des données.

- Pour partager cette vision unique, elle doit être rendue consultable. Dans la Figure 4, **le référentiel** est illustré par une base de données. Néanmoins, le référentiel n'est pas nécessairement une base de données physique, il peut aussi consister en une base de données logique. En effet, deux possibilités sont envisageables :

  a. Soit la vision unique est stockée dans une nouvelle base de données spécifiquement dédiée à la gestion des données de référence (*Intégration physique*).

  b. Soit la vision unique est consultable via un annuaire de données qui fait le lien entre les consommateurs et les fournisseurs de données (*Intégration logique*). Les données restent stockées dans les bases de données des fournisseurs de données qui déterminent la vue unique.

  Dans les deux cas, le référentiel doit souvent jouer le rôle de médiateur pour faciliter la consolidation des données et les échanges de données entre fournisseurs et consommateurs.

- La **qualité des données** intégrées au référentiel est primordiale car :

  a. toutes les applications consommatrices de ces données doivent pouvoir les réutiliser en toute confiance,

  b. la propagation d'erreurs pourrait mettre à mal l'ensemble des applications qui consultent ces données.

  Tout comme dans le cas d'une pandémie, la contamination des données peut se propager très rapidement à l'ensemble du système.

La vitesse de propagation d'une pandémie liée à la mauvaise qualité des données dépend principalement du nombre d'applications qui vont consulter les données contaminées et de la fréquence de consultation. Potentiellement, le risque est très élevé car une donnée peut-être partagée entre de nombreuses applications en quasi temps réel.

La situation idéale serait d'identifier les problèmes de qualité à la source et de les corriger avant de les diffuser. Cependant, cette solution n'est pas facilement réalisable en pratique car, dans l'absolu, on ne peut pas garantir une qualité des données irréprochable. Poser l'hypothèse que l'on partage des données qui seront toujours d'un niveau de qualité élevé n'est pas réaliste. La perfection n'existe pas, surtout en qualité de données, même si on doit tendre vers celle-ci. C'est pourquoi, lorsque l'on partage des données, on doit toujours tenir compte du fait que les données échangées ne seront jamais parfaites. Des mécanismes doivent être mis en place afin d'éviter la propagation d'erreurs a priori et de corriger celles a posteriori.

Avant d'être intégrées au référentiel, les données doivent être nettoyées via des outils d'intégration et d'amélioration de la qualité des données. La correction des anomalies détectées impliquera dans les cas non triviaux l'intervention de personnes métier qui devront soit apporter les corrections nécessaires, soit accepter certaines anomalies dont le coût de correction serait trop élevé par rapport au bénéfice réalisé.

- Ces échanges de données devront être chapeautés par une organisation en charge de la **gouvernance des données** c'est-à-dire du bon déroulement des échanges de données que ce soit au niveau de la gestion des anomalies, de la gestion des accès au référentiel ou du respect de la vie privée.

3. Créer et partager la vue unique ne suffit pas, il faut aussi pouvoir la gérer et la faire évoluer dans le temps en fonction des besoins métier. Un référentiel n'est pas qu'une simple banque de données logique ou physique, c'est aussi une application qui permet de gérer les échanges de données entre les différentes applications (fournisseuses et consommatrices) qui y sont connectées.

   Par exemple, le référentiel doit être capable de garder pour chaque donnée de référence les liens entre les identifiants (primary key) utilisés chez les fournisseurs, chez les consommateurs et dans le référentiel (si intégration physique).

   La figure 5 illustre le référentiel en tant qu'application proprement dite échangeant des données via un middleware sous un format standardisé avec d'autres applications. Ce type de référentiel permet de mutualiser les efforts en terme de :

   a. Traduction des données (mappings).
   b. Standardisation des données.
   c. Synchronisation des données.
   d. Validation et de correction des données.
   e. Gouvernance transversale et collaborative.

   Une difficulté majeure est la gestion des modèles de données qui évoluent dans le temps aussi bien au niveau du référentiel que des fournisseurs de données. Les données de référence sont souvent valorisées différemment, selon des versions successives.

Dans la Figure 5, on imagine facilement que les formats des données échangées (#1, #2, #3) ne vont pas rester inchangés et qu'ils évolueront forcément dans le temps en fonction des besoins. Ce constat s'applique aussi au standard du référentiel, même si, par définition, il évoluera moins vite. Cette situation apparaît régulièrement lorsqu'une nouvelle réglementation entre en vigueur et qu'une période de transition, de cohabitation est nécessaire avec l'ancienne réglementation. Dès lors, la gestion des versions ne doit pas se limiter uniquement au contenu des données mais doit aussi concerner les méta-données (modèles de données). On doit être capable de connaître la valeur d'une donnée il y a trois mois, mais également le format des données utilisé à l'époque.

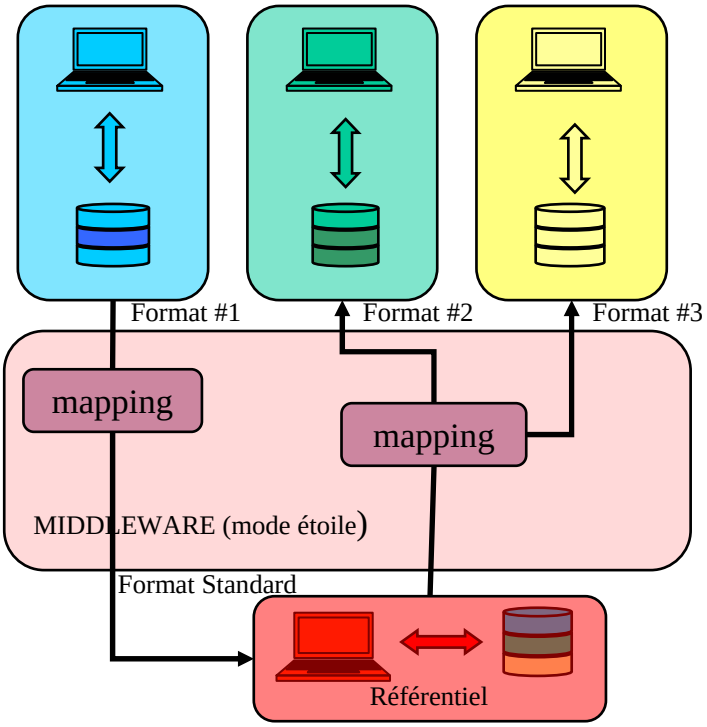


Figure 5 : MDM : Création d'un référentiel

Par exemple, si une gestion de versions efficace (Data Versioning [Elmasri, 06]) n'est pas mise en place et qu'une réglementation impose l'ajout d'une colonne dans une table, cela implique la mise à jour des anciennes données et pose la question des valeurs qui devront être insérées dans cette nouvelle colonne. À l'inverse, lorsque l'on supprime une colonne dans une table, les données seront elles simplement supprimées ou doivent-elles être préservées ?

Une technique utilisée pour résoudre ce type de problème est l'utilisation d'History Tables. Une History Table est créée pour chaque nouveau modèle de données tel qu'il était d'application au moment de l'insertion de lignes dans la table. La contrepartie de cette gestion plus fine des versions est la complexité accrue des requêtes de consultation des données dites historiques, puisqu'à tout moment il faut connaître quelle History Table correspond à la version qui nous intéresse. Le nombre d'History Tables pouvant croître très rapidement.

2.2. Les architectures MDM

Les architectures MDM sont des architectures d'échange qui déterminent où les données de référence vont être stockées et comment elles vont être partagées entre les différents fournisseurs et consommateurs de données. Chaque architecture possède ses avantages et ses inconvénients. Le choix d'une architecture est souvent délicat, car ce choix peut évoluer au cours du temps et dans certaines situations une combinaison de différentes architectures peut se révéler nécessaire.

2.2.1. Répertoire Virtuel (Registry)

La première architecture consiste à créer un **annuaire de données** dont le rôle principal est de rediriger les requêtes des consommateurs de données vers les fournisseurs de données adéquats (voir Figure 6).

Son activité principale consiste à gérer un index reprenant les clefs d'accès aux données sources. Cette architecture offre une plateforme d'échange qui s'occupe d'aiguiller les messages vers les fournisseurs de données sans tenir compte de leur contenu. Cette plateforme fonctionne principalement en *mode Pull*.

L'objectif est de créer un point d'entrée unique auquel se référer pour consulter les données de référence et ainsi permettre leur **intégration logique** alors qu'elles demeurent dans les applications sources d'origine. L'annuaire de données tient à jour une liste des différents pointeurs permettant de localiser les données de référence. L'accès aux données sera autorisé ou non suivant les politiques de gestion des accès établies au niveau du référentiel mais également au niveau des applications sources.

Un annuaire de données est généralement doté de fonctionnalités plus avancées que le simple aiguillage. Un référentiel de données a la capacité de décomposer et de rediriger les requêtes des consommateurs vers les fournisseurs de données adéquats, pour ensuite rassembler les différents résultats obtenus et les renvoyer au consommateur initial. Il joue alors le rôle d'intermédiaire.

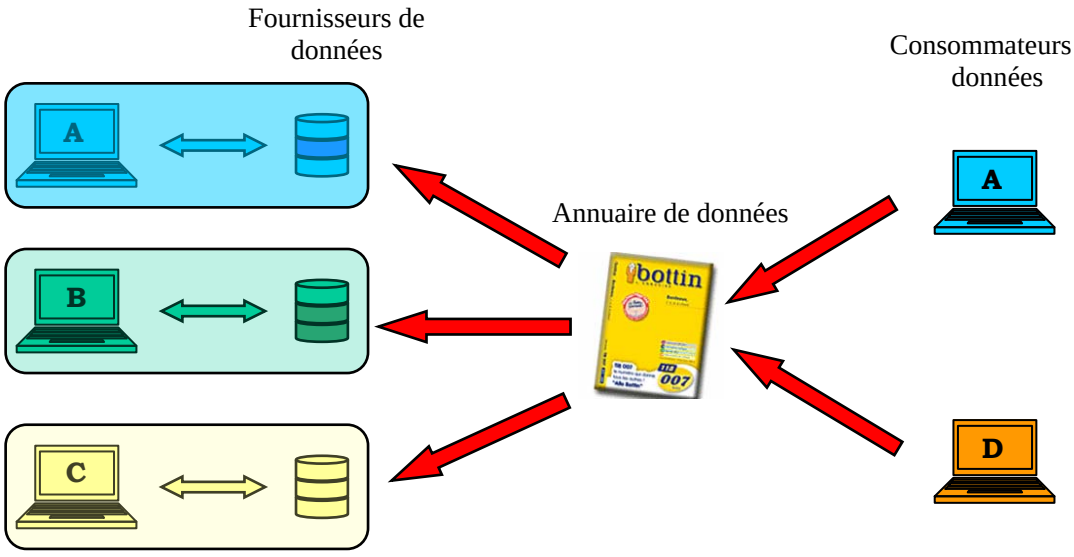


Figure 6 : MDM : Répertoire virtuel

Avantages

Le principal *avantage* de cette architecture est sa mise en place relativement transparente pour les utilisateurs finaux. Les changements à apporter aux applications existantes sont minimaux :

- Les fournisseurs de données continuent à gérer leurs données avec les mêmes applications et les mêmes bases de données qu'auparavant.
- L'annuaire s'occupe uniquement du transfert des données et non du contenu.
- Aucune migration des données n'est nécessaire, l'application doit simplement pouvoir répondre à des requêtes de consultation de données.
- Les systèmes sources restent complètement indépendants et l'utilisation de la plateforme d'échange est facultative.
- Sans annuaire de données l'application peut toujours consulter ses propres données et en échanger sans passer par l'annuaire central de données. Dans ce cas de figure, le consommateur de données est limité à son propre annuaire de données.

Inconvénients

Le principal *inconvénient* de cette architecture est qu'elle cache la nécessité d'avoir un modèle commun des données de référence. L'absence de ce modèle commun induit un manque d'uniformisation des fonctions de gouvernance des données de référence. En effet, chaque application fournisseuse ou consommatrice conserve ses propres modèles de données, ses propres règles de validation, de correction et ses propres outils spécifiques de gestion des données.

De plus, cette approche ne décloisonne que très légèrement les données en érigeant de fragiles passerelles entre les silos de données. En effet, chaque application garde, potentiellement, une copie des données de référence qui peuvent évoluer de manière différente au cours du temps. À terme, on arrive à une situation où il devient impossible d'identifier la source de la donnée de référence. Différentes versions d'une même donnée de référence coexistent avec des incohérences et des contradictions.

Dès lors, un autre problème fondamental est le manque de synchronisation de ces données. On pourrait imaginer un patient fervent adepte du shopping thérapeutique qui consulte de nombreux médecins en leur fournissant des informations contradictoires sur son état de santé. Si le médecin ne prend pas lui-même l'initiative de confronter ses données avec d'autres sources d'information, les incohérences pourraient persister. L'annuaire central n'avertira pas des changements qui ont été apportés par d'autres médecins, il permet simplement d'identifier les endroits où des informations sur un patient peuvent être demandées. Le médecin généraliste aurait la charge de demander ces informations, de les trier, d'évaluer leur pertinence, de relever et de résoudre les incohérences. Malheureusement, une fois que ces incohérences auront été résolues, elles ne seront pas automatiquement diffusées aux autres médecins ayant un lien thérapeutique avec ce patient. En cas de doute, chaque médecin devra contacter la source authentique.

Ce type d'architecture implique un niveau de gouvernance minimale pour les données et augmente ainsi le risque d'échanger des données de manière anarchique. L'implication des médecins et leur rigueur seront des facteurs déterminants dans le succès d'une architecture de ce type.

2.2.2. Centralisation (Repository)

À l'autre extrême, se trouve l'architecture dite de « **centralisation** ». Cette architecture n'intègre plus les données logiquement mais physiquement. Les données de référence et les attributs nécessaires au bon fonctionnement des applications sont centralisés dans une base de données unique (voir Figure 7). Cette DB centrale devient la seule source de vérité. Une seule et même application peut être utilisée pour l'acquisition, la validation et la consultation des données. Dès lors, fournisseurs et consommateurs de données utilisant la même application centrale pour gérer ou interroger les données de référence.

L'échange des données peut s'effectuer suivant les *modes Pull ou Push*. C'est-à-dire que :

- soit les consommateurs doivent faire explicitement la demande pour obtenir l'information (mode Pull),
- soit la DB centrale prend l'initiative d'avertir les consommateurs lorsqu'une donnée a été modifiée (mode Push).

Dans le cadre des dossiers patients électroniques minimaux, ceux-ci seraient centralisés en un endroit unique. Les médecins n'auraient plus accès en local à leurs dossiers patients mais devraient se connecter à une base de données (DB) centrale. Soit une application spécifique est fournie aux prestataires de soins afin de pouvoir modifier et accéder en temps réel aux dossiers patients stockés dans la DB centrale. Soit les applications existantes qui accèdent, créent et assurent la maintenance des dossiers patients sont modifiées afin d'utiliser directement les données de la DB centrale.

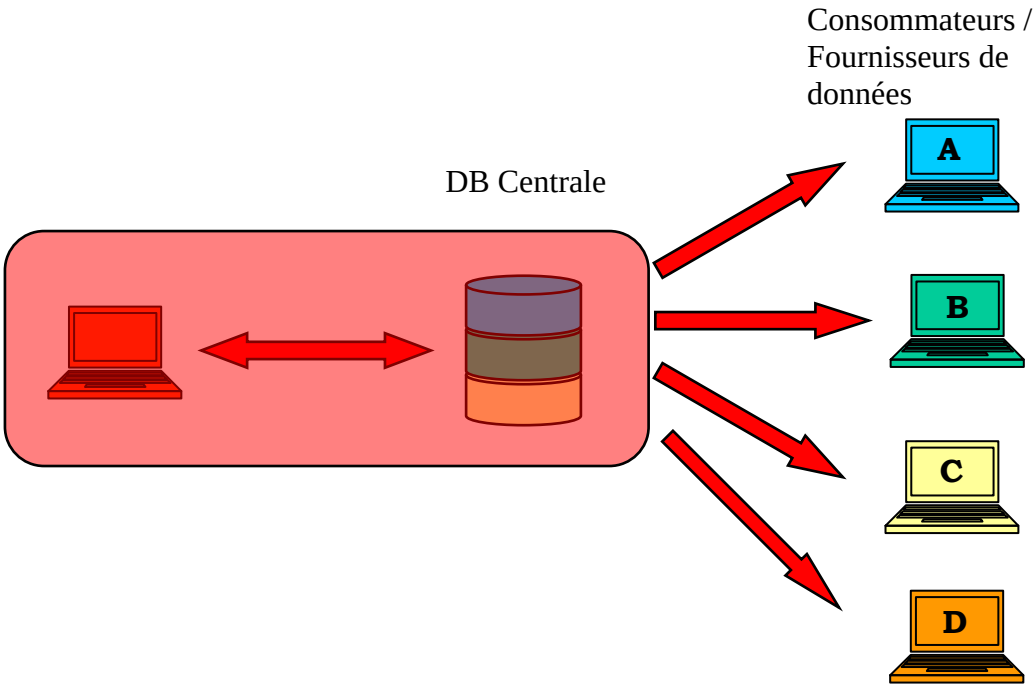


Figure 7 : MDM: Architecture de Centralisation

Avantages

Le principal avantage de cette solution est d'avoir une application unique pour gérer les données de référence, les rendre accessibles et ainsi éviter la duplication et l'hétérogénéité des données. L'objectif est d'éliminer les données dupliquées entre différentes applications (chez différents médecins) et ainsi éviter de devoir

synchroniser ces différentes copies. La DB centralise la gestion des requêtes, les transformations, les enrichissements, l'intégration de données et la gestion des anomalies. Les mêmes processus de création, de modification et de correction des données de référence sont utilisés par tous.

À chaque patient serait donc associé un et un seul dossier patient auquel les prestataires de soins auront accès via un point d'entrée unique. On évite ainsi que plusieurs dossiers patients circulent pour le même patient avec des informations incomplètes voire contradictoires. Lorsqu'un changement est apporté au dossier patient, il est directement visible, au même moment, par tous les médecins autorisés à consulter ce dossier ou une partie de ce dossier. Centraliser les dossiers patients permet d'éviter la duplication des données, de diminuer la charge de travail liée à la gestion des données et d'assurer la sécurité et le contrôle des accès aux données de manière réellement efficace. Une architecture centralisée offre la possibilité d'une gouvernance forte des données avec des procédures uniformisées beaucoup plus difficiles à mettre en œuvre lorsque les données sont éparpillées dans plusieurs applications.

Inconvénients

Cette centralisation entraîne également des inconvénients majeurs. Les plus importants sont le coût de la mise en place de cette architecture et l'acceptation du principe de la centralisation des données dans une base de données unique (syndrome du Big Brother).

Avant même de pouvoir centraliser les données, il est indispensable de les consolider. Cette consolidation est une tâche complexe nécessitant d'identifier tous les fournisseurs de données et de résoudre les conflits qui vont inévitablement apparaître. L'identification et la résolution de ces conflits doivent être supervisées par le métier conformément à l'existant. Par exemple, la fusion de deux dossiers médicaux concernant un même patient peut mettre en lumière des différences aussi bien au niveau des coordonnées du patient que de la liste des substances auxquels il a déjà présenté des réactions allergiques.

Les données de référence doivent également être migrées vers la DB centrale. Une refonte majeure des applications existantes est donc nécessaire. Cette refonte n'est pas toujours possible car :

- le code source n'est pas forcément disponible,
- les licences n'autorisent pas la modification du code,
- le coût du reengineering des applications est trop important.

En règle générale, la complexité de la DB centrale et principalement de son modèle de données risquent de devenir difficile à maîtriser. Ce modèle aura tendance à grandir de manière démesurée afin de prendre en compte les exigences de chaque consommateur de données. La structure du contenu d'un dossier patient doit être définie au préalable, de manière claire et standardisée. Une évolution anarchique de cette structure doit être prohibée. L'intérêt collectif doit primer sur les intérêts individuels. La définition de cette structure de données standard est un problème plus organisationnel que technique.

Ce type d'architecture entraîne aussi des conséquences importantes sur l'infrastructure qui transporte et stocke les données de référence. La DB devient une base de données opérationnelle qui sera interrogée simultanément et en permanence par de nombreuses applications. L'utilisation de la DB centrale et de la plate-forme d'échange deviennent obligatoire pour les applications consommatrices et fournisseuses de données. Dès lors, cette plateforme doit être capable d'assurer un haut niveau de disponibilité des données et doit pouvoir supporter une forte augmentation du trafic sur le réseau suivant le type et la quantité de données échangées. Dans cette situation, les consommateurs et

fournisseurs de données deviennent fortement dépendants de la DB centrale et de la plateforme d'échange de données. Si la DB centrale est corrompue ou si la plateforme d'échange est surchargée ou indisponible, la consultation ou la création de dossiers patients deviendra impossible.

2.2.3. Coopération (Hybrid)

Au croisement de ces deux architectures est née une architecture intermédiaire dite de « **coopération**». Les objectifs de cette architecture sont :

- de diminuer la charge de travail liée à la mise en place d'une architecture de centralisation,
- de diminuer la complexité liée à la gestion et à l'évolution d'un répertoire virtuel,
- de minimiser les impacts sur les applications fournisseuses et
- de créer une DB commune et neutre contenant une version intégrée des données de référence (voir Figure 8).

Les fournisseurs de données doivent se synchroniser avec cette DB commune et l'avertir de tout changement apparu sur les données de référence (*mode Push*). Par contre, la consultation des données peut s'effectuer en *mode Push ou Pull :*

- Soit la DB commune avertit ses consommateurs des changements apportés aux données de référence qui les intéressent (mode Push).
- Soit les consommateurs doivent régulièrement faire des demandes de consultation des données afin de vérifier si leurs données sont à jour (mode Pull).

Les données de référence sont dupliquées mais un point d'entrée unique existe pour consulter une version intégrée des données de référence. Les fournisseurs de données gardent le contrôle sur les données qu'ils fournissent à la DB commune et peuvent devenir transparents pour les consommateurs qui désire consulter la donnée avant de savoir qui la fournit.

Dans le cadre de la gestion des dossiers patients électroniques minimaux, cela impliquerait notamment que :

- la DB commune serait gérée par les médecins généralistes.
- les prestataires de soins synchroniseraient leurs données médicales avec le médecin généraliste qui se chargerait via le référentiel :
    - d'intégrer les données provenant de différentes sources,
    - d'éliminer les incohérences,
    - de mettre à disposition et de diffuser ces informations de meilleure qualité.

Les prestataires de soins voulant consulter un dossier patient pourront s'adresser soit directement au médecin généraliste soit comme auparavant à un prestataire de soins spécifique. Cependant, si on ne passe pas par la DB commune (c'est-à-dire, le médecin généraliste), les informations reçues n'auront pas forcément été intégrées avec d'autres données médicales concernant le même patient. Consulter le point central permet d'éviter un échange anarchique des données et d'obtenir une vue globale et unifiée sur celles-ci.

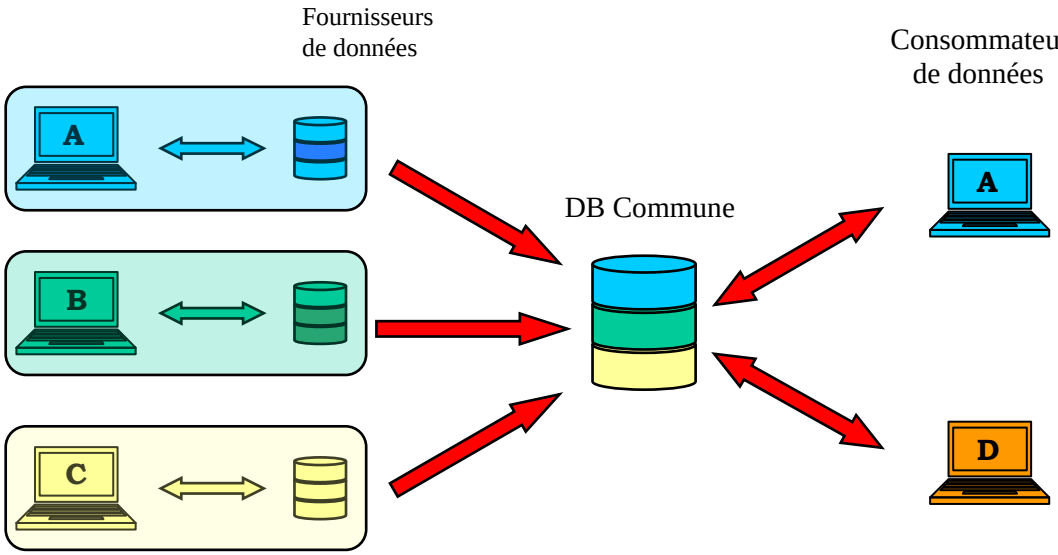


Figure 8 : MDM: Architecture de Coopération

Avantages

Le principal avantage de cette solution est de mettre à disposition des consommateurs de données une source de référence regroupant des données intégrées. L'objectif étant d'avoir le moins d'impact possible sur les systèmes existants tout en améliorant la qualité des données échangées. Une grande flexibilité est accordée au niveau de la gouvernance des données.

Chaque consommateur de données, suivant les autorisations qui lui ont été octroyées, a donc la possibilité de demander une version intégrée des données à la DB Commune. Cette intégration des données est réalisée soit par l'organisme neutre gérant la DB Commune, soit directement par les fournisseurs de données. L'aspect collaboratif est très important ici car sans la participation des fournisseurs de données à l'alimentation de la DB Commune, ce référentiel ne constituera qu'un silo de données supplémentaire.

Dans les cas des dossiers électroniques médicaux minimum, cette intégration sera effectuée par les médecins généralistes qui reçoivent les données fournies par les différents hôpitaux et médecins spécialistes. Cependant, dans tous les cas, ce travail d'intégration sera désormais directement profitable à tous les prestataires de soins et permettra ainsi d'améliorer la qualité des données du patient.

Inconvénients

Bien qu'il semble que cette architecture tire profit des avantages des deux architectures précédentes, il est important de souligner ses inconvénients. Premièrement, contrairement aux autres architectures, les données de référence sont une nouvelle fois dupliquées. Ironiquement, cette duplication risque de créer un nouveau silo de données. Deuxièmement, comme dans le cas de l'architecture de centralisation, la complexité de la DB commune dépend de la quantité de données à stocker et du nombre potentiel d'utilisations de ces données. Enfin, la synchronisation des données n'est toujours pas garantie. Néanmoins, cette synchronisation se limite à vérifier la cohérence entre les données des fournisseurs et celles se trouvant dans la DB commune. On évite donc que tous les fournisseurs de données doivent se synchroniser deux à deux comme dans le cas du répertoire virtuel.

2.2.4. Comment choisir son architecture MDM ?

Chacune des trois architectures MDM que nous avons présenté possède des avantages et des inconvénients. En résumé, l'architecture de centralisation semble la plus efficace opérationnellement car les applications consommatrices ont toujours accès à une seule source de données de référence consistante et à jour. Cependant, le coût de la mise en place d'une telle architecture est extrêmement élevé et les impacts sur les applications fournisseuses de données sont souvent trop importants voir même radicaux. Les deux autres architectures autorisent la duplication des données ce qui implique, d'une part, une latence entre la mise à jour des données sources et la mise à jour des données de référence et, d'autre part, l'obligation de mettre en place des techniques complexes assurant la consistance et la synchronisation des données.

Le choix d'une architecture de type **répertoire virtuel** semble le plus indiqué lorsque :

- la gouvernance des données est faible,
- les données de référence évoluent peu,
- le nombre de sources authentiques différentes pour la même donnée est limité et la cohérence entre ces différentes sources est importante,
- le niveau de disponibilité requis pour les données de référence n'est pas primordial,
- la complexité des requêtes de consultation des données de référence est faible,
- un retour sur investissement à court terme est primordial.

Le choix d'une architecture de type **centralisation** semble le plus indiqué lorsque:

- la gouvernance des données est forte,
- le nombre d'applications impliquées dans le projet est limité,
- le contrôle sur les applications fournisseuses est important,
- la contextualisation des données de référence n'est pas un besoin capital pour les applications consommatrices,
- la valeur ajoutée apportée par un répertoire central opérationnel justifie les coûts et le temps nécessaires à sa mise en place.

Le choix d'une architecture de type **coopération** semble le plus indiqué lorsque :

- le niveau de gouvernance sur les données doit être flexible,
- le contrôle sur les applications sources n'est pas garanti,
- la création d'une source de données neutre est nécessaire,
- le niveau de disponibilité des données est important,
- la cohérence des données apporte une grande valeur ajoutée,
- l'architecture de type répertoire virtuel doit évoluer de manière incrémentale vers une architecture plus centralisée.

Le choix d'une architecture MDM peut aussi être déterminé par les objectifs métier liés au projet. Nous considérons ici sept objectifs (voir Table 4) :

- Améliorer le **contrôle** des données. Le contrôle des données est directement lié au niveau de gouvernance souhaité.

a. Dans le cas de l'architecture de type répertoire virtuel, le contrôle se limite aux échanges de données.

b. Dans le cas de l'architecture de type centralisation, le contrôle concerne à la fois les échanges, le stockage et le contenu des données.

c. L'architecture de type coopération contrôle l'échange des données et offre une grande flexibilité en ce qui concerne le contrôle du contenu des données et de leur stockage. On peut établir différents niveaux de contrôle suivant le type de données concernées. Par exemple, d'un côté aucun contrôle des données médicales et de l'autre côté un contrôle poussé de la qualité des données administratives et de leur intégration (détection des doublons, standardisation, analyse des formats, …).

- Améliorer la **disponibilité** des données. La disponibilité dépend de nombreux paramètres. Le premier est le nombre d'intervenants par lesquels les données vont transiter. Pour améliorer la disponibilité, il faut éviter de faire des appels en cascade à différents services d'accès aux données provenant de différentes applications.

  a. Dans le cas de l'architecture de type répertoire virtuel, on rajoute un intermédiaire. Le répertoire virtuel est dépendant du niveau de disponibilité des sources authentiques. Le niveau de disponibilité du répertoire virtuel s'alignant au mieux sur le niveau de disponibilité le plus bas des fournisseurs de données.

  b. Dans le cas de l'architecture de type centralisation, le niveau de disponibilité dépend uniquement de la DB centrale.

  c. Dans le cas de l'architecture de type coopération, le niveau de disponibilité des données dépend de la DB Commune mais également de la synchronisation avec les fournisseurs de données qui est rarement garantie en temps réel.

- Améliorer la **cohérence**/qualité des données. La cohérence des données n'est pas assurée avec l'utilisation d'un répertoire virtuel. De plus, la qualité des données échangées n'est pas connue. Ainsi des données de mauvaise qualité risquent d'être diffusées vers les applications consommatrices. L'architecture de coopération permet de créer une vue unique cohérente et intégrée des données de référence. Cependant, la synchronisation entre les fournisseurs de données et la vue unique n'est pas toujours garantie. L'architecture de centralisation élimine ce problème de synchronisation et assure l'accès à des données de référence uniques, cohérentes et à jour.

- Améliorer l'**enrichissement** des données. L'enrichissement et l'intégration des données sont fortement dépendants des données de départ. Si les données à enrichir sont incohérentes ou de mauvaise qualité le résultat de l'intégration sera forcément discutable. C'est pourquoi les architectures de type coopération et centralisation devraient être privilégiées si l'objectif principal est de favoriser un enrichissement optimal des données de référence.

- Préserver l'**indépendance** des applications fournisseuses de données. Plus la gestion des données de référence se centralise, moins l'indépendance des applications fournisseuses pourra être préservée.

a. Le répertoire virtuel implique que les applications demeurent totalement indépendantes, même si elles doivent pouvoir répondre à des requêtes de consultation des données.

b. La coopération, comme son nom l'indique, nécessite une volonté des applications à coopérer pour partager les données via une DB commune. Ceci n'a aucun impact sur leurs propres bases de données.

c. Par contre, la centralisation implique une dépendance forte des applications vis-à-vis de la DB qui devient non seulement commune mais aussi centrale et unique.

- Minimiser les changements nécessaires à la **mise en place** de la solution MDM. La mise en place d'un répertoire virtuel est nettement moins onéreuse que la mise en place d'une architecture de centralisation. Dans le premier cas de figure, la difficulté est de créer un annuaire de données performant. Dans le deuxième cas de figure, la difficulté est d'adapter radicalement les applications existantes qui devraient revoir complètement la gestion et l'implémentation de leurs accès aux DB.
- Diminuer les **coûts de gestion** des données. Une fois que la solution MDM a été mise en place, l'architecture de type centralisation favorise les économies d'échelle. Les données ne sont plus dupliquées et aucun mécanisme de synchronisation n'est plus nécessaire. Lorsque les données de référence évoluent, les changements sont apportés une et une seule fois. Dans le cas du répertoire virtuel, chaque application continue à devoir assurer la cohérence et la qualité de ses données de manière locale. On échange des données mais on ne mutualise pas les efforts pour leur gestion. Dans le cas de la coopération, on mutualise les efforts pour gérer les données, mais on les duplique dans une base de données commune, ce qui augmente inévitablement les coûts de gestion.

	Répertoire Virtuel	Coopération	Centralisation
Contrôle	-	+	++
Disponibilité	-	+	++
Cohérence	-	+	+++
Enrichissement	-	++	++
Indépendance	++	-	---
Mise en place	++	-	---
Coûts de gestion	--	-	+++

Table 4 : MDM: Choisir son architecture

3. MDM : La mise en place

Selon Gartner, d'ici à 2012, le taux d'échec des projets MDM s'élèvera à 50%. Même si les technologies MDM ne sont pas encore complètement matures, le principal problème est le manque crucial de Best Practices liées à la mise en place d'un projet MDM. Dès lors, la plupart des entreprises sont confrontées à des défis importants notamment en ce qui concerne la définition des règles de gouvernance et des critères de qualité. Afin de remédier à cette situation, voici quelques Best Practices préconisées lors de la mise en place d'un projet MDM.

Comme tout projet de développement d'un système d'information (SI), un projet MDM comprend un certain nombre d'étapes successives. L'objectif de cette section n'est pas d'exposer en détail les étapes classiques d'un projet de développement logiciel mais de mettre l'accent sur les particularités des projets MDM en mettant en avant les tâches spécifiques à ceux-ci et en les illustrant sur notre étude de cas. Le caractère transversal et multi-organisationnel des projets MDM exige une attention particulière. Dans la suite de cette section, nous regroupons les étapes spécifiques aux projets MDM suivant trois phases (voir Figure 9) : l'analyse (Section 3.1), la conception (Section 3.2) et l'implémentation (Section 3.3). Le processus de développement d'un référentiel est itératif ; c'est-à-dire qu'à tout moment, lors des phases de conception ou d'implémentation, un retour en arrière vers les phases précédentes est toujours possible.

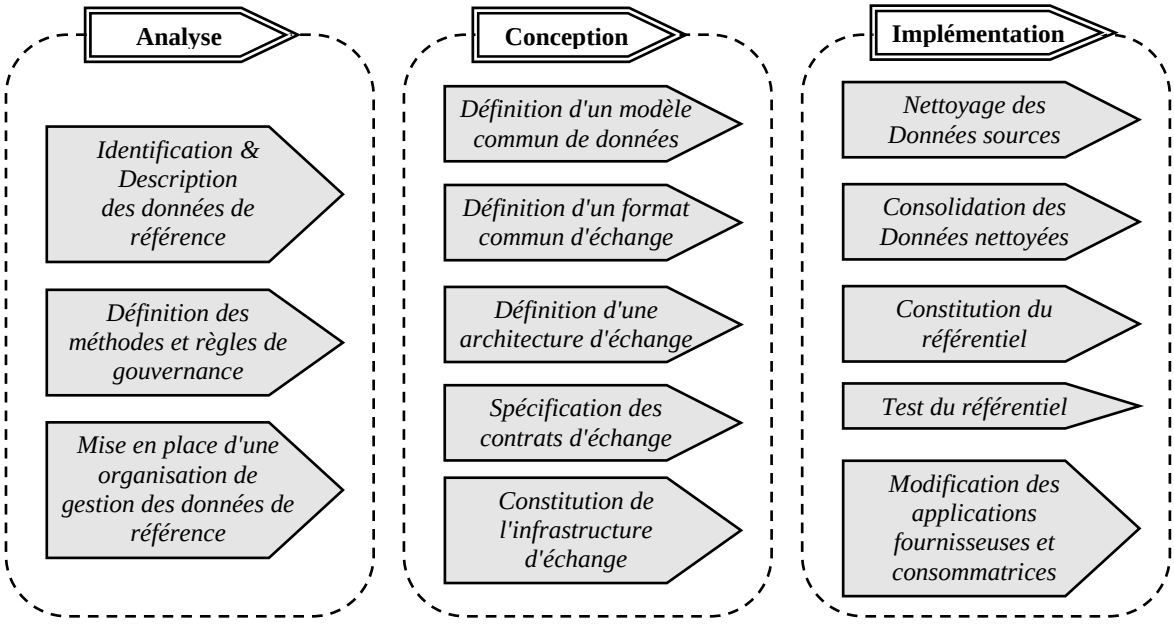


Figure 9 : MDM: Les étapes à suivre

3.1. Phase d'analyse

Aborder l'approche MDM comme un problème avant tout technologique est une erreur qui peut être lourde de conséquence. L'approche MDM c'est essentiellement comprendre des processus et des systèmes métier complexes et mettre en place des (nouveaux) processus afin de gérer leurs données. La première question à se poser n'est pas de savoir si l'approche MDM se basera sur une solution IBM, Oracle, ou SAP en suivant ou non les principes d'une architecture SOA. Au départ, il faut identifier quelles données vont être partagées, pourquoi elles doivent être partagées, qui va en être la source authentique, où vont-elle être stockées, qui en est le propriétaire, qui peut y accéder et quels sont les obstacles (techniques, conceptuels ou politiques) qui pourraient empêcher le partage de ces données (Section 3.1.1). Ensuite, il faut s'interroger sur les règles à appliquer afin d'assurer la gestion et la maintenance de ses données de référence (Section 3.1.2). Enfin, il faut déterminer les parts de responsabilité dans la gestion journalière des données de référence et favoriser la coordination des différents intervenants en créant une organisation spécifiquement dédiée à cette tâche (Section 3.1.3).

3.1.1. Identifier et décrire les données de référence.

Selon Forrester, les projets MDM ne couvriront pas l'ensemble des données mais se focaliseront sur les données critiques avec des problèmes de qualité visibles. De tout évidence, le périmètre des données de référence doit être délimité avec précaution. Cette première étape d'analyse est généralement révélatrice de la complexité d'un projet MDM. L'objectif est de définir un périmètre optimal pour les données de référence et d'obtenir un aperçu détaillé de leur statut actuel. Certaines organisations identifient des centaines de structures de données différentes avec des centaines d'interprétations tout aussi différentes.

L'identification des données de référence dépend principalement du domaine d'application, des besoins métier et du type de données échangées. Généralement, une donnée de référence possède les caractéristiques suivantes :

1. Une donnée de référence a une **valeur métier** importante. Il faut distinguer les données critiques pour le métier des données moins sensibles tant du point de vue de leur *qualité* que de leur *disponibilité*.
2. Une donnée de référence est réutilisée, **partagée** entre différentes applications et/ou échangée avec des tiers.
3. La **durée de vie** d'une donnée de référence est considérée comme longue. En opposition aux données dites transactionnelles, une donnée de référence a généralement un long cycle de vie. Cette durée dépend principalement du métier considéré. De plus, un même type de données peut avoir des cycles de vie et des rythmes de mise à jour différents suivant les contextes d'utilisation de cette donnée. Par exemple, les données concernant un patient ou un dossier patient ont potentiellement un long cycle de vie et sont plus sujettes aux changements. Tandis qu'une prescription ou les résultats d'un examen médical ont un cycle de vie court et ne sont pas sujets au changement.
4. Le **volume** des données de référence est suffisamment élevé. L'utilité de la création d'un référentiel dépend de la quantité de données qui vont y être stockées. Gérer trois ou cent mille patients nécessite différents types de stratégies et d'outils. Outre le volume des données, le volume des transactions est également à prendre en considération.

Une fois les données de référence identifiées, la maîtrise des données de référence dépend de notre capacité à les expliciter et à les faire valider par les acteurs impliqués. Expliciter ces données de référence consiste essentiellement à décrire :

- les structures des données et les relations entre elles,
- les méta-données (Attribut, type de données, valeurs autorisées, valeurs par défaut, les contraintes, les dépendances, etc.) [Elmasri, 06],
- la manière dont ces données sont interprétées (glossaire métier),
- le cycle de vie des données avec les processus de création, de modification (correction) et de suppression des données,
- qui sont les fournisseurs et les consommateurs de ces données,
- les cas d'utilisation,
- les exigences de qualité par rapport aux données (fraîcheur, disponibilité, complétude, …).

Exemple SumEHR

Dans notre étude de cas, l'identification des données de référence paraît évidente. Les données de référence peuvent être considérées comme les données du dossier médical minimal dont la structure a été définie dans le standard SumEHR. Suivant les critères d'identification que nous avons définis plus haut, le dossier médical minimal en satisfait la plupart :

- les dossiers médicaux minimaux ont une haute valeur ajoutée d'un point de vue métier afin de pouvoir assurer la continuité des soins,
- ces dossiers médicaux vont être partagés entre différents prestataires de soins suivant des règles très précises.
- un dossier médical est sujet aux modifications et a une durée de vie potentiellement longue (Espérance de vie des hommes et des femmes en Belgique).
- le volume de données est potentiellement très élevé (toutes personnes recevant des soins en Belgique).

Dans un dossier médical, on distingue, pour des raisons évidentes de protection de la vie privée, les données « administratives » et « médicales ». Cette distinction apparaît clairement dans le message SumEHR et particulièrement dans la définition *folder* (voir Section 1.4.4) qui sépare les données administratives (démographiques) concernant le patient (firstname, familyname, birthdate, sex, address(es), telecom(s), usuallanguage) et ses données médicales (vaccination(s), médicament(s), allergie(s), risque(s), antécédents, etc.) reprises dans la *transaction*. Même si la frontière entre données médicales et administratives n'est pas toujours évidente à déterminer, toutes deux peuvent être considérées comme des données de référence. Évidemment, des mesures particulières devront être prises pour les données médicales mais les principales différences apparaîtront au niveau des règles de gouvernance et de l'architecture du référentiel que nous présenterons dans la suite. Par exemple, aucun intégration des données médicales ne sera autorisée. À l'heure actuelle, les médecins généralistes jouent les rôles de centralisateur et d'intégrateur des données reprises dans le SumEHR. C'est à partir de leur dossier médical informatisé qu'est généré un dossier médical minimal respectant le format SumEHR.

Les médecins généralistes sont donc la source authentique pour les données SumEHR. Mais n'existe-t-il pas d'autres fournisseurs potentiels de données ? Les hôpitaux ? Les médecins spécialistes ? et pourquoi pas le registre national ou l'INAMI ?

Dans ce cadre, la plateforme eHealth pourrait jouer un rôle essentiel d'intermédiaire

entre les institutions publiques et les médecins généralistes. Ce type de service (voir Figure 10) permettrait aux médecins généralistes de se concentrer sur les données médicales et de réduire leurs tâches administratives liées aux données administratives des patients voire des autres médecins.

Lorsque l'on analyse le standard SumEHR (voir Figure 10) on remarque que certaines données pourraient, via la plateforme eHealth, être fournies par le registre national (le signalétique des patients) et/ou de l'INAMI (le signalétique des médecins). Cette idée n'est pas neuve, car la plateforme eHealth offre déjà des services aux prestataires de soins qui permettent d'accéder à certaines données du registre national. Un point d'attention à considérer est comment intégrer de manière optimale et automatique ces données dans les SumEHRs gérés par les médecins généralistes.

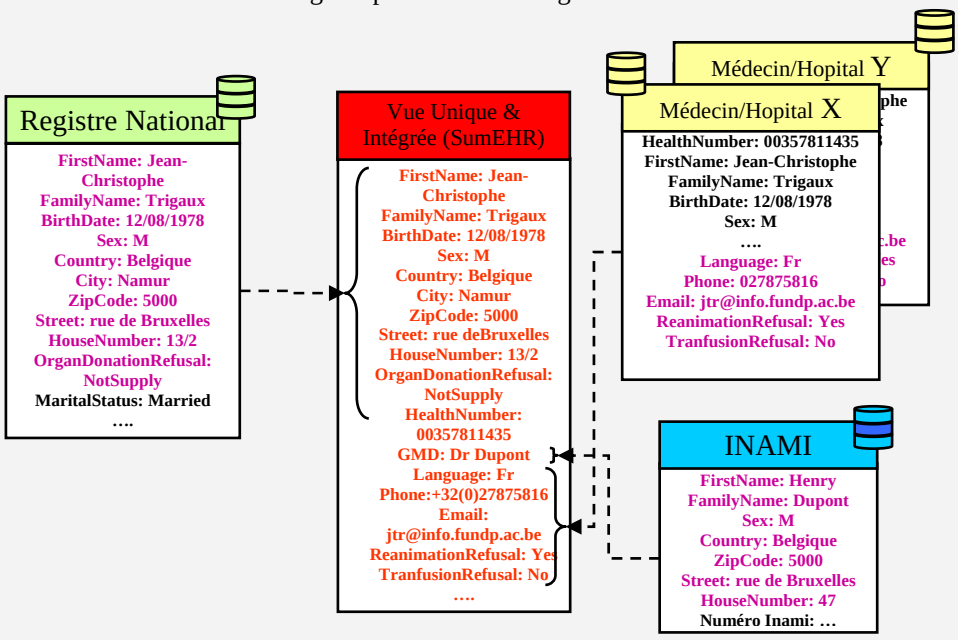


Figure 10 : SumEHR: Exemples de fournisseurs potentiels de données

En ce qui concerne la description des données de référence, le format SumEHR définit la structure des données de référence et de certaines de leurs méta-données. L'utilisation de codifications ou thesaurus médicaux (ICD10, ICD9, ICPC2) nous en apprend plus sur la signification des données médicales. Cependant, à ce stade, nous n'avons pas trouvé d'informations suffisamment détaillées concernant les relations existant entre ces données, la signification des données (mise à part les thesaurus médicaux), les cycles de vie des données ainsi que les différents scénarii d'utilisation envisagés pour les données de référence. Ces éléments permettraient de clarifier le contexte du projet et d'avoir en main les éléments concrets qui permettraient de diriger les choix de conception dans les phases suivantes.

3.1.2. Déterminer les méthodes et les règles de gouvernance

Sans une gouvernance efficace et appropriée, les initiatives MDM rencontreront des difficultés principalement liées à des conflits politiques et à la résistance des fournisseurs de données. Cette résistance se justifie par le manque d'une explicitation claire des règles régissant l'utilisation de leurs données et sur la

responsabilité qu'ils devront endosser. L'objectif est de définir une stratégie de gouvernance dès les premières phases du projet de manière à augmenter ses chances de réussite. Définir une stratégie de gouvernance consiste à :

- Définir les règles de qualité concernant la consistance, la précision et la complétude des données.
- Définir des indicateurs de mesure de la qualité des données.
- Définir les moyens mis en œuvre pour détecter les anomalies.
- Définir les règles d'arbitrage pour traiter ces anomalies.
- Définir les règles d'intégration, d'élimination des doublons.
- Définir les règles de sécurité en terme de protection et d'accessibilité des données.
- Définir un processus de maintenance et de gestion du changement afin de préserver la qualité des données et d'assurer leur disponibilité.

Exemple SumEHR

Dans le cas du projet SumEHR, la stratégie de gouvernance devrait être élaborée en collaboration étroite avec les principaux acteurs et en accord avec les réglementations légales. Des exemples de règles à affiner seraient :

1. Respecter la loi du 22 août 2002 relative aux droits du patient.
2. Garantir le secret professionnel et préserver la relation de confiance entre les médecins et les patients.
3. Les *données à caractère privé* doivent être chiffrées de bout en bout.
4. Seuls les *médecins* possédant un *lien thérapeutique* avec le patient peuvent consulter (déchiffrer) les données qui leur sont adressées.
5. Pas de répudiation possible après l'envoi et/ou la réception de données médicales.
6. Les patients sont identifiés par leur numéro NISS. Lorsqu'ils ne possèdent pas de numéro NISS, un numéro pourra être généré via le registre Bis de la BCSS.
7. Les données administratives d'un dossier patient doivent être synchronisées entre les différents acteurs de soins de santé et de la meilleure qualité possible.
8. etc.

Une distinction claire entre les règles applicables aux données administratives et celles applicables aux données médicales doit être établie. Une description détaillée de ces règles de gouvernance est primordiale. Il faut expliciter pour chaque règle quelles données sont concernées et quelles restrictions leurs sont imposées tant du point de vue de leur contenu que de leur accessibilité. Par exemple, il faudrait identifier quelles sont les données à caractère privé ?, comment reconnaît-on un médecin? (diplômé / diplômé et praticien), comment définit-on le lien thérapeutique entre un patient et son médecin ?, quand un lien thérapeutique expire-t-il ?, Peut-on intégrer des données médicales ?, Peut-on fusionner des données médicales ?

L'objectif du référentiel est d'arriver à (semi)-automatiser la vérification de la satisfaction de ces règles. En conséquence, la formalisation de ces règles en des termes qui puissent être interprétés par un ordinateur est importante. Par exemple, le référentiel pourrait vérifier (via eHealth) automatiquement auprès de l'INAMI et du SPF santé qu'un médecin a bien reçu le diplôme de docteur en médecine et qu'il possède également un numéro INAMI lui permettant d'exercer son métier sur le territoire belge.

3.1.3. Mettre en place une organisation pour gérer les données de référence

Au terme de la phase d'analyse, il est nécessaire de mettre en place une organisation afin de mener la suite du projet et d'assurer la gestion journalière des données de référence ainsi que le respect et l'évolution des règles de gouvernance. L'objectif est de constituer une équipe de personnes en charge de la gestion du référentiel aussi bien pour sa constitution que sa maintenance.

Cette organisation est dotée de différentes prérogatives :

- La gestion et la centralisation des données de référence,
- L'élaboration et l'évolution des règles de gouvernance,
- La supervision des données de référence (utilisation, accès, préservation de la cohérence et du caractère privé, etc.),
- L'arbitrage des problèmes relatifs au référentiel et aux données gérées par celui-ci,
- Faciliter l'utilisation du référentiel et le valoriser.

Ce groupe de personnes est généralement composé des contributeurs à l'élaboration des règles de gouvernance. Toutefois, il faut préserver son caractère transversal et la mixité des profils business et IT. Dans la suite de cette section, nous décrirons différents rôles à jouer au sein de cette organisation. Cependant, ces rôles doivent être reconsidérés au sein de chaque organisation :

- ***Sponsor***. Le Sponsor détermine le plan stratégique et met à disposition les moyens (ressources et personnel) à la mise en place de la solution MDM.
- ***Business Data Owner***. Le Business Data Owner participe à
  - La validation et la définition de la structure et de la sémantique des données,
  - La spécification des niveaux de qualité et de sécurité.
  - La définition des règles concernant la création, modification et suppression des données de référence dont il est le propriétaire.

  Le Business Data Owner collabore étroitement avec les gestionnaires des processus métier qui ont un impact sur le cycle de vie de ses données.
- ***Data Steward***. Le data Steward joue le rôle de coordinateur et assure la gestion journalière des données de référence. Il gère opérationnellement la qualité des données et supporte la mise en œuvre des règles de gouvernance. Il est garant de l'intégrité des données de référence et du support aux Data Users.
- ***Data User***. Soit le Data User est limité à une simple consultation des données, soit il possède aussi des prérogatives en termes de création/modification/suppression de données. Habituellement, un Data User est également un Data Owner pour un autre type de données. Dans tous les cas, il participe à la définition des scenarii d'utilisation et spécifie ses exigences notamment au niveau de la disponibilité et de la qualité des données.
- ***Data Architect***. Le Data architect participe à la définition des architectures applicatives et techniques des solutions MDM. Il collabore étroitement avec les Business Data Owners afin de définir les modèles de données et d'assurer leur cohérence et standardisation. Il s'attache au respect des normes et standards et favorise l'interopérabilité entre les différents systèmes partageant les données.

En résumé, le Sponsor s'occupe de l'établissement des budgets et de la définition de la stratégie générale. Le Business Data Owner s'occupe en détail de ses données. Le Data Steward s'occupe de l'utilisation correcte du référentiel et

vérifie la bonne synchronisation et intégration des données. Le Data User définit ses exigences quant à l'utilisation des données. Enfin, le Data Architect met à disposition son expertise technique et métier afin de favoriser la convergence vers une solution la plus interopérable possible.

Exemple SumEHR

Dans le cas du projet SumEHR, de nombreux acteurs peuvent intervenir dans l'échange des dossiers médicaux. Dès lors, constituer une organisation suffisamment représentative est une étape extrêmement délicate mais nécessaire afin d'assurer une gestion efficace et éthique de l'échange de dossiers médicaux minimum dans le cadre de la loi. Les principaux membres de cette organisation devraient représenter :

1. les patients,
2. les médecins généralistes,
3. les prestataires et institutions de soins (médecins spécialistes, hôpitaux, etc.),
4. l'ordre des médecins, la commission de la vie privée,
5. les réseaux de santé,
6. les organismes publiques liés aux soins de santé (SPF santé, INAMI, etc.),
7. les sociétés informatiques développant des logiciels de gestion des dossiers patients à destination des médecins généralistes et des hôpitaux (WINDOC, Health One, Corilus, SoSoeme, Prodoc, Fisimed, Socrate, etc.),
8. la plateforme sécurisée d'échange de données médicales eHealth,
9. etc.

La question délicate est donc de déterminer dans cette organisation qui jouent quel(s) rôle(s). Nous n'avons évidemment pas de réponses satisfaisantes à cette question mais nous avons des questions complémentaires à soulever. Est-ce qu'un rôle peut-être joué par un seul acteur ? Qui sera (seront) le(s) sponsor(s) d'un projet de cette envergure ? Qui est le propriétaire des données médicales ? Le patient et/ou le médecin généraliste ? Le médecin généraliste a dans les faits un rôle de centralisateur des données mais quelle est l'intersection acceptable entre le rôle du médecin généraliste et le rôle de Data Steward ?

3.2. Phase de conception

Une fois la phase d'analyse terminée, différentes décisions doivent être prises afin de concevoir le référentiel. Dans un premier temps, il est nécessaire de définir un modèle commun (Section 3.2.1) pour les données de référence. Ce modèle permettra à tous les intervenants de comprendre la structure et la sémantique des données qui vont être échangées. Dans un deuxième temps, un format standard d'échange doit être défini afin de servir de langage commun entre les différents interlocuteurs (Section 3.2.2). Ensuite, il faut déterminer l'architecture d'échange (Section 3.2.3) et spécifier les contrats d'échanges (Section 3.2.4) établis entre les différents intervenants. Enfin, une infrastructure d'échange (Section 3.2.5) doit être proposée afin de pouvoir satisfaire au mieux les règles de gouvernance et vérifier leur mise en œuvre correcte.

3.2.1. Définir un modèle commun des données de référence

L'objectif de cette étape est de définir une modèle logique standard pour les données de référence explicitant leurs attributs, le type des attributs, les valeurs par défaut, les valeurs autorisées, les contraintes, les relations entre les données, la signification des données, etc.

Le premier risque est de vouloir satisfaire tous les acteurs en incluant la majorité des attributs des données sources. Les données de référence deviennent alors très détaillées et extrêmement complexes à produire, à maintenir et à utiliser.

Le second risque est de créer des données de référence dont l'interprétation n'est pas claire pour tous les utilisateurs. Des techniques telles que la modélisation des données de référence, la constitution d'une ontologie et/ou la constitution d'un glossaire métier sont fortement recommandées afin de minimiser les ambiguïtés et de s'assurer de la convergence des points de vue sur les données de référence.

Exemple SumEHR

À notre connaissance, il n'existe actuellement pas de modèle commun des données incluses dans un dossier médical minimal. La description d'un SumEHR fournie sous un format XML/XSD ne suffit pas pour décrire la structure et la sémantique des données. Une description XML permet de structurer des données avec des tags et de spécifier leur format (caractère, entier, date, …) mais elle ne décrit aucunement les relations entre les données, ni leur signification d'un point de vue métier.

Une première amélioration serait de recréer à partir de ce format XML un modèle de données (entité-relation par exemple) et de lui associer un glossaire métier. La validation de ce modèle par les acteurs clés permettra sans doute de soulever des incohérences et/ou des non-dits impactant les hypothèses ou les choix qui auront été pris lors de la phase d'analyse.

3.2.2. Standardiser le format d'échange des données de référence

L'étape préalable à l'échange effectif de données est la définition d'un format standard (généralement basé sur les technologies XML/XSD) qui servira de langage commun pour toutes les applications fournissant ou consommant ces données. L'objectif est de définir un langage commun pour toutes les applications leur permettant de pouvoir comprendre, publier et retraiter les données de référence. Ce langage commun est généralement défini à partir du modèle commun de données et détermine la structure des données et leur format. Deux alternatives sont envisageables pour définir ce type de langage :

1. le standard est un **langage pivot** permettant la traduction d'un format de données vers un autre. L'avantage de cette solution est que ce standard reste transparent pour les applications. La solution MDM traduit les données dans le format spécifique à l'application consommatrice.
2. le standard est un **langage commun** et la solution MDM fournit simplement les données de référence sous ce format qui devra pouvoir être compris et donc exploité par toutes les applications.

La traduction des données vers le standard peut donc être gérée :

- soit au niveau de la solution MDM,
- soit en interne par les applications consommatrices.

La deuxième solution est généralement préférée, surtout lorsque les données sont échangées entre différentes organisations, car les formats spécifiques à chaque application fournisseuse évoluent généralement plus rapidement que le standard lui-même.

L'élaboration d'un standard est un problème délicat en soi, quel que soit le domaine considéré. De nombreux conflits aussi bien politiques qu'économiques peuvent freiner l'adoption d'un standard commun. Chaque acteur espère que le standard se rapprochera le plus possible de la solution qu'il a ou qu'il prévoit d'implémenter.

Exemple SumEHR

Le cœur même du projet SumEHR est la définition d'un format d'échange standard pour les dossiers patients. Ce format d'échange est considéré comme un langage commun qui, en réalité, est une transaction Kmehr (Kindmessages for electronic healthcare record, standard belge). Kmehr et donc SumEHR se basent sur les technologies XML/XSD. À titre d'illustration, nous avons repris dans les deux cadres suivants l'extrait d'une description XSD et d'un fichier XML définissant un SumEHR :

1. Dans le premier cadre se trouve un exemple de message XML contenant les données fictives d'un patient conforme au schéma XSD défini dans le point suivant.
2. Dans le second cadre se trouve une description XSD simplifiée spécifiant la structure des données concernant un patient tel que définie dans Kmehr. Brièvement, on constate par exemple que :
   a. À la 8 et 9 ème ligne, un patient possède un « familyname » qui est une donnée de type string.
   b. À la 10ème ligne, un patient possède une « birthdate » dont la structure est elle-même définie dans une autre description XSD (mot clé « ref »).

```
<patient>
        <id S="ID-PATIENT" SV="1.0">49112002395</id>
        <id S="LOCAL" SL="Sumehr-Love" SV="1.06.02">3800</id>
        <firstname>Jean-Christophe</firstname>
        <familyname>TRIGAUX</familyname>
        <birthdate>
                <date>1978-12-18</date>
        </birthdate>
        <sex>
                <cd S="CD-SEX" SV="1.0">male</cd>
        </sex>
        <address>
                <cd S="CD-ADDRESS" SV="1.0">home</cd>
          <country>
                <cd S="CD-COUNTRY" SV="1.0">be</cd>
          </country>
          <zip>5000</zip>
          <city>NAMUR</city>
          <street>rue de bruxelles</street>
          <housenumber>13</housenumber>
        </address>
        <telecom>
               <cd S="CD-ADDRESS" SV="1.0">home</cd>
               <cd S="CD-TELECOM" SV="1.0">phone</cd>
               <telecomnumber>027875816</telecomnumber>
        </telecom>
        <usuallanguage>fr</usuallanguage>
</patient>
```

```
1 <xsd:element name="patient" type="personType"/>
2 <xsd:complexType name="personType">
3         <xsd:sequence>
4                 <xsd:element name="id" type="ID:ID-PATIENT"
5                   maxOccurs="unbounded"/>
6                 <xsd:element name="firstname" type="xsd:string"
7                   maxOccurs="unbounded"/>
8                 <xsd:element name="familyname"
9                   type="xsd:string"/>
10                <xsd:element ref="birthdate" minOccurs="0"/>
11                <xsd:element ref="sex"/>
12                <xsd:element ref="address" minOccurs="0"
13                   maxOccurs="unbounded"/>
14                <xsd:element ref="telecom" minOccurs="0"
15                  maxOccurs="unbounded"/>
16                <xsd:element name="usuallanguage"
17                  type="xsd:language" minOccurs="0"/>
18                <xsd:element name="text" type="DT:textType"
19                   minOccurs="0"  maxOccurs="unbounded"/>
20        </xsd:sequence>
```

3.2.3. Définir une architecture pour échanger les données de référence

Bien que les solutions MDM puissent prendre de nombreuses formes, la plupart se basent sur les mêmes types d'architectures. Généralement, trois types d'architectures MDM sont privilégiés : le répertoire virtuel (registry), la centralisation (repository) et la coopération (hybrid) (voir Section 2.2). Des considérations à la fois techniques et business entrent en ligne de compte dans la sélection d'une architecture.

Le critère le plus discriminant est souvent le niveau de contrôle sur les données. L'architecture de centralisation permet un contrôle fin et poussé des données de référence, de leur qualité et de leur intégration. À l'opposé, l'architecture de type répertoire virtuel limite le contrôle aux échanges de données et à la redirection des demandes d'accès vers les données. Entre les deux se situe l'architecture de coopération. Une description de ces architectures ainsi que leurs avantages et inconvénients respectifs ont été détaillés dans la Section 2.2.

Exemple SumEHR

Dans le cadre des différentes pistes de réflexion que nous avançons concernant l'échange de dossiers médicaux minimum, l'objectif principal est que les médecins conservent la gestion complète et entière des données médicales dont ils ont la charge. Dés lors, différentes propositions sont envisageables dans le contexte de l'approche MDM et de ses architectures types.

Par exemple, la mise en place d'un annuaire de données permettrait de préserver le lien entre le patient et les endroits où une requête pourrait être envoyée afin d'obtenir certaines données les concernant (suivant des règles d'accessibilité strictes). Cet annuaire de données permettrait ainsi aux médecins généralistes d'alimenter les dossiers médicaux intégrés (DMI) qu'ils gèrent pour leurs patients et à partir de ceux-ci générer, en tant que source authentique, des SumEHR qui seront ensuite partagés avec les différents consommateurs de données. Dès lors, le médecin généraliste pourrait à la fois :

- consulter les données médicales de ses patients chez d'autres prestataires de soins,
- fournir les dossiers médicaux minimaux sous le format SumEHR en tant que source authentique.

On souligne ici le rôle prépondérant que pourrait jouer le médecin généraliste qui est le garant et le centralisateur des données concernant ses patients. Dans ce contexte, le choix de l'architecture est une problématique importante qui doit encore être approfondie.

En ce qui concerne l'architecture de type **centralisation** il est évident qu'elle doit être écartée car à l'heure actuelle il existe une volonté forte que la gouvernance des données médicales reste décentralisée au niveau des médecins généralistes.

En ce qui concerne l'architecture de type **répertoire virtuel**, elle pourrait se justifier pour les raisons suivantes :

1. La gouvernance des données est extrêmement faible :
    - le seul contrôle autorisé sur les données médicales est la vérification que leur caractère privé est préservé. C'est-à-dire que les données sont chiffrées de bout en bout et que seules les personnes autorisées peuvent y

accéder. La majorité des contrôles doit donc être effectuée au niveau des échanges de données : leur accessibilité, leur disponibilité, etc.

- La synchronisation des données n'est pas une préoccupation majeure. Dans un premier temps, un SumEHR est considéré comme un instantané qui ne sera pas nécessairement stocké au niveau des applications consommatrices. Dans ce cas de figure, la consommation de données n'entraînera donc pas la création de doublons.
- La qualité des données est une préoccupation majeure mais qui sera gérée par les fournisseurs de données et non par le référentiel.
- Le référentiel ne doit donc s'occuper que de la centralisation des requêtes et de leur redirection. Le référentiel pourrait éventuellement vérifier le bon chiffrement des données afin d'éviter que des applications fournisseuses envoient des données non ou mal chiffrées.

2. La préservation de l'indépendance des applications fournisseuses de données. Principalement, l'indépendance des applications utilisées par les médecins.

Toutefois, si le choix d'une architecture de type répertoire virtuel était pris, cela entraînerait également des conséquences importantes qu'il ne faudra pas négliger :

1. La gestion des doublons sera délicate. Même si la consommation de données n'entraîne pas la création de doublons, il faudra éviter au maximum qu'il existe plusieurs fournisseurs de données pour les mêmes données patient au sein d'un réseau de santé et entre réseaux de santé. Si cette situation devait se produire, il faudra :
   - identifier ces doublons,
   - assurer la cohérence de ces doublons via des mécanismes de synchronisation ou d'élimination.
2. Les problématiques de synchronisation, d'intégration, de gestion du changement et de qualité devront être adressées localement dans chaque application et donc potentiellement par les médecins généralistes avec l'aide de leurs outils de gestion médicale.
3. Un haut niveau de disponibilité des données sera difficilement atteignable. En effet, le niveau de disponibilité des données baissera en fonction du nombre d'intervenants dans l'échange de données et/ou du nombre d'appels en cascade de services web. Il est difficilement imaginable qu'un médecin accepte de rendre disponible 24/24h l'application qu'il a installée sur son ordinateur dans son cabinet.
4. Le niveau de gouvernance sur les données administratives sera également faible. Ces données n'étant pas la préoccupation principale des médecins il y a de fortes chances que leur qualité varie. Pourtant ces données administratives sont importantes notamment pour :
   - identifier un patient et donc son dossier médical,
   - se rendre au domicile du patient en cas d'urgence,
   - envoyer les factures à la bonne adresse.

Le passage d'une architecture de type répertoire virtuel vers une architecture de type **coopération** peut se justifier lorsque :

1. Le niveau de disponibilité des données devrait augmenter. La disponibilité des données ne dépendrait plus des outils de gestion médicaux spécifiques aux médecins mais d'une base de données neutre à laquelle ces outils se synchroniseraient.
2. Une différenciation du niveau de gouvernance entre les données médicales et administratives est nécessaire.
3. La détection et la gestion de doublons entre différentes applications doivent être améliorées.

Pour ces raisons, la **proposition** de l'approche MDM serait de préconiser à moyen terme la création de référentiels pour les dossiers médicaux minimum au sein des différents réseaux de santé où les médecins généralistes, en tant que source authentique, géreraient les données concernant leurs patients.

Ces référentiels se baseraient sur une architecture de type coopération afin de minimiser les impacts sur les applications existantes des médecins. La principale contrainte pour ces applications serait de synchroniser leurs données (uploader leur SumEHR) avec la base de données neutre. Comme nous l'avons vu précédemment ce référentiel pourrait même faciliter la vie des médecins en leur offrant des services limitant la (re)saisie et la vérification manuelles des données administratives de leurs patients. Différents référentiels de données vont donc apparaître et coopérer afin d'échanger ces données médicales. Chaque réseau de santé posséderait son propre référentiel de données.

Pour les *données administratives*, eHealth pourrait jouer le rôle d'intermédiaire entre les différents réseaux de santé et les institutions publiques (INAMI, Registre National, etc.). eHealth pourrait offrir des services similaires à un référentiel de type répertoire virtuel pour centraliser les requêtes des réseaux de santé concernant les données administratives de leurs patients et les rediriger aux institutions compétentes (voir Figure 11).

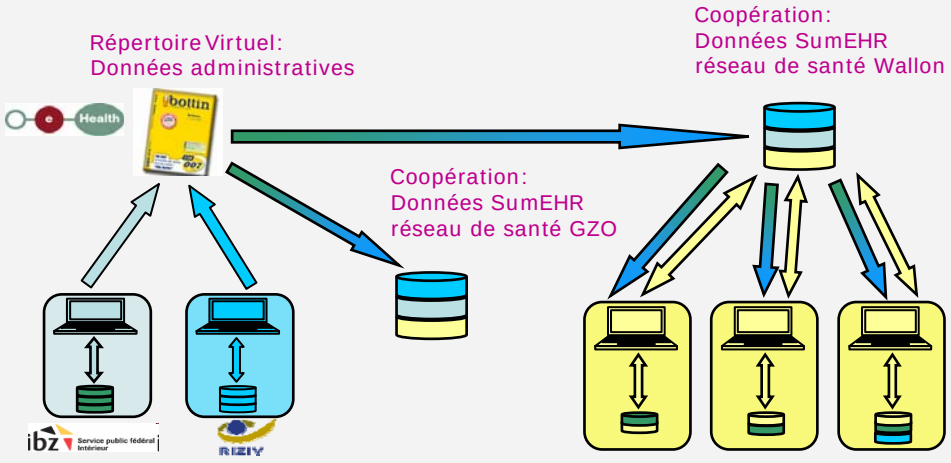


Figure 11 : SumEHR: Proposition d' architecture pour les données administratives

De plus, il serait nécessaire de connecter les différents référentiels des réseaux de santé entre eux. Par exemple, si je consulte habituellement mon médecin généraliste dans la région de Namur mais que, malheureusement, je suis grièvement blessé suite à un accident à Gand, j'aimerais que le médecin urgentiste de Gand puisse avoir accès à mon dossier médical.
eHealth pourrait cette fois-ci jouer un rôle d'intermédiaire entre les différents réseaux

de santé. Ehealth offrirait une service minimum permettant simplement de localiser le(s) réseau(x) de santé qui posséderai(en)t un dossier patient concernant un patient. C'est un service typique pour un annuaire de données ou répertoire virtuel (voir Figure 12). Le médecin urgentiste de Gand pourrait alors contacter directement le réseau de santé wallon pour demander mon dossier patient.

Cet annuaire de données peut se construire soit :

- en demandant aux réseaux de santé d'annoncer qu'ils possèdent un dossier médical pour tel patient.
- en analysant les requêtes de consultation envoyées par les réseaux de santé à eHealth. On prend ici l'hypothèse selon laquelle lorsqu'un réseau de santé demande des données administratives sur un patient, cela implique qu'il gère un dossier médical le concernant.
- en combinant les deux approches précédentes.

Ce référentiel pourrait être utilisé par les réseaux de santé pour obtenir les dossiers patients qui ne sont pas gérés en leur propre sein. Ce référentiel permettrait également de faciliter l'identification de doublons éventuels entre réseaux de santé (c'est-à-dire plusieurs dossiers patients pour le même patient dans différents référentiels). La manière dont les doublons seront résolus devra être définie au niveau des règles de gouvernance.

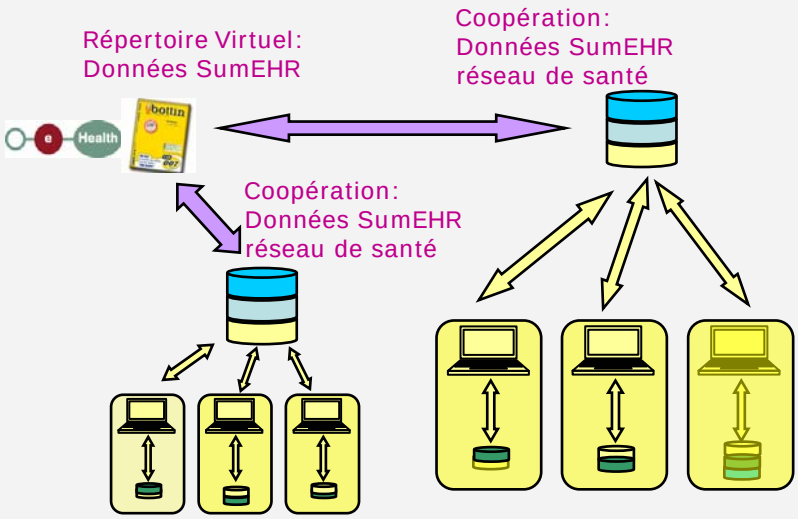


Figure 12 : SumEHR:Proposition d'Architecture pour les données médicales

3.2.4. Spécifier les contrats d'échange

L'élaboration des contrats d'échange est généralement une étape obligatoire pour spécifier la qualité des échanges effectués entre les différents partenaires. Ces contrats d'échange déterminent le niveau de service sur lequel s'accordent les parties concernées afin de partager et administrer les données de référence.

Durant cette étape, on explicite de manière contractuelle un Service Level Agreement (SLA) qui définit essentiellement :

- Le propriétaire du service.
- Les modalités de l'échange (output, input, type de données, fréquences, volumétrie, dates de mise à disposition,…)

- La qualité du service fourni (disponibilité, temps de réponse, engagement sur la qualité des données, …)
- Les procédures relatives au traitement des non-conformités (délais de dépannage, instances d'arbitrage, mesures de reprise,…)

Exemple SumEHR

Dans le cas du projet SumEHR, chaque intervenant offrant des services à destination des patients, des prestataires de soins ou des réseaux de santé devrait fournir un SLA pour chacun des services qu'il offre. Par exemple, la plateforme eHealth offre aux prestataires de soins un service de non répudiation des messages envoyés via sa plateforme, nommé « timestamping ». Voici un extrait de ce SLA[6] concernant les exigences en terme de disponibilité (Figure 13) :

	Objectifs
Définition	• Le service Timestamping ou TSA est considéré comme disponible lorsque le test suivant se déroule correctement : o Une requête est envoyée au service web TSA. o TSA répond à cette requête avec une date précise et une signature. o La date et la signature sont enregistrées dans la base de données. o Il est vérifié si le traitement se déroule correctement. • La consultation du Timestamping ou TSC (consultation des données du TS par les hôpitaux et l'INAMI) est considérée comme disponible lorsque le test suivant se déroule correctement : o La requête est exécutée dans la base de données o Le statut est contrôlé • Les objectifs ci-après concernent l'environnement de production. • Les interventions planifiées n'ont pas d'impact sur les résultats de la disponibilité (l'indisponibilité n'est pas prise en compte). •
Méthode de mesure	• Pour mesurer la disponibilité du service Timestamping, une simulation contrôlant les fonctionnalités susmentionnées est réalisée toutes les 10 minutes. •
Formule de calcul	• $Beschikbaarheid = \frac{(D \times U) - O}{(D \times U)}$ o D = nombre de jours par mois où l'environnement doit être disponible o U = nombre d'heures par jour où l'environnement doit être disponible o O = nombre d'heures d'indisponibilité dans la période de disponibilité o
Période de calcul	• La disponibilité est calculée et rapportée mensuellement. Les actions correctives sont entreprises en fonction de ces chiffres. • L'évaluation définitive a lieu annuellement. •
Résultat à atteindre pour TSA et TSC	99,9 % du lundi au dimanche, de 0:00 à 24:00 = 8,76 heures d'indisponibilité possible par an

Figure 13 : SumEHR: un exemple de SLA

[6] https://wwwacc.ehealth.fgov.be/binaries/website/fr/pdf/SLA-Timestamping-FR-Final.pdf

3.2.5. Constituer l'infrastructure d'échange des données de référence

Une fois que la phase d'analyse permet d'avoir une vue globale de la situation et qu'une architecture d'échange a été choisie, il est nécessaire de mettre en place une infrastructure adaptée et efficace. L'objectif est de construire une infrastructure qui supportera l'architecture choisie et satisfera aux exigences notamment en terme de gouvernance, de scalabilité et de reliability. Cette infrastructure se base sur différents outils et technologies :

- Gouvernance des données
  - Modélisation des données et des flux de données
  - Gestion des processus de validation des données et de gestion des anomalies (Système de workflow et de gestion de règles)
- Qualité des données
  - Profiling et Monitoring
  - Détection des doublons (Matching)
  - Standardisation
  - Enrichissement et Intégration
- Stockage des données
  - Repository
  - Sécurité
  - Gestion des versions
- Distribution des données
  - Data distribution platform
  - Mode Pull ou Push
  - Temps réel ou batch
  - Chiffrement, anonymisation, timestamping, …
  - Accessibilité (UAM), disponibilité, temps de réponse, détection des goulots d'étranglement, …

Les outils proposés actuellement sur le marché (voir Section 4.2) nécessitent d'être adaptés (ou configurés) afin de se conformer aux choix architecturaux, aux données de référence, aux plates-formes et aux applications existantes au sein des organisations. À l'heure actuelle, malgré les tentatives d'intégration des outils, un seul outil n'est pas suffisant pour répondre à l'ensemble des besoins MDM (voir Chapitre 4). Dans la majorité des cas, des développements en interne seront nécessaires, surtout en ce qui concerne la gouvernance des données.

Exemple SumEHR

Dans le cadre de notre scénario pour le projet SumEHR et de l'architecture de coopération que nous avons proposée, les choix concernant l'infrastructure doivent être pris par les réseaux de santé. Ces choix leur sont spécifiques et varieront sans doute suivant leur infrastructure existante. D'un point de vue purement MDM, ces infrastructures devraient mettre à disposition une plateforme de distribution de données (Data Distribution Platform), des outils d'intégration et de gestion de la qualité (Data Quality & Integration) ainsi que des outils de gouvernance (Data Governance).

L'infrastructure devrait s'attacher à (1) faciliter l'interopérabilité entre les différents réseaux de santé et (2) tenir compte des exigences en terme de gouvernance de données, d'amélioration de la qualité des données et de distribution des données.

Dans cette optique, la plateforme eHealth apporte des alternatives intéressantes en ce qui concerne le transport et la sécurisation des données, c'est-à-dire la Data Distribution Platform. En effet, eHealth met à disposition des réseaux de santé une plateforme qui permet d'échanger des données entre prestataires de soins de manière sécurisée.

Une Data Distribution Platform est en quelque sorte un convoyeur qui va éviter tout problème pouvant surgir une fois que les données sortent d'un hôpital ou d'une application médicale en général. Ce type de Data Distribution Platform s'occupe des échanges de données et non du contenu des données. Dans le cadre des échanges de données médicales, une Data Distribution Platform devrait pouvoir fournir les services minimum suivants :

- Vérifier les identités du destinateur et du destinataire ;
- Vérifier que les données médicales sont protégées, c'est-à-dire qu'elles sont correctement chiffrées et que seul le destinataire peut les lire ;
- Assurer le transport des données au bon destinataire dans un délai raisonnable ;
- Garder une trace des échanges de données effectués via la plateforme de distribution de données ;
- Assurer la non répudiation des données (timestamping).

Grâce à eHealth, différents scénarii seraient donc envisageables pour les réseaux de santé :

1. Soit ils utilisent leur propre plateforme d'échange de données inter et intra réseaux de santé ;
2. Soit ils utilisent leur propre plateforme de distribution de données pour échanger leurs données en interne et ils utilisent la plateforme eHealth pour échanger des données entre réseaux de santé ;
3. Soit ils utilisent la plateforme eHealth pour tous leurs échanges de données inter et intra réseaux de santé.

3.3. Phase d'implémentation

Une fois que l'infrastructure et les outils la supportant sont opérationnels, il est temps de les utiliser pour produire les données de référence et les partager effectivement entre les différents fournisseurs et consommateurs de données. Ce processus d'implémentation est généralement itératif et a pour objectif final de fournir en output un référentiel de données respectant l'architecture choisie (voir Section 3.2.3) et satisfaisant les SLA spécifiés (voir Section 3.2.4) lors de la phase de conception.

3.3.1. Nettoyer et transformer les données sources

L'objectif de cette étape est d'améliorer la qualité des données sources et de se conformer au format standard d'échange défini durant la phase de conception.

Le nettoyage et les transformations appliquées aux données sources sont similaires aux opérations d'Extract Transform and Load (ETL) utilisées afin d'alimenter un entrepôt de données. Des exemples typiques de transformations sont la normalisation des formats de date, l'insertion de valeur par défaut, la correction des codes postaux, etc.

Le nettoyage comprend également la détection de doublons pouvant faire leur apparition au sein d'une même base de données. La plupart des outils et technologies sont capables d'identifier un grand nombre d'erreurs potentielles. Néanmoins, en ce qui concerne leur correction, seules les plus triviales peuvent être prises en charge de manière automatisée. Certaines erreurs nécessiteront toujours des investigations humaines déterminant les actions à prendre en accord avec le métier.

L'utilisation d'outils adaptés à l'analyse de larges bases de données est fortement recommandée, notamment, pour les fonctionnalités liées au data profiling, data standardisation, data matching, data cleansing (cfr Data Quality).

3.3.2. Consolider les données sources

L'objectif de la consolidation est d'identifier les doublons existant entre différents fournisseurs de données ou au sein du même fournisseur de données pour ensuite les fusionner. La consolidation des doublons est une étape délicate qui doit être menée par le métier et supportée par des outils facilitant l'identification et la résolution des doublons. Des outils sont nécessaires car la phase d'identification peut nécessiter l'analyse et la comparaison de milliers de records.

Néanmoins, même avec des outils, la tâche demeure complexe car si on fusionne trop de données on perd de l'information et, d'un autre côté, si on oublie certains doublons on risque des désynchronisations et l'apparition d'incohérences entre les données utilisées par les différentes applications. Dans tous les cas, les bonnes pratiques préconisent de toujours conserver l'historique des versions afin d'éviter que des opérations de fusionnement ou de défusionnement deviennent irréversibles.

De plus, l'élimination des doublons ne suffit pas, il est nécessaire d'étudier comment les relations existant entre les données vont évoluées lors de la consolidation. C'est pourquoi, les algorithmes de matching et d'intégration sont à la fois sophistiqués et complexes.

Dans le contexte des soins de santé, une contrainte importante est de proscrire la fusion de faux positifs. Par exemple, on doit éviter que deux dossiers patients qui en réalité concernent deux patients distincts ne soient fusionnés par erreur. Les solutions MDM offrent des outils de matching qui calculent différents niveaux de confiance associés à l'identification de doublons. Ces niveaux permettent de déterminer à partir de quel seuil l'élimination des doublons peut être gérée automatiquement ou alors si elle nécessite l'intervention d'un gestionnaire de données. Deux principaux facteurs peuvent fortement limiter la consolidation des données :

1. la mauvaise *qualité des données* sources risque de fortement alourdir le processus de consolidation et de diminuer la qualité du résultat.
2. la consolidation des données sources doit parfois être réduite au strict minimum afin de respecter une *privacy policy* qui pourrait limiter l'utilisation des données à des cas très particuliers.

Cette étape n'est pas obligatoire suivant l'architecture privilégiée, mais fortement conseillée car, sans consolidation, le risque de partager des données incohérentes augmente avec le nombre de données échangées via le référentiel. De plus, certaines anomalies ne sont identifiables qu'après la consolidation de données provenant de sources différentes. La consolidation permet dans une certaine mesure de faire du cross-checking entre plusieurs fournisseurs de données.

3.3.3. Constituer le référentiel

Suivant l'architecture et l'infrastructure d'échange choisies, différentes politiques sont envisageables pour constituer un référentiel. Une fois que le nettoyage, la transformation et potentiellement la consolidation des données ont été effectués, il faut déterminer comment elles vont être stockées et rendues disponibles. Les données de référence peuvent être stockées à leur emplacement initial, ou être dupliquées et/ou migrées vers une base de données spécifiquement dédiée au stockage des données de référence. La migration des données de référence n'est pas improbable et, dans ce cas de figure, il est primordial de minimiser les impacts sur les applications sources. Enfin, il faut offrir différents services (Data Services) permettant de consulter les données du référentiel, de les filtrer, de les créer, de les supprimer, de les mettre à jour, de vérifier certaines anomalies (automatiquement ou manuellement), etc.

Prototype

Dans le cadre de notre étude de cas, nous avons implémenté un prototype de référentiel dont le but est d'illustrer de manière concrète comment un référentiel de données peut être construit et quelles sont les principales fonctionnalités d'un outil MDM. Dans ce contexte, nous avons dû faire un certain nombre d'hypothèses simplificatrices, réductrices qui ne correspondent pas à la réalité. Ces hypothèses concernent principalement le scénario envisagé pour le prototype ainsi que les principaux outputs des phases d'analyse et de conception :

- Scénario du référentiel : considérons trois hôpitaux (Charleroi, Namur, Tournai) qui possèdent chacun des données hétérogènes me concernant (voir Figure 14). Mon médecin généraliste qui a un rôle de centralisateur et d'intégrateur des données me concernant intègre ces différentes données et rend cette version intégrée et résumée disponible aux autres prestataires de soins en utilisant le format SumEHR.

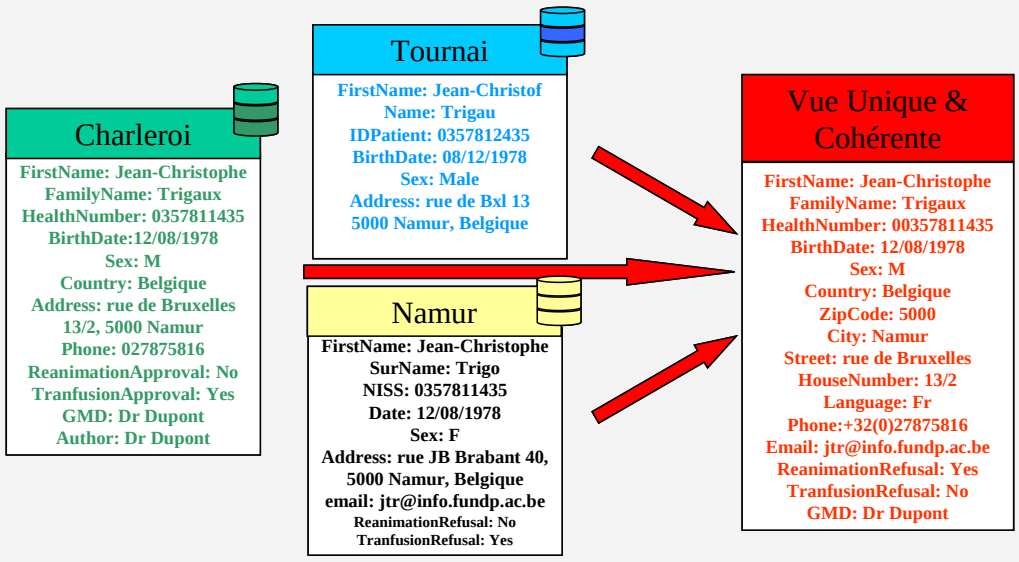


Figure 14 : SumEHR : Scénario

- Standard d'échange (consommation) : SumEHR
- Architecture : Coopération (hybrid)
- Infrastructure :

  Notre choix concernant l'infrastructure s'est orienté vers des outils open source (voir Figure 15) qui se basent sur les principes architecturaux du SOA et sur le langage de programmation JAVA. Ce choix permet la réalisation d'un prototype à moindre coût offrant la possibilité d'illustrer rapidement les concepts du MDM.

  Nous avons sélectionné les outils open source développés par Sun (aujourd'hui racheté par Oracle) dans le cadre du projet Mural, avec :

  - Data distribution Platform :
    - Mode Pull et temps réel
    - Enterprise Service Bus : OpenESB [OpenESB]
    - Serveur d'applications : GlassFish [GlassFish]
  - Data repository : MySQL [MySQL]
  - Data Quality and Governance : MDM Solution: Mural – Master Index Data Manager (MIDM) [Mural]

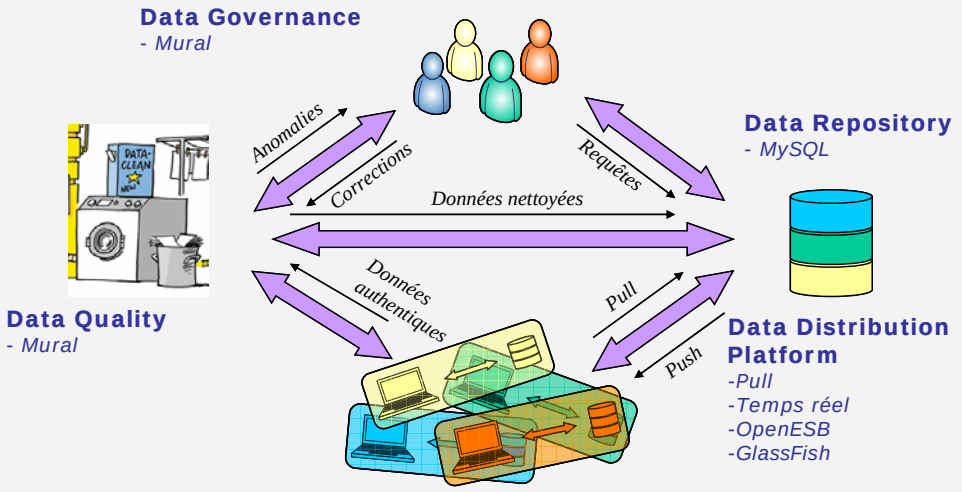


Figure 15 : SumEHR: Infrastructure Open Source

Cependant, nous attirons une nouvelle fois l'attention du lecteur sur le fait que ces hypothèses sont très discutables et qu'elles dépendent fortement des règles de gouvernance qui devront être établies par les experts du domaine. Par exemple, l'intégration des données médicales est fortement limitée et consiste essentiellement à compléter le dossier médical avec par exemple de nouveaux diagnostics, de nouveaux médicaments prescrits, de nouveaux résultats d'examens médicaux, etc. Mais il nous paraît important de pouvoir associer ces nouveaux « évènements » au bon patient et de pouvoir identifier a posteriori les applications fournisseuses de chaque donnée.

De plus, en pratique, de nombreux paramètres sont à prendre en compte dans le choix de l'infrastructure et des outils qui y sont associés. Les principaux paramètres étant l'intégration avec l'existant, l'adéquation avec les besoins métier, la pérennité des fournisseurs d'outils, leurs coûts d'achat et de maintenance. De nombreux autres outils MDM existent (voir section 4.2) et la solution que nous proposons ici n'est certainement pas la plus adaptée dans tous les contextes et certainement pas dans le cas des échanges de données médicales. Mais elle permet d'illustrer les principales fonctionnalités d'un outil MDM facilitant la mise en place d'un référentiel de données. N'oublions pas que l'important n'est pas la solution technologique, ce n'est qu'un support pour l'équipe qui va devoir gérer le référentiel de données avec tous ces aspects organisationnels.

La démarche générale que nous avons suivie pour constituer ce prototype à l'aide de l'outil Mural est la suivante :

1. Dans un premier temps, l'outil Mural (via la plateforme de développement NetBeans) nous demande en input un modèle des données qui seront gérées par le référentiel. Nous avons élaboré ce modèle de données (voir Annexe 9.1.1) ainsi que les différentes codifications (Code Module) qui y sont associées à partir du standard SumEHR.
2. Dans un second temps l'outil Mural nous demande de définir les règles de matching (voir Annexe 9.1.2) qui permettront d'identifier avec un certain degré de probabilité les doublons apparaissant dans les données de référence.
3. Dans un troisième temps, l'outil Mural génère automatiquement à partir de ce modèle et de ces règles de matching trois éléments importants :
    a. Le code SQL de la base de données associée au référentiel.
    b. Les Data Services permettant de détecter les doublons, de consulter, filtrer, corriger, fusionner, défusionner, etc., les

données de référence (voir Annexe 9.1.9).

c. Une interface web faisant directement appel à ces services webs pour gouverner ces données, gérer leur synchronisation et vérifier les différentes transactions qui ont eu lieues.

Le code SQL est ensuite injecté dans une base de données MySQL, l'application composée des différents Data Services est déployée sur le serveur d'application GlassFish et l'interface web est rendue accessible.

4. Dans un quatrième temps, nous devons intégrer ces trois éléments générés et établir les différentes règles de sécurité régissant l'accès à la base de données, l'utilisation des Data Services et l'accessibilité à l'interface web.
5. Enfin, différentes fonctionnalités de synchronisation de données (broadcasting, feeding, etc.) et de traduction entre différents formats de données peuvent être ajoutées au référentiel en se basant essentiellement sur le langage BPEL [BPEL].

Le référentiel résultant de cette implémentation offre différentes fonctionnalités que nous illustrons avec les scénarii suivant :

1. Les **fournisseurs de données** (médecins ou hôpitaux) peuvent envoyer et consulter les données du référentiel en quasi temps réel en utilisant les différents Data Services déployés sur le serveur d'application.
2. Le **référentiel** peut détecter automatiquement certains doublons et incohérences. Suivant la probabilité établie par les règles de matching l'outil peut automatiquement fusionner des doublons ou demander au Data Steward de prendre cette décision ou non (voir Annexe 9.1.3).
3. Le **Data Steward** (mon médecin généraliste) peut :
   a. Fusionner des doublons (voir Annexe 9.1.4),
   b. Identifier les sources d'une donnée de référence (voir Annexe 9.1.5),
   c. Vérifier les transactions qui ont eu lieu (voir Annexe 9.1.6),
   d. Défusionner des données qui n'auraient pas dû l'être (rollback) (voir Annexe 9.1.7),
   e. Lier les données de référence à des sources authentiques qui deviendront par défaut la source de vérité lorsqu'un conflit apparaîtra entre des sources contradictoires (voir Annexe 9.1.8).

Dans ce scénario, pour rappel fictif, la charge de travail demandée au Data Steward est telle que le médecin généraliste ne peut la prendre en charge seul. Notre objectif était de garder le médecin généraliste au centre de la gestion des données de ses patients. Ce rôle impliquerait notamment d'intégrer les données qu'il reçoit de manière électronique des différents fournisseurs de données (autres prestataires de soins, registre national, etc.).

3.3.4. Tester et évaluer le référentiel

Une fois le référentiel constitué et avant de le mettre en production, il est nécessaire de tester la disponibilité des données, les mécanismes de synchronisation, la charge sur les bus de données, les temps de réponse, la fiabilité. Les utilisateurs de données doivent vérifier que les données qu'ils réceptionnent sont correctement interprétables et contextualisables suivant les

besoins qu'ils ont spécifiés. Dans cette étape on ne vérifie pas que l'infrastructure, on vérifie également :

- la qualité des données et la manière dont elles ont été consolidées,
- le résultat du nettoyage et de la consolidation et leur conformité par rapport aux attentes des utilisateurs du référentiel,
- l'utilisabilité du référentiel par rapport aux scénarii d'utilisation et services spécifiés au préalable,
- la disponibilité des données auprès des fournisseurs de données.

3.3.5. Modifier les applications fournisseuses et consommatrices

L'objectif est de faire évoluer les applications fournisseuses et consommatrices en fonction des choix architecturaux et d'infrastructure pour tenir compte du référentiel et satisfaire les contrats d'échanges. Les principales améliorations à apporter sont :

- La capacité à envoyer et recevoir des messages conformes au standard d'échange communément accepté.
- La capacité des applications à mettre en place des mécanismes de synchronisation avec les données de référence.
- La capacité des applications fournisseuses de données à améliorer la qualité de leurs données en interne avant de les exporter vers les données de référence.
- La capacité des applications à d'abord consulter les données de référence avant de créer un nouveau record.

4. MDM : Le support logiciel

Ce chapitre a pour objectif de présenter comment des outils logiciels peuvent supporter un référentiel de données aussi bien pour sa mise en place que pour sa maintenance. Dans un premier temps, nous présenterons les principales fonctionnalités logicielles attendues afin de faciliter la gestion et la maintenance d'un référentiel de données (Section 4.1). Dans un second temps, nous analyserons le marché des outils MDM (Section 4.2). Enfin, nous identifierons les principaux critères à tenir en compte lors de la sélection d'une solution MDM (Section 4.3).

4.1. Fonctionnalités attendues

L'objectif d'un outil MDM consiste à regrouper l'ensemble des données de référence au moyen d'un référentiel standardisé qui met à disposition les données de référence et favorise leur pilotage via une interface commune. Afin d'atteindre cet objectif, les outils MDM devraient fournir un ensemble de fonctionnalités répondant aux exigences et principes du MDM (voir Figure 16). Ces fonctionnalités sont regroupées suivant trois catégories : les fonctionnalités à destination des utilisateurs métier (Section 4.1.1), des administrateurs métier (Section 4.1.2) et des administrateurs techniques (Section 4.1.3).

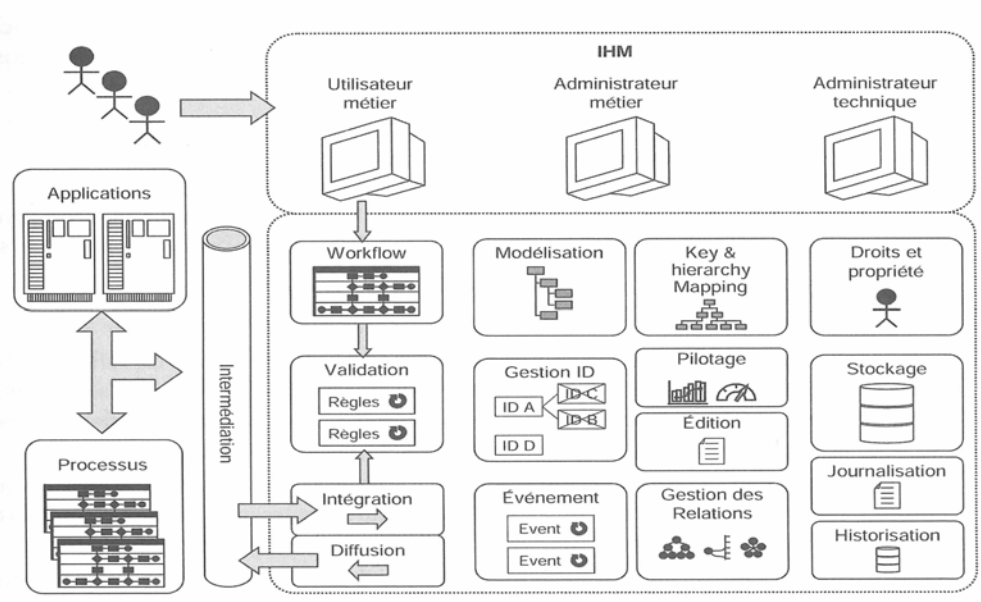


Figure 16 : MDM: Fonctionnalités logicielles [Régnier et al., 09]

4.1.1. Utilisateur métier

La gestion des données de référence requiert une forte implication des utilisateurs métier. Les données de référence doivent être reliées à des processus métier qui vont manipuler et transformer ces données tout au long de leur cycle de vie. Ces processus sont très utiles lors de la saisie des données et l'élaboration d'écrans de saisie généralement autogénérés à partir du modèle de données.

L'acquisition des données peut aussi se réaliser de manière automatique par chargement de message unitaire ou en masse. Cette acquisition en masse peut impliquer certaines transformations sur la donnée initiale. En effet, les données acquises peuvent être « corrigées » par exemple en mettant en conformité les formats ou en éliminant les doublons. Ces transformations sont généralement effectuées à l'aide d'outils de gestion de la qualité (Data Quality Tools[7]) qui sont capables de détecter des anomalies et d'en corriger certaines.

Les anomalies restantes étant redirigées suivant un workflow défini à l'avance, on a la possibilité de normaliser des formats de date, d'insérer des valeurs par défaut ou de corriger des codes postaux. Dans tous les cas, les données reçues doivent être conservées afin de pouvoir à tout moment revenir à la situation initiale.

Lors de l'acquisition des données, une première phase de validation doit déjà être entreprise. Toute donnée entrant dans le périmètre de la solution MDM doit être validée. Les utilisateurs métier sont les principaux acteurs pouvant vérifier la cohérence et la qualité des données. Cette validation à la source peut dans une certaine mesure être automatisée via la définition et l'exécution de règles de validation lors de l'acquisition des données. Ces règles peuvent essentiellement porter soit sur le format des données (règles syntaxiques) soit sur la valeur des données (règles de gestion), soit sur les liens entre les données (règles de cohérence).

En résumé, les utilisateurs métier attendent d'une solution MDM qu'elle offre :

- Acquisition manuelle avec
  - génération automatique d'écrans de saisie
  - Complétion automatique
- Acquisition automatique avec
  - chargement par messages unitaires
  - chargement en masse avec conservation des données initialement reçues
  - détection des données de référence commune
- Validation semi-automatique au fil de l'eau avec
  - vérification des règles syntaxiques
  - vérification des règles de gestion
  - vérification des règles de cohérence
- Correction semi-automatique au fil de l'eau avec
  - Des outils de Data Quality permettant notamment
    - la mise en conformité des formats
    - le dédoublonnage

[7] Smals Research Publication: Data Quality Tools - Evaluer et améliorer la qualité des données (2007), consultable à l'adresse suivante : http://documentation.smals-mvm.be/.

- Des moteurs de Workflow et Business-Rules adaptés pour l'acquisition et la validation des données

4.1.2. Administrateur métier

L'utilisation du référentiel dépend fortement de la manière dont il a été construit, de la manière dont il évolue et de l'adéquation de la structure des données par rapport aux besoins métier. Cet administrateur métier aussi appelé le « Data Steward » doit prendre en charge différents rôles :

1. Le premier rôle est de créer et maintenir un modèle de données commun qui permettra à tous les utilisateurs du référentiel de pouvoir consulter et retraiter ces données. Les aspects liés à la modélisation des données, à la gestion des relations et des hiérarchies entre les données sont primordiaux.
2. Le second rôle est de gérer l'évolution du référentiel en assurant le pilotage des données, le contrôle de la qualité des données, l'édition des données et la gestion de l'annuaire de données reliant les identifiants du référentiel aux identifiants des systèmes sources.

En résumé, les administrateurs métier attendent d'une solution MDM qu'elle offre:

- Modélisation des données et des méta-données avec
  - modélisation et persistance des modèles
  - définition des objets métier et alignement sémantique
- Gestion des relations et hiérarchies entre données
- Pilotage des données et des transactions avec
  - fonctions de recherche avancées dans le référentiel
    - les hiérarchies
    - les relations
    - les critères métier
  - suivi d'indicateurs de qualité des données
  - suivi des transactions avec un système de logging et de reversing
  - tableaux de suivi
  - analyse d'impact d'une modification de procédure et/ou de modèle
- Edition des données et méta-données du référentiel
- Annuaire de données avec
  - gestion des identifiants du référentiel et des instances
  - transcodification des paramètres et des identifiants (contextualisation)

4.1.3. Administrateur technique

La gestion du référentiel implique d'une part le métier et d'autre part l'IT. L'administrateur technique a pour rôle principal d'assurer le développement

et la maintenance de la plateforme technique supportant le référentiel. Cette plateforme doit offrir différentes fonctionnalités :

1. Des fonctionnalités de stockage, d'historisation et de journalisation des données. La solution MDM doit permettre de stocker à fois les données mais aussi les méta-données et les modèles de données correspondants.

   L'historisation et la gestion des versions sont également capitales afin de supporter la gouvernance et le pilotage des données. On doit toujours pouvoir revenir à la (les) version(s) précédente(s) et étudier différents indicateurs dans le temps. On doit aussi pouvoir faire cohabiter différentes versions en même temps.

   Par exemple, le processus de souscription d'une assurance est adapté successivement selon les versions de la réglementation en vigueur. Dans le cadre d'un audit ou d'une analyse des données a posteriori, il est primordial de garder la trace des actions effectuées sur les données via la journalisation de ces données.

2. Des fonctionnalités d'accès et de diffusion des données. La solution MDM doit permettre de vérifier les droits d'accès individuels, suivant la définition des rôles, pour chaque étape du processus de gestion des données de référence. Elle doit aussi répercuter correctement tout changement effectué sur le référentiel aux fournisseurs et utilisateurs de données concernés. Avant la diffusion de données il est généralement nécessaire de les contextualiser ;

   - soit parce que le modèle de données utilisé par le consommateur est différent,
   - soit parce que les formats et les règles sont différentes (ex : par pays, par organisation),
   - soit parce que les valeurs des attributs sont différentes (ex : multilinguisme).

   Le référentiel doit donc être capable de fonctionner sous un mode événementiel et ainsi réagir de manière appropriée à la détection d'un changement ou d'une anomalie, à la correction et à la synchronisation des données. Une gestion événementielle permet de déclencher des actions tels que la synchronisation mais également le pilotage, l'émission d'alertes ou la mise à jour d'informations obsolètes.

3. Des fonctionnalités d'administration et de maintenance du référentiel. La solution MDM doit permettre de définir les droits d'accès. Ces droits d'accès dépendent de la donnée, de son cycle de vie, des attributs que l'on demande à consulter et du rôle joué par le demandeur. La solution MDM doit offrir une gestion fine des versions. En effet, l'historisation « classique » des données (versioning in time) ne suffit pas, il faut aussi permettre de gérer simultanément des versions concurrentes de différents modèles pour une même donnée (versioning in space).

   Par exemple, si une loi est modifiée on doit être capable de gérer simultanément les données de l'ancien et du nouveau régime. Le « versioning in space » consiste à tenir en compte de l'évolution des modèles de données (attribut, structure) répondant aux besoins métier afin de pouvoir gérer différentes versions du modèle de données simultanément.

En résumé, les administrateurs techniques attendent d'une solution MDM qu'elle offre:

- Stockage, historisation et journalisation des données
  - Gestion des versions (données et méta-données)
  - Cohabitation des versions
- Accès et de diffusion des données
  - Vérification des droits d'accès.
  - Synchronisation des données
  - Contextualisation des données
  - Gestion événementielle
- Administration et de maintenance du référentiel
  - Définition des droits d'accès et des rôles

4.2. Analyse de marché des outils MDM

Le marché des outils MDM est actuellement dominé par les éditeurs dits « généralistes » tel que Oracle [Butler, 09b], SAP [Halpern, 07], IBM et Tibco. Les éditeurs de plus petite taille tel que Initiate, DataFlux, Orchestra-Network, etc. se concentrent sur des solutions adaptées à des secteurs métier spécifiques. Le marché des outils MDM est en pleine consolidation avec des rachats successifs entre acteurs du marché. Le plus récent et le plus remarqué est le rachat de Sun par Oracle.

Les leaders comme IBM, Oracle et SAP vont continuer à compléter leur offre, tandis que les acteurs plus petits vont rechercher des partenariats pour étoffer leurs offres de services et atteindre plus de clients. La plupart des outils se tourneront vers des approches plus génériques et multi-domaines permettant de prendre en charge tout type de données avec des solutions capables de gérer plus de 500 millions de données de référence. Les principaux fournisseurs d'outils MDM font évoluer leurs solutions MDM verticales (CDI, PIM) vers des solutions MDM plus horizontales.

La tendance chez IBM et Oracle est l'homogénéisation de leurs offres MDM en intégrant mieux leurs offres SOA et métier avec les différentes solutions MDM acquises grâce à leur politique de rachat. De manière plus générale, les éditeurs de solutions MDM envisagent d'investir principalement leurs efforts dans l'amélioration des services liés à :

- La qualité des données afin de garantir le partage de données fiables dans l'ensemble des processus de l'organisation.
- La gestion des risques avec des processus de prévention des fraudes et de minimisation des risques.
- La gouvernance des données afin de faciliter la mise en place et l'évolution de règles de gouvernance adaptées et gérées par l'organisation.

Aaron Zornes du *MDM institute*[8] prévoit que le marché des outils MDM se stabilisera en 2012, une fois que les équipes en charge du domaine MDM dans les entreprises auront acquis l'expérience et la maturité nécessaires à la mise en oeuvre de ces solutions et de leur intégration [Zornes, 09].

Sur le marché actuel qui est en constante évolution, il existe une profusion d'outils MDM. Nous ne pouvons donc présenter ici qu'un instantané de ce marché dont la pérennité n'est que de courte durée. Dans une premier temps, nous reprenons la catégorisation des produits MDM (voir Figure 17) proposée dans [Régnier et al., 09]. Cette catégorisation sépare trois types de produits MDM :

1. Product Information Management (PIM).
2. Customer Data Integration (CDI).
3. les outils génériques et multi-domaines (Générique).

La taille des bulles n'est pas signifiante en terme de part de marché, elles symbolisent la généricité des solutions du point de vue de leurs modèles de données. Cette catégorisation s'estompera au fur et à mesure que le marché gagnera en maturité. Les éditeurs, comme Oracle, par exemple, font converger de plus en plus leurs solutions afin d'offrir une plateforme de gestion unifiée.

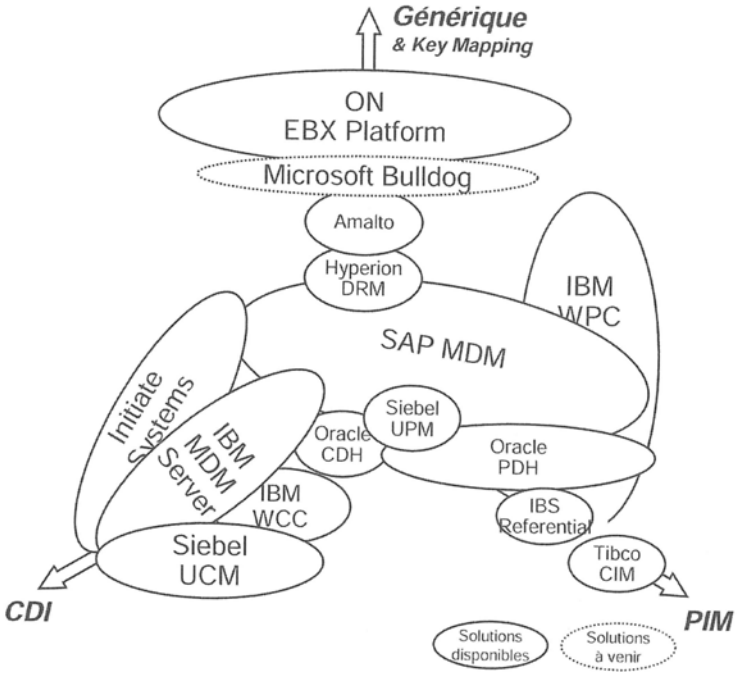


Figure 17 : Catégorisation des outils MDM [Régnier et al., 09]

Dans un second temps, nous présentons un panorama plus détaillé des principaux éditeurs d'outils MDM avec leur(s) produit(s) et une brève description de ceux-ci (voir Table 5).

[8] http://www.tcdii.com/mdm/aboutMDMinstitute.html

Editeur	Produit	Description
Amalto technologies	Xtentis MDM	Solution MDM spécialisée dans l'intégration de données financières et rachetée en juin 2009 par Talend (voir infra).
Dataflux (SAS)	MDM Customer Data Integration	Solution MDM préparamétrée pour gérer les données clients et intégrable avec les outils data quality de Dataflux.
Data Foundations	Onedata for Master Data Management	Solution MDM avec un modèle de données ouvert assurant les services de réconciliation sémantique entre les entités métier.
IBM	Websphere Product Center	Websphere Product Center est un catalogue de référencement de produits issu de l'acquisition de Trigo mi 2004. Il propose néanmoins un modèle générique et présente des outils de collaboration, de workflow ainsi que des écrans préconfigurés pour l'acquisition des données.
	Infosphere MDM Server	Infosphere MDM Server issus de l'acquisition de DWL à la mi 2005 est axé sur les clients avec un modèle de données relativement fermé. Cette solution est également intégrée à des solutions d'alimentation et de qualité de données issues d'Ascential (MQ Serie, Information Server).
Initiate Systems	Initiate Master Data Service	L'éditeur américain propose différentes solutions MDM se basant sur des modèles de données spécifiques à certains domaines (Initiate Customer, Initiate Citizen, Initiate organisation, Initiate Patient, Initiate Provider).
Microsoft	Bulldog	Bulldog est la solution MDM en cours de développement chez Microsoft issue de l'acquisition de Stratature en 2007. Bulldog devrait être intégré à Sharepoint et sa date de sortie devrait coïncider avec la prochaine release d'Office.
Oracle	Product Information management	Product Information management est une solution PIM résultant de l'acquisition de Retek.
	Customer data Integration	Customer data Integration est une solution CDI résultant de l'acquisition de Siebel.
	Hyperion DRM	Hyperion DRM résulte de l'acquisition de Hyperion qui lui même avait acquis Razza. Cette solution est principalement axée sur la gestion des méta-données.
	Master Data Management Suite (Sun)	Solution MDM intégrée à la suite SOA Javacaps développée par Sun et qui a été récemment acquise par Oracle.
	Mural (Sun)	Version OpenSource de la solution Master Data Management Suite et fortement intégré à Glassfish, OpenESB et à l'environnement de développement de Sun (NetBeans). Sun ayant été racheté par Oracle, son avenir est incertain et dépendra de la politique d'Oracle concernant le MDM.

Orchestra Networks	EBX.Platform	EBX.Platform est une solution basée sur un modèle générique de données et spécialisée dans la génération de code à partir des modèles (Model Driven Development), la gestion de versions et les systèmes de workflow. Orchestra collabore également avec Ilog afin de compléter sa solution avec un moteur de règles puissant (Irule).
SAP	SAP Netweaver MDM	La solution MDM résulte de l'intégration de la solution MDM de A21 et de la solution EAI de SAP (Netweaver XI).
Siperian	MDM Hub (TM)	La solution MDM était initialement dédiée au monde pharmacologique. Elle se base néanmoins sur une approche générique et orientée modèles permettant la génération automatique de code. Cette solution est également disponible sur Appexchange de salesforce.com.
Smartco	Smart Financial Data Hub	Comme son nom l'indique, cette solution est issue du monde financier et spécialisée dans la gestion des données de référence liées au marché, aux localisations, aux transactions et aux analyses financières.
Talend	Talend Open MDM	À l'origine, Talend était spécialisé dans les outils open source liés à l'intégration et à la qualité des données. Suite au rachat de la solution de Master Data Management du Français Amalto Technologies, spécialisé dans les échanges B2B, l'ambition de Talend est d'offrir une version open source couvrant l'intégralité des besoins d'une solution MDM. C'est-à-dire : • l'intégration de données (ETL) : Talend Integration Suite ; • la qualité des données : Talend Open Profiler pour l'analyse et Talend Data Quality pour le nettoyage des données ; • le MDM : nouvelle solution bâtie autour du noyau de la solution Amalto Xtensis.
Teradata	Teradata MDM	La solution MDM de Teradata est fortement associée aux technologies d'entrepôt de données développées par Teradata. Cette solution est caractérisée par une hypercentralisation des données produits essentiellement adaptée au secteur de la distribution et de l'industrie.
Tibco	Collaborative Information Manager 7.2.1	Collaborative Information Manager est une solution MDM initialement orientée produits mais qui tend vers une solution plus générique. Cette solution résulte essentiellement de l'acquisition de Velosel et de son intégration avec le middleware de Tibco. La solution s'intègre également avec les solutions d'intelligence artificielle de Netrics permettant : • d'améliorer considérablement la recherche de doublons d'enregistrements au sein de grands ensembles de données grâce à une modélisation mathématique avancée.

		• d'éviter aux utilisateurs de devoir développer leurs propres règles de dédoublonnage grâce à l'autoapprentissage proposé par Netrics.

Table 5 : MDM: Panorama des outils

Le coût des licences d'une solution MDM complète et intégrée varie entre 1 et 2 millions de $. Ce prix doit être toutefois nuancé car les éditeurs offrent généralement la possibilité d'acheter certains composants logiciels séparément. De plus, les logiciels des éditeurs généralistes tels que Oracle, IBM ou SAP sont déjà présents dans la plupart des grandes entreprises et administrations, ce qui leur permet d'alléger sensiblement le prix de ces licences.

Le coût des licences n'est évidemment pas le seul à devoir être pris en considération. Une étude[9] sponsorisée par Microsoft auprès de 188 entreprises principalement en Amérique du nord (59%) et en Europe (20%) révèle que :

- les coûts d'implémentation d'une solution MDM s'élèvent en moyenne à 7 millions de $ avec une médiane de 3.5 millions de $.
- les coûts de maintenance d'une solution MDM s'élèvent en moyenne à 8 équivalents temps plein avec une médiane de 4,5 équivalents temps plein.

4.3. Quel outil MDM choisir ?

Vu le coût relativement conséquent des solutions MDM, on peut dès le départ s'interroger sur la nécessité d'une telle solution et sur notre capacité à développer in house ce même type de solutions à moindre coûts.

Il n'existe pas de réponses immédiates à ces questions. Tout dépend du contexte, de la situation de départ (compétence, infrastructure), des investissements que l'on est prêt à consentir, du niveau de contrôle sur le code des applications fournisseuses et consommatrices de données, des fonctionnalités qui nous semblent prioritaires, de l'efficacité attendue et du caractère spécifique ou non de la solution envisagée.

Dans le contexte de la sécurité sociale, certains principes du MDM et différentes technologies liées au MDM sont déjà appliqués. Citons, par exemple, l'architecture de type répertoire virtuel (BCSS, eHealth), l'architecture de type coopération (Athena), la data quality (Data Quality Cel), les entrepôts de données, les technologies d'intégration (ETL), les technologies BI (Cognos, SAS), etc. Au niveau de la gouvernance des données, différentes initiatives très intéressantes ont vues le jour, telles que :

- l'application glossaire en production depuis 2001 élaborant un dictionnaire de données avec gestion des versions, workflow de validation afin de documenter les messages XML DmfA et DRS.
- Le prototype FALCO en tant que système de knowledge management pour la gestion des anomalies DmfA.

À l'heure actuelle, les outils MDM sur le marché proposent différentes technologies qui ont l'avantage d'être de plus en plus intégrées et de privilégier l'échange de données en temps réel avec des outils performants d'amélioration de la qualité et d'intégration des données. Cependant, ces outils pêchent

[9] http://www.it-director.com/business/content.php?cid=11194

crucialement au niveau de la gouvernance des données. En effet, la gouvernance des données possède une dimension organisationnelle et humaine qui peut difficilement être prise en compte par les fournisseurs d'outils MDM. L'idéal serait de pouvoir rapprocher nos projets de gouvernance de données avec les différentes technologies MDM existantes ou futures.

Pour sélectionner une solution MDM, outre les coûts et les fonctionnalités supportées par la solution, différents critères sont généralement envisagés :

- **Intégrabilité** – La solution MDM s'intègre-t-elle facilement à l'infrastructure existante ?
- **Généricité** – La solution MDM peut-elle prendre en compte tout type de données de référence ?
- **Précision** – La solution MDM fournit-elle des règles de matching suffisamment fins pour correspondre à la réalité du terrain ? (Minimiser le nombre de faux positifs et négatifs).
- **Scalability et performance** – La solution MDM est-elle capable d'évoluer en fonction de vos besoins sans sacrifier la performance ?
    - La solution est-elle capable de prendre en charge l'évolution des modèles de données aussi bien au niveau du référentiel qu'au niveau des fournisseurs de données (gestion des versions) ?
    - La solution est-elle capable de faire cohabiter différentes versions des modèles de données ? Suite à un changement légal, est-elle capable de gérer à la fois les données qui sont régies par l'ancien régime et celles régies par le nouveau ?
    - La solution est-elle capable de supporter des fluctuations importantes au niveau du volume de données échangées et des transactions effectuées ?
- **Vitesse et facilité d'implémentation** – Combien de temps et quelle énergie seront nécessaires pour rendre opérationnel la solution MDM ?
- **Expérience et expertise du fournisseur** – Est-ce que le fournisseur de la solution MDM possède suffisamment d'expérience et de know-how pour satisfaire les besoins spécifiques à votre métier ?

5. MDM : Comment faciliter l'approche SOA ?

Ce chapitre a un double objectif : (1) examiner les liens entre les approches MDM et SOA et (2) analyser comment et pourquoi le MDM pourrait faciliter le SOA. Dans un premier temps, nous rappellerons brièvement ce qu'est l'approche SOA (Section 5.1) et ensuite nous ferons le rapprochement ente les approches MDM et SOA (Section 5.2).

5.1. L'approche SOA

Le SOA[10] est un paradigme d'architecture logicielle destiné à augmenter l'interopérabilité, l'agilité et la réutilisabilité logicielles principalement au niveau des processus. Dès 1996, le terme SOA a été défini comme "*A Service-Oriented Architecture is an enterprise-scale IT architecture for linking resources on demand. These resources are represented as business-aligned services which can participate and be composed in a value-net, enterprise, or line of business to fulfill business needs. The primary structuring element for SOA applications is a service as opposed to subsystems, systems, or components*"[11].

Cependant, le SOA et le développement des services génériques ne résolvent les problèmes d'hétérogénéité qu'en surface. Si aucune mesure n'est prise, l'hétérogénéité subsiste au niveau des données. Une architecture SOA offre une plateforme intégrée pour accéder de façon générique aux données métier. Cependant, sans une uniformisation de la structure et des modèles sémantiques des données dans les applications, l'hétérogénéité subsiste au niveau des données. Dès lors, les informations remontées par ces services restent incohérentes. Dans ces conditions, le déploiement de services métier à valeur ajoutée ne peut se faire que si la qualité des données qu'ils utilisent est garantie.

En effet, selon Forrester, "*the harmonization of product and customer data provides a cornerstone on the road to SOA. With true master data, Web services and the related business processes will become more accurate, timely, and efficient, leading to improved ROI on existing investments as well as improved business intelligence*" [12].

[10] Smals Research Publication: Service Oriented Architecture: Software als een dienst (2006)
[11] http://www.ibm.com/developerworks/library/ws-soa-design1/
[12] http://www.tibco.com/software/master-data-management/default.jsp

5.2. Rapprochement entre MDM et SOA

Dans le cadre du SOA, l'attention a tendance à se focaliser sur la création et la mise à disposition de services web. Cependant, si les données qu'ils manipulent sont inconsistantes, tant les applications que les processus métier transversaux à plusieurs systèmes ne pourront fournir les résultats et les bénéfices escomptés. C'est pourquoi, le MDM est généralement considéré comme une approche facilitant la mise en place du SOA. Toutefois, les deux approches restent complémentaires :

- le SOA a besoin du MDM pour structurer, uniformiser et consolider les données hétérogènes.
- le MDM a besoin du SOA pour partager et distribuer les données consolidées entre applications hétérogènes via des Data Services.

Sans le MDM, le SOA offre des services rationalisés mais dont les données sont dispersées et pour lesquels aucune cohérence sémantique n'est garantie. Le développement et la maintenance de ces services deviennent donc dépendants des connexions établies avec les différentes sources de données potentiellement hétérogènes.

Lorsque les données de référence sont dupliquées dans plusieurs systèmes, le SOA favorise l'intégration des données de manière standard en utilisant les technologies XML et la diffusion de ces données intégrées en utilisant les technologies ESB [Bonnet, 07].

D'une part le MDM veut rationaliser la gestion des données métier et, d'autre part, le SOA veut rationaliser la gestion des services métier. Dans les deux cas, cette rationalisation se base sur les mêmes principes :

1. **Réutilisabilité :** le SOA favorise la réutilisation des services métier par les différents processus de l'entreprise. Le MDM favorise le partage des données entre applications hétérogènes.
2. **Annuaire et référencement :** le SOA expose les services métier à différents consommateurs via son annuaire de services. Le MDM expose les données de référence à différents consommateurs via son référentiel de données.
3. **Abstraction :** le SOA permet de masquer les détails de l'implémentation via ses services. Le MDM masque la complexité des modèles de données via son référentiel.
4. **Alignement IT/Métier :** que ce soit le SOA, le MDM ou le BPM, on se rapproche de plus en plus de la dimension métier. On décrit des processus métier (BPM) implémentés via des services métier (SOA) qui manipulent des données métier (MDM).

L'abstraction des données via l'utilisation de Data Services est le principe le plus profitable aux deux approches. L'abstraction de données permet :

1. de mettre à disposition les données de manière indépendante aux données physiques,
2. de découpler les données physiques et les services proprement dits,
3. d'émuler la donnée désirée ainsi que son format sans devoir recréer ou restructurer les données physiques,
4. de définir une couche d'abstraction de données qui répondra au mieux aux exigences métier traduites dans les services métier,
5. de faciliter l'intégration des données à un niveau logique.

L'alignement entre l'IT et le métier est également un élément en faveur de la complémentarité des approches SOA, MDM et BPM (voir Figure 18). L'approche MDM favorise la flexibilité des processus métier qui se basent désormais sur des données métier cohérentes et unifiées. Suivant les exigences métier, les processus métier peuvent évoluer indépendamment des données métier.

L'avantage pour une organisation est de pouvoir assurer la pérennité de ses données métier quel que soit les processus qui vont les manipuler. Cette séparation entre processus et données métier prend tout son sens lorsque les données d'une organisation sont utilisées par les processus d'une autre organisation.

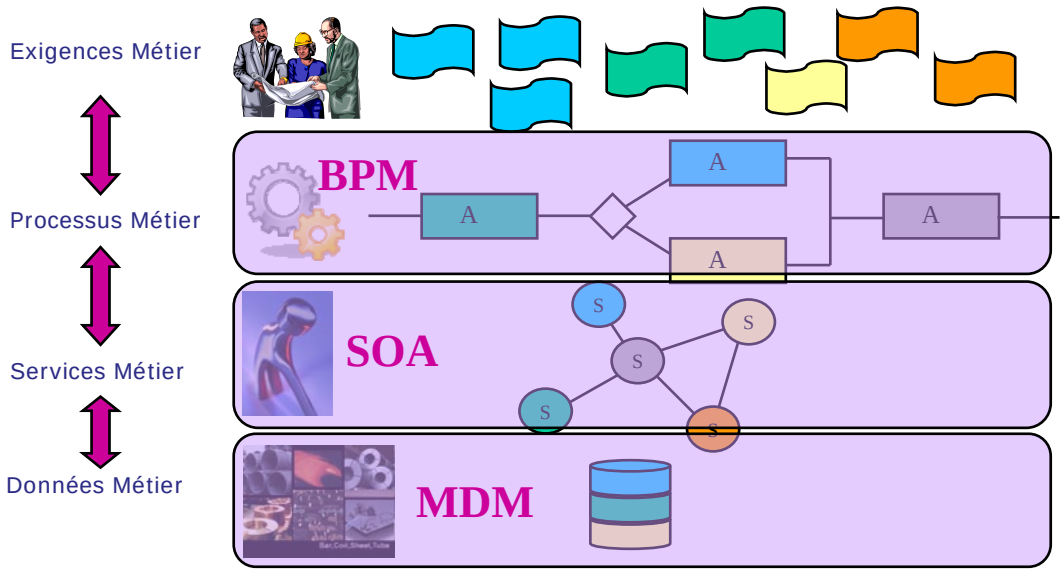


Figure 18 : Alignement MDM, SOA, BPM

Le MDM et le SOA sont deux approches complémentaires qui contribuent l'une à l'autre. Une question délicate est donc de distinguer clairement les contributions de chacune d'entre elles. On pourrait aussi reformuler la question en s'interrogeant sur les circonstances dans lesquelles il serait judicieux d'appliquer l'approche MDM dans un contexte SOA et inversement.

- Quand l'approche SOA a-t-elle besoin de l'approche MDM ?
  - Est-ce que mes services sont impactés par des changements dans mes bases de données ?
  - Est-ce que mes services font appel à deux ou plusieurs bases de données pour extraire ou mettre à jour la même donnée ?
  - Est-ce que mes analystes et développeurs ont besoin de comprendre les modèles complexes de données pour pouvoir s'interfacer avec un nouveau système ?
  - Est-ce que deux ou plusieurs bases de donnés consultées possèdent des données redondantes ?
  - Existe-t-il des données dupliquées, incomplètes, manquantes, hétérogènes entre différentes bases données consultées ?

  - Existe-t-il des data services offrant une interface unique pour consulter et mettre à jour les données de référence ?
  - Est-ce que mes services utilisent la même définition sémantique pour les donnés de référence ?
  - Est-ce que des services intermédiaires de vérification et de mise en conformité de la qualité des données vous paraissent utiles ?

- Quand l'approche MDM a-t-elle besoin de l'approche SOA ?
  - Est-ce que mon référentiel de données supporte XML en tant que format standard et extensible de description de données ?
  - Est-ce que mon référentiel de données met à disposition les données de référence via des services de recherche, de consultation, de mise à jour, etc. ?
  - Comment mes consommateurs de données peuvent-ils consulter les données stockées dans mon référentiel ?
  - Comment mes fournisseurs de données peuvent-ils mettre à jour les données stockées dans mon référentiel ?
  - Est-ce que mon référentiel est capable de partager ses données en quasi temps réel ?
  - Est-ce que mon référentiel de données peut échanger des méta-données avec d'autres référentiels de données ?

6. Conclusion

Tout au long de cette étude, nous avons présenté les enjeux de l'approche MDM, ses principes fondamentaux, les différentes étapes à suivre pour mettre en place cette approche et comment des outils logiciels pouvaient la supporter.

Nous avons vu que l'approche MDM intègre différentes méthodes et technologies afin de limiter la perte de contrôle sur nos données critiques et de rationaliser leur gestion lorsqu'elles sont partagées entre plusieurs applications. En effet, l'enjeu principal du MDM est de faciliter la gestion des données transversalement à différentes applications en mettant en place une organisation de circonstance supportée par un référentiel de données.

Un référentiel de données consiste essentiellement en une application qui supervise la gestion d'une banque de données alimentée par plusieurs fournisseurs de données et consultable par différents consommateurs. Ce référentiel se focalise sur les données à haute valeur ajoutée dont la qualité et l'accessibilité sont cruciales pour les partenaires métier. Ces données sont aussi appelées données de référence ou master data. L'objectif du référentiel est d'intégrer et d'uniformiser les différentes données reçues ou collectées pour ensuite les rendre facilement accessible. Nous avons vu que cette intégration peut être réalisée de manière logique (annuaire de données) ou physique (DB commune ou centralisée).

Pourtant ce référentiel n'est qu'un support technique. Il faut avant tout mettre en place une organisation capable de créer et maintenir des processus efficaces de gestion des données de référence (**stratégie de gouvernance**). L'aspect technologique ne joue qu'un rôle mineur dans l'approche MDM, dont la difficulté principale réside dans son caractère multi-organisationnel. L'essentiel est de s'accorder sur un modèle commun des données qui vont être incluses dans le référentiel, sur la manière dont ces données doivent être interprétées (sémantique) et sur les règles de gouvernance associées à ces données. Cela inclut à la fois les aspects liés à la qualité, la gestion des versions, la sécurité et le caractère privé des données.

La **définition** de cette stratégie de gouvernance dépend fortement de notre capacité à évaluer la situation existante concernant nos données. L'utilisation des techniques de Data Profiling pourrait être très profitable dans cette situation. Le data profiling consiste à examiner différentes sources de données existantes (bases de données, fichiers,...) et à collecter des statistiques et des informations sur ces données. Ces informations permettent d'avoir une vue globale sur les données, de répertorier les données redondantes ou celles qui pourraient être réutilisées, de mesurer la qualité des données, d'identifier les données à risque pour lesquelles une stratégie adaptée devrait être déployée, etc.

L'**application** de cette stratégie de gouvernance dépend de notre capacité à réutiliser des services communs (Data Services) permettant d'assurer la gestion des données de manière uniforme. Dans le cadre d'une architecture de type SOA, les data services sont des services réutilisables qui préservent l'intégrité des données de référence tout au long de leur cycle de vie (de leur capture jusqu'à leur destruction) même lorsque différentes applications les manipulent. L'objectif est de distinguer clairement les services implémentant la logique métier et les services spécifiquement dédiés à la gestion des données (accessibilité, enregistrement, consultation, modification, destruction, contrôle, etc.).

Le respect des principes MDM et l'utilisation d'un référentiel de données permettraient de se réapproprier ses données métier, de les enrichir et d'assurer leur pérennité, indépendamment des processus qui les manipulent. L'approche MDM permet de mutualiser les efforts pour assurer la synchronisation, le partage et la qualité des données de référence à travers les différents silos d'informations en quasi temps réel.

Cette capacité à partager de l'information est critique pour les grandes multi nationales ainsi que pour les gouvernements. Que ce soit dans le cadre des soins de santé, de la sécurité sociale ou de la sécurité du territoire, l'échange de données entre les organismes concernés est crucial. Les critiques adressées au gouvernement américain par les différents rapports d'étude concernant les attentats du 11 septembre 2001 sont un exemple révélateur. Ces critiques soulignaient le manque d'échange d'information entre les systèmes informatiques des agences de renseignement américaines (FBI, CIA et NSA) ainsi qu'à l'intérieur des départements de ces mêmes agences.

Outre les défis récurrents à une approche MDM, les solutions MDM adaptées au secteur public sont aussi fortement sujettes aux règles et réglementations légales dont la transcription dans un référentiel peut se révéler fastidieuse. La problématique principale se situe donc au niveau de la gouvernance des données.

L'approche MDM est directement liée au partage des données entre acteurs de la sécurité sociale et/ou des soins de santé. Aussi bien la BCSS que eHealth ont été des précurseurs dans le domaine. Chaque jour, de nombreuses banques de données gèrent et partagent des données de référence. Quelques exemples révélateurs sont les banques de données gérant la signalétique des citoyens (registre national), la signalétique des travailleurs (SIGeDIS), la carrière des employés, l'identification des entreprises (BCE), les vaccinations, les dons d'organes, la description des médicaments, etc.

Dans ce contexte, plusieurs technologies et principes relatifs au MDM sont déjà appliqués. Citons, par exemple, les principes liés à l'architecture de type répertoire virtuel (BCSS, eHealth), l'architecture de type coopération (Athena), la data quality (Data Quality Cel) et les techniques liées aux entrepôts de données, aux technologies d'intégration (ETL), aux technologies d'analyse de type BI (Cognos, SAS), aux mécanismes de synchronisation (mutation DmfA), etc. Au niveau de la gouvernance des données, différentes initiatives très intéressantes ont également vues le jour, telles que :

- L'application glossaire, en production depuis 2001, élaborant un dictionnaire de données avec notamment gestion des versions et workflow de validation afin de documenter les messages XML DmfA et DRS.
- Le prototype FALCO en tant que système de knowledge management pour la gestion des anomalies DmfA.

Clairement, l'approche MDM est au cœur de l'eGovernment et de eHealth. L'appropriation des concepts MDM par les institutions gouvernementales représente une opportunité de faciliter la collaboration entre les consommateurs

et les fournisseurs de données, de faciliter la mise en place d'une approche SOA, d'améliorer les services associés aux banques de données et de mutualiser les efforts en terme de synchronisation des données, d'amélioration de leur qualité et de gestion des anomalies.

En ce qui concerne les outils MDM, ils ne nous semblent pas, à l'heure actuelle, suffisamment matures. Leur principale valeur ajoutée serait l'intégration de l'ensemble des technologies qui sont nécessaires à la gestion d'un référentiel de données dans un outil unique. Même si la complémentarité des technologies pourrait devenir un avantage indéniable, leur intégration est encore compliquée. Le support lié à la gouvernance des données est pour l'instant la pierre angulaire qui manque crucialement aux outils MDM. Enfin, signalons également que l'utilisation de ces outils doit être envisagée avec précaution car leur coût d'achat est relativement important et leur intégration avec les applications existantes peut s'avérer très délicate.

7. Références

[Bell, 08] Michael Bell, Service-Oriented Modeling (SOA): Service Analysis, Design, and Architecture, 2008.

[Bonnet, 07] Pierre Bonnet , Master Data Management & SOA, Orchestra Networks, 2007.

[Bonnet, 09] Pierre Bonnet, Management des données de l'entreprise: Master Data Management et modélisation sémantique, Orchestra Networks, 2009.

[Bouzeghoub, 09] Intégration des données d'entreprise, Formation Capgemini, 2009

[BPEL] http://www.oracle.com/technology/products/ias/bpel/index.html

[Butler, 09a] David Buttler, Oracle Master Data Management: Executive Overview, Oracle, 2009.

[Butler, 09b] David Buttler, Oracle Master Data Management: Technical Overview, Oracle, 2009.

[Datactics, 05] Case Study: Bank of Ireland (NI), Datactics, url: http://www.datactics.com/Resources/case-studies/Bank-of-Ireland, 2005.

[Davies, 09] Marcus Davies, Case study: Using MDM to inject Agility into Healthcare, Master Data Management Summit Europe 2009, 2009.

[Elmasri, 06] Elmasri, R. & Navathe, S. B. Fundamentals of Database Systems (5th Edition) Addison Wesley, 2006.

[Faucher, 07] Hubert Faucher & Philippe Latapie, Un exemple de décisionnel de très grande volumétrie : le SID de l'UNEDIC, les rencontres CIO du décisionnel, 2007.

[Ferguson, 08] Mike Ferguson, Getting Started With Master Data Management, DataFlux, 2008.

[GlassFish] https://glassfish.dev.java.net/

[Halpern, 07] Master Data Management : Extracting value from your most important intangible asset, SAP, 2007.

[Initiate, 03] Success Stories: Capital Health Finds Cure for Linking and Sharing Data, Initiate, url: http://www.initiate.com/customers/success_stories/Pages/CapitalHealthfindscureforlinkingandsharingpatientdata.aspx , 2003.

[Mural] https://mural.dev.java.net/

[MySQL] http://www.mysql.com

[OpenESB] https://open-esb.dev.java.net/

[Régnier et al., 09] Régnier-Pécastaing, F.; Gabassi, M. & Finet, J., Dunod (ed.), MDM: Enjeux et méthodes de la gestion des données, Logica management Consulting, 2008.

[Shankar, 09] Ravi Shankar, Seven Ways to Reduce IT Costs with Master Data Management, Siperian, 2009.

[Zornes, 09] Aaron Zornes, MDM Boot Camp, Master Data Management Summit Europe 2009, 2009.

8. Glossaire

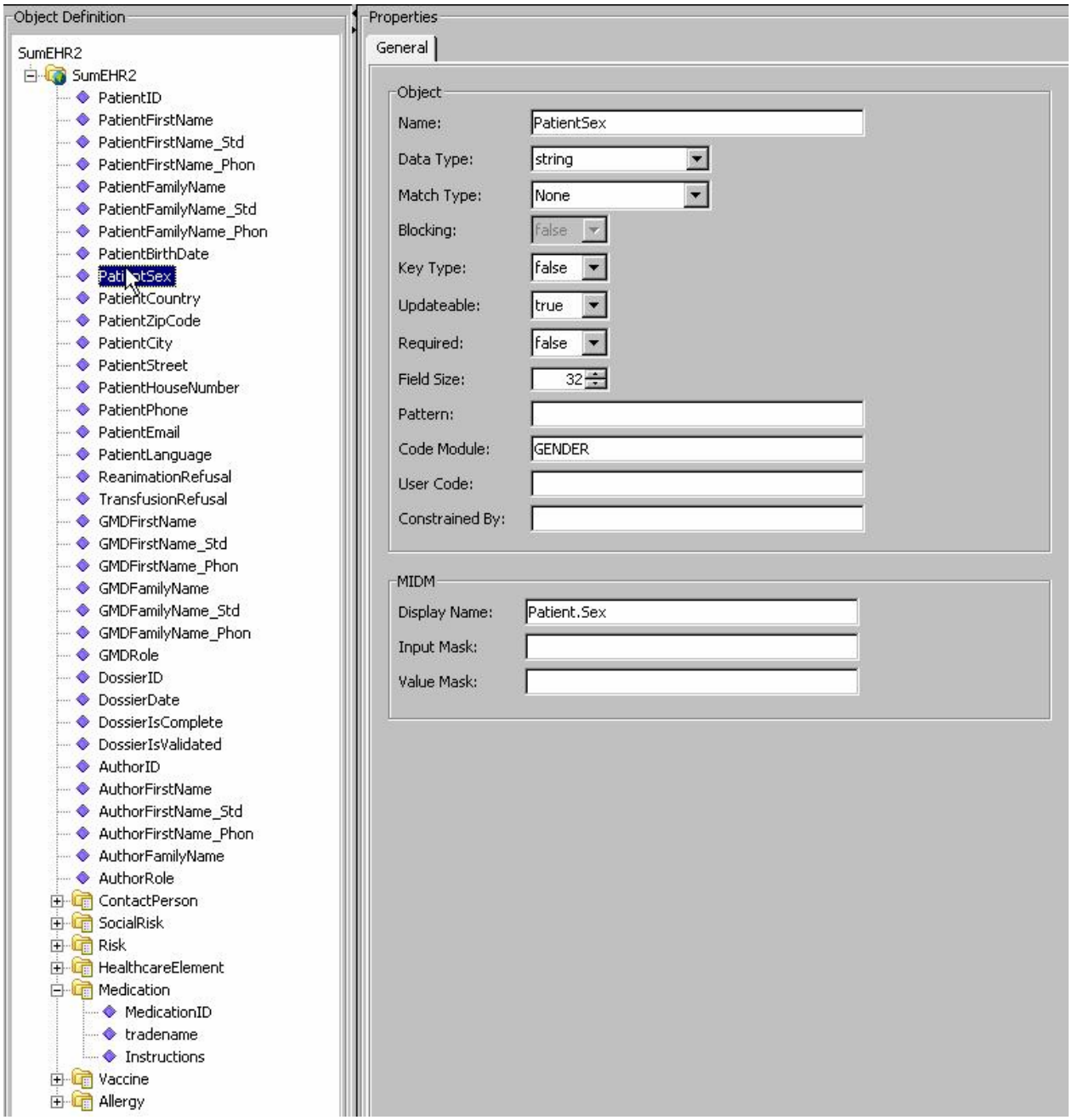
BPM	Business Process Management – Terme générique désignant à la fois la modélisation et la traduction, à l'aide d'un « moteur », des processus modélisés dans la réalité de l'entreprise.
BI	Business Intelligence – Techniques d'organisation et outils de travail pour la gestion des informations et des documents de pilotage de l'entreprise.
CDI	Customer Data Integration – Combinaison de technologies, processus et services nécessaires pour créer et maintenir une vue exacte, complète et à jour des clients.
EAI	Entreprise Application Integration – Bus de communication permettant des échanges de messages entre applications, permettant d'éviter la multiplication des échanges « pont à point ».
EII	Entreprise Information Integration – Catégorie de logiciels qui permet l'intégration de données éparpillées dans une base de données virtuelle.
ERP	Enterprise Ressource Planning – Catégorie de logiciels qui permet de gérer l'ensemble des processus opérationnels d'une entreprise, en intégrant l'ensemble des fonctions de cette dernière comme la gestion des ressources humaines, la gestion comptable, financière, mais aussi la vente, la distribution, l'approvisionnement, le commerce électronique.
ESB	Enterprise Service Bus – Bus applicatif permettant de router de manière sécurisée l'appel à des services.
ETL	Extract Transform Loading - Technologie informatique middleware (intergicielle) permettant d'effectuer des synchronisations massives d'information d'une base de données vers une autre.
MDM	Master Data Management – Démarche qui se focalise sur la création et la maintenance d'un référentiel de données facilitant la gestion et le partage des données de référence.
PIM	Product Information Management - Combinaison de technologies, processus et services nécessaires pour créer et maintenir une vue exacte, complète et à jour des produits.
Registry	Annuaire d'objets informatiques ne contenant que des informations structurées.
Repository	Annuaire d'objets informatiques contenant des descriptions complémentaires non structurées.
SLA	Service Level Agreement – Engagement sur la disponibilité et la qualité des données.
SOA	Service Oriented Architecture – Modèle d'architecture applicative mettant en œuvre des connexions en couplage lâche entre divers composants logiciels (ou services).
UML	Unified Modelling Language - Langage graphique de modélisation des systèmes d'information qui se veut standard et orienté objet.
Web service	Composant applicatif indépendant accessible par l'entremise d'une interface bien définie.
XML	eXtensible Markup Language – Langage de description sous forme de balises permettant de décrire une structure de données.
XSD	XML Schema Definition – Langage de description décrivant de manière structurée le type de contenu, la syntaxe et la sémantique d'un document XML.

9. Annexe

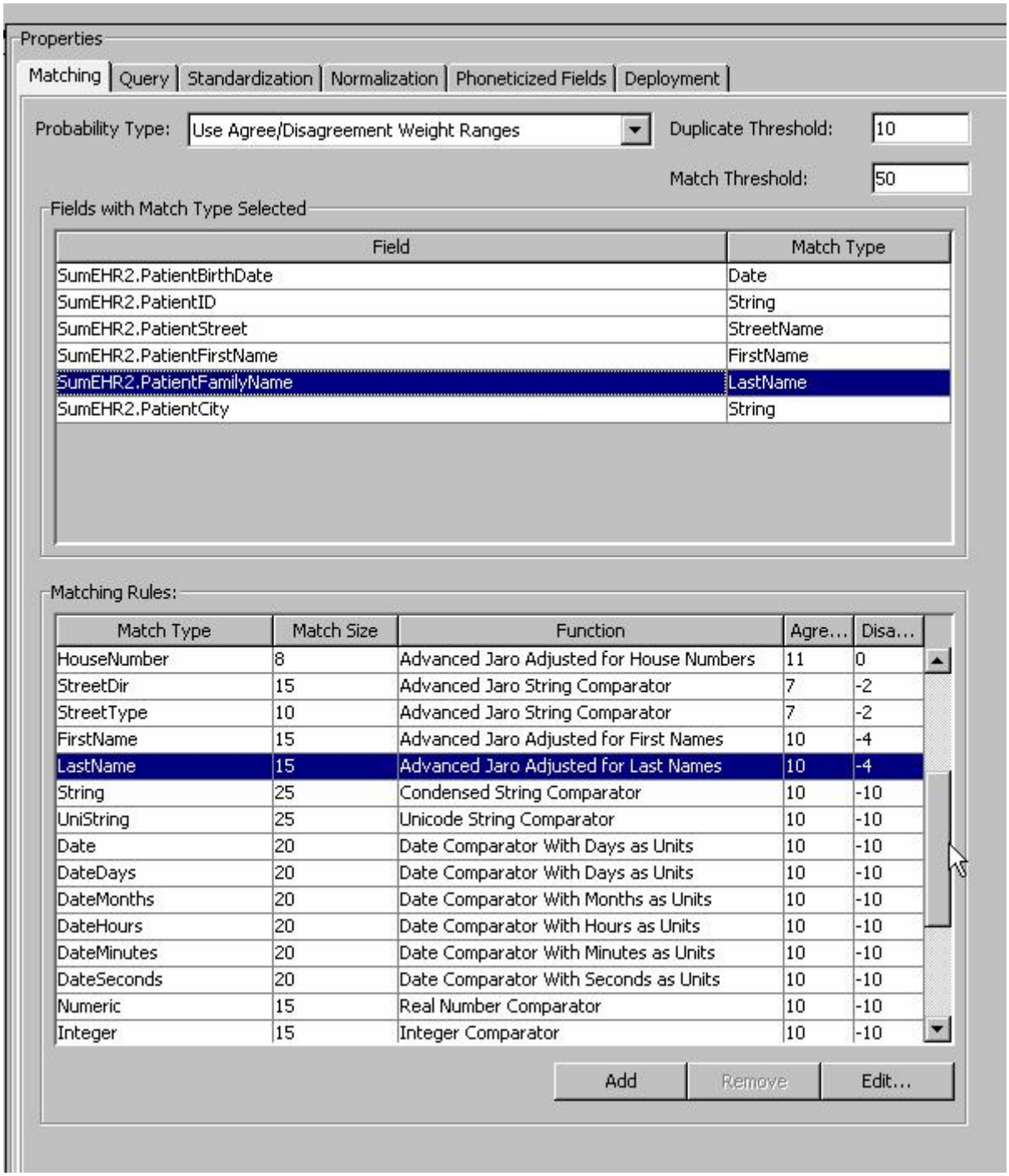
9.1. Captures d'écran du Prototype

1. Élaboration du modèle de données
2. Élaboration de règles de matching
3. Détection de doublons
4. Fusionnement de doublons
5. Identification des sources
6. Historique des transactions
7. Défusionnement de faux positifs
8. Définition des sources authentiques
9. Quelques Data Services du référentiel

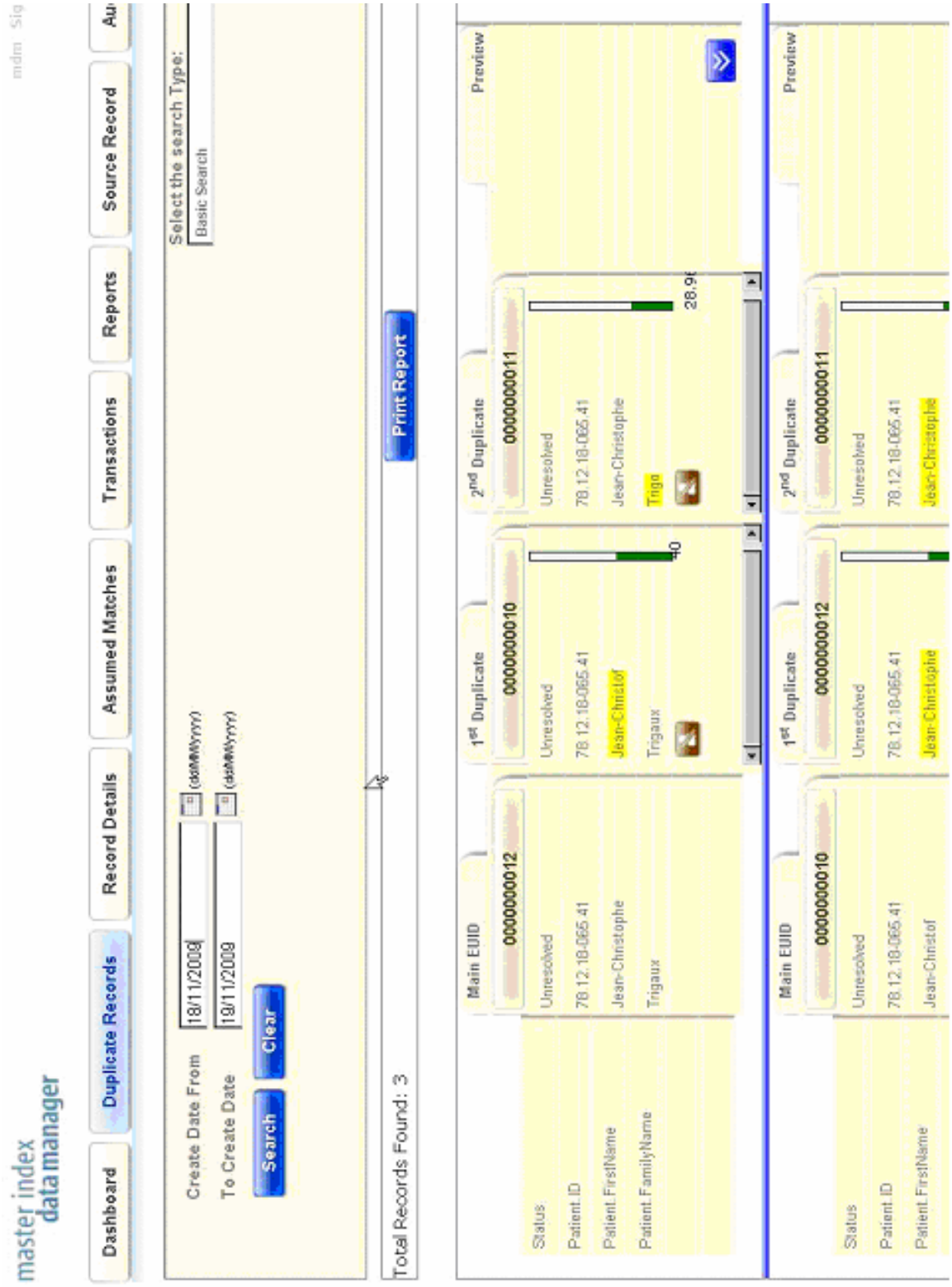
1. Élaboration du modèle de données



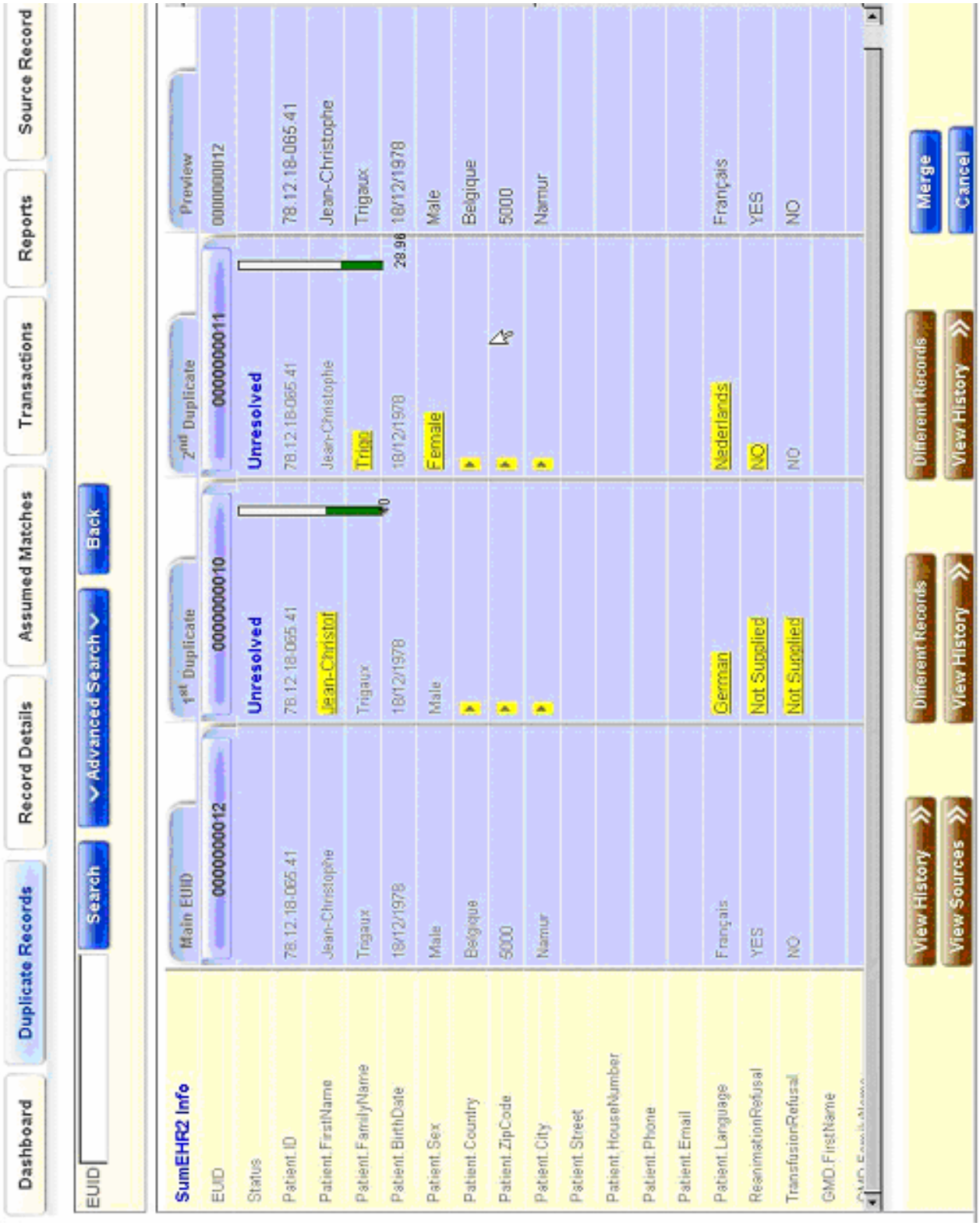
2. Élaboration de règles de matching



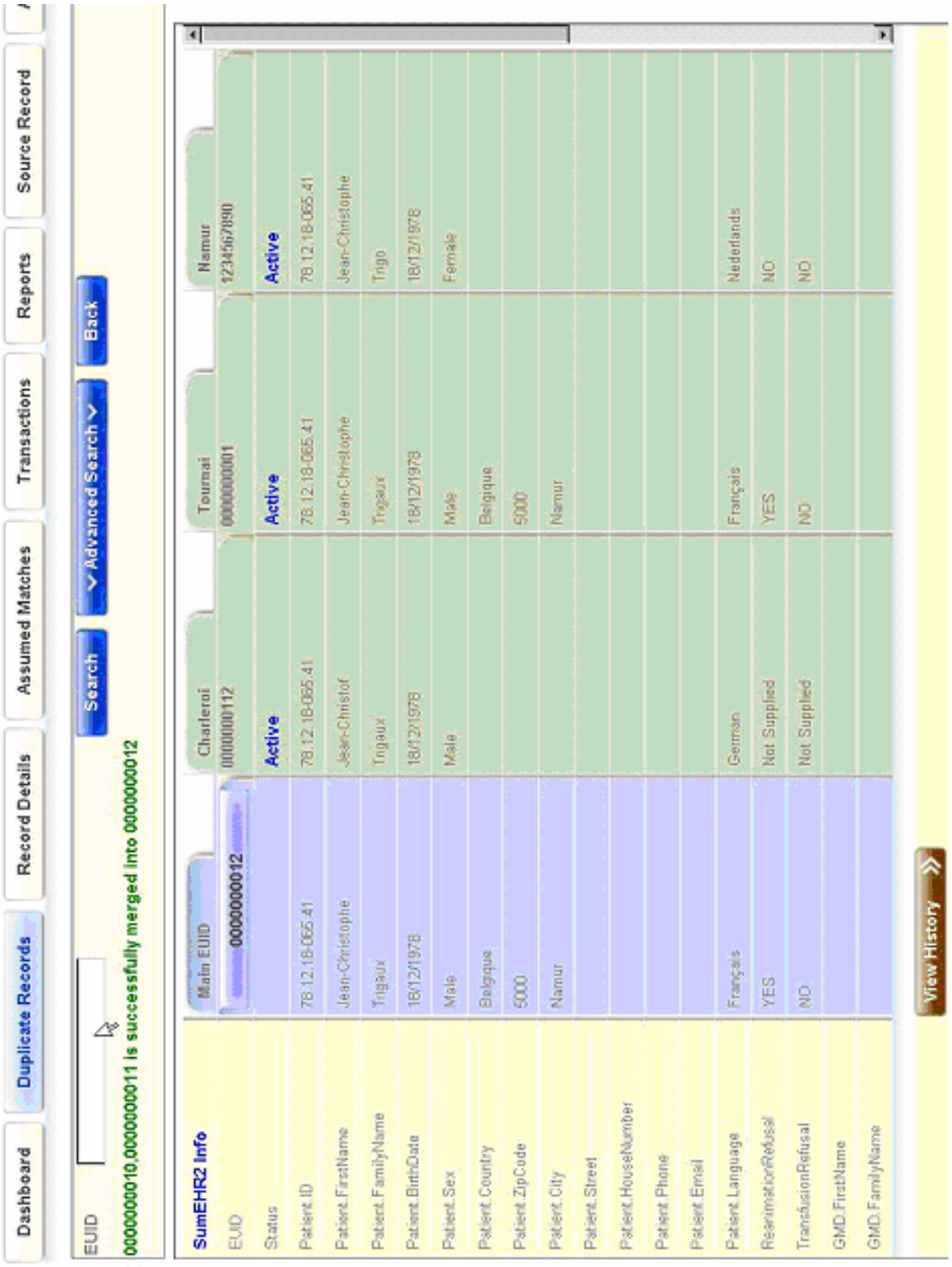
3. Détection de doublons



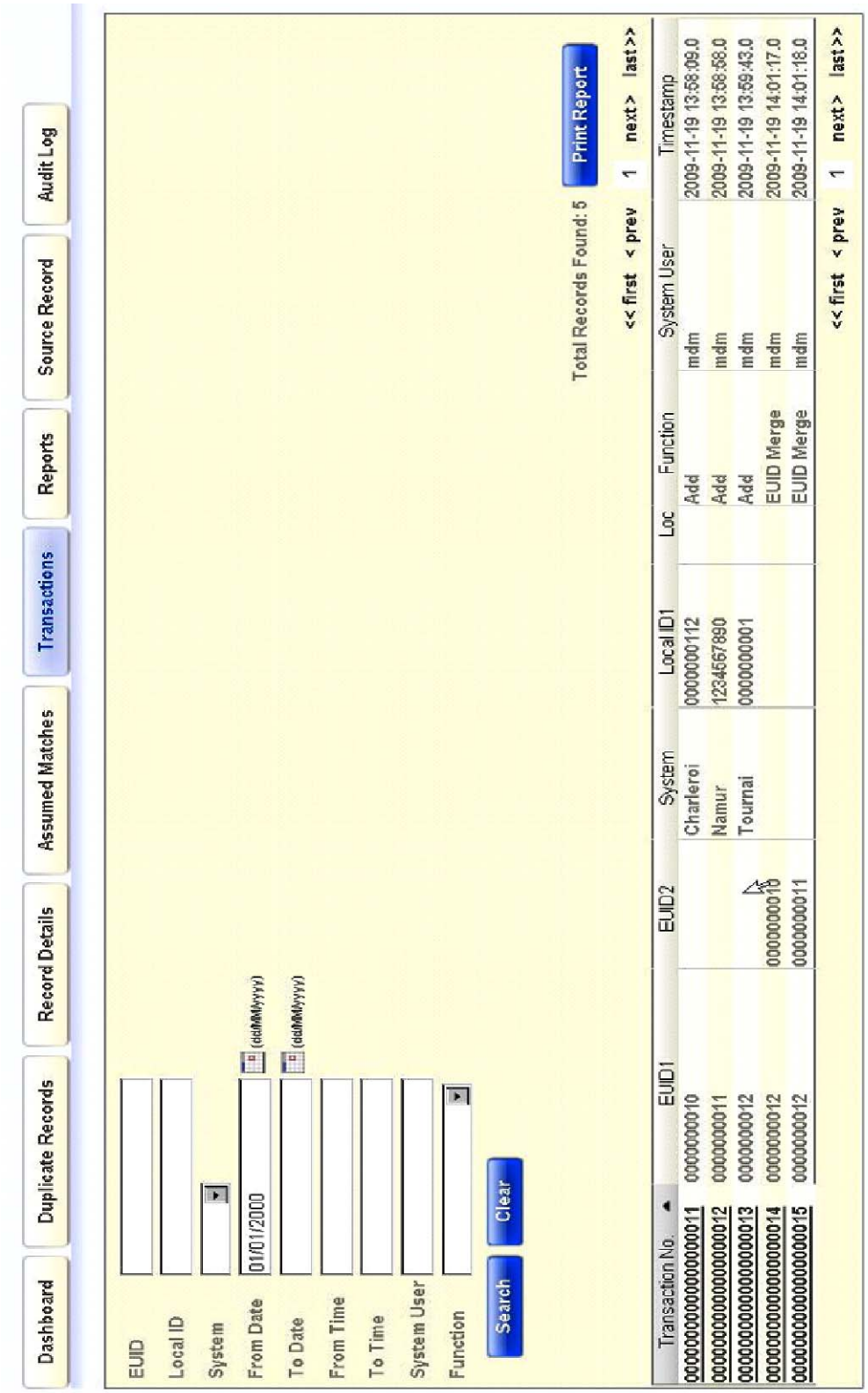
4. Fusionnement de doublons



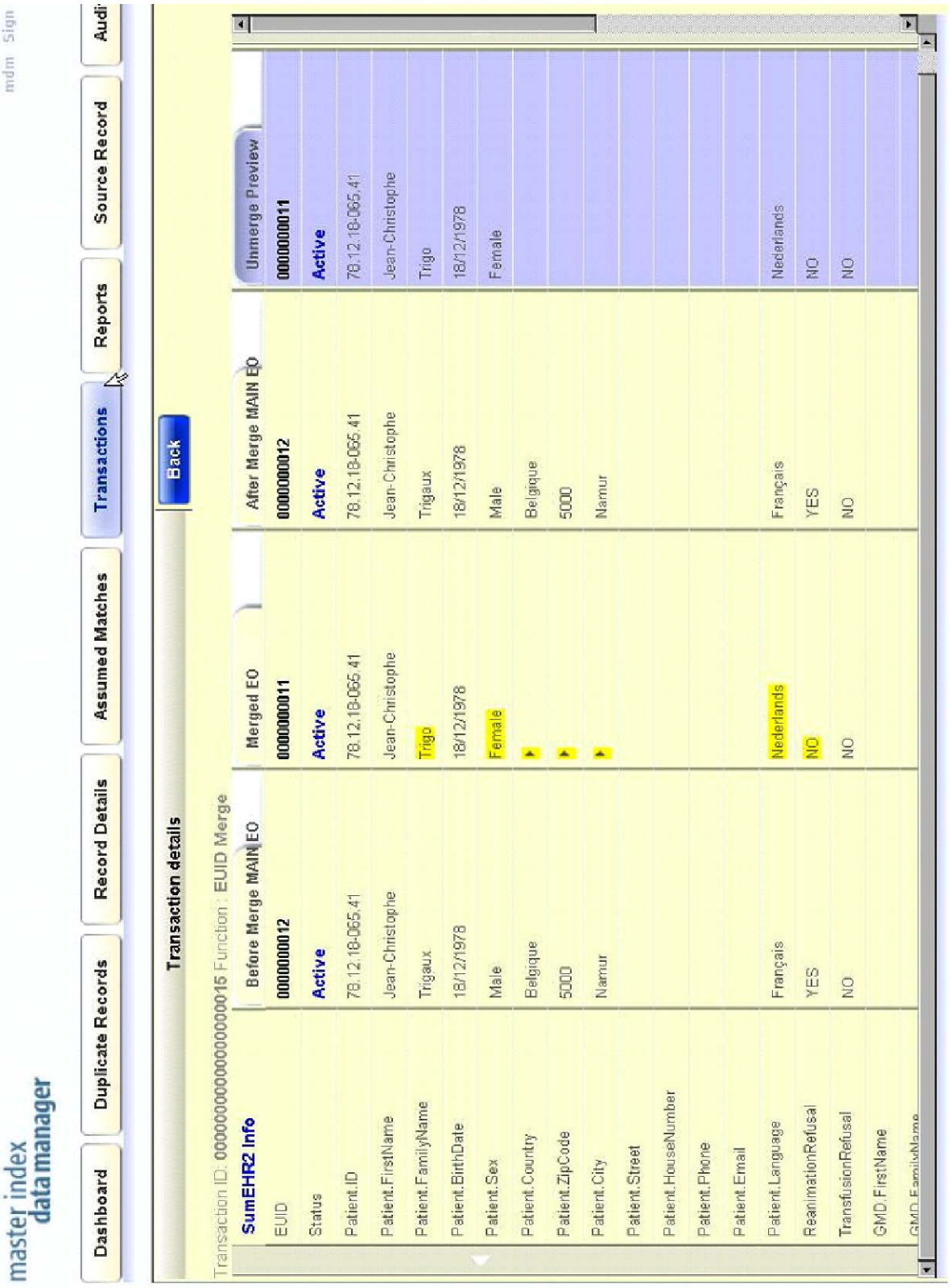
5. Identification des sources



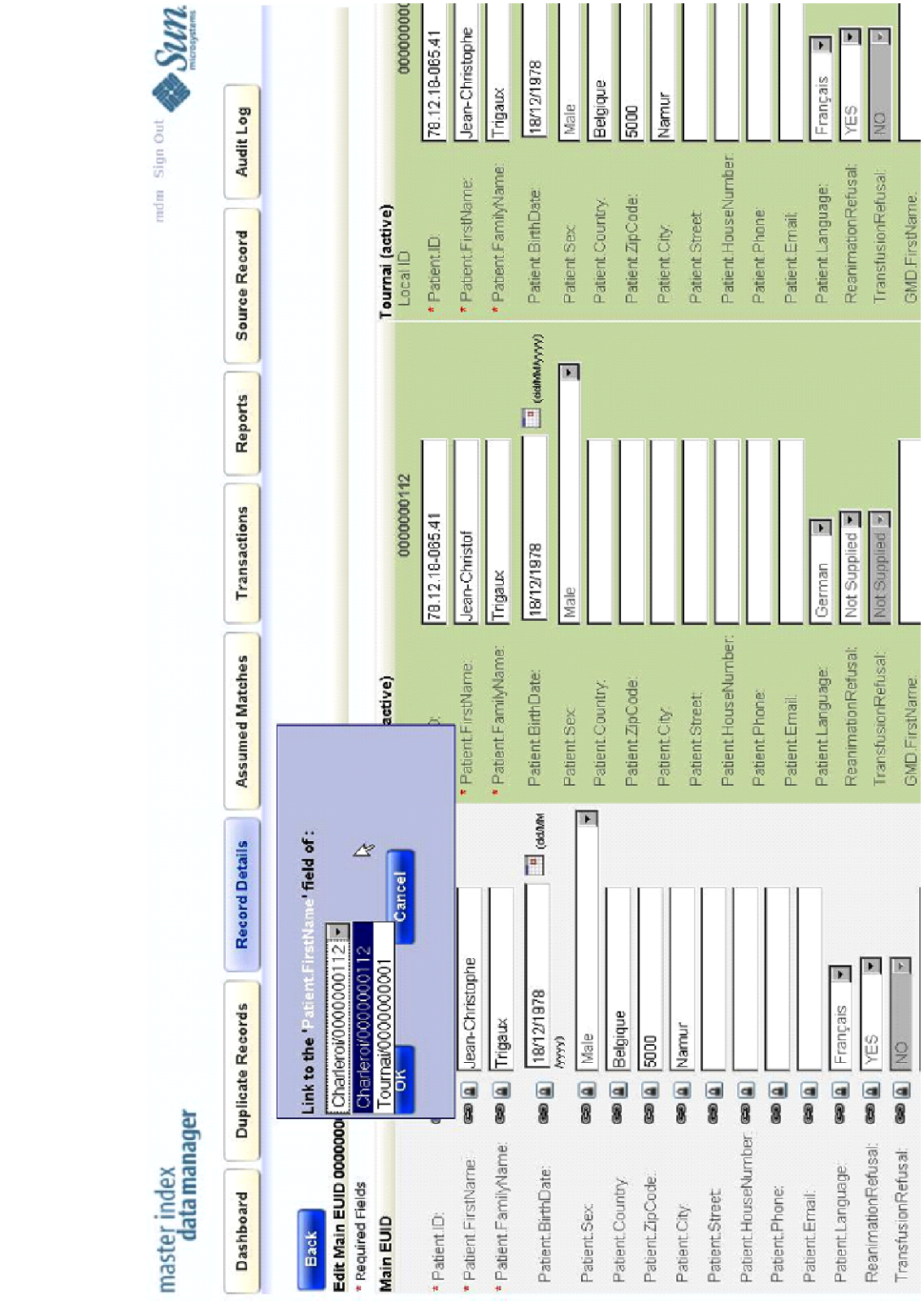
6. Historique des transactions



7. Défusionnement de faux positifs



8. Définition des sources authentiques



9. Quelques Data Services du référentiel

getEUID

Parameters | Output | Faults | Description

Parameter Name	Parameter Type
systemCode	java.lang.String
localid	java.lang.String

updateSystemRecord

Parameters | Output | Faults | Description

Parameter Name	Parameter Type
sysObjBean	com.sun.mdm.index.webservice.SystemSumEHR2

updateEnterpriseRecord

Parameters | Output | Faults | Description

Parameter Name	Parameter Type
eoBean	com.sun.mdm.index.webservice.EnterpriseSumEHR2

addSystemRecord

Parameters | Output | Faults | Description

Parameter Name	Parameter Type
euid	java.lang.String
sysObjBean	com.sun.mdm.index.webservice.SystemSumEHR2

mergeEnterpriseRecord

Parameters | Output | Faults | Description

Parameter Name	Parameter Type
fromEUID	java.lang.String
toEUID	java.lang.String
calculateOnly	boolean

getSystemRecord

Parameters | Output | Faults | Description

Parameter Name	Parameter Type
systemCode	java.lang.String
localid	java.lang.String