ASSURER LA QUALITE DES DONNEES : UN DEFI PERMANENT POUR LES SYSTEMES D'INFORMATION, BASES ET ENTREPOTS DE DONNEES

Laure Berti-Équille

Résumé : Les problèmes de qualité des données tels que les doublons, les incohérences, les valeurs manquantes, incomplètes, obsolètes ou peu fiables surviennent de façon permanente et systématique dans les bases, entrepôts de données ou systèmes d'information. Les décideurs ne peuvent plus aujourd'hui ignorer les conséquences de la non-qualité de leurs données (ou leur qualité médiocre) sur les prises de décision et les coûts financiers considérables qu'elle engendre. L'objet de cet article est de faire un bref tour d'horizon des problématiques de recherche, des solutions proposées et des techniques employées pour analyser, contrôler et corriger la qualité des données.

Mots-Clés : Qualité des données, métriques, détection de doublons, nettoyage de données

1. INTRODUCTION

Les problèmes de qualité des données stockées dans les bases, les entrepôts de données ou les systèmes d'information se propagent de façon endémique à tout type de données et dans tous les domaines d'application : données gouvernementales, commerciales, industrielles ou scientifiques [6][7][13]. Il s'agit en particulier d'erreurs sur les données, de doublons, d'incohérences, de valeurs manquantes, incomplètes, obsolètes, peu fiables, etc. Les conséquences de la non-qualité des données (ou leur qualité médiocre) sur les prises de décision et les coûts financiers que celle-ci engendre sont aujourd'hui considérables [32][33][38]. À titre indicatif, un récent rapport de l'institut américain *The Data Warehousing Institute* (*TDWI*) indique que les problèmes de qualité des données coûtent actuellement plus 600 milliards de dollars par an à l'économie américaine…

Avec la multiplication des sources d'informations disponibles et l'accroissement des volumes et flux de données potentiellement accessibles, la qualité des données et, au sens large, la qualité des informations n'ont cessé de prendre une place de premier plan tant au niveau académique qu'au sein des entreprises (en particulier outre-Atlantique : AT&T, Hewlett Packard) : il n'est plus question de négligence ou de « laisser-faire » (c'est-à-dire, utiliser aveuglément les données sans en connaître la qualité et les laisser se dégrader). L'urgence, unanimement reconnue, est de proposer des solutions pratiques aux multiples problèmes de non-qualité des données. Évaluer préalablement et contrôler la qualité des données stockées dans les systèmes d'information, bases et entrepôts de données est crucial afin de :

- proposer aux décideurs et utilisateurs des mesures objectives et une expertise critique de la qualité des données,
- permettre à ceux-ci de relativiser la confiance qu'ils pourraient accorder aux données et leur permettre, ainsi, de mieux en adapter leur usage,
- évaluer finalement la validité et l'intérêt des connaissances extraites avec une assurance sur les décisions prises à partir des données.

En effet, l'analyse des données, l'extraction de connaissances à partir des données et la prise de décision peuvent être réalisées sur des données inexactes, incomplètes, ambiguës et de qualité médiocre [27]. Mais on peut alors s'interroger sur le sens à donner aux résultats de ces analyses et remettre en cause, à juste titre, la qualité des connaissances ainsi « élaborées » et le bien-fondé des décisions prises.

L'objet de cet article est de faire un bref tour d'horizon des problématiques de recherche et des techniques employées pour analyser, contrôler et corriger la qualité des données. L'article tentera également de souligner que : 1) l'impact et les coûts de la non-qualité des données (tout comme sa méconnaissance) retentissent à chaque étape du processus de traitement des données (modélisation des données, collecte/import, stockage, gestion, intégration, interrogation, et analyse des données), et 2) de nombreuses techniques peuvent être combinées pour consolider et améliorer la qualité des données.

2. QUELQUES CAUSES DE LA NON-QUALITE DES DONNEES

Les causes des problèmes de qualité des données sont d'origines très diverses :

- lors de la modélisation conceptuelle des données, lorsque les définitions des attributs n'ont pas été suffisamment bien structurées ou normalisées, que le schéma de la base n'a pas été validé ou qu'il manque des contraintes d'intégrité ou des procédures pour maintenir la cohérence des données,
- lorsque les interprétations des données et des informations sont incohérentes, notamment à cause de différences nationales ou culturelles dans l'usage de certains codes ou symboles,

- lors du développement logiciel : les besoins peuvent ne pas avoir été complètement spécifiés, ils ont pu évoluer et des erreurs peuvent avoir été introduites lors du processus de développement, notamment entre les phases d'analyse et de conception du système d'information,
- lorsque la méthode de saisie des données n'a pas été bien conçue et qu'il manque des procédures de vérification systématique ou des techniques de mesure et de détection d'erreurs ou de doublons : les erreurs humaines sont alors facilement introduites,
- lorsque la sécurité physique ou logique du système d'information laisse à désirer : les données peuvent être corrompues volontairement,
- lorsque les entreprises n'ont pas les moyens de traquer l'âge de leurs données, de les mettre à jour, de les enrichir ni, encore, d'éliminer les données devenues obsolètes,
- lors de l'intégration ou de la post-intégration de plusieurs sources d'informations hétérogènes : les données des différents systèmes peuvent être contradictoires ou incohérentes, et les heuristiques d'appariement et d'intégration peuvent s'avérer inadaptées pour chacun des systèmes dont la qualité des données n'est d'ailleurs pas homogène localement (certaines informations sont plus précises que d'autres ou plus critiques pour l'entreprise et donc traitées avec plus ou moins d'attention),
- lors de la conversion, refonte ou migration des systèmes : les programmes de conversion peuvent introduire de nouvelles erreurs ; la rétro-ingénierie peut ne pas considérer (ou perdre) tout ou partie du contexte de définition, de production ou d'usage des données,
- lorsque des données répliquées sur différents sites ne sont pas correctement ou assez rapidement mises à jour : les copies secondaires ne sont plus cohérentes avec les copies primaires.
- etc.

Le Tableau 1 récapitule à titre indicatif quelques-uns des principaux problèmes de qualité des données pouvant survenir à chaque étape de leur traitement ainsi que les solutions potentielles proposées aussi bien dans le domaine académique qu'industriel et commercial.

ÉTAPES DE TRAITEMENT	SOURCES DE PROBLÉMES	SOLUTIONS POTENTIELLES
Création des données	Entrée manuelle Absence de standards pour les contenus et les formats Entrée de doublons Approximations Contraintes matérielles ou logicielles Erreurs de mesure	*Approche Préemptive* : Architecture pour la gestion de processus et workflows : (audits, intendance des données - "*data stewards*") Approche *Rétrospective* : – *Nettoyage de données* : élimination des doublons, "*merge/purge*", appariement des noms et adresses, jointure approximative – *Diagnostic* : détection automatique des erreurs et exceptions
Collecte / import des données	Destruction ou mutilation d'information par des pré-traitements inappropriés Perte de données : *buffer overflows*, problèmes de transmission Absence de vérification	- Utiliser des techniques de fouille de données pour vérifier que toutes les données ont été correctement transmises - Vérifier la transmission, l'intégrité des données, leur format - Suivi de données - Édition de données (*data publishing*) - Agrégation de données et *data squashing*
Stockage des données	Absence de méta-données, de mises à jour Modèles et structures de données inappropriés Modifications *Ad hoc* Contraintes matérielles ou logicielles	– Méta-données – Planification et customisation par domaine – Profilage des données, moniteur de navigation dans les données
Intégration des données	Multiple sources de données Synchronisation temporelle Systèmes de données non classiques Facteurs sociologiques Jointures *Ad hoc* Appariements aléatoires Heuristiques	– Exiger des estampilles temporelles précises pour assurer la cohérence logique et temporelle des jeux de données, lignage des données – Outils commerciaux pour : la migration des données et le nettoyage de données, le profilage de données – Outils académiques pour faire les appariements et les jointures entre jeux de données
Recherche et analyse des données	Erreur humaine Contraintes sur la complexité de calcul Contraintes logicielles, incompatibilité Problèmes de passage à l'échelle, de performances et de confiance dans les résultats Utilisation de boîtes noires pour l'analyse Attachement à une famille de modèles (statistiques) Expertise insuffisante d'un domaine Manque de familiarité avec les données	– Planification : évaluer le problème par rapport à l'équipement et vice versa – Fouille de données exploratoire (*Exploratory Data Mining* - EDM) – Plus grande responsabilité des analystes – Analyse en continu plutôt que ponctuel – Échantillonnage plutôt qu'analyse complète – Boucle de rétro-action

Tableau 1 : Qualité des données : problèmes et solutions proposés

3. APPROCHES PREVENTIVES : MESURER LA QUALITE DES DONNEES

La qualité des données peut être mesurée selon différents critères objectifs et quantitatifs qui constituent des indicateurs utiles pour contrôler la qualité, prévoir et planifier des actions correctives à appliquer aussi bien au sein de la base ou entrepôt de données qu'aux sources qui l'alimentent en entrée. La qualité des données peut être représentée et visualisée sous la forme d'un tableau d'indicateurs facilitant la compréhension et la mise en œuvre d'une stratégie d'amélioration de la qualité des données à chaque étape de la chaîne de traitement et réduire ainsi le phénomène « *garbage in garbage out* » (c'est-à-dire la propagation des erreurs et anomalies sur les données de leur entrée jusqu'aux résultats en sortie du système).

De nombreuses dimensions ont pu être proposées pour caractériser les multiples facettes de la qualité des données. Les plus fréquemment mentionnées dans la littérature sont : l'exactitude, la complétude, l'actualité, la cohérence dont des définitions générales sont les suivantes :

- l'exactitude se mesure en détectant le taux de valeurs correctes dans la base de données (soit par rapport à un référentiel consolidé : par exemple, une autre base de données de référence ou une vérité terrain, soit par vérification de contraintes spécifiques,
- la complétude se mesure en détectant le taux de valeurs manquantes dans la base,
- l'actualité se mesure en détectant le taux de valeurs obsolètes dans la base de données par rapport à une date fixée ; la fraîcheur se mesure par une comparaison de la date de saisie ou d'entrée dans le système à la date courante [36],
- la cohérence se mesure par rapport à un ensemble de contraintes (ou règles métier) en détectant les données de la base qui ne les satisfont pas.
-

Bien d'autres dimensions de la qualité (plus de 300) ont pu être proposées dans la littérature [23][1][32][33][39][38][34] pouvant être aussi bien mesurées objectivement qu'évaluées subjectivement par un expert. Dans la Figure 1 extraite de [3] on présente, à titre illustratif, quelques autres dimensions pouvant caractériser la qualité des données. Citons quatre grandes catégories de dimensions caractérisant la qualité des données :

- la première relative à la qualité intrinsèque des données avec des dimensions telles que la précision des données, la complétude des champs, la cohérence, l'unicité, l'exactitude définies précédemment,
- la seconde relative à la qualité de la gestion des données par le système (par exemple, la sécurité de l'accès, l'accessibilité des données, leur réutilisabilité, etc.),
- la troisième relative à la qualité de la représentation, en particulier la qualité du modèle des données sous-jacent (par exemple, la concision du modèle, le caractère approprié du format, la documentation du modèle, etc.),
- et, enfin, la quatrième catégorie de dimensions : *i)* relatives à l'utilisateur (par exemple, l'adéquation avec ses centres d'intérêt), *ii)* relatives au domaine d'application (par exemple, la criticité, la pertinence par rapport à une requête, le coût, etc.), *iii)* relatives au référentiel temporel (par exemple, l'actualité, la fraîcheur, la traçabilité, la variabilité des données, etc.) ou enfin *iv)* relatives à un état de la connaissance (par exemple, la réputation de la source, la crédibilité, etc.).

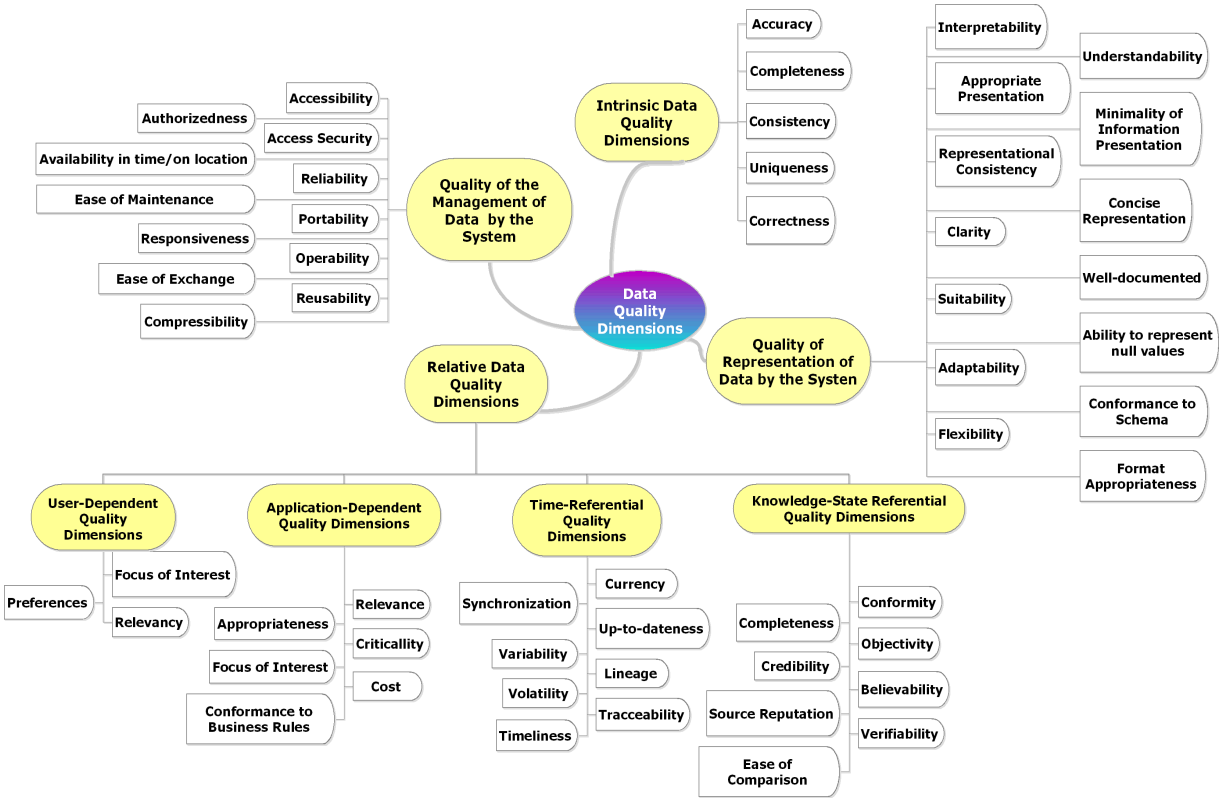

Figure 1 : *Quelques dimensions de la qualité des données*

Intéressons-nous dans la suite de cet article aux dimensions pour lesquelles des métriques ont été proposées, c'est-à-dire celles qui sont mesurables objectivement.

Métriques pour évaluer la qualité d'une modélisation conceptuelle de type Entité-Association

Au niveau conceptuel, il est intéressant d'analyser la complexité du modèle de données à l'origine du schéma d'une base de données relationnelle, et ce, principalement pour en évaluer la maintenabilité. Ces mesures sont des supports quantitatifs afin de comparer des alternatives de conception et permettre l'identification des problèmes de conception qui ont nécessairement un impact plus ou moins direct sur la qualité des données stockées. Quelques-unes des métriques proposées dans la littérature, et en particulier dans l'ouvrage très riche de Piattini et Coll. [28] sont présentées dans le Tableau 2. Ce sont des mesures objectives et subjectives de la complexité et des facteurs de qualité d'un modèle conceptuel de type Entité-Association. La validation de ces métriques reste aujourd'hui un problème de recherche ouvert.

AUTEURS	MÉTRIQUES	VALIDATION	
		THÉORIQUE	EMPIRIQUE
Gray *et al.* 1991 [9]	Complexité structurelle du modèle : $E=\sum_{i=1}^{n} Ei$ avec *n* entités *Ei* Complexité pour l'entité *i* : $Di=Ri*(a*FDA+b*NFDA)$ avec 0 < *a* <= *b*, *Ri* = nombre d'associations, *FDAi* = nombre d'attributs en dépendances fonctionnelles, *NFDAi* = nombre d'attributs sans dépendance fonctionnelle	Non	Non
Moody *et al.*, 1998 [21]	**Complétude** : <nombre d'items du modèle ne correspondant pas aux spécifications des utilisateurs, nombre de spécifications non représentées dans le modèle, nombre d'items du modèle qui correspondent aux spécifications mais qui sont mal définis, nombre d'incohérences dans la modélisation> **Intégrité** : <nombre de contraintes non satisfaites par les données, nombre de contraintes incluses dans le modèle qui ne correspondent pas exactement à la réalité modélisée> **Flexibilité : <**nombre d'éléments dans le modèle sujets à modification, coût estimé des modifications, importance stratégique des modifications> **Compréhensibilité** : <estimation par l'utilisateur du caractère compréhensible et interprétable du modèle> **Correction** : <nombre de violations des conventions du modèle de données, nombre de violations des formes normales, nombre d'instances redondantes dans le modèle> **Simplicité** : <nombre d'entités et associations ; somme pondérée ($aN^E + bN^R + cN^A$), où N^E est le nombre d'entités, N^R le nombre d'associations et N^A le nombre d'attributs> **Intégration** : <nombre de conflits avec le modèle de données commun, nombre de conflits avec les systèmes existants> **Implémentabilité** : <estimation du risque technique, estimation du risque de planification, , estimation du coût de développement, nombre d'éléments du niveau physique inclus dans le modèle>	Non	Non
Genero *et al.*, 2000 [10]	**NE** : nombre total d'entités dans le modèle ; **NA** : nombre total d'attributs d'entités et d'associations (simples ou composés); **DA** : nombre d'attributs dérivés (i.e., attributs dont la valeur peut être déduite ou calculée) ; **CA** : nombre total d'attributs composés ; **MVA** : nombre total d'attributs multi-valués ; **NR** : nombre total d'associations dans le modèle ; **M:NR** : nombre total d'associations M:N ; **1:NR** : nombre total d'associations 1:N et 1:1 ; **N-AryR** : nombre total d'associations N-aires ; **BinaryR** : nombre total d'associations binaires ; **NIS-AR** : nombre total d'associations IS_A (généralisation/spécialisation) ; **RefR** : nombre total d'associations cycliques ; **RR** : nombre total d'association redondantes	Oui	Partiellement
Piattini *et al.* , 2000 [27]	**RvsE :** rapport entre le nombre d'associations *NR* et le nombre d'entités *NE* du modèle $RvsE=\left(\frac{NR}{NR+NE}\right)^2$ avec *NR* + *NE* > 0. **DA :** rapport entre le nombre d'attributs dérivés *NDA* et le nombre maximal d'attributs dérivés possibles *NA* : $DA=\frac{NDA}{NA-1}$ avec *NA* > 1. **CA :** rapport entre le nombre d'attributs composés *NCA* et le nombre d'attributs total *NA* : $CA=\frac{NCA}{NA}$ avec *NA* > 0. **RR :** rapport entre le nombre d'associations redondantes *NRR* et le nombre d'associations *NR* : $RR=\frac{NRR}{NR-1}$ avec *NR* > 1. **M:NRel :** rapport entre NM:NR, le nombre d'associations M:N sur le nombre d'associations *NR* : $M:NRel=\frac{NM:NR}{NR}$ avec *NR* > 1. **FLeaf :** mesure de la complexité des hiérarchies (IS_A) : $FLeaf=\frac{NLeaf}{NE}$ avec *NLeaf* : nombre d'entités filles dans une hiérarchie généralisation/spécialisation et *NE* nombre d'entités dans chaque hiérarchie, *NE* > 0. **IS_Arel** : nombre moyen de supertypes directs et indirects par entité non racine *ALLSup* : $IS\_ARel=FLeaf-\frac{RLeaf}{ALLSup}$	Non	Partiellement

Tableau 2 : *Métriques utilisées pour évaluer la complexité et la qualité d'un modèle conceptuel de type Entité-Association*

Métriques pour évaluer la qualité d'une base de données relationnelle

On peut rapidement classer les métriques de qualité des données en trois catégories :
1- celles mesurables par simples comptages,
2- celles basées sur une vérification de contraintes ou de règles métier définies *a priori*,
3- celles nécessitant des traitements statistiques plus élaborés.

Parmi les nombreuses métriques proposées dans la littérature [38] relevant des deux premières classes, citons en particulier les travaux de Naumann et Coll. [24][23] (voir Tableau 3). Ces derniers définissent plusieurs critères de qualité objectifs et subjectifs permettant d'établir des stratégies d'exécution de requêtes dans une architecture de type médiateur/adaptateurs (incluant la création, la sélection et l'ordonnancement de plans de requêtes selon la qualité des données et des sources). Dans ce contexte, plusieurs sources de données hétérogènes peuvent répondre à tout ou partie d'une requête. Ainsi, la qualité du résultat d'une requête est une fonction (fusion) de la qualité des sources et des données qui le compose.

AUTEURS	DEFINITION DES METRIQUES	
Naumann, Leser 1999 [23]	Spécifiques à la source de données	**Facilité de compréhension** : jugement de l'utilisateur de 1 à 10 basé sur la présentation des données **Réputation** : jugement de l'utilisateur de 1 à 10 basé sur des préférences personnelles et son expérience **Fiabilité** : score de 1 à 10 basé sur la précision des méthodes expérimentales de recueil et/ou de production des données **Actualité** : fréquence de mise à jour mesurée en nombre de jours
	Spécifiques aux requêtes	**Disponibilité** : pourcentage de temps pendant lequel la source de données est accessible **Prix** : prix d'une requête en US dollars **Temps de réponse** : temps d'attente moyen en secondes par requête **Précision** : pourcentage de données sans erreur **Pertinence** : pourcentage d'objets du monde réel représentés par la source de données
	Spécifiques aux attributs	**Complétude** : pourcentage de valeurs non nulles
Motro, Rakov 1997 [20]	**Complétude** : proportion des informations correctement stockées D par rapport au monde réel modélisé W: $C=\frac{\lvert D\cap W\rvert}{\lvert W\rvert}$ **Précision** (soundness) : proportion des informations correctement stockées : $S=\frac{\lvert D\cap W\rvert}{\lvert D\rvert}$ **Homogénéité de la qualité d'une vue** : différentiel moyen entre la précision d'une vue v et celle de toutes ses sous-vues possibles v_i $Hs(v)=\frac{1}{N}\sum_{i=1}^{N}\lvert S(v)-S(v_i)\rvert$	
Labrinidis, Roussopoulos 2001 [17]	**Fraîcheur** : probabilité d'accéder une version à jour d'une vue v de la base de données sur une période donnée T=[ti,tj] : $Fresh(v)=\frac{1}{T}\times\int_{ti}^{tj} f(d_i)^t$ avec $f(d_i)^t=\begin{cases}0, si la donnée di est obsolète au temps t\\1, si la donnée di n'est pas obsolète au temps t\end{cases}$	

Tableau 3 : *Quelques métriques pour évaluer des dimensions de la qualité des données*

La qualité des données n'est jamais définitivement acquise et elle évolue à chaque instant. En effet, à chaque action sur les données, peuvent être introduites des erreurs. Ne rien faire et laisser « vieillir » les données engendre également des problèmes de non-qualité. Les travaux tels que ceux de Labrinidis et Roussopoulos [18] sur la fraîcheur des données proposent une mesure de probabilité de la fraîcheur des données.

Approche statistique et fouille de données exploratoire

Concernant la dernière classe de métriques nécessitant des traitements statistiques plus élaborés, citons celles issues de la fouille de données exploratoire (*Exploratory Data Mining - EDM*)[7]. Il s'agit d'un ensemble de techniques statistiques proposant des résumés et des visualisations pour détecter des problèmes dans un ensemble de données avant de réaliser des analyses plus coûteuses. Elle a pour objectifs d'être largement applicable, rapide dans ses temps de réponse, simple à mettre en œuvre et ses résumés sont faciles à stocker, à mettre à jour et à interpréter. Elle révèle des informations qui permettent de faire des hypothèses, notamment sur la distribution des données ou les corrélations entre tel ou tel attribut (par exemple, une distribution gaussienne des valeurs d'un attribut *A* qui est lui-même lié de façon linéaire aux attributs *B* et *C* de la base). La fouille de données exploratoire peut être soit dirigée par les modèles et faciliter l'utilisation de méthodes paramétriques (modèles log-linéaires, par exemple), soit être dirigée par les données. Les résumés seront typiquement des moyennes, écarts-types, médianes ou autres quantiles sur plusieurs échantillons de l'ensemble des données. Ils permettent de caractériser le centre de la distribution des valeurs d'un attribut représentatif de la population, quantifier l'étendue de la dispersion des valeurs de l'attribut autour du centre ou, encore, décrire la dispersion (forme, densité, symétrie, etc.).

Concernant les techniques d'analyse sur des données manquantes [35], la méthode d'imputation par régression décrite par Litlle et Rubin [19] est très utilisée, d'autres méthodes telles que celle de Monte Carlo par chaînes de Markov (*MCMC*) permettent de simuler des données en leur supposant une distribution normale multivariée. D'autres références traitant des données manquantes sont décrites dans le tutorial de Pearson [27].

Concernant les données isolées (*outliers*) : les techniques de détection employées sont des graphes de contrôle, des techniques basées respectivement *i)* sur un modèle, *ii)* sur une comparaison, *iii)* sur des méthodes géométriques de mesure de distance à l'ensemble des données [15], *iv)* sur la distribution (ou la densité) de la population des données avec la notion d'exception locales (*local outliers*) [2] ou, encore, *v)* basées sur la déviation dans une série temporelle de données. D'autres tests de « *goodness-of-fit* » tels que celui du *Chi2* permettent de vérifier l'indépendance des attributs ou encore le test de Kolmogorov-Smirnov de mesurer la distance maximum entre la distribution supposée des données et la distribution empirique calculée à partir des données. Ces tests univariés permettent de valider des techniques d'analyse et des hypothèses sur les modèles employés. D'autres tests plus complexes et multivariés sont également proposés dans [7] (*DataSphere* : pyramides, hyper-pyramides et test de Mahalanobis pour des distances entre moyennes multivariées).

4. APPROCHES CORRECTIVES PAR DES TECHNIQUES DE BASE DE DONNEES

Nettoyage des données par extraction et transformation

L'intérêt des métriques sur la qualité des données est de pouvoir les exploiter à des fins d'amélioration de la qualité de la base. Le nettoyage de données fait partie des stratégies d'amélioration automatique de la qualité des données et se décompose en trois étapes [30][41][42] : 1) auditer les données afin de détecter les incohérences, 2) choisir les transformations pour résoudre les problèmes de non-qualité, et enfin 3) appliquer les transformations choisies au jeu de données. Parmi de nombreux outils académiques d'extraction et de transformation des données (*ETL – Extraction-Transformation-Loading*) dont ceux figurant dans le Tableau 3, citons Potter's Wheel [31], ARKTOS [37], AJAX [9], BELLMAN [8] qui permettent l'extraction de structures et expressions régulières, la transformation de valeurs de données (par application de fonctions de formatage), la transformation de l'ensemble des valeurs de n-uplets (lignes) et d'attributs (colonnes) d'une base de données relationnelle.

Prototype	***Descriptif***	***Particularités***
Potter's wheel [31]	Détection et correction d'anomalies transformations des données (*add, drop, merge, split, divide, select, fold, format*)	Interactivité, inférence de la structure
Ajax [9]	Langage déclaratif basé sur des opérateurs logiques de transformations (*mapping, view, matching, clustering, merging*)	Séparation logique/physique, 3 algorithmes pour l'appariement
Arktos [37]	Méta-modèle unique permettant de couvrir toutes les activités *ETL* d'un entrepôt (architecture, gestion de la qualité, etc.)	Distinction entre le méta-modèle et les niveaux conceptuel, logique et physique
Intelliclean [20]	Détection et correction d'anomalies par utilisation d'une base de règles (*duplicate identification, merge, purge, stemming, soundex, stemming, abbreviation*)	Difficulté de passage à l'échelle
Telcordia's tool [5]	Outil paramétrable pour l'appariement des enregistrements (*record linkage)* selon des fonctions de distance et d'appariement. Les règles d'appariement peuvent être générées, testées avec des techniques statistiques ou par apprentissage automatique	Pré-traitement avec élimination des « stop-words »

Tableau 4 : *Prototypes de recherche dédiés au nettoyage des données*

Potter's Wheel [31] est un système *ETL* interactif permettant de visualiser les effets des transformations sur les données. Par un mécanisme d'analyse sur les domaines, il permet également de trouver des incohérences dans les valeurs d'attributs. Les opérations de transformation possibles dans Potter's Wheel sont récapitulées dans la Figure 2.

TRANSFORMATION	DÉFINITION FORMELLE
Formatage	$\phi(R, i, f) = \{(a_1,\ldots, a_{i-1}, a_{i+1},\ldots, a_n, f(a_i))\mid (a_1,\ldots, a_n) \in R\}$
Ajout	$\alpha(R, x) = \{(a_1,\ldots, a_n, x)\mid (a_1,\ldots, a_n) \in R\}$
Suppression	$\pi(R, i) = \{(a_1,\ldots, a_{i-1}, a_{i+1},\ldots, a_n)\mid (a_1,\ldots, a_n) \in R\}$
Copie	$\kappa((a_1,\ldots, a_n), i) = \{(a_1,\ldots, a_n, a_i)\mid (a_1,\ldots, a_n) \in R\}$
Fusion	$\mu((a_1,\ldots, a_n,), i, j, glue) = \{(a_1,\ldots, a_{i-1}, a_{i+1}, \ldots, a_{j-1}, a_{j+1},\ldots, a_n, a_i \oplus \text{glue} \oplus a_j)\mid (a_1,\ldots, a_n) \in R\}$
Division	$\delta((a_1,\ldots, a_n), i, \text{pred}) = \{(a_1,\ldots, a_{i-1}, a_{i+1}, \ldots, a_n, a_i, \text{null})\mid (a_1,\ldots, a_n) \in R \wedge \text{pred}(a_i)\} \cup \{(a_1,\ldots, a_{i-1}, a_{i+1}, \ldots, a_n, \text{null}, a_i)\mid (a_1,\ldots, a_n) \in R \wedge \neg\text{pred}(a_i)\}$
Partage	$\omega((a_1,\ldots, a_n,), i, splitter) = \{(a_1,\ldots, a_{i-1}, a_{i+1},\ldots, a_n, \text{left}(a_i, splitter), \text{right}(a_i, \text{splitter}))\mid (a_1,\ldots, a_n) \in R\}$
Repliement	$\lambda(R, i_1, i_2,\ldots, i_k) = \{(a_1,\ldots, a_{i_1-1}, a_{i_1+1},\ldots, a_{i_2-1}, a_{i_2+1},\ldots, a_{i_k-1}, a_{i_k+1},\ldots, a_n, a_{il})\mid (a_1,\ldots, a_n) \in R \wedge 1 <= l <= k\}$
Sélection	$\sigma(R, \text{pred}) = \{(a_1,\ldots, a_n)\mid (a_1,\ldots, a_n) \in R \wedge \text{pred}((a_1,\ldots, a_n))\}$

Notation : R est une relation avec n attributs, i et j sont les indices des attributs ; a_i représente la valeur d'un attribut pour un tuple donné; x et *glue* sont des valeurs, f est une fonction de transformation d'une valeur en une autre ; $x \oplus y$ concatène x et y ; splitter est une position dans une chaîne de caractères ou une expression régulière ; left(x, splitter) est la partie gauche de x après avoir partagé la chaîne de caractères à la position indiquée par la variable splitter ; pred est une fonction retournant un booléen.

Exemple de transformation des données d'une table:

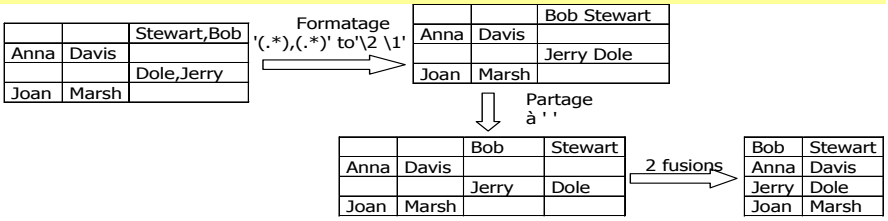

Figure 2 : *Opérations de transformation utilisées pour le nettoyage des données*

BELLMAN [8] fournit une suite d'outils ETL et un outil de profilage de base de données. D'autres opérations, telles que le clustering et l'appariement sont employées pour la détection et l'élimination de doublons : *AJAX* [9] est une extension du langage déclaratif SQL permettant de spécifier chaque transformation de données (*matching, merging, mapping, clustering*) nécessaire au processus de nettoyage des données. Ces transformations normalisent les formats de données et détectent les paires d'enregistrements qui se rapportent le plus probablement au même objet. Cette étape d'élimination des doublons est appliquée si des données approximativement redondantes sont trouvées et un appariement multi-table calcule des jointures par similarité entre des données distinctes ce qui permet leur consolidation. Nous renvoyons le lecteur au panorama des travaux proposé par Rahm et Do [30] ainsi qu'aux articles de Winkler [41][42] pour plus d'informations.

Jointures approximatives et élimination des doublons

Dans le cas d'une intégration de plusieurs bases de données relationnelles, il est nécessaire d'associer plusieurs tables au moyen de jointures pour lesquelles souvent on ne dispose pas de clés communes exactes. Lors d'une recherche de doublons sur une seule table, il est nécessaire de procéder par auto-jointure même si les clés identifient de façon unique plusieurs n-uplets de la table, ces n-uplets pouvant décrire pourtant la même réalité. Dans ces deux cas, la technique de jointure approximative est alors recommandée [12][16]. C'est également le cas lorsque les clés contiennent des informations imprécises. Par exemple, pour apparier les données des clients d'une base sur les facturations et celles d'une base de données sur les installations de télécommunications, les noms et adresses peuvent être écrits de différentes façons (« FT » ou « France Télécom » ; « Avenue du Général de Gaulle » ou « av. Gal de Gaule ») et il peut être difficile de faire l'appariement sur les noms ou adresses. Si, en revanche, le numéro de téléphone est le même, on pourra supposer qu'il s'agit bien de la même personne, c'est pourquoi il s'avère nécessaire d'abord de standardiser certains attributs (adresses, abréviations, etc.) et d'examiner les informations qui corroborent ou non une hypothèse d'appariement sur l'ensemble des attributs disponibles.

Parfois très spécifique à l'application, la technique de jointure approximative consiste à regrouper et trier les n-uplets par groupes selon une fonction de hachage sur les valeurs d'un ou plusieurs attributs (par exemple, utilisant les premières lettres ou consonnes des noms propres). Les n-uplets qui se trouvent dans les mêmes groupes sont candidats à l'appariement et pour chaque paire de candidats, une fonction d'appariement basée sur la mesure d'une distance est calculée. Seuls les paires de plus haut score sont effectivement appariées (ou assimilées à des doublons). La Figure 3 récapitule la méthode classique de recherche de doublons.

1. Pré-traitement des données (standardisation des attributs, des abréviations, structuration des adresses)
2. Fonction permettant de réduire l'espace de recherche
   - Tri ou hâchage selon une clé
   - Examen par Fenêtrage (Multiple) (*Windowing*)
3. Choix d'une fonction de comparaison permettant d'exprimer la distance entre les paires telle que :
   - Identité, distance simple ou complexe
   - Distance pondérée par la fréquence ou dirigée par des règles
   - Distance de Hamming, distance d'édition, distance de Jaro
   - Comparaison N-grams
   - Soundex
4. Choix d'un modèle de décision
   - Méthodes probabilistes : avec/sans ensemble d'apprentissage
   - Méthodes basées sur des règles et connaissances du domaine
5. Vérification de l'efficacité de la méthode

Figure 3 : *Méthode générique pour la recherche de doublons*

Une implémentation de la technique de jointure approximative est décrite dans [5]. Pour des domaines d'attributs textuels, l'appariement des chaînes de caractères (*string matching*) calcule une distance comptabilisant le nombre d'opérations d'édition (telles que l'ajout, la suppression d'un caractère ou le changement de lettre) nécessaires pour transformer une chaîne de caractères en une autre. Par exemple, « Telecom » a une distance d'édition de 2 de « Telefon ». Les chaînes de caractères dont la distance d'édition est inférieure à un seuil fixé seront appariées. L'ensemble des algorithmes d'appariement de chaînes de caractères sont détaillés dans [25][26]. Le Tableau 5 présente les différentes mesures de similarité pouvant être employées.

Distances	**Principales caractéristiques**
Distance de Hamming	Applicable à des champs numériques fixes (N°Sécu, Code Postal) mais ne prend pas en compte les ajout/suppressions de caractères
Distance d'édition	Calcul du coût minimal de conversion (par ajout/suppression/ remplacement de caractères)
Algorithme de Jaro	Caractères communs et transposés : (C/L1 +C/L2 +(2C-T)/2C)/3 avec C nombre de caractères communs, L1 et L2 longueurs des deux chaînes de caractères à comparer, T nombre de transpositions de caractères
Distance N-grams	$\sqrt{\sum_{\forall x} \lvert f_a(x) - f_b(x) \rvert}$ Somme du nombre de caractères communs sur toutes les sous-chaînes de caractères x de longueur N présents dans les chaînes a et b
Soundex	Première lettre du mot puis encodage des consommes sur 3 caractères tel que : B,F,P,V -> 1 | C,G,J,K,Q,S,X,Z ->2 | D,T -> 3 | L -> 4 | M,N -> 5 | R -> 6 (ex. "John" et "Jan" sont encodé J500 ; "Dupontel" est encodé D134)

Tableau 5 *: Distances de similarité pour comparer les chaînes de caractères pour la détection de doublons*

Pour des données semi-structurées en XML, l'appariement d'arbres est une technique analogue à l'appariement de chaîne de caractères par le calcul d'une distance d'édition, mais elle est beaucoup plus coûteuse [17] : le calcul d'une distance d'édition sur n caractères est en $O(n^2)$ alors que le calcul d'une distance d'édition sur l feuilles d'arbre est en $O(l^4)$. Toutefois, une approche intéressante d'appariement et de nettoyage de données XML a été récemment proposée par Weiss et Naumann dans [40].

5. CONCLUSION ET PERSPECTIVES

De nombreux cas, décrits dans la littérature et dans la presse scientifique, révèlent de nombreuses situations alarmantes liées aux enjeux de la (non-)qualité des données dans les bases, entrepôts de données et systèmes d'information commerciaux, médicaux, du domaine public ou de l'industrie. Les approches mises en œuvre jusqu'ici sont *ad hoc*, fragmentées et spécifiques à des domaines d'application relativement cloisonnés. Des solutions théoriquement fondées et validées en pratique sont aujourd'hui très attendues pour évaluer et contrôler la qualité des données. Plusieurs verrous scientifiques dans ce domaine ont été identifiés [7][1] et ils expliquent d'ailleurs les limitations des outils actuellement disponibles :

- l'hétérogénéité et la diversité des données multi-sources à intégrer : les données sont extraites de différentes sources d'informations puis intégrées alors qu'elles possèdent différents niveaux d'abstraction (d'une donnée brute à un agrégat). Les données sont intégrées au sein d'un même jeu de données qui contient donc potentiellement une superposition de plusieurs traitements statistiques. Ceci nécessite une très grande précaution dans l'analyse de ces données. Les jointures entre les différents jeux de données sont difficiles (quand elles ne sont pas faussées) de par le problème de l'identification non-ambiguë des données et l'impact des données manquantes,
- les volumes de données manipulées et le problème de passage à l'échelle des techniques de mesure et de détection : si certaines méthodes statistiques sont tout à fait adaptées pour fournir des résumés de la qualité de données numériques en grands volumes, elles s'avèrent inefficaces sur des données multidimensionnelles de grandes dimensions,
- la richesse et la complexité des données telles que les séries temporelles, les données extraites de pages *web* (*web-scrapped*) et les données multimédias (combinant texte, audio, vidéo, image) pour lesquelles on ne dispose pas (ou très peu) de métriques de la qualité,
- la cohérence logique et temporelle entre les données et les méta-données décrivant la qualité avec notamment des problèmes de rafraîchissement de données/méta-données : les bases de données modélisent une portion du monde réel en constante évolution. Or, dans le même temps, la qualité des données peut devenir obsolète. Le processus d'estimation et de contrôle doit être constamment reconduit en fonction de la dynamique du monde réel modélisé ainsi que des changements des centres d'intérêt et de l'attention particulière portée à certaines données (car des données jugées critiques à un instant donné ne le restent pas indéfiniment).

Pour conclure, les multiples problèmes évoqués dans cet article offrent plus que jamais d'intéressantes perspectives de recherche pour les différentes communautés scientifiques travaillant autour de la qualité des données en statistiques, bases de données, ingénierie de la connaissance, gestion de processus, etc. Pour les entreprises et industriels, détenteurs de données en masse, la qualité des données peut demeurer un épineux problème qui se pose de façon récurrente, les guidant ponctuellement vers des choix pragmatiques et souvent à court terme par manque de moyens et d'appuis hiérarchiques. Il est alors clair que pour apporter des solutions concrètes, opérationnelles sur le long terme et théoriquement fondées, des collaborations étroites entre le monde académique et les industriels sont une nécessité.

7. RÉFÉRENCES

[1] C. Batini, T. Catarci et M. Scannapiceco : *A survey of data quality issues in cooperative information systems* ; tutorial présenté à International Conference on Conceptual Modeling (ER), 2004.

[2] M. Breunig, H. Kriegel, R. Ng et J. Sander : *LOF: Identifying density-based local outliers* ; International Conference ACM SIGMOD, pp. 93-104, 2000.

[3] L. Berti-Equille : Modelling and Measuring Data Quality for Quality-Awareness in Data Mining, Chapitre de livre soumis à *Quality Measures in Data Mining*, F. Guillet et H. Hamilton (Ed.), à paraître 2005.

[4] D. Carlson : *Data stewardship in action* ; DM Review, mai 2002.

[5] F. Caruso, M. Cochinwala, U. Ganapathy, G. Lalk et P. Missier : *Telcordia's database reconciliation and data quality analysis tool* ; International Conference on Very Large databases (VLDB), pp. 615-618, 2000.

[6] J. Celko et J. McDonald : *Don't warehouse dirty data* ; Datamation, 41(18), 1995.

[7] T. Dasu et T. Johnson : *Exploratory data mining and data cleaning* ; Wiley, 2003.

[8] T. Dasu, T. Johnson, S. Muthukrishnan et V. Shkapenyuk : *Mining database structure or, How to build a data quality browser* ; Proceedings of ACM SIGMOD Conference, 2002.

[9] H. Galhardas, D. Florescu, D. Shasha, E. Simon et C. Saita : *Declarative data cleaning: Language, model, and algorithms* ; International Conference on Very Large Databases (VLDB), pp. 371-380, 2001.

[10] R. Gray, B. Carey, N. McGlynn et A. Pengelly : *Design metrics for database systems* ; BT Technology Journal, 9(4), 1991, pp. 69-79.

[11] M. Genero, M. Piattini, C. Calero et M. Serrano: *Measures to get better quality databases* ; ICEIS 2000, pp. 49-55, juillet 2000.

[12] M. Hernandez et S. Stolfo : *Real-world data is dirty: Data cleansing and the merge/purge problem* ; Data Mining and Knowledge Discovery, 2(1), pp. 9-37, 1998.

[13] K. Huang, Y. Lee et R. Wang : *Quality information and knowledge management* ; Prentice Hall, New Jersey, 1999.

[14] T. Johnson et T. Dasu : *Comparing massive high-dimensional data sets* ; International Conference KDD, pp. 229-233, 1998.

[15] E. Knorr et R. Ng : *Algorithms for mining distance-based outliers in large datasets* ; International Conference on Very Large Databases (VLDB), pp. 392-403, 1998.

[16] N. Koudas et D. Srivastava : *Approximate joins: Concepts and techniques* ; tutorial donné à International Conference on Very Large Databases (VLDB), 1363, 2005.

[17] N. Koudas, D. Srivastava, H. Jagadish, S. Guha et T. Yu : *Approximate XML joins* ; International Conference ACM SIGMOD, 2002.

[18] A. Labrinidis et N. Roussopoulos : *Update propagation strategies for improving the quality of data on the Web* ; International Conference on Very Large Databases (VLDB), 2001.

[19] R. J. A. Little et D. B. Rubin : *Statistical analysis with missing data* ; Wiley, New-York, 1987.

[20] W. L. Low, M. L. Lee et T. W. Ling : *A knowledge-based approach for duplicate elimination in data cleaning* ; Information System, Vol. 26 (8), 2001.

[21] A. Motro et I. Rakov : *Not All answers are equally good: Estimating the quality of database answer ; in. Flexible answering systems*, Kluwer Academic Press, pp.1-21, 1997.

[22] L. Moody, G. Shanks et P. Darke : *Improving the quality of entity relationship models - Experience in research and practice* ; International Conference on Conceptual Modeling, pp. 255-276, 1998.

[23] F. Naumann : *Quality-driven query answering for integrated information systems* ; Lecture Notes in Computer Science, vol. 2261, Springer, 2002.

[24] F. Naumann, U. Leser et J. Freytag : *Quality-driven integration of heterogeneous information systems* ; International Conference on Very Large Databases (VLDB), 1999.

[25] G. Navarro : *A guided tour to approximate string matching* ; ACM Computer Surveys, 33(1), pp. 31-88, 2001.

[26] G. Navarro, R. Baeza-Yates, E. Sutinen et J. Tarhio : *Indexing methods for approximate string matching* ; IEEE Data Engineering Bulletin, 24(4), pp. 19-27, 2001.

[27] R. K. Pearson : *Data mining in face of contaminated and incomplete records* ; SIAM International Conference on Data Mining, 2002.

[28] M. Piattini, M. Genero, C. Calero, C. Polo et F. Ruiz : *Database quality* ; in Chapitre 14, Advanced Database Technology and Design, Artech House, pp. 485-509, 2000.

[29] M. Piattini, C. Calero et M. Genero (réds.) : *Information and database quality* ; The Kluwer International Series on Advances in Database Systems, Vol. 25, 2002.

[30] E. Rahm et H. Do : *Data cleaning: Problems and current approaches* ; IEEE Data Engineering Bulletin 23(4), pp. 3-13, 2000.

[31] V. Raman et J. M. Hellerstein : *Potter's Wheel: an Interactive data cleaning system* ; International Conference on Very Large Databases (VLDB), 2001.

[32] T. Redman : *Data quality for the information age* ; Artech House Publishers, 1996.

[33] T. Redman : *Data quality: The field guide* ; Digital Press (Elsevier), 2001.

[34] M. Scannapieco, A. Virgillito, M. Marchetti, M. Mecella et R. Baldoni : *The DaQuinCIS architecture: a platform for exchanging and improving data quality in cooperative information systems* ; in Information Systems, Elsevier, 2004.

[35] J. L. Schafer : *Analysis of incomplete multivariate data* ; Chapman & Hall, 1997.

[36] D. Theodoratos et M. Bouzeghoub : *Data currency quality satisfaction in the design of a data warehouse* ; Special Issue on Design and Management of Data Warehouses, International Journal on Cooperative Information Systems, 10(3), pp. 299-326, 2001.

[37] P. Vassiliadis, Z. Vagena, S. Skiadopoulos et N. Karayannidis : *ARKTOS: A tool for data cleaning and transformation in data warehouse environments* ; IEEE Data Engineering Bulletin, 23(4), pp. 42-47, 2000.

[38] R. Wang : *Journey to data quality* ; Advances in Database Systems, vol. 23, Kluwer Academic Press, Boston, 2002.

[39] R. Wang, V. Storey et C. Firth : *A framework for analysis of data quality research* ; IEEE Transactions on Knowledge and Data Engineering, 7(4), pp. 670-677, 1995.

[40] M. Weiss et F. Naumann : *DogmatiX tracks down duplicates in XML* ; Proc. of the 2005 ACM SIGMOD International Conference on Management of Data, Baltimore, MA, États-Unis, juin, 2004.

[41] W. E. Winkler : *Data cleaning methods* ; International Conference KDD, 2003.

[42] W. E. Winkler : *Methods for evaluating and creating data quality* ; Information Systems, Vol. 29, no. 7, 2004.