Smals

Evaluer et améliorer la qualité des données

Data Quality: Tools

Clients & Services
Section Recherches

Date : Septembre 2007
Deliverable : 2007/TRIM3/02
Statut : Final
Auteurs : Dries Van Dromme, Isabelle Boydens, Yves Bontemps

Koninklijke Prinsstraat 102
1050 Brussel
Rue du Prince Royal 102
1050 Bruxelles

Tel : 02/787.57.11
Fax : 02/511.12.42

Management Summary

La qualité des données représente pour beaucoup d'organisations un défi de taille. Elle est considérée par les bureaux d'analystes Gartner et Butler Group comme un point critique pour le succès des initiatives SOA, la mise en place de systèmes de Business Intelligence, de Customer-Relationship Management, entre autres. Mais surtout, comme l'a souligné le premier deliverable consacré à cette thématique (« Data Quality : Best Practices »), la qualité de l'information est stratégique car elle désigne l'adéquation relative des données aux objectifs qui leur ont été assignés. De fait, au sein des administrations, des données inadéquates ou non pertinentes peuvent entraîner des effets extrêmement négatifs sur les plans financiers ou « business ». Ces effets peuvent toucher le traitement des dossiers des citoyens, les décisions stratégiques du management, les initiatives de données entre administrations, ou encore la construction de sources authentiques, pour ne citer que quelques exemples.

Comme l'a montré l'étude « Data quality : Best practices », il est primordial d'agir, de manière continue, à la source des concepts et flux d'information alimentant un système d'information. En effet, si l'on se contente de corriger les données inadéquates, sans traiter les causes, on se trouve face à un travail aussi inutile qu'infini. Un système d'information est un fleuve et un travail exclusif de correction des valeurs inadéquates n'endigue pas l'arrivée régulière de nouvelles données non pertinentes. Toutefois, en complément de cette approche, il peut être crucial de disposer d'outils intervenant au sein des bases de données pour deux raisons. D'une part, il faut pouvoir traiter le passé : données inadéquates (doubles, incohérences) déjà incluses dans les bases de données. D'autre part, le traitement à la source ne garantit pas dans l'absolu l'absence de saisie de valeurs inadéquates (émergence de doubles suite à des erreurs orthographiques, par exemple).

Dans ce contexte, un marché d'outils dédiés à l'analyse et à l'amélioration de la qualité des bases de données s'est fortement développé depuis plusieurs années. Il a d'ailleurs été reconnu comme un marché à part entière par Gartner, qui lui a consacré un premier « Magic Quadrant » en avril 2006. Ce rapport détaille l'offre actuelle en la matière : « profiling » (audit formel d'une base de données), « standardisation » des données et « matching » (détection de doublons et d'incohérences au sein d'une ou plusieurs sources). Sur la base d'un case study relatif à une base de données administrative « grandeur nature », il montre les avantages des « data quality tools » par rapport à un développement « in house » : qu'il s'agisse du temps de développement, de la richesse algorithmique (quantité de règles réutilisables) ou encore, du recours à des bases de connaissances multilingues régulièrement mises à jour (concernant les adresses, par exemple). En conclusion, ces outils, qui offrent également un traitement plus souple et rapide en cas de « change request », paraissent indispensables pour toute organisation au sein de laquelle la qualité de l'information est considérée comme un facteur crucial.

Contenu

But et structure du document

Afin d'évaluer et d'améliorer la qualité des bases de données, il est important de mener « à la source » une stratégie d'amélioration continue (voir deliverable « Data quality ; best practices »). En complément de celle-ci, des outils peuvent toutefois s'avérer indispensables afin de traiter les problèmes du passé (doubles, incohérences, …) et de détecter les difficultés à la saisie (erreurs orthographiques sources de confusion, …).

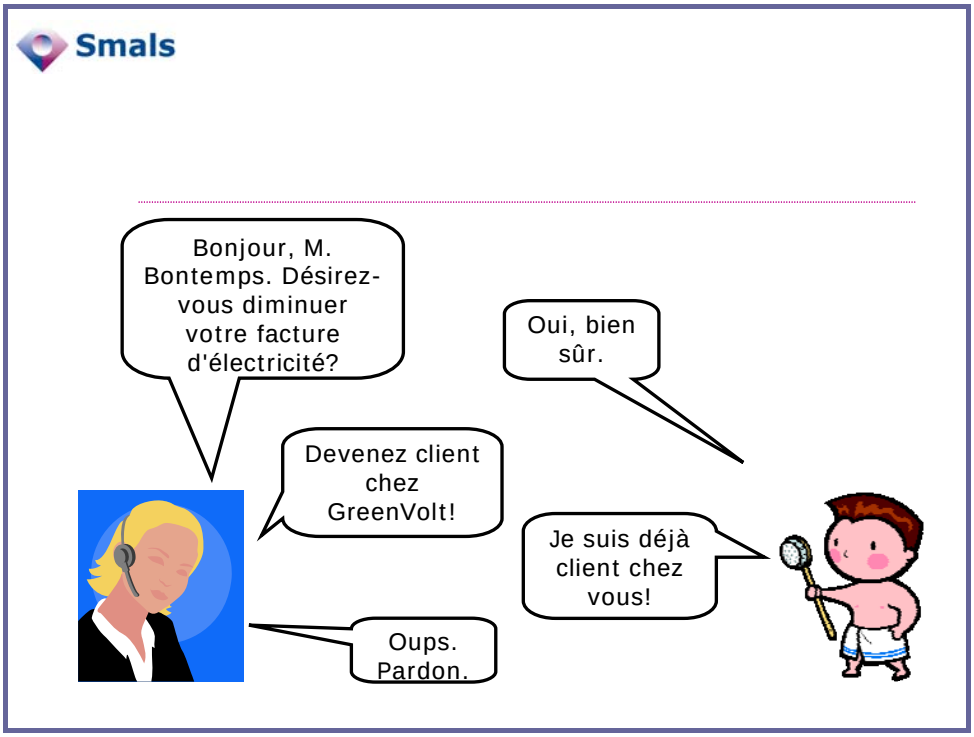
Cette étude a pour objectif de présenter les différentes fonctionnalités qu'offrent ces outils, tant sur le plan des fondements techniques que sur celui des services. Sur la base d'un « case study » illustrant les questions que soulèvent les systèmes d'information au sein de l'e-government (détection de doubles réintroduits a posteriori dans le Répertoire des employeurs de l'ONSS), il a pour objet de voir si le recours aux outils est efficace et s'il est plus avantageux en termes de fonctionnalités et de « coûts-bénéfices » qu'un développement « in house ». Sur la forme, le présent rapport propose une formule inédite puisque le cœur de celui-ci est constitué de commentaires correspondant aux slides de la séance d'information du 21 septembre 2006.

Après une illustration de la problématique de la qualité de l'information, en guise d'introduction (chapitre 1), les concepts de base associés aux « Data Quality tools » sont présentés selon la famille de fonctionnalité concernée (chapitre 2) :

- Profiling (point 2.1.) : audit formel de données en vue de tester leur adéquation aux méta-données correspondantes. Sur la base des résultats de cet audit, des mesures correctrices peuvent être menées (adaptation des méta-données ou des valeurs incohérentes).
- Standardisation (point 2.2.) : application de règles permettant de s'assurer que toutes les données soient encodées suivant les mêmes conventions, y compris des types complexes (adresses, nom) dans un contexte multilingue et avec l'aide de bases de connaissances d'adresses recouvrant des zones internationales (ou de bases de connaissance internes).
- Matching (point 2.3.) : comparaison d'enregistrements au sein d'un fichier ou entre bases de données concurrentes en vue de détecter les incohérences ou doublons et d'en améliorer la qualité.
- Monitoring (point 2.4.) : suivi dans le temps des indicateurs de qualité spécifiés à l'aide des méthodes 1 à 3.

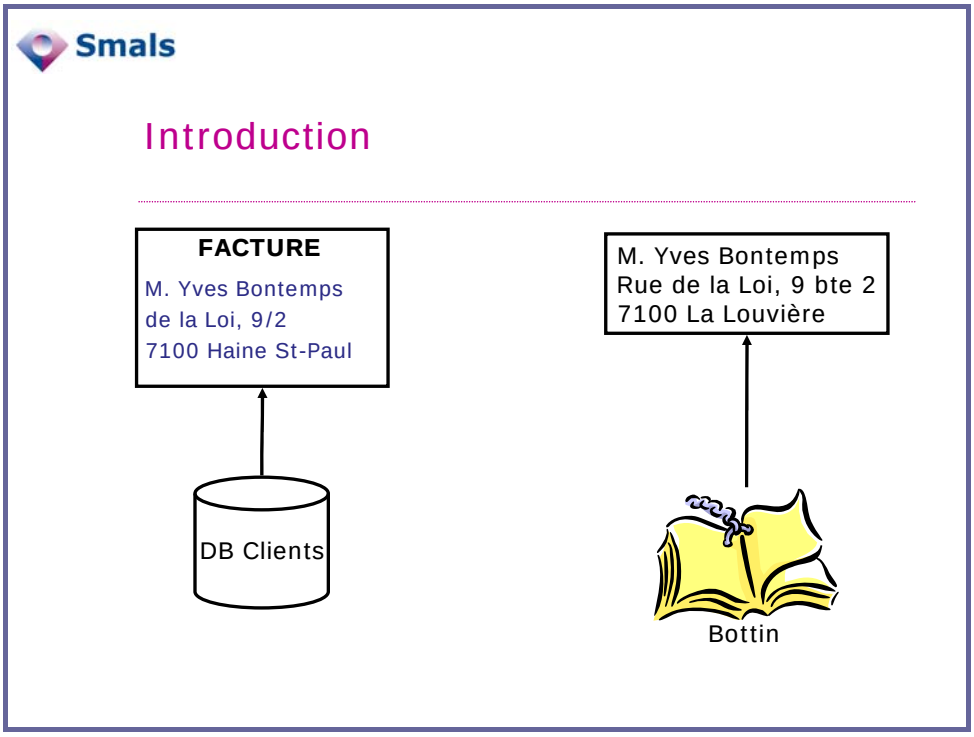
Le point III est consacré à l'étude des outils commerciaux en la matière, sur la base du case study évoqué plus haut. L'architecture au sein de laquelle ces outils peuvent s'inscrire (en « batch » ou en ligne) est également abordée. Le point IV reprend les conclusions principales de l'étude.

1. Qualité des données : rappel de la problématique



La question de la qualité de l'information, de l'adéquation aux usages (on parle de « fitness for use ») intervient concrètement dans tous les domaines de la vie courante, scientifique ou administrative.

Comme l'illustre le dessin ci-dessus, la « non-qualité » peut affecter la crédibilité d'une entreprise en cas d'erreur sur le nom ou sur l'adresse…



En effet, comment confier la gestion de ses comptes en toute confiance à une banque, si celle-ci se montre incapable de gérer correctement votre adresse ? La « non-qualité » a un impact d'autant plus important que le système d'information concerné est un instrument d'action sur le réel. Ainsi, dans le domaine de la sécurité sociale belge, plus de 35 milliards d'euros sont annuellement prélevés et redistribués : la qualité des bases de données correspondantes est donc cruciale. Il en va de même dans le secteur militaire où des erreurs dans les bases de données cartographiques peuvent avoir un impact sur l'orientation des bombardements pendant une guerre…

A partir de quelques exemples (voir encadrés suivants), il est utile de rappeler les enjeux de la notion de « fitness for use », abondamment analysée dans la première partie méthodologique de cette thématique (« Data quality : best practices »).

La qualité de l'information ne renvoie jamais à la perfection de celle-ci mais à son adéquation relative à un ensemble de besoins donnés. La tolérance à l'erreur variera en fonction des enjeux : dans le domaine statistique, par exemple, des anomalies résiduelles peuvent être tolérées et faire l'objet d'un redressement ultérieur. Par contre, dans le domaine de l'administration fédérale, chaque enregistrement d'une base de données, correspondant à un citoyen, doit faire l'objet d'un traitement équitable et minutieux. On se trouve donc toujours face à un arbitrage de type « coût-bénéfice ». Cet arbitrage peut varier dans le temps, avec l'évolution des besoins (liés dans l'e-administration à l'évolution de la loi : d'une période à l'autre, le champ d'assujettissement à l'administration fédérale pourra en effet varier). Les trois exemples qui suivent sont récents : ils illustrent la problématique dans le domaine de l'armement, du vote électronique et des fichiers de police.

Smals

Introduction
Impacts/enjeux

- National Firearms Licensing Management Systems (UK)
  - *"During the pilot there were a number of data quality issues, which meant the system was returning errors, so the system was declined"*
  - *"If the Home Office really is incapable, over a period of eight years, of computerizing something as straightforward as a few hundred thousand firearms records, then it does suggest that they do not have a hope of making a success of the introduction of the national identity card scheme"*
- http://www.computing.co.uk/computing/news/2148820/delay-gun-register

Smals

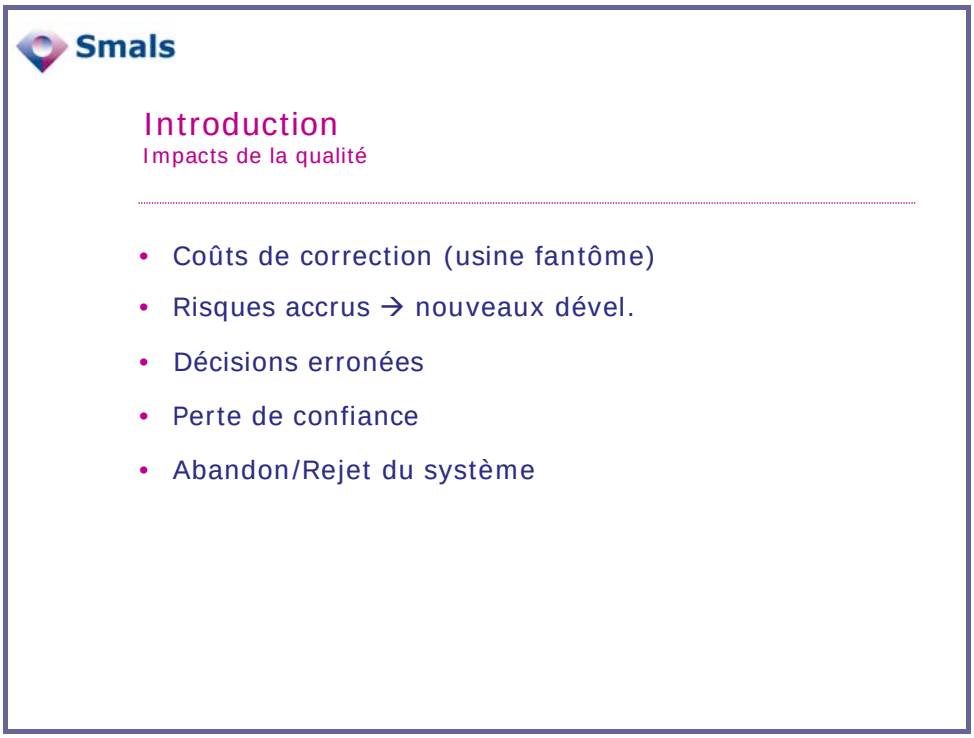
Introduction
Impacts/enjeux

- Voter's registration system in California
  - Registration system of all voters, based on identification (driver's license). Checked against Calif. Dept of Motor Vehicles database.
  - "*The rigorous system will reject applications whose data doesn't exactly match the confirming documents. Even small discrepancies, such as a missing middle initial, could cause an application to be rejected.*"
  - "*The voter database has "been a disaster for anyone who is trying to register for the first time or reregister because they moved, got married and need to change their name or change parties,"*
  - http://www.computerworld.com/databasetopics/data/story/0,10801,110353,00.html

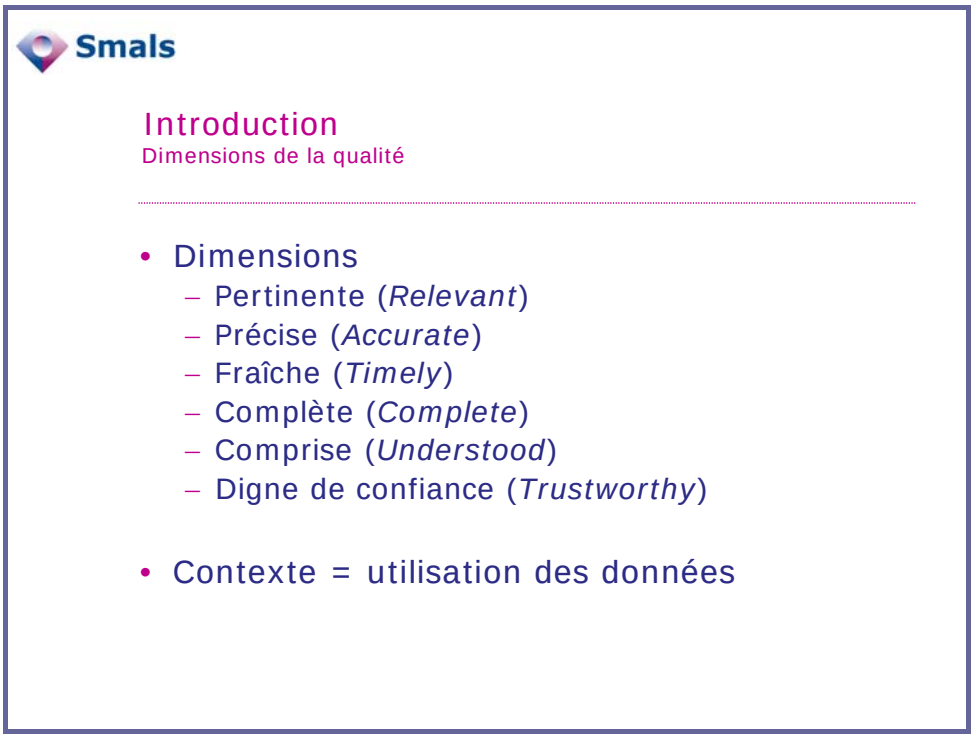
Smals

Introduction
Impacts/enjeux

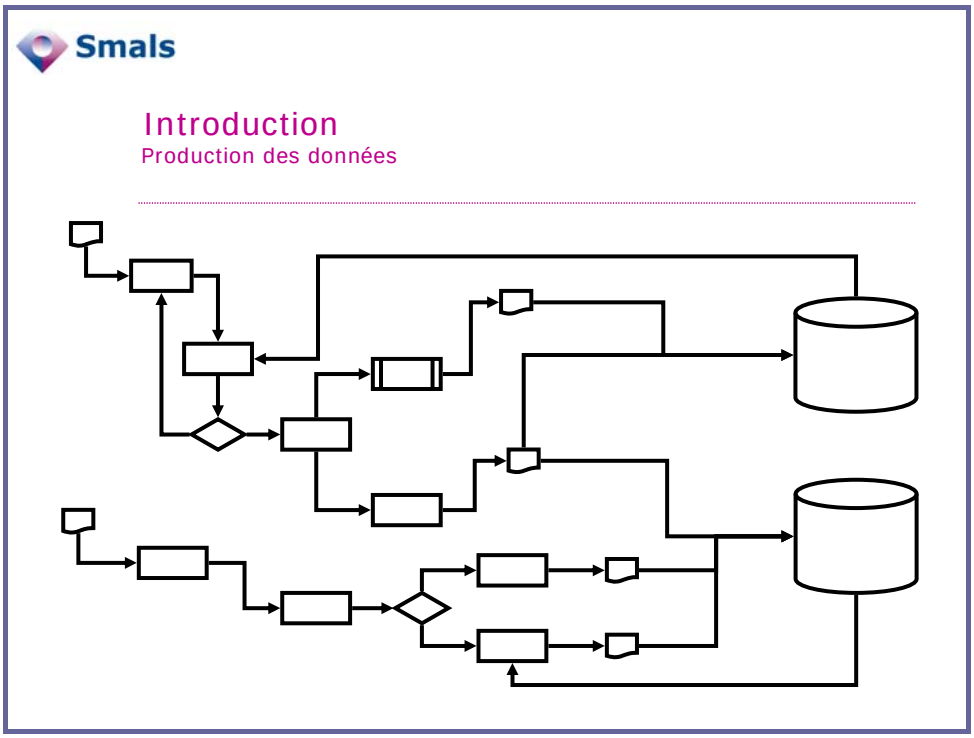
- Criminal Records Bureau (UK)
  - Check that someone has no criminal record prior to appointment (esp. unsupervised contacts with children).
  - *"The Criminal Records Bureau's first and foremost priority is to help protect children and vulnerable adults"*
  - *"The Criminal Records Bureau is only as effective as the information it can access."*
  - *Liberal Democrat home affairs spokesman Nick Clegg said the errors took "Home Office incompetence to new absurd levels". He added: "This latest fiasco will erase the last bit of public confidence in the Home Office."*
  - *http://news.bbc.co.uk/1/hi/uk/5001624.stm*



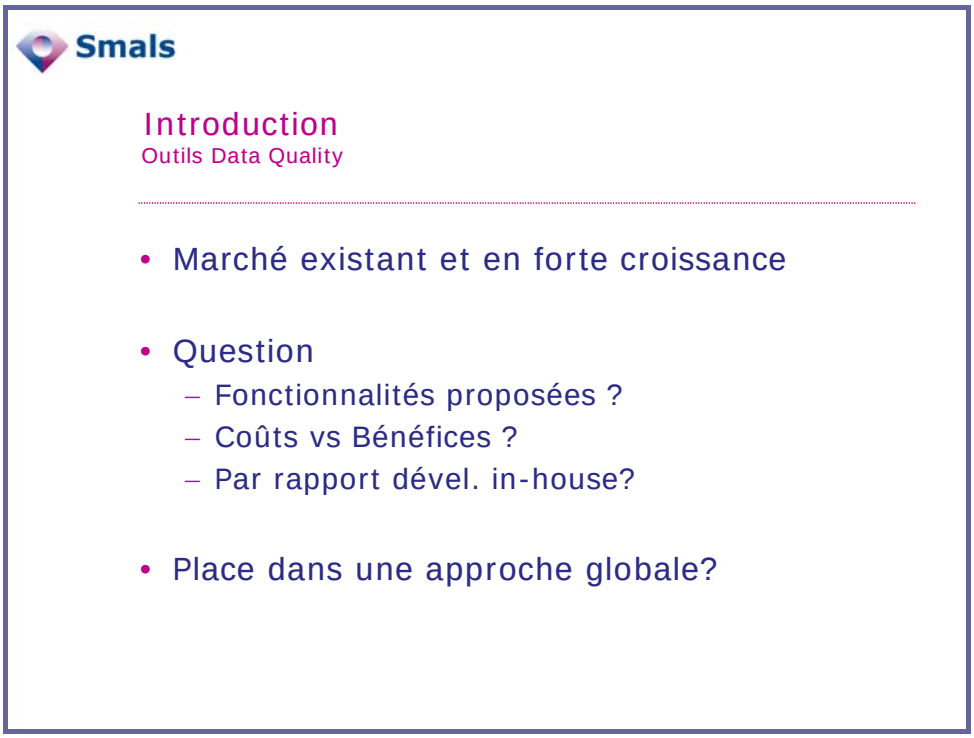
En tant qu'instruments d'action sur le réel, les systèmes d'information peuvent avoir un impact très négatif en cas de qualité insuffisante des données : tant au sein du système d'information, qu'au niveau des utilisateurs.



Comme l'a développé la première partie de l'étude (« Data quality : best practices »), la qualité, de par sa nature relative, revêt plusieurs dimensions, la principale étant la « pertinence » d'une base de données. Cette première dimension est non quantifiable. D'autres dimensions seront plus ou moins stratégiques selon les besoins, tantôt quantifiables (exemple : la fraîcheur de la mise à jour de l'information par rapport à une date donnée), tantôt non (exemple : le fait que les données soient fiables et crédibles). Dans certains cas, un arbitrage se posera entre dimensions : plus l'information est fraîche moins elle est potentiellement valide car peu de temps aura été consacré aux tests et aux corrections. Sur la base de cette analyse, des choix seront effectués en fonction du contexte et des enjeux.



Ainsi que le premier rapport sur le thème de la qualité des données (« Data quality : best practices ») l'a longuement montré, une fois les difficultés à l'origine des problèmes de qualité identifiées (concepts mal définis, flux d'information générant des doublons), il faut agir à la source. En effet, un système d'information est un fleuve. La mise en place exclusive de contrôles et tests formels au sein d'une collection de données nettoie ponctuellement le fond du fleuve mais n'endigue pas l'émergence de nouveaux flux d'information de qualité douteuse. Il s'agira donc de préciser les définitions, de documenter les systèmes d'information, d'établir des procédures claires et univoques en vue de l'alimentation du système. Toutefois, il peut être utile, en complément, d'agir au sein des bases de données, via les « data quality tools », objet du présent deliverable. Et ce, pour deux raisons. En premier lieu, il faut pouvoir traiter les problèmes issus du passé (présence de doubles, d'incohérences, …) qu'un « reengineering » des processus ne permettrait pas de supprimer a posteriori. En second lieu, même si l'on mène une politique rigoureuse et continue d'examen et d'amélioration de la qualité « en continu », il peut être pertinent, en complément, de mettre en place des outils destinés à accompagner la détection formelle et le traitement de données de mauvaise qualité. Cela s'avère crucial, par exemple en vue de prévenir et traiter l'émergence de n-uplets suite à la saisie multiple sous des graphies distinctes du nom d'un même citoyen.

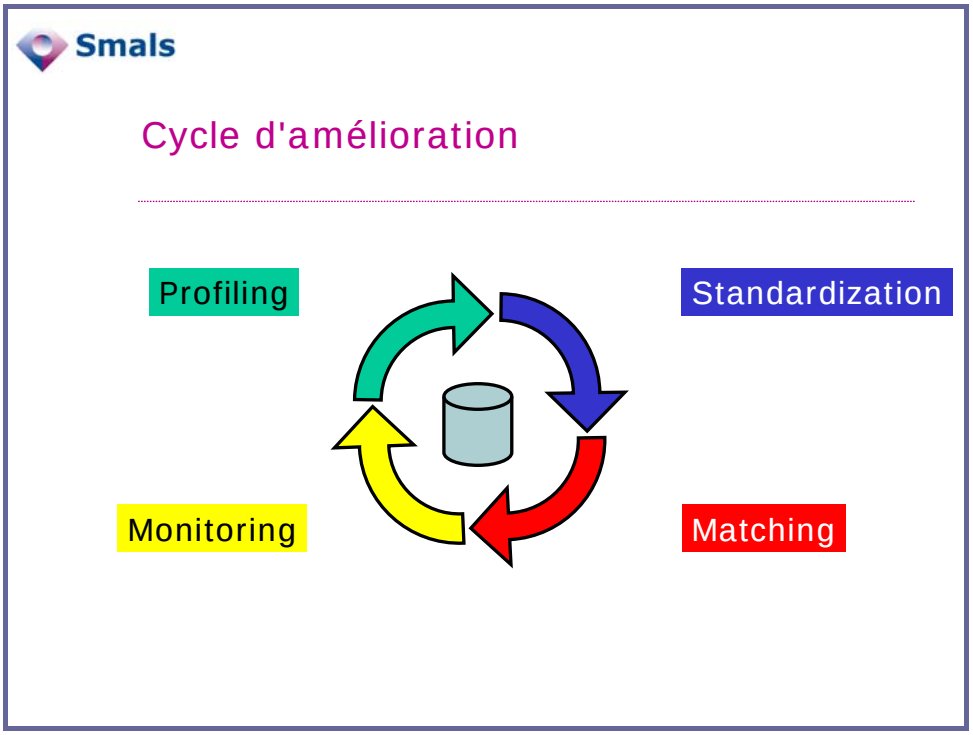


Les « data quality tools » correspondent en effet à un marché existant et en forte croissance (comme en témoignent les références de Gartner et Butler Group citées au seuil de ce rapport). Plusieurs questions se posent donc dans le contexte des bases de données de l'e-government :

- Quelles fonctionnalités concrètes ces outils proposent-ils réellement ?
- Le recours à ces outils est-il plus avantageux, en termes de coûts-bénéfices, qu'un développement « in house » ?

Nous tenterons de répondre à ces deux questions, d'une part en examinant les fonctionnalités inhérentes aux techniques incluses dans les outils, d'autre part en nous appuyant sur une consultation informelle du marché qui permettra de tester plusieurs outils sur la base d'un « case study » réel typique de notre domaine d'application : le Répertoire des Employeurs de l'Office National de la Sécurité Sociale. Ce répertoire comporte diverses caractéristiques complexes, comme le multilinguisme des adresses et des dénominations. Il s'agira de voir comment les outils du marché permettent d'y détecter des doubles. Nous reviendrons plus loin sur cette expérience et verrons ensuite comment intégrer, le cas échéant, les outils examinés dans une approche globale.

2. « DQ Tools » : cycle d'amélioration, concepts et techniques



Comme décrit dans le 1er paragraphe du concept « Fitness for Use », une qualité sans faille (une situation de la base de données dans laquelle toutes les données sont correctes à 100%) est impossible à atteindre. Dans la majorité des cas, les processus existants alimentant une base de données entraînent au contraire une qualité insuffisante ; par conséquent les données ne conviennent pas suffisamment à l'utilisation à laquelle on les destine (même dans le cas où des efforts louables ont été investis afin de détecter et d'améliorer des erreurs à des moments précis).

Viser la qualité des données doit donc constituer un processus continu d'amélioration. Ce processus inclut un cycle permettant de manier une base de données qui se trouve en continuel état de fluctuation (*entries* et *updates*), de gérer les exigences de qualité variables, les attentes des utilisateurs et les changements dans un environnement en évolution (comme en témoignent les modifications législatives dans le secteur public).

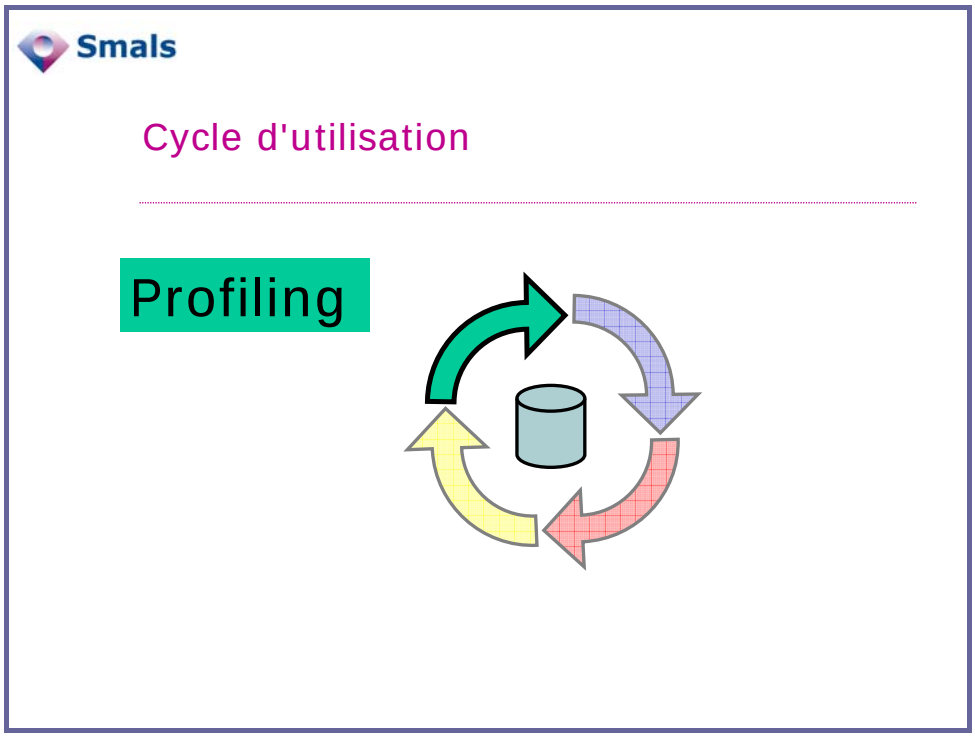
En raison de la taille des bases de données modernes, il n'est en règle générale plus possible de faire ceci manuellement et il convient d'automatiser. A cette fin, les data quality tools apportent un soutien potentiel au responsable de la gestion et de la qualité d'une base de données. Ceux-ci incluent en effet un grand nombre de fonctionnalités (les plus automatisées possible) ainsi qu'un ensemble de connaissances incorporées (par exemple des codes postaux, des noms de rue,…), offertes par le biais d'une interface utilisateur.

D'un point de vue global, l'on peut subdiviser les fonctionnalités généralement offertes par les data quality tools en 4 concepts :

- Profiling (point 2.1),
- Standardisation (point 2.2),
- Matching (point 2.3),
- Monitoring (point 2.4).

Chacun de ces concepts sera éclairci plus loin.

2.1. Data Profiling



Définition

Le Data Profiling est *l'utilisation de techniques analytiques dans le but de découvrir la structure, le contenu et la qualité* ***réels*** *d'une collection de données*[1].

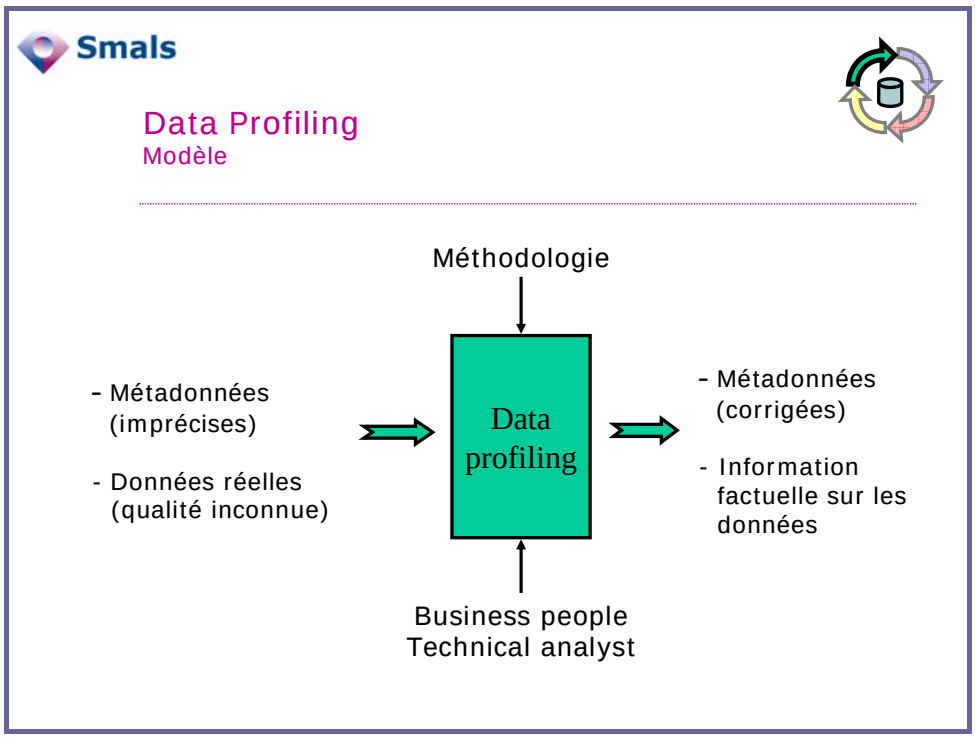
A cette fin, le Data Profiling emploie comme input tant les données elles-mêmes que toutes les méta-données connues correspondant aux données.

L'output est constitué de méta-données formellement exactes (corrigées et exhaustives) et d'informations supplémentaires sur le contenu et la qualité des données.

Le Data Profiling peut donc être appliqué afin de découvrir la présence de données inadéquates (inexactes) dans une base de données ou de méta-données à corriger. Ces problèmes découverts forment alors les Data Quality *issues* devant être résolues.

Ainsi, le Data Profiling forme le début logique d'un cycle d'amélioration de la qualité et peut également former la base du développement de nouvelles applications sur la base des données analysées (les données sont-elles bien adaptées à la nouvelle utilisation qu'on veut en faire) ou de la migration de données vers d'autres projets.

[1] Olson J., Data Quality : the Accuracy Dimension. Elsevier : The Morgan-Kaufmann Series in Database Management, 2002.



Les méta-données peuvent être inadéquates pour les raisons suivantes :
- elles sont incomplètes ou inexistantes ;
- erronées ou vagues et imprécises ;
- obsolètes ;
- …

Les données peuvent être inadéquates pour les raisons suivantes :
- elles sont invalides ;
- contradictoires par rapport aux méta-données ;
- le contenu diffère de ce que l'on attendait ;
- …

Le Data Profiling suit une approche précise, méthodologique afin de parvenir, sur la base de méta-données et des données réellement présentes dont on ne connaît en général pas la qualité, à des méta-données corrigées et complètes d'une part, et à des informations les plus exhaustives possibles sur les données adéquates et inadéquates, d'autre part.

Pour que cette approche réussisse, ce qui est découvert par le Data Profiling doit être validé avec la connaissance *business*. C'est pourquoi une équipe de Data Profiling sera typiquement composée tant d'analystes techniques que de personnes disposant d'un profil *business* pertinent. Dans tous les cas, l'équipe devra être compétente dans les deux domaines.

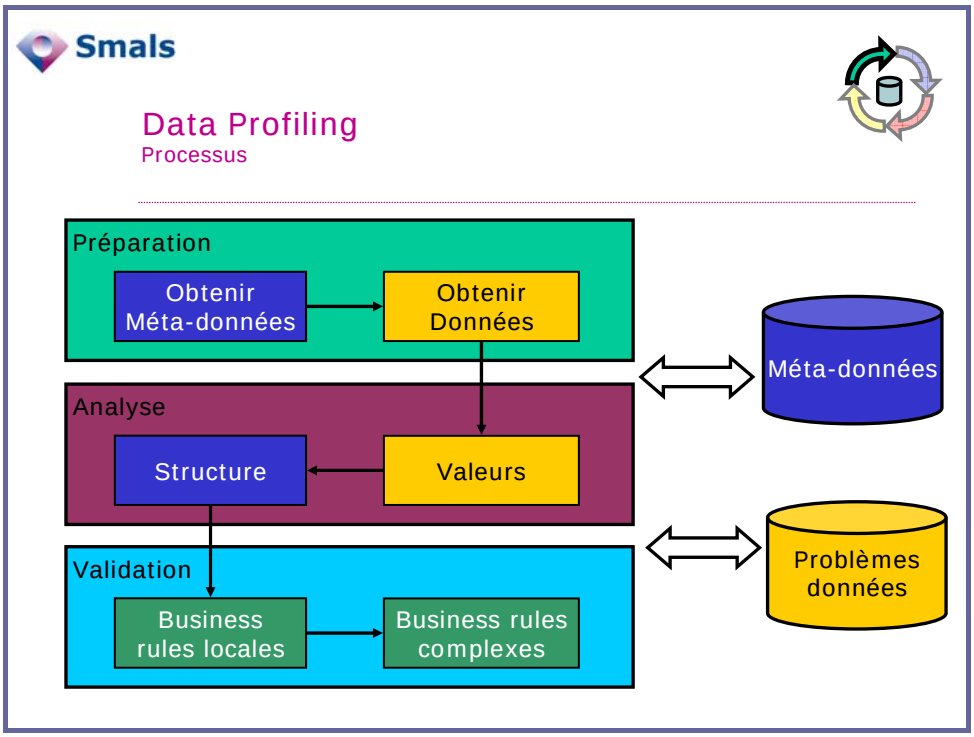
Notons que le Data Profiling vise donc des **informations** factuelles sur (l'exactitude formelle) des **données**, par exemple :
- 10% des champs NUM_TEL sont vides ;
- le domaine autorisé de HEURES_REF est [3600 - 4000],

et **non** des informations **déduites** des données :
- les travailleurs du secteur de la construction ont sensiblement plus d'accidents du travail.

On peut découper le processus de Data Profiling en plusieurs phases :

- La préparation ;
- L'analyse :
- La validation.

Ces principes ont été expérimentés avec succès (en l'absence de tool et donc, de manière restreinte) chez Smals, au sein de la « Data Quality Cel » en vue de réaliser le profiling d'un sous-ensemble de la KBO (Kruispunt Bank voor Ondernemingen) du SPF Economie.



Préparation

La préparation est loin d'être banale et comprend :
- l'extraction des données nécessaires (il est impossible d'effectuer les analyses dans un environnement de production) et les transformer dans une forme normale, dans l'hypothèse où l'on dispose de bases relationnelles[2] (théoriquement, la troisième forme normale est la meilleure) ;
- la collecte des méta-données et de la documentation ainsi que leur formalisation en vue d'un traitement exécutable ultérieur (voir plus loin) ;
- l'identification et le rassemblement des parties participantes (composition de l'équipe de Data Profiling, composée tant d'analystes techniques que de spécialistes du domaine d'application).

Cette étape est parfois la plus exigeante et celle qui prend le plus de temps.

Les méta-données et la documentation peuvent provenir de sources très différentes :
- Systèmes de gestion de l'information
  - Dictionnaires des données (ex : glossaires)
  - Répertoire de méta-données
- Définitions des données
  - Copybooks COBOL
  - Catalogues
- Logiques/Règles business
  - Programmes
  - Analyses fonctionnelles
  - Instructions aux utilisateurs
- Personnes
  - DBA, Architecte de données, Analyste business, utilisateurs

[2] ELMASRI R. et NAVATHE S. B., Fundamentals of Database Systems. New York : Pearson Addison Wesley, 2007 (5ème édition), p. 325-367.

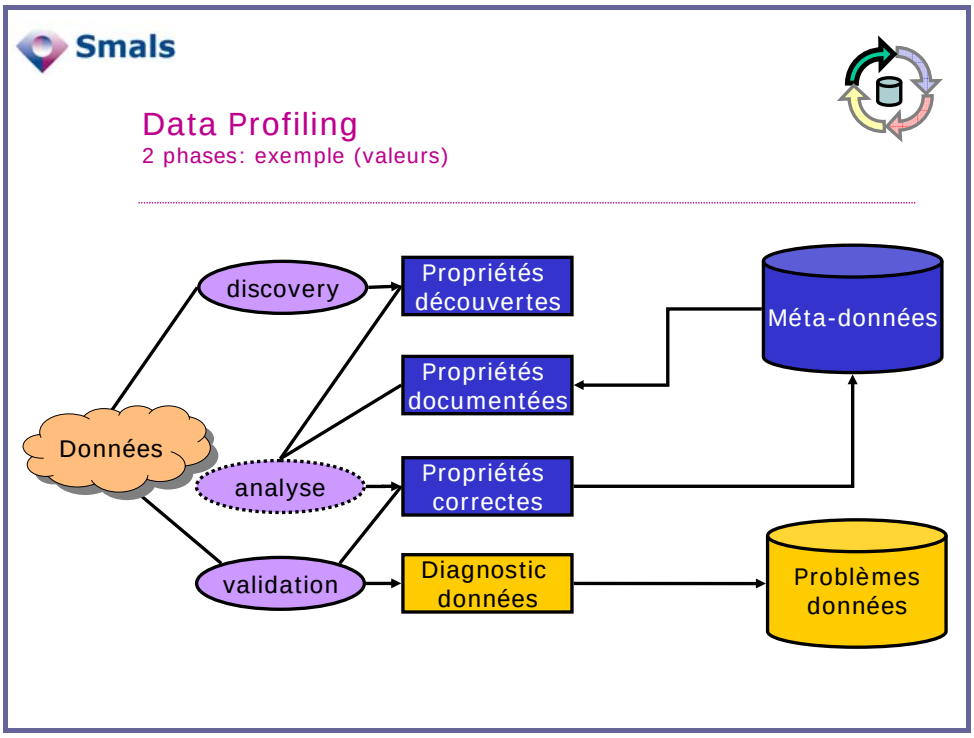
Analyse & Validation

Les analyses poursuivies dans le processus de Data Profiling ne peuvent être considérées indépendamment des validations indispensables. Les analyses peuvent être subdivisées en fonction de leur centre d'intérêt, portant d'une part sur les *valeurs* constatées par attribut, d'autre part sur la *structure* de la base de données. Dans ce cas, on différencie à chaque fois deux étapes :

- Une phase de *Discovery*, lors de laquelle on découvre de manière automatisée, bottom-up, *ce qui est* (sur la base des données réellement présentes) par rapport à *ce qui est permis ou visé* (ce dernier cas découle plutôt des méta-données et de la connaissance business) ;
- dans une seconde étape
  1) on vérifiera les caractéristiques découvertes sur la base des données à la lumière des caractéristiques documentées dans les méta-données (assertion testing),
  2) on analysera les différences et
  3) au moyen d'une validation offerte par des spécialistes *business*, on parviendra à des caractéristiques formellement correctes (complétées et corrigées) et à des diagnostiques les plus exhaustifs possibles indiquant quelles données sont en contradiction avec les caractéristiques correctes.

Une validation poussée permet de découvrir, de vérifier, de formaliser et de documenter des *business rules* (qu'elles soient simples, locales ou complexes) là où cela n'a pas encore été fait.

1. Accent mis sur les valeurs



Les méta-données liées aux valeurs des attributs décrivent en général les caractéristiques suivantes :

Automatiquement vérifiable		Exige une connaissance *business*
- Type	- Unicité	- Domaine (liste de valeurs valides)
- Longueur/précision	- Séquences	- Signification *business*
- Null autorisé	- Formatage	- Conventions d'encodage

Les analyses effectuées à ce niveau sur les valeurs des attributs sont regroupées sous la **Column Property Analysis**.

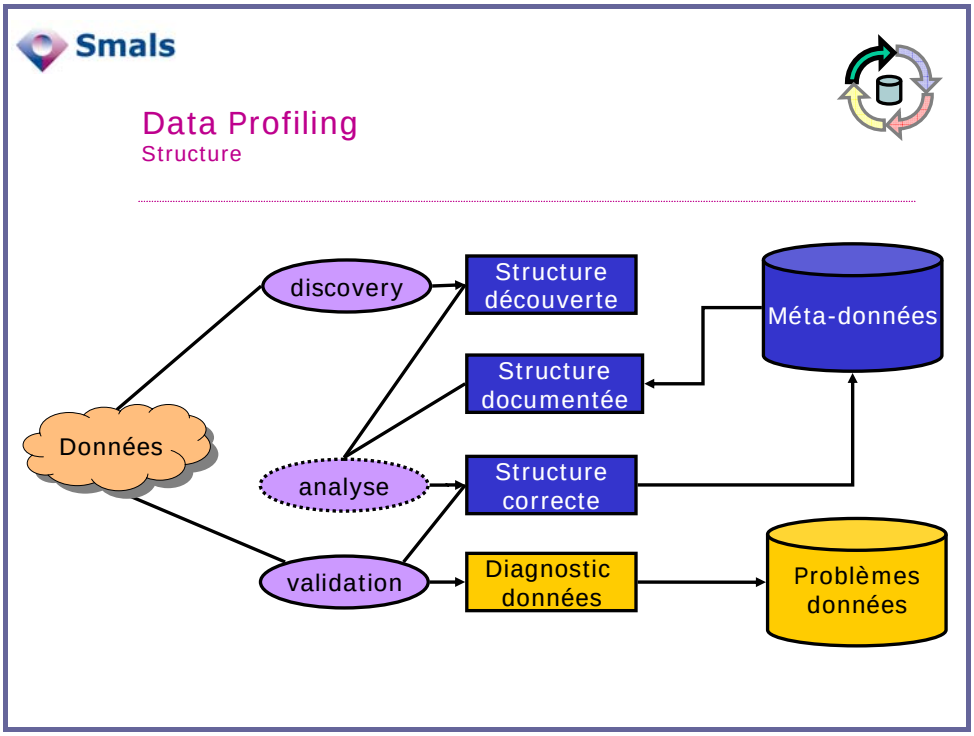
Les diagnostics suivants peuvent être composés sans peine à l'aide d'un logiciel de Data Profiling (analyses intégrées et automatisées) :

- Attributs non utilisés ou peu utilisés ;
- Types de données spécifiques (date, nombres entiers,…) ;
- Valeurs constatées
  - Valeurs en contradiction avec les méta-données / column properties ;
- Fréquence et distribution des valeurs constatées
  - Valeurs inutilisées ;
  - Outliers (~ valeurs inhabituellement grandes ou petites) ;
  - Statistiques descriptives (min, max, médiane, moyenne, écart-type, distribution, …) ;
- Représentations incohérentes (« sans numéro » est représenté parfois par « S/N », parfois par « z-n », etc.) ;
- Représentations de NULL (vide, « N/A », etc) ;
- *Patterns* (voir plus loin l'exemple de Data Profiling).

En plus, les logiciels de Data Profiling (faisant souvent partie des solutions de Data Quality) supportent le « *drill down* », c'est-à-dire le chargement et l'affichage de tous les

records pour lesquels un attribut analysé obtient une valeur définie, ou affiche un *pattern* défini, etc. - le tout accessible grâce à un simple clic de souris.

2. Accent mis sur la structure



L'analyse de la structure se sert de règles définissant comment des attributs (colonnes) s'organisent les uns par rapport aux autres pour former des entités (tableaux) et comment des tableaux s'organisent les uns par rapport aux autres pour former des *business objects*.

D'un autre côté, il existe aussi des analyses qui se concentrent sur la *structure* de la base de données. Les principales caractéristiques vérifiées grâce à une analyse de structure sont les suivantes :

- Identifiants (primary keys), par exemple :
  - ONSS_NR
- Primary & foreign key pairs
- Redundant data columns
- Column synonyms
- Relations (*jointures*)
  - Par exemple, FORM_JUR est un code documenté dans la table annexe à la table EMPLOYEUR, « FORMES_JURIDIQUES »
- Dépendances fonctionnelles (les valeurs d'un ou plusieurs champs déterminent les valeurs d'un ou plusieurs autres champs), par exemple :
  - INS_CODE → VILLE

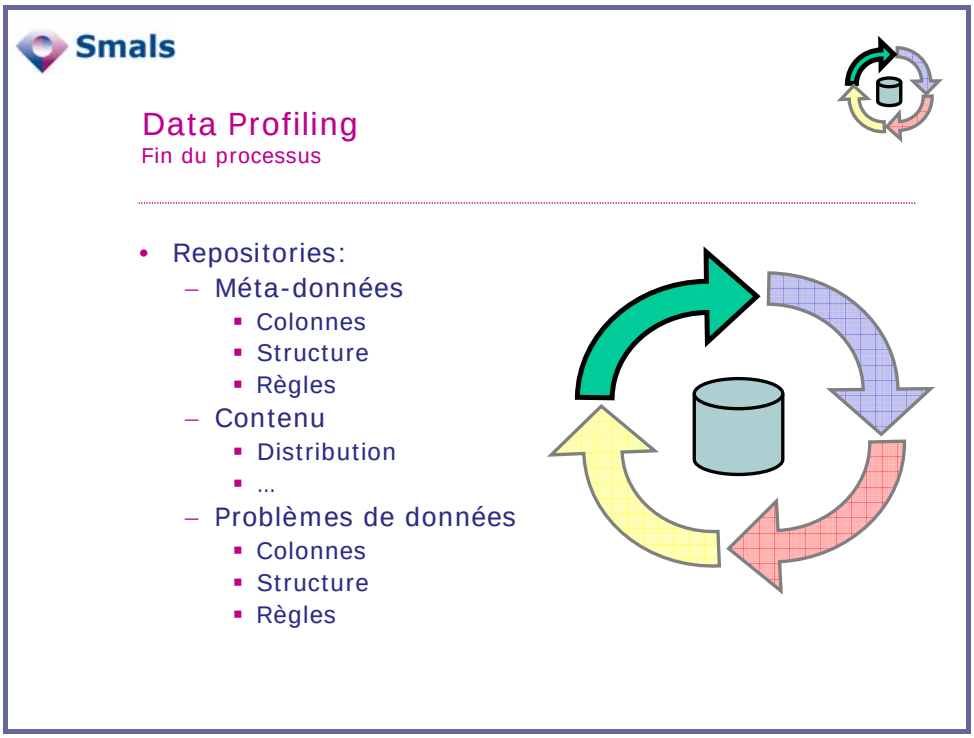
***Business rules* simples et complexes : validation**

Une fois que toutes les valeurs invalides dans les colonnes ainsi que toutes les infractions sur les règles de la structure ont été identifiées, il y a aussi lieu d'examiner les règles qui prescrivent que les valeurs provenant de différentes colonnes doivent former une combinaison acceptable. S'il s'agit d'une combinaison de valeurs de colonne correspondant au même *objet business*, on parlera d'une *business rule* locale ou simple. S'il s'agit de conditions concernant les valeurs des colonnes de différents *objets business*, on parlera d'une *business rule* complexe.

Valide ≠ Correct
Une erreur courante : ce n'est pas parce qu'une valeur de colonne est *valide* qu'elle est aussi correcte, car même si elle appartient au domaine autorisé de cette colonne, il peut y avoir une business rule selon laquelle des combinaisons de valeurs de colonnes correctes décrivent tout de même une situation incorrecte.

Dans cette phase, l'analyste de Data Profiling et l'analyste business se réunissent afin de rassembler les business rules (qu'elles soient documentées ou non) et de spéculer sur toutes sortes de conditions qui devraient être remplies par les données. Celles-ci sont alors converties en une logique exécutable (par exemple SQL) afin de vérifier dans quelle mesure les conditions sont valables et d'extraire les enregistrements qui sont en infraction.

Ainsi, à condition qu'une validation poussée soit effectuée, les méta-données peuvent être corrigées et complétées et il est possible d'indiquer quelles données sont en infraction. Cette information est organisée dans un répertoire central et forme la base pour des actions de data quality devant être entreprises.



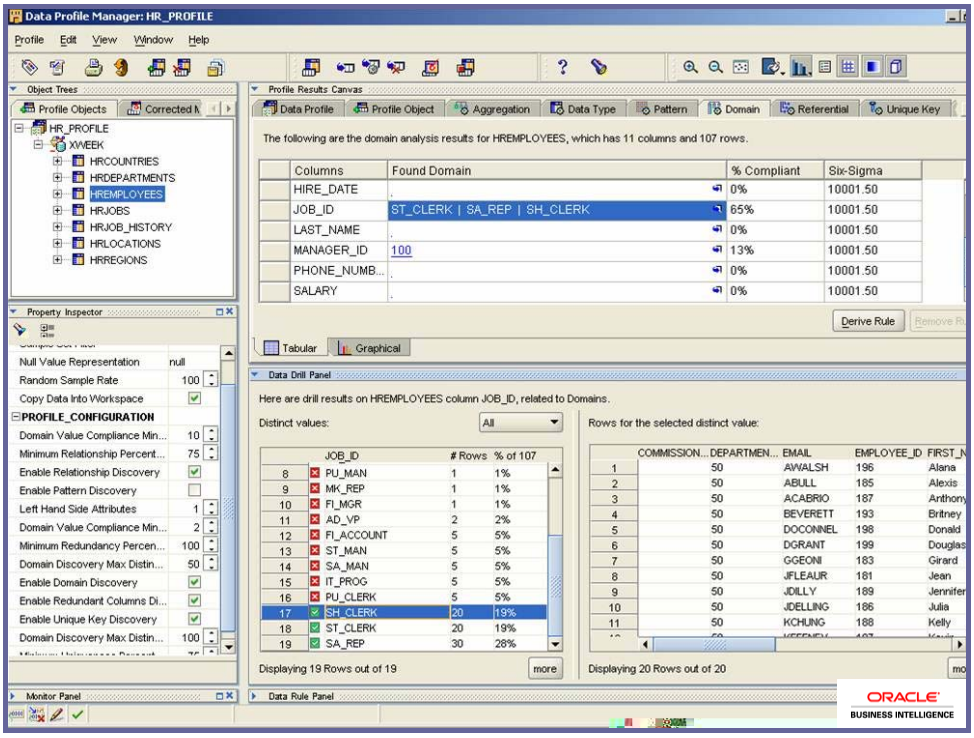
Data Profiling : un logiciel est-il nécessaire ?

Développer soi-même une méthode ad hoc pour effectuer ce qui précède comporte clairement les inconvénients suivants :

- un énorme investissement (en temps et en ressources humaines) pour générer le code et un grand nombre de sollicitations de la base de données d'une complexité croissante (par exemple SQL statements) ;
- un manque de support méthodologique :
  → risque de manquer des informations importantes;
- certaines analyses seront impossibles suite à la taille des bases de données modernes en combinaison avec la complexité des analyses :
  - dépendances fonctionnelles,
  - synonymes,
  - drill-down,
  - …
- il est difficile et coûteux de prendre en considération les changements de spécification en cours de projet (méthode du « trial and error »).

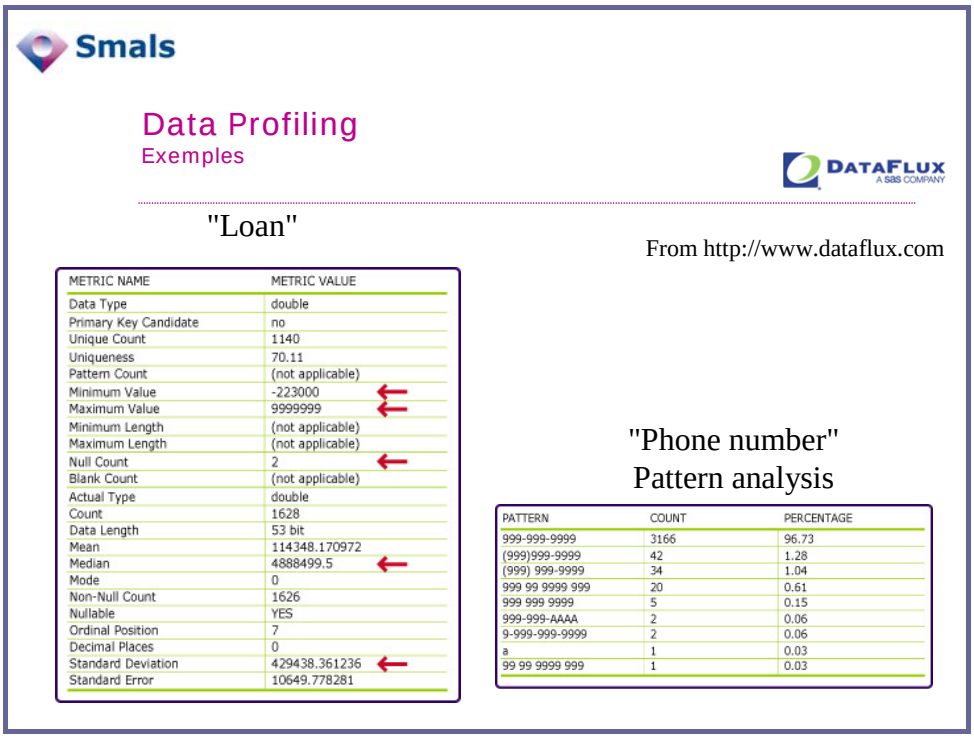
Les logiciels dont c'est le rôle d'effectuer de telles analyses surmontent ces restrictions.

Logiciel de Data Profiling : exemples



Une interface utilisateur utile propose toutes les fonctionnalités à portée de main, facilite l'inspection visuelle et permet le « drill down ». Il est aussi possible d'effectuer des corrections.

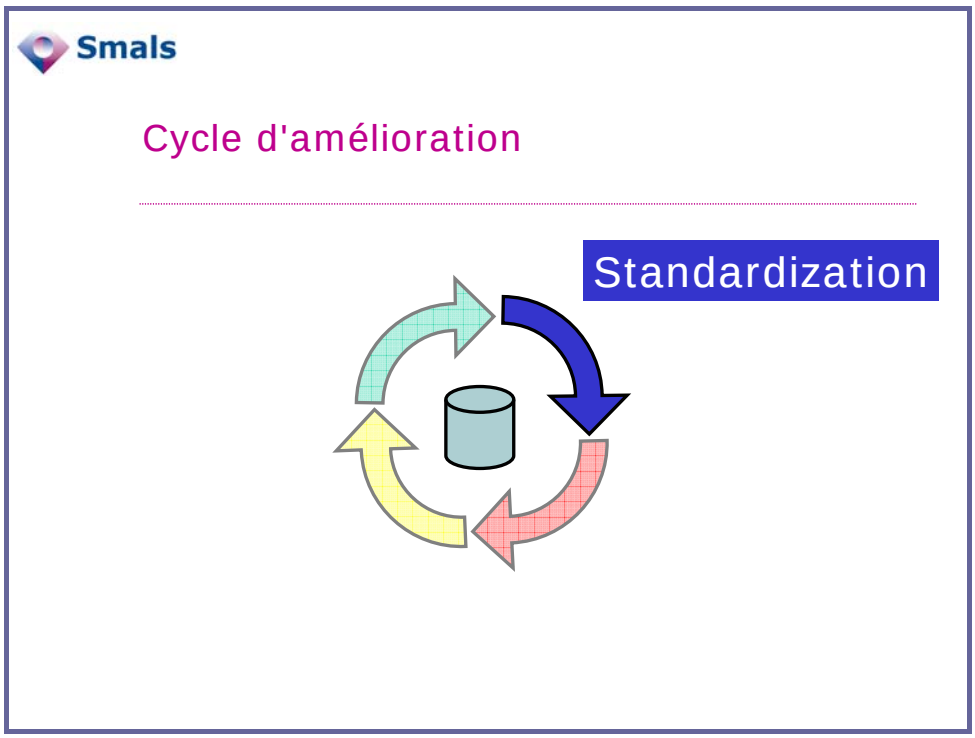
En général, les logiciels permettent également de spécifier des business rules et de les vérifier quant au contenu (*assertion testing*).



En un seul coup d'œil, on saisit à partir d'une certaine colonne quel est l'état du contenu de la base de données, par exemple :

- type d'une donnée,
- données statistiques sur la validité relative des valeurs,
- ***patterns*** et
- problèmes, par exemple :
  - valeurs négatives pour le montant d'un prêt,
  - caractères alphanumériques pour un numéro de téléphone.

2.2. Standardisation des données



Définition

Une deuxième fonctionnalité dans le cycle de l'amélioration de la qualité des données est la standardisation des données, que l'on peut définir le mieux au moyen des objectifs suivants :

- disposer de conventions univoques pour une représentation correcte des données (= standards) pour tous les attributs ;
- corriger la représentation des données, afin qu'elles suivent le standard (les diagnostics du Data Profiling indiquent dans une grande mesure là où la correction est nécessaire) ;
- corriger certains problèmes, identifiés par les activités de Data Profiling ;
- *parsing* et *enrichment* (enrichissement) des données :
  - la (re)structuration des champs qui sont non structurés, structurés de manière erronée ou *overloaded*, au moyen de la subdivision en unités significatives (*parsing*) ;
  - l'ajout de connaissance avec une *business rule* - comme des calculs, des annotations (dans le cas où une adresse est invalide, un numéro de maison inexistant, …), l'information provenant de *lookup tables* (codes postaux de tableaux annexes, données géographiques et démographiques, données de logiciels commerciaux, Enterprise Reference Data ou Master Data) ;
- former la base pour un Data Matching (cette technique est détaillée plus loin) plus performant en facilitant le matching champ par champ pour les champs standardisés.

Domaines restreints et domaines complexes

Le « domaine », couvrant les valeurs autorisées d'un attribut, peut être *restreint* ou *complexe* :

- Standardisation dans le cas de domaines *restreints* :

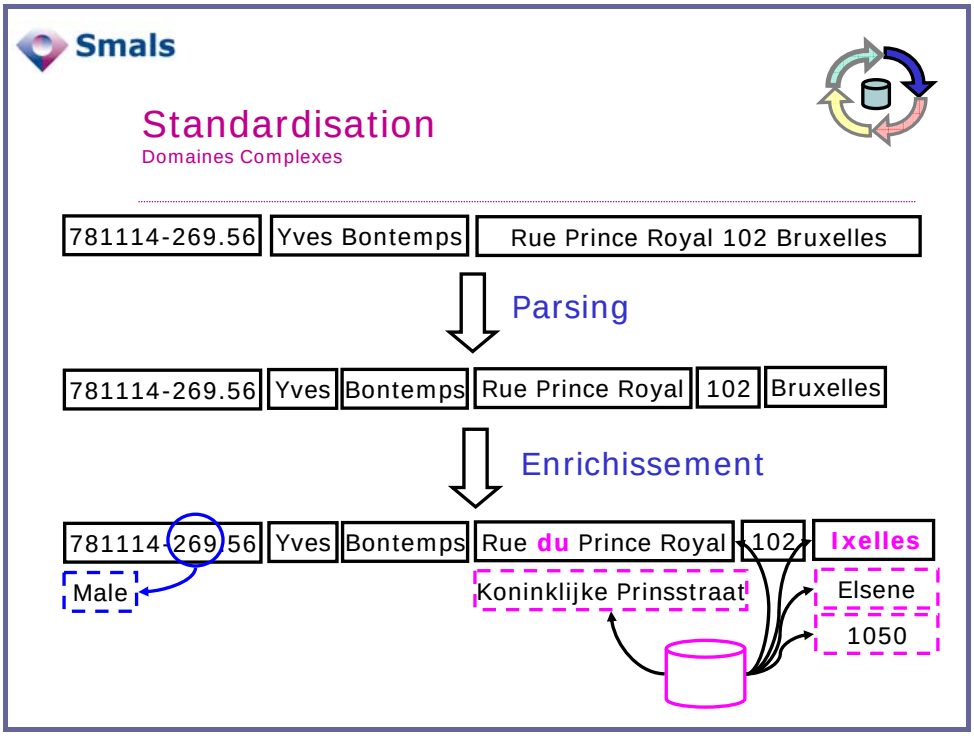
  Un domaine *restreint* peut se décrire à l'aide d'une liste finie et exhaustive de valeurs autorisées qui sont univoques. Si l'on impose un standard pour la représentation des données dans un domaine restreint, la transformation nécessitée pour passer des représentations alternatives à la représentation standard serait facilement définissable et exécutable. Ceci peut se faire aisément dans un outil de Profiling.

  Une liste fermée de codes postaux constitue un exemple typique de domaine restreint. Plusieurs données de la DmfA ont ainsi un domaine restreint (code travailleur, code prestation, activité par rapport au risque, …), comme en témoignent les « annexes structurées ».

- Standardisation dans le cas de domaines *complexes* (noms, adresses, etc.) :

  La standardisation est dans ce cas bien plus difficile et doit être soutenue par des fonctionnalités supplémentaires comme le *parsing* et l'*enrichment*, avec des informations spécifiques aux domaines et avec des données référentielles externes.

Parsing & Enrichment



Dans l'exemple ci-dessus, un champ d'adresse composé est subdivisé grâce au *parsing* en parties significatives (nom de la rue, numéro, commune) puis *enrichi* avec le nom de la rue dans l'autre langue nationale, avec le code postal et la commune.

Grâce à une connaissance spécifique du domaine, un numéro de registre ayant une structure connue (*pattern*) peut être subdivisé afin d'en retirer de l'information qui peut servir à l'*enrichissement* :

Pattern : 99.99.99-999.99

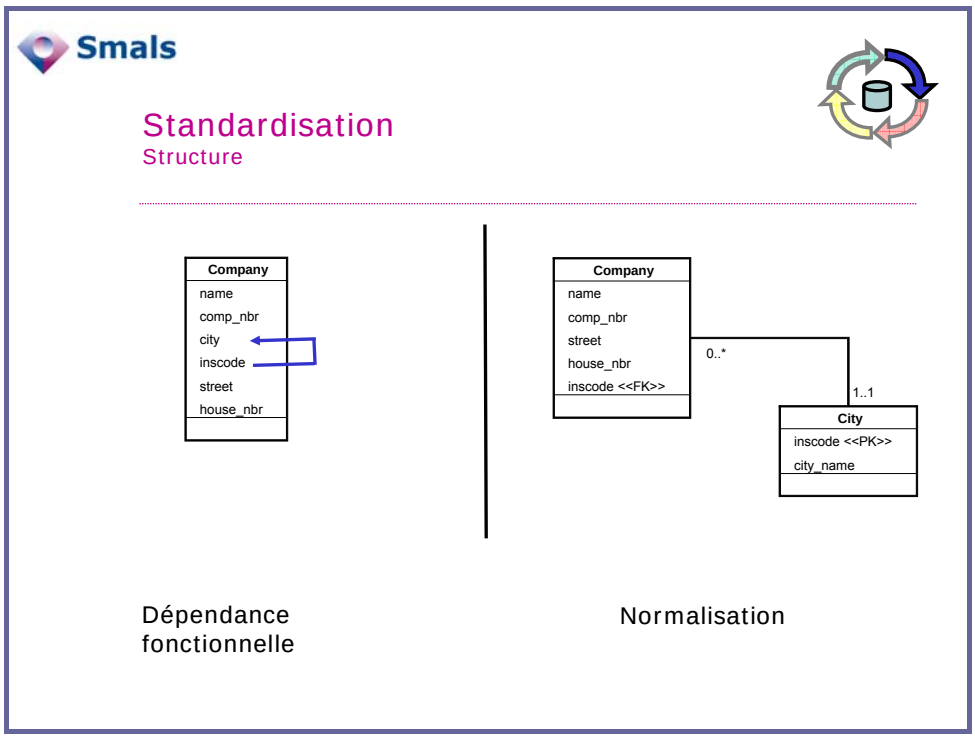
Sémantique : YY.MM.DD-CJN.CD (Compteur Journalier des Naissances, Check Digit)

Enrichissement : CJN mod 2 = 0 → sexe = 'M'

CJN mod 2 = 1 → sexe = 'F'

Date de Naissance = DD '/' MM '/' YY

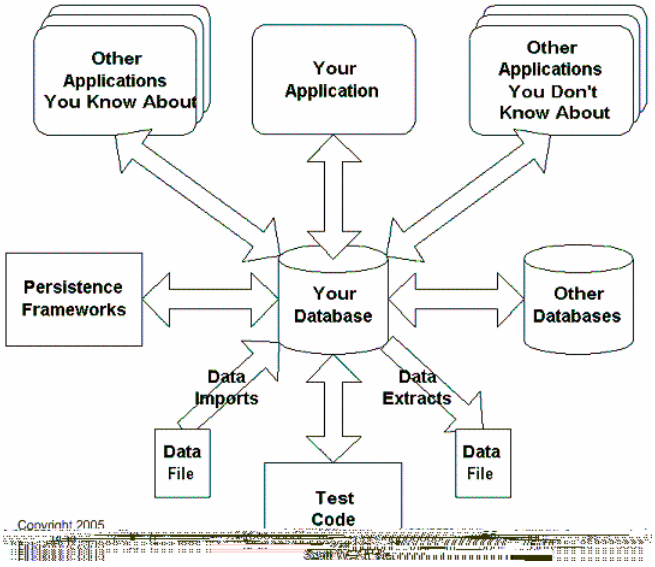
Standardisation de la structure de la base de données



La standardisation peut également se concevoir par rapport à la structure d'une base de données. Les structures standard pour les bases de données sont des « formes normales »[3].

Cela dit, imposer une forme normale implique potentiellement une modification du schéma de la base de données au niveau opérationnel.

Bien qu'idéalement, cette modification soit souhaitable, ceci est impossible d'un point de vue pratique, en raison des liens étroits qu'entretient une base de données avec différentes autres bases de données, programmes et applications. Cela exigerait une co-évolution de toutes les structures liées (qui ne sont pas toujours toutes connues). En outre, il n'existe pas de logiciel spécifique qui pourrait soutenir cet effort.



[3] ELMASRI R. et NAVATHE S. B., Fundamentals of Database Systems. New York : Pearson Addison Wesley, 2007 (5ème édition), p. 325-367.

Logiciels commerciaux pour la Standardisation

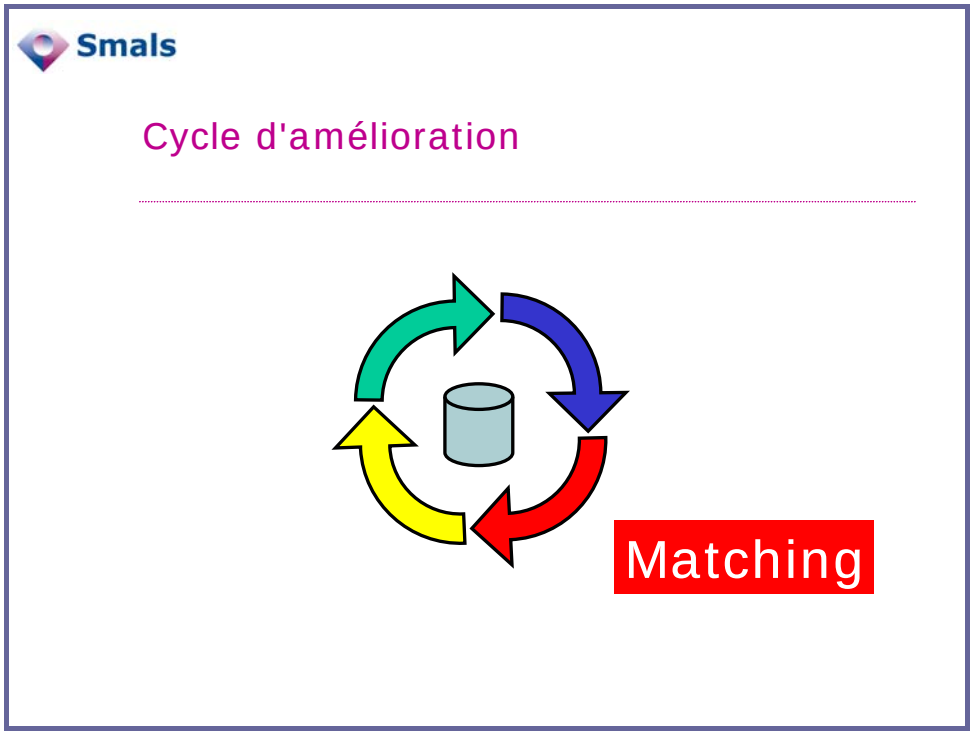
Le développement d'une logique exécutable en état d'effectuer la Standardisation des données revient donc à l'encodage d'une multitude de règles et de connaissance business, indispensable à un *parsing* et *enrichment* corrects. Ceci signifie un énorme effort.

Afin de soutenir les opérations de Standardisation des données, les logiciels de Data Quality incorporent des bases de connaissances spécialisées avec

- une abondance en règles applicables (50.000+),
- des dizaines d'années/homme d'expérience,
- des *lookup tables* (tableaux associatifs) avec de l'information géométrique et démographique,
- un support (bases de connaissance) dans des contextes et régions spécifiques (pays, langue, culture),
- la reconnaissance de patterns, des expressions régulières, grammaire,…

A l'avenir (dans 2 à 3 ans), suite aux progrès de l'Intelligence Artificielle, avec des modèles de probabilité (modèles Hidden Markov) et une capacité d'apprentissage, il sera possible d'envisager des mécanismes s'adaptant eux-mêmes au contexte (dans les limites d'un cadre formel).

2.3. Data Matching



Définition

Le Data Matching ou Record Linkage fait référence à la recherche de records relatifs à une même entité.

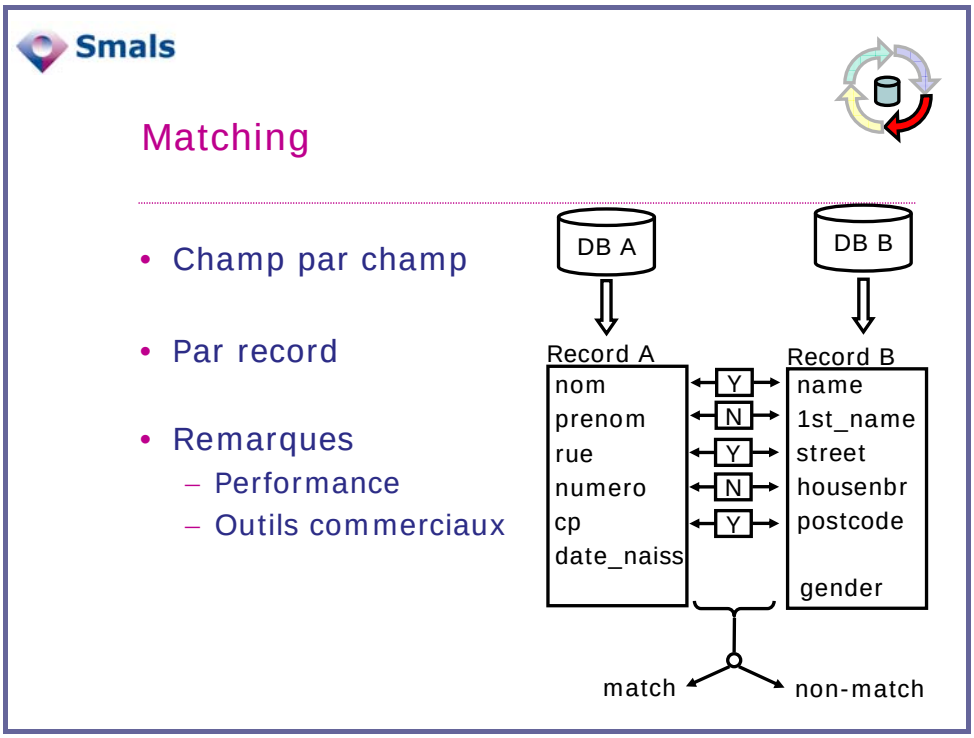
Comme les bases de données sont rarement formellement totalement valides et cohérentes, une même entité peut être présente dans plusieurs records et souvent aussi de manière différente et imparfaite. On parle alors de ***doubles et d'incohérences* dans une base de données**.

Il est en outre possible que différentes bases de données (avec une structure logique donnée) encodent (modélisent) la même entité ou le même ensemble d'entités de manière différente. Ceci cause donc des **incohérences entre les bases de données**. On compte parmi ces phénomènes l'exemple où une entité est contenue dans une base de données mais pas dans une autre et vice versa.

Les objectifs du Data Matching sont donc les suivants :

- Déduplication d'une base de données, en détectant les doublons ;
- Data Integration :
  - déterminer l'information adéquate en présence de plusieurs sources de données qui existent en concurrence ;
  - déduire de l'information et détecter les incohérences en reliant des bases de données ;
- Eviter l'introduction de nouveaux doublons ou incohérences (online matching à l'entrée).

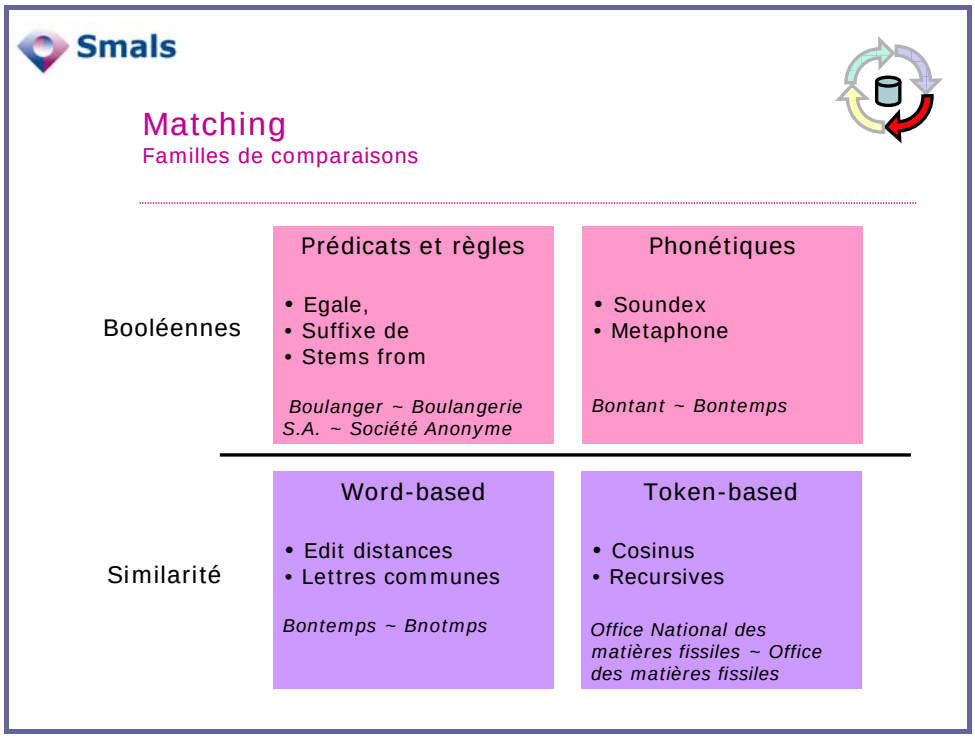
Matching



Comme nous l'avons vu, le Data Matching permet de comparer deux sources de données, afin d'y repérer les records communs (ou de détecter une série de records dans laquelle une sous-série cruciale de données est équivalente). Si l'on compare alors les sources de données et que l'on souhaite détecter les doubles (records redondants dans un seul répertoire). La comparaison entre les deux sources se fait généralement en deux phases :

- Lors d'une première étape, la **similarité** entre les records est définie **champ par champ**.
- Lors d'une seconde étape, les similarités champ par champ sont **agrégées** pour parvenir à un résultat global : pour chaque paire de records possible, il est décidé s'il s'agit d'un *match*, certainement *pas un match*, ou *peut-être un match*, et il peut être indiqué pourquoi.

Ces techniques seront approfondies plus loin.



Familles de comparaisons

Les nombreuses méthodes permettant d'opérer des comparaisons champ par champ peuvent globalement être subdivisées en familles. Nous en décrirons certaines d'entre elles, sans vouloir être exhaustifs.

D'une part, nous avons les méthodes **booléennes**, qui en résultat d'une comparaison débouchent toujours sur un *match* ou un *non match* (0 ou 1). Il s'agit soit d'algorithmes phonétiques, soit de méthodes consistant en une combinaison de prédicats (logiques) et de règles.

D'autre part, il existe les méthodes comparatives dont le résultat se présente sous forme de **score de similarité**, autorisant une appréciation *match* ou *non match* plus subtile. Ces méthodes peuvent être *word-based* (comparaison entre des *strings* de mots individuels) ou *token-based* (comparaison entre des *strings* de mots multiples).

Examinons maintenant quelques exemples de ces familles.

Comparaisons booléennes - Prédicats et règles

La première famille de comparaisons consiste en une collection de relations booléennes (affirmations logiques / prédicats) introduites et combinées selon un ensemble de règles pour parvenir à un résultat *match* ou *non match* :

Relations booléennes	**Exemples**
is-equal-to	Bontemps ~ Bontemps
is-a-prefix-of	Bon ~ Bontemps
is-a-suffix-of	Temps ~ Bontemps
is-an-abbreviation-of	S.A. ~ Société Anonyme
stems-from	dental ~ dent
…	

Les règles employant de telles relations pour parvenir à une affirmation :

1) sont parfois intégrées dans des outils commerciaux (le cas échéant, parfois, sous forme black-box, de sorte qu'on ignore parfois comment l'outil parvient à cette conclusion) ; ou
2) doivent être développées ; cette solution est possible à condition de posséder suffisamment de connaissances du domaine.
   - Point positif : cela peut être une bonne occasion d'exploiter les connaissances du domaine ;
   - Point négatif : le développement sera long et pénible et la maintenance très onéreuse.

Comparaisons booléennes - Algorithmes phonétiques

Les méthodes phonétiques reposent sur le principe que bon nombre d'erreurs et d'incohérences sont la conséquence de la retranscription de l'oral vers l'écrit.

Prenons l'exemple du nom « Bontemps » :

- Montand
- Bontant
- Bonton
- Bontand
- Beautemps, …

Pour y remédier, ces méthodes partent du principe suivant :

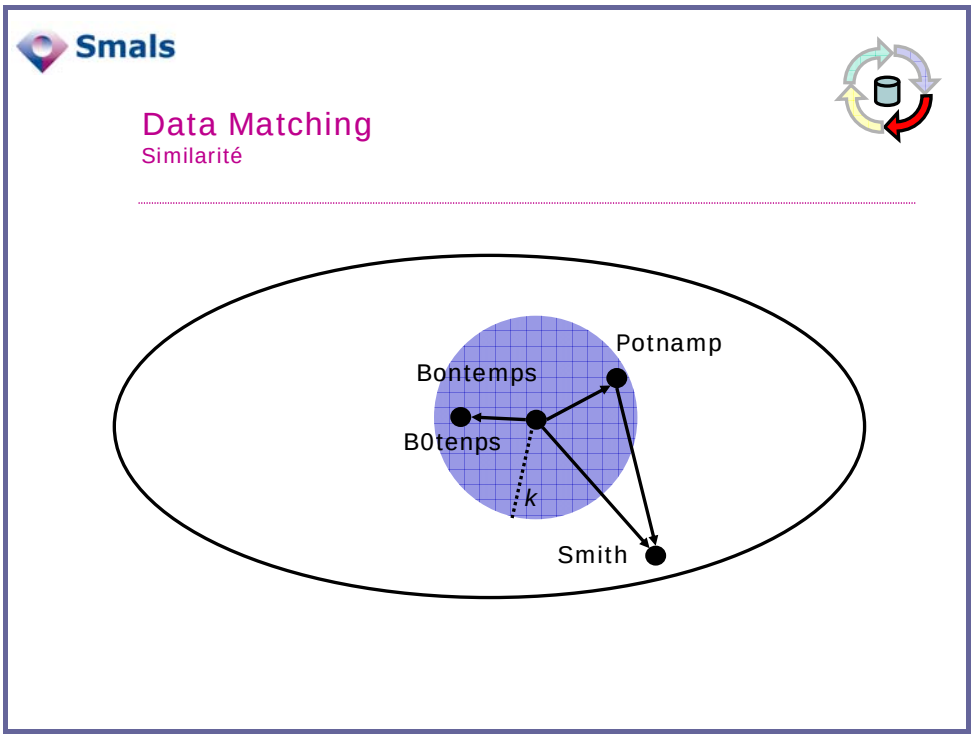
> Si d'un mot *m* la représentation de sa prononciation *p(m)*
> est égale à
> la représentation de la prononciation *p(m')* d'un autre mot m',
> alors
> *m ~m'* : les mots *m* et *m'* sont considérés comme désignant la même chose.

Exemple : Russel Soundex Algorithm (1918)[4]

Algorithme	**Exemple**	
	- Bontemps	- Bondant
- Garder la première lettre	B	B
- Supprimer a, e, h, i, o, u, w, y	BNTMPS	BNDNT
- Codage selon les six règles suivantes : 1: B,F,P,V 2: C,G,J,K,Q,S,X 3: D,T 4: L 5: M,N 6: R	B53512	B5353
- Garder les 4 premiers symboles (*padding* avec 0 : si le résultat compte moins de 4 caractères, rajouter des 0)	B535	B535

Il existe de nombreuses variantes[5].

[4] Soundex : http://west-penwith.org.uk/misc/soundex.htm
[5] A. J. Lait & B. Randall "An Assessment of Name Matching Algorithms" Dept of Computing Science, University of Newcastle upon Tyne : http://homepages.cs.ncl.ac.uk/brian.randell/Genealogy/NameMatching.pdf



Méthodes basées sur la « similarité »

Les méthodes comparatives générant un score de similarité offrent une appréciation plus fine que les méthodes booléennes (0 ou 1 ; *match* ou *non match*) car elles reposent sur le principe qu'il existe une similarité variable entre par exemple (voir illustration ci-dessus) :

- Bontemps
- B0tenps
- Potnamp
- Smith

La difficulté réside parfois dans la définition d'une valeur seuil (rayon *k*) à partir de laquelle la similarité est interprétée comme *match* ou *non match*. Comme nous l'avons dit précédemment, ces méthodes peuvent être *word-based* (comparaison entre des *strings* de mots individuels) ou *token-based* (comparaison entre des *strings* de mots multiples).

Méthodes de similarité word-based : « Edit Distance »

La méthode de similarité « Edit Distance » définit la distance entre deux mots comme *le nombre « d'opérations » nécessaires pour passer d'un mot à l'autre.* Les opérations autorisées sont les suivantes :

- Insertion (I)
- Effacement (D)
- Substitution (S)

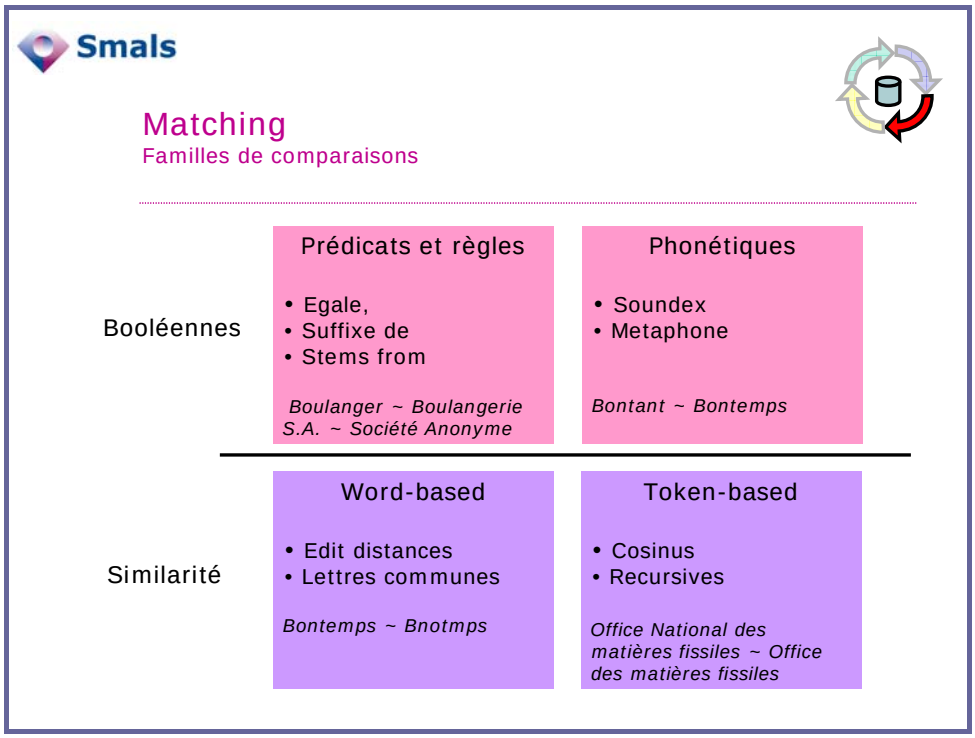
Exemple : pour calculer la distance en vue d'un matching entre le nom correct (« Bontemps ») et un nom incluant des fautes de frappe (« Bnotmps ») :

- B**n**ontemps (I)
- Bnotemps (D)
- Bnotmps (D)

Trois opérations sont nécessaires. Par conséquent, la distance est égale à 3.

Nous pouvons toutefois y attribuer des poids plus subtils, en accord avec les difficultés typiques liées à l'OCR (par exemple la substitution de O pour C) ou les fautes de frappe habituelles, ou en fonction de la disposition des touches sur le clavier (sanctionner moins sévèrement les substitutions de lettres proches l'une de l'autre sur le clavier), etc.

Il existe de nombreuses approches alternatives (voir annexe).



Méthodes de similarité token-based

Ces méthodes permettent de comparer des *strings* de plusieurs mots. Elles ont pour caractéristique d'être **insensibles à l'ordre** dans lequel les mots apparaissent[6].

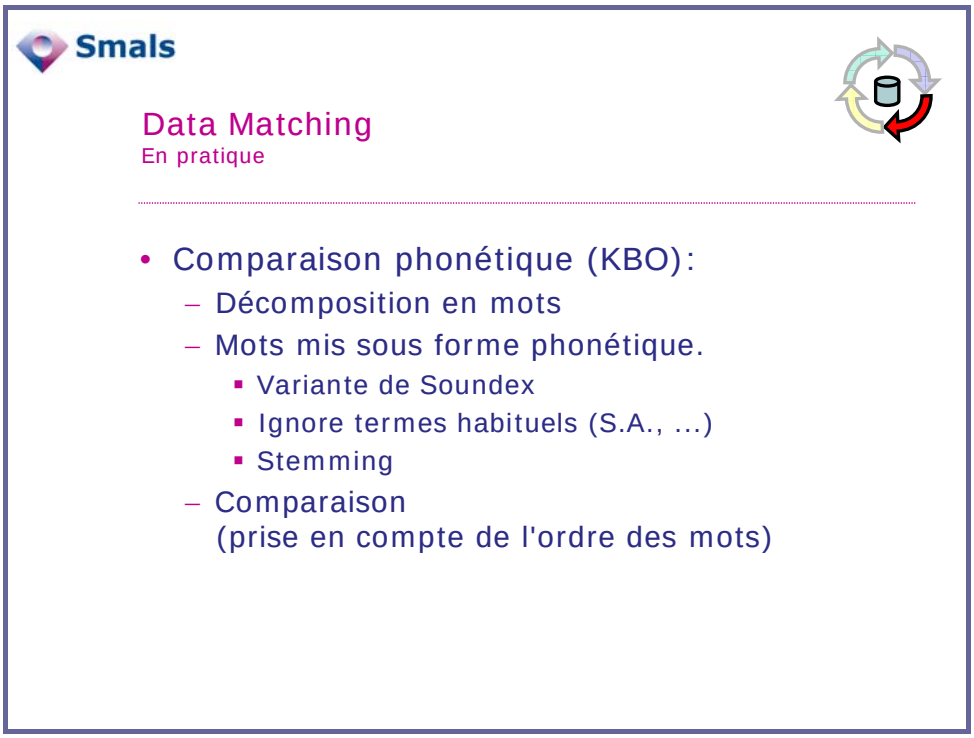
Souvent, on utilise des **schémas de pondération,** tels que TF-IDF (Term Frequency / Inverse Document Frequency), où les mots plus rares sont plus importants dans l'appréciation finale : *match* / *non match* (exemple : dans le contexte d'institutions fédérales, les mots *Matière*, *Fissiles*, *Déchets*, *Radioactives* sont les plus importants dans la comparaison entre « Organisme pour la Gestion des Déchets Radioactifs et des Matières Fissiles » et « Office National des Matières Fissiles et Déchets Radioactifs »).

Dans la pratique, les méthodes de similarité token-based sont généralement **récursives** (exemple : « Soft TF-IDF[7] ») :

- dans une première phase, elles utilisent une technique word-based (voir plus haut) pour déterminer si deux « tokens » des *strings* à comparer sont identiques :
  - Office ~ Ofice
- dans une deuxième phase, ces résultats sont combinés avec une technique token-based.

[6] Exemples typiques : « Cosinus » et « Jaccard ».
- Le cosinus de l'angle entre des vecteurs multidimensionnels dont les dimensions sont formées par les mots est utilisé comme mesure de la similarité.
- La similarité Jaccard est rendue par la proportion du nombre de mots communs (intersection) dans l'ensemble des différents mots (union).

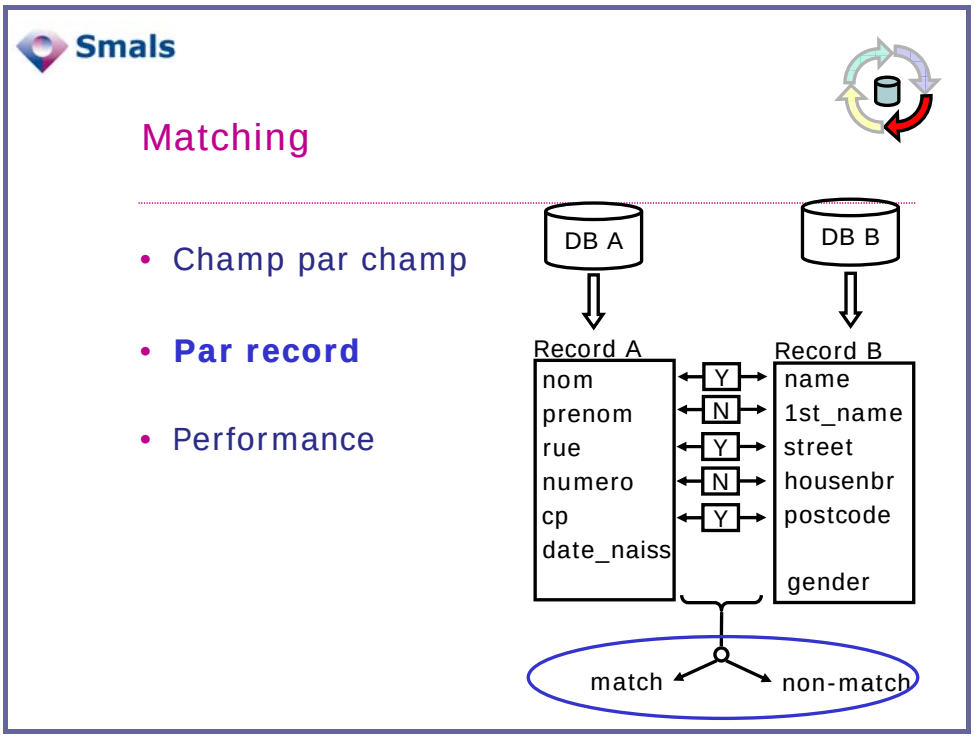
[7] COHEN, W., RAVIKUMAR, P., AND FIENBERG, S. 2003. A comparison of string distance metrics for name-matching tasks. In The IJCAI Workshop on Information Integration on the Web (IIWeb). Acapulco, Mexico. http://www.cs.cmu.edu/~wcohen/postscript/ijcai-ws-2003.pdf



Data Matching dans la pratique - Comparaison phonétique à la BCE

Le schéma Data Matching suivant est présent à la BCE en tant que fonctionnalité pouvant être utilisée pour les recherches :

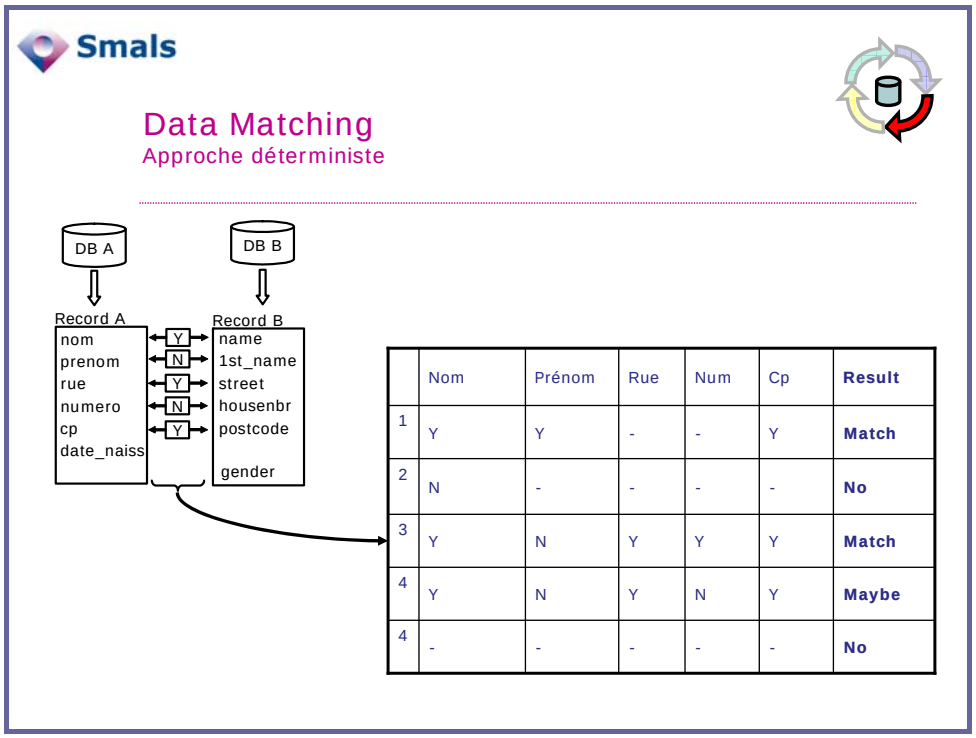
- les records avec strings sont subdivisés en mots individuels ;
- les mots sont transcrits sous forme **phonétique**
  - avec une variante de l'algorithme *Soundex* (voir plus haut) ;
  - il est aussi procédé à un « stemming » (réduction des mots à leur radical).
- la comparaison est réalisée
  - **sans** tenir compte de **termes fréquents** (comme S.A., …), qui tantôt sont mentionnés, tantôt non ;
  - **mais l'ordre des mots** joue **bien** un rôle.



Data Matching *par record* - « Record Linkage »

Ce qui précède n'est que la moitié de la question : une fois les comparaisons champ par champ accomplies, les résultats sur les records complets doivent être combinés pour déboucher sur une évaluation *match* ou *non match* (également *maybe* dans certains cas).

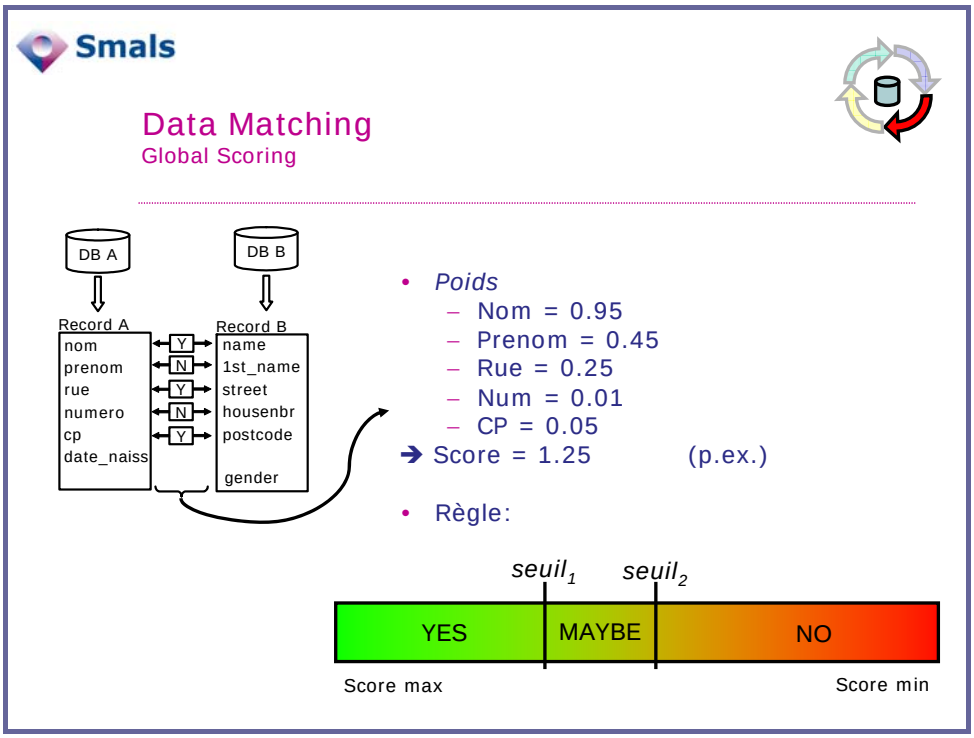
Globalement, nous distinguons deux approches : l'une *déterministe*, l'autre *probabiliste*. Nous les présentons successivement ci-dessous.



	Nom	Prénom	Rue	Num	Cp	Result
1	Y	Y	-	-	Y	**Match**
2	N	-	-	-	-	**No**
3	Y	N	Y	Y	Y	**Match**
4	Y	N	Y	N	Y	**Maybe**
4	-	-	-	-	-	**No**

Data Matching - Approche déterministe

Dans cette approche, l'appréciation *match* / *non match* peut être décrite au moyen d'une table de vérité : on énumère de manière déterministe les combinaisons de comparaisons champ par champ (dans cette approche, il s'agit normalement de résultats booléens) qui débouchent respectivement sur *match*, *non match* et *maybe*. Ceci peut être interprété comme un ensemble de *règles business* indiquant s'il s'agit ou non d'un *match*. L'ordre des règles dans le tableau de vérité peut avoir de l'importance le cas échéant.



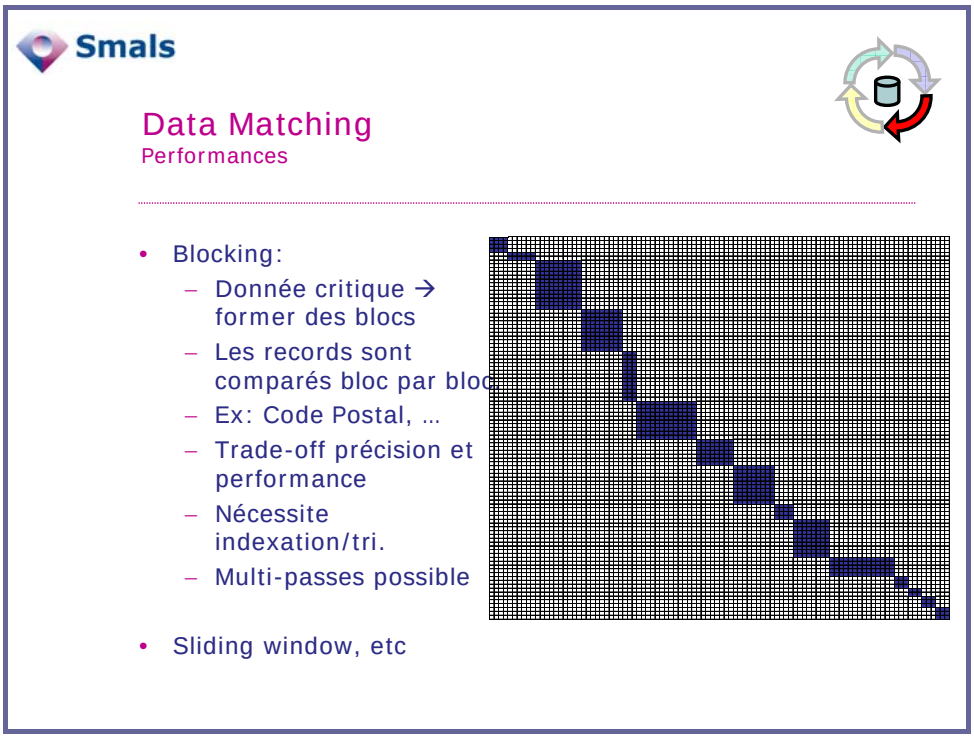
Data Matching - Approche probabiliste (« *Global Scoring* »)

Si l'on attribue à chacun des sous-critères champ par champ un poids relatif (voir l'exemple dans la figure ci-dessus), on obtient le score de *matching* global (par paire de *records*) représentant la somme pondérée des scores de comparaison champ par champ. Dans l'approche probabiliste, les comparaisons champ par champ sont souvent des scores de similarité, donc exprimées en variables continues. Le résultat n'est donc pas simplement 0 ou 1, ce qui laisse de la place à l'interprétation. Dans la pratique, on ne travaille ainsi pas avec un seuil, mais avec deux seuils sur l'échelle continue (voir également la figure ci-dessus). On obtient ainsi une zone grise, dans laquelle l'algorithme ne peut déterminer si une comparaison est *match* ou non. Ces cas doivent donc absolument être marqués d'un drapeau pour vérification ou manipulation manuelles au moyen d'une autre technique.

Les logiciels commerciaux proposent généralement cette approche probabiliste, dans une ou plusieurs variantes. De par leur expérience, les spécialistes du *business domain* peuvent ainsi attribuer et affiner les poids relatifs et attendre des résultats sans devoir posséder des connaissances algorithmiques.

Un exemple d'une telle technique probabiliste, souvent considérée comme fondamentale dans la littérature, est la méthode proposée par *Fellegi & Sunter,* en 1969. Cette méthode tient compte de l'incertitude qui peut exister et exprime celle-ci au moyen deux probabilités :

- m = la probabilité que la méthode prononce un *match*, lorsqu'il s'agit réellement d'un match ;
- u = la probabilité que la méthode prononce un *match*, lorsqu'il ne s'agit pas réellement d'un match (i.e. la probabilité d'un *false positif*).



Data Matching - Performance

Exécuter un matching consiste généralement à comparer chaque record d'une source avec chaque record d'une autre source (ou à comparer tous les records avec chaque autre record en cas de détection de doublons dans un seul répertoire). Par exemple, si les répertoires à comparer comportent un nombre de records de l'ordre de 10^5 (cent mille), le nombre de comparaisons à effectuer sera de l'ordre de 10^{10} (dix milliards).

Or, même avec l'infrastructure hardware la plus moderne, une telle opération **est loin d'être négligeable**. Tant au niveau de la mémoire qu'au niveau du temps de traitement, cela peut occasionner un problème insurmontable. Aussi des **optimisations sont-elles nécessaires** dans les méthodes et logiciels modernes de Data Matching. L'une des possibilités d'optimisation les plus répandues est le *Blocking*.

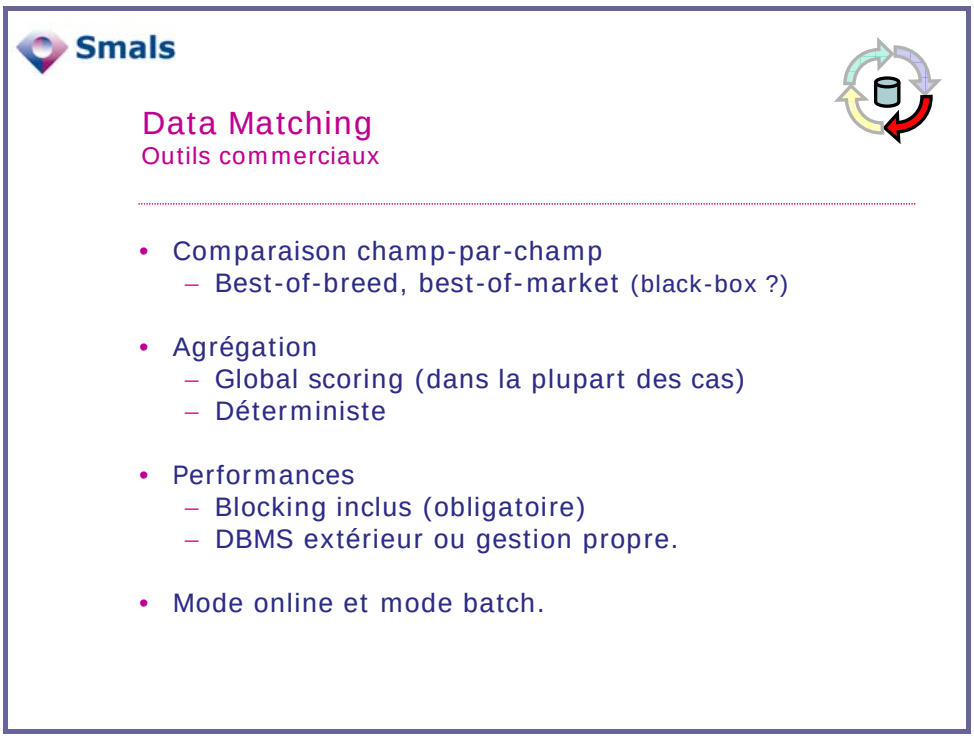
Technique d'optimisation : ***Blocking*** - principe de base :

- blocage sur la base d'une donnée critique (par exemple : code postal) :
  - pour chacune des valeurs présentes dans la colonne critique, les records possédant la même valeur pour cette colonne sont réunis dans des blocs (voir figure ci-dessus) ;
  - une indexation et un tri de tous les records sont donc nécessaires.
- comparaison de records deux par deux dans chaque bloc :
  - seuls les records d'un même bloc (donc avec une même valeur pour la donnée critique) sont comparés deux par deux, et ce pour chaque bloc. Dans le cas des codes postaux, seules les entreprises auxquelles correspond le même code postal seront comparées deux par deux.
- choix ou identification d'une ou plusieurs « données critiques » :
  - à l'aide de techniques automatisées ou sur la base de connaissances du domaine, les « données critiques » pour le *Blocking* doivent donc être choisies de telle manière que l'on puisse espérer qu'aucun *match* ne soit possible entre les records qui revêtent différentes valeurs pour la « donnée critique » ;

- si possible, on choisira donc de préférence les données « précises », « stables », « fiables ». Cependant, étant donné qu'une certitude est impossible, il y aura probablement un *trade-off* entre précision et performance.

Il existe ici un grand nombre de variantes, d'affinements et d'alternatives[8].

[8] Winkler, William E. 2006. Overview of Record Linkage and Current Research Directions. http://www.census.gov/srd/papers/pdf/rrs2006-02.pdf

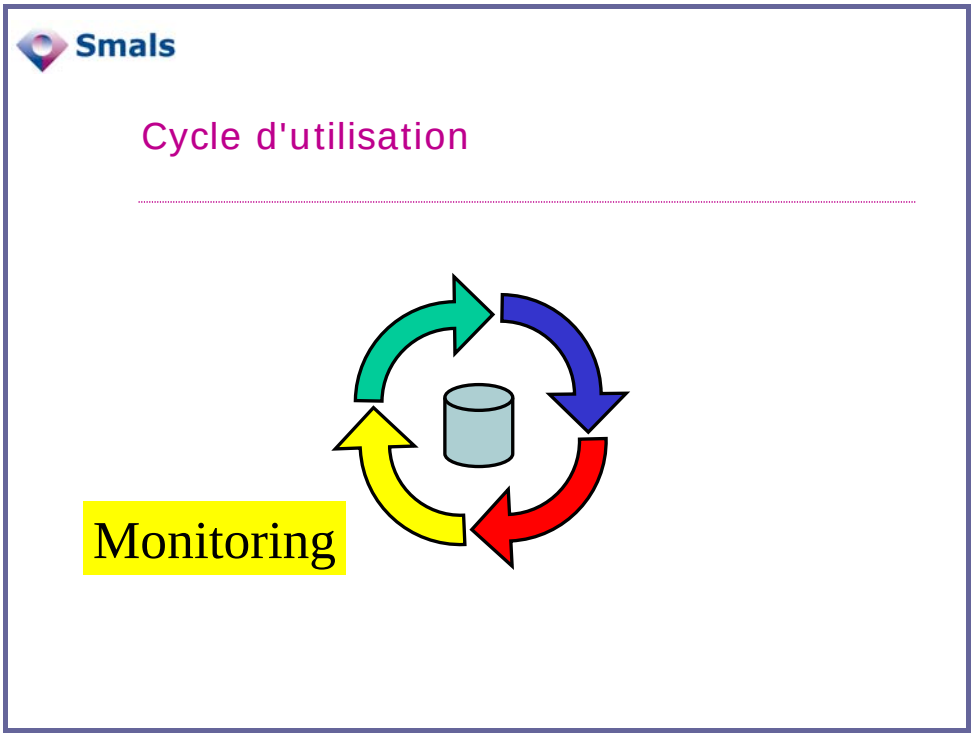


Data Matching - Logiciels commerciaux

Divers logiciels commerciaux proposent le Data Matching. En règle générale, ils ont les propriétés suivantes :

- ils comportent les comparaisons champ par champ best-of-breed, state of the art :
  - elles offrent souvent plusieurs alternatives ;
  - certaines sont des méthodes « black-box » : on ne sait pas exactement quel algorithme exécute la comparaison.
- la fusion de comparaisons champ par champ en un matching de records peut s'opérer de plusieurs façons :
  - la majorité des logiciels comportent au moins une technique « global scoring » probabiliste ;
  - certains englobent des techniques déterministes ou permettent d'élaborer des schémas déterministes.
- tous les logiciels comportent des techniques dont la performance est optimisée (grâce aux efforts de nombreuses années/homme) et supportent le *Blocking* sous l'une ou l'autre forme ;
- ils sont compatibles avec des DMBS (database management systems) externes ;
- certains disposent de leur propre DBMS ;
- ils peuvent fonctionner en *batch mode* (détection a posteriori des doublons et des incohérences dans un même répertoire ou entre plusieurs répertoires) et en *online mode* (dans une application en ligne, un nouveau record ou une mise à jour fait l'objet d'un *matching* avec le contenu d'une banque de données dans un délai acceptable).

2.4. Data monitoring



Data Monitoring - « *mesurez* et vous *saurez* »

Pour compléter le cycle d'amélioration de la qualité des données, il **faut** d'une manière ou d'une autre mesurer et examiner l'évolution de la qualité des données, et en suivre l'évolution sur une base régulière. Le Data Monitoring a pour but de réaliser cet objectif

- à l'appui d'indicateurs bien définis ;
- de manière automatisée ;
- de façon régulière, planifiée ;
- pour que la qualité présente puisse être connue et rapportée (« score boards ») ;
- pour que la qualité atteinte grâce à des initiatives d'amélioration puisse être préservée ;
- pour que les données responsables du dépassement de certains seuils de qualité et de la violation de *règles business* puissent immédiatement être *signalées* de manière à ce que les *actions qui s'imposent puissent être entreprises* ;
- pour que des tendances puissent être identifiées ;
- pour que des anomalies de phénomènes cycliques puissent être découvertes ;
- pour que les coûts puissent être estimés plus justement.

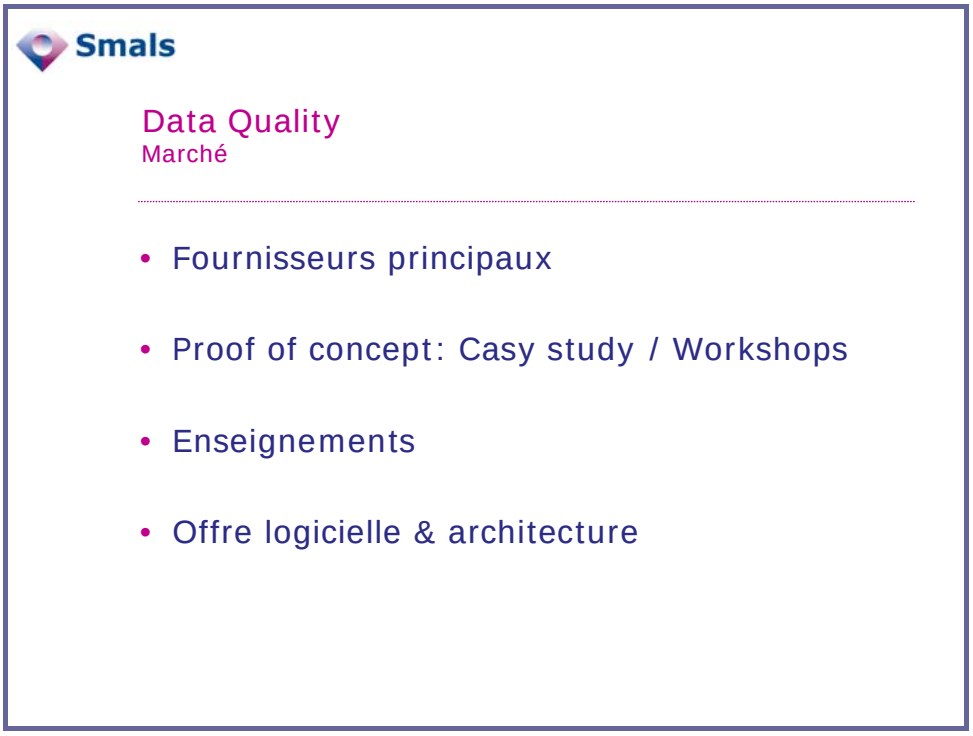
Les fonctionnalités imaginables sont légion. Dans la pratique, elles sont offertes dans une mesure variable par des logiciels commerciaux. Vu les perfectionnements possibles dans ce domaine, on assiste à une intégration croissante des fonctionnalités de monitoring sur le marché.

Il est par ailleurs essentiel que les logiciels de monitoring puissent aisément s'intégrer dans l'architecture IT et les systèmes « Decision Support » existants.

Data Monitoring - Lien avec le Data Profiling

Le lien avec le Data Profiling est notable. Les mêmes règles business, dont les violations doivent être détectées via des analyses de Data Profiling, doivent être contrôlées par le module de Data Monitoring. Les profils d'analyse Data Profiling doivent pouvoir être stockés et relancés régulièrement (i.e. monitoring).

3. DQ Tools: marché et « case study »



Dans ce chapitre, nous verrons :

- un aperçu des principaux fournisseurs de Data Quality Tools sur le marché en 2006, sachant qu'il n'existe pas de solution complète en open source pour l'ensemble de ces fonctionnalités;
- un *« case study : détection des doublons dans le répertoire des employeurs »* : la flexibilité et la capacité de détection des doublons du logiciel Data Quality de quatre fournisseurs sélectionnés ont été testées dans des workshops distincts sur la base de cette étude de cas;
- quelques enseignements, tirés des démonstrations ;
- un aperçu de l'architecture globale des solutions Data Quality et de l'offre de fonctionnalités (généralement similaires).

Fournisseurs principaux

Chaque année, la société Gartner (www.gartner.com) détermine les principaux fournisseurs pour un certain nombre de domaines-clés de l'IT. Le case study qui suit se base sur le Magic Quadrant for Data Quality Tools d'avril 2006.

Sur ce marché se démarquent deux phases de consolidation :

- les fournisseurs qui existaient sur le marché avec des solutions Data Profiling sont devenus des fournisseurs de solutions Data Quality ;
- les fournisseurs de solutions Data Quality deviennent des fournisseurs de solutions intégrales BI (Business Intelligence) et ETL (Extract-Transform-Load).

La tendance observée peut signifier que le Data Quality devient un service couramment employé pour la gestion quotidienne des bases de données.

Participants au *case study* - 4 workshops

Les fournisseurs qui ont été invités au *case study* sont les suivants:

- Firstlogic (Business Objects) ;
- Trillium Software ;
- DataFlux (SAS) ;
- Human Inference ;

Selon Gartner, ce sont tous des leaders du marché ou du moins des acteurs visionnaires. On trouvera d'autres logiciels du marché dans ce domaine cités par Gartner (*Magic Quadrant for data quality tools*, 2007) : Informatica, DataLever, Innovative Systems, …

***Case study* : détection des doublons dans le répertoire des employeurs**

Comme nous le savons, le répertoire des employeurs de l'Office National de Sécurité Sociale recèle des « doublons » et de nouveaux doublons s'y introduisent potentiellement par le biais des divers processus qui alimentent la banque de données.

Dans leur travail quotidien, les utilisateurs *business* du répertoire se heurtent parfois à des problèmes qui témoignent de la présence de doublons. Les doublons fortuitement, et manuellement découverts, sont conservés dans un historique. On ne dispose donc en aucun cas d'une liste exhaustive des doublons. La capacité de détection des doublons des solutions Data Quality des quatre fournisseurs sélectionnés a été testée en **réintroduisant dans le répertoire les doublons** découverts et préalablement supprimés. La banque de données ainsi altérée a été analysée par le logiciel des quatre fournisseurs, dans des workshops séparés. Les doublons découverts ont ainsi pu être comparés avec les doublons connus.

Afin d'éprouver la flexibilité et l'utilisabilité des solutions, les données n'ont été mises à la disposition des fournisseurs qu'au début des workshops. Avant cela, ils n'ont reçu qu'une description circonstanciée de la banque de données.

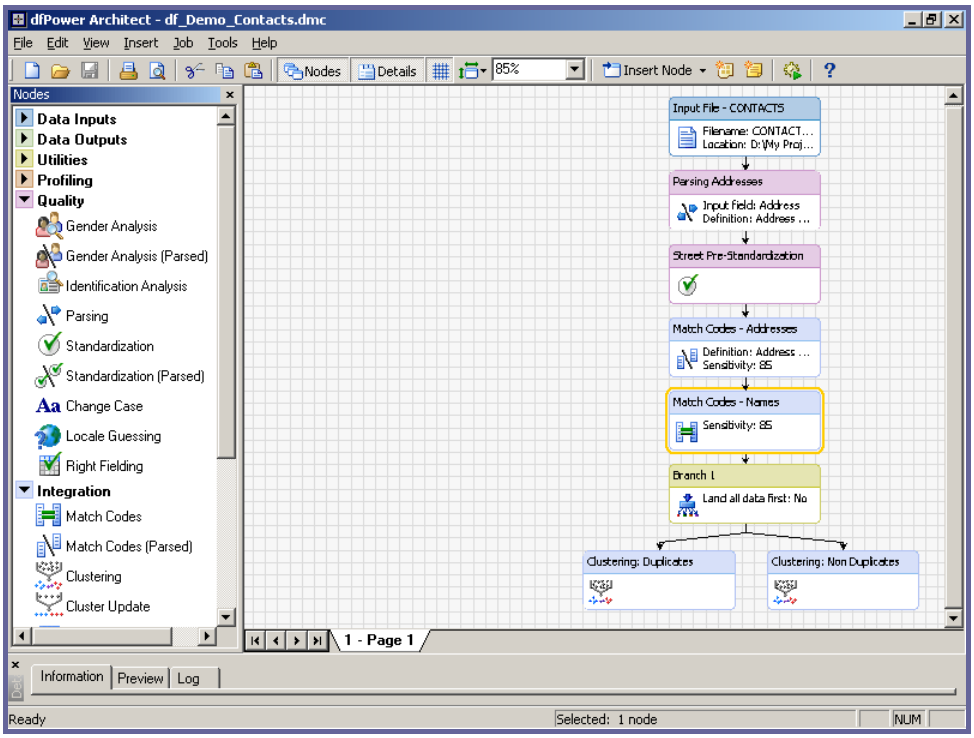
Ordre de grandeur du répertoire des employeurs : +/- 230.000 records.

Complexité : 10 colonnes ont été jugées intéressantes pour la détection des doublons. Ci-dessous, vous trouvez les colonnes avec des domaines complexes, comme :

- dénomination (exemple : « SmalS-MvM », …)
- adresse (exemple : « Rue du Prince Royal 102, 1050 Bruxelles »)
- forme juridique (exemple : « A.S.B.L. », « asbl », « VZW », …)
- …

La journée de workshop s'est déroulée comme suit :

- Profiling (2h)
- Matching (4h)
- Debriefing (1h30)



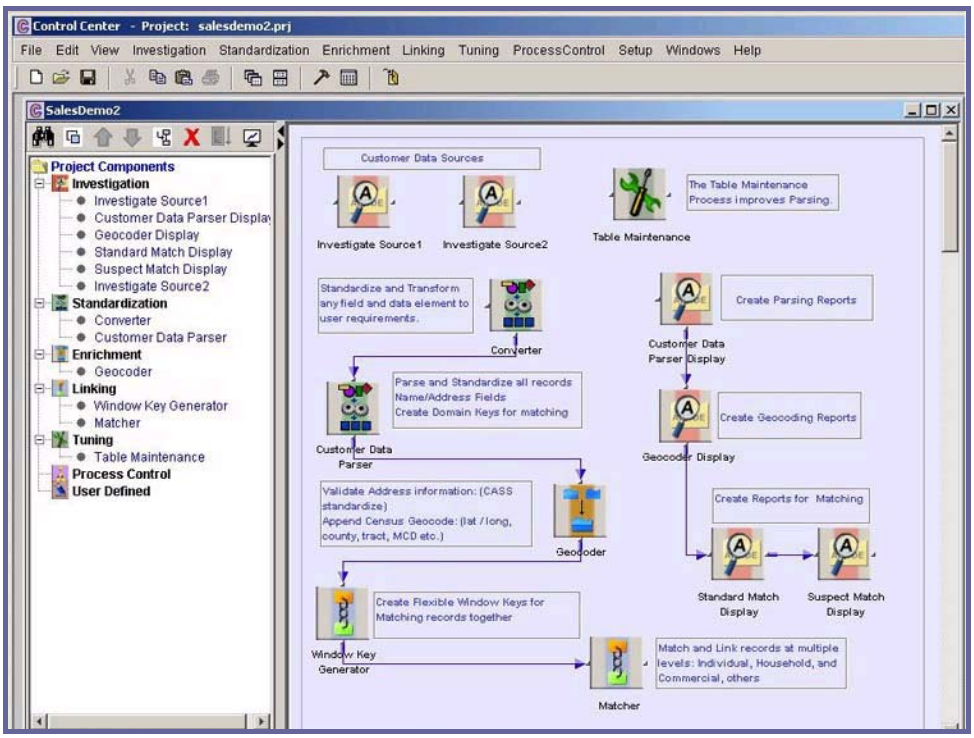
Démonstration du processus de matching, mis au point durant le workshop

A l'aide d'une interface utilisateurs graphique, intuitive, un processus de *matching* pour la détection des doublons a été élaboré sous forme de série de tâches successives (blocs) :

- input ;
- « parsing » des informations d'adresse (voir p.30 « Parsing & Enrichment ») ;
- standardisation des champs d'adresse « parsés » (via la *knowledge base* avec des informations d'adresse correctes, standardisées pour une certaine région - en l'espèce la Belgique) ;
- préparation du matching via un codage des informations d'adresse ;
- préparation du matching via un codage des informations de dénomination ;
- matching proprement dit, permettant de détecter :
  - les doublons ;
  - les records pour lesquels (sur la base de ce processus et de ces informations) il n'a pas été découvert de redondance.

S'ils le souhaitent, les utilisateurs expérimentés peuvent judicieusement adapter les paramètres *par défaut* de certains blocs.

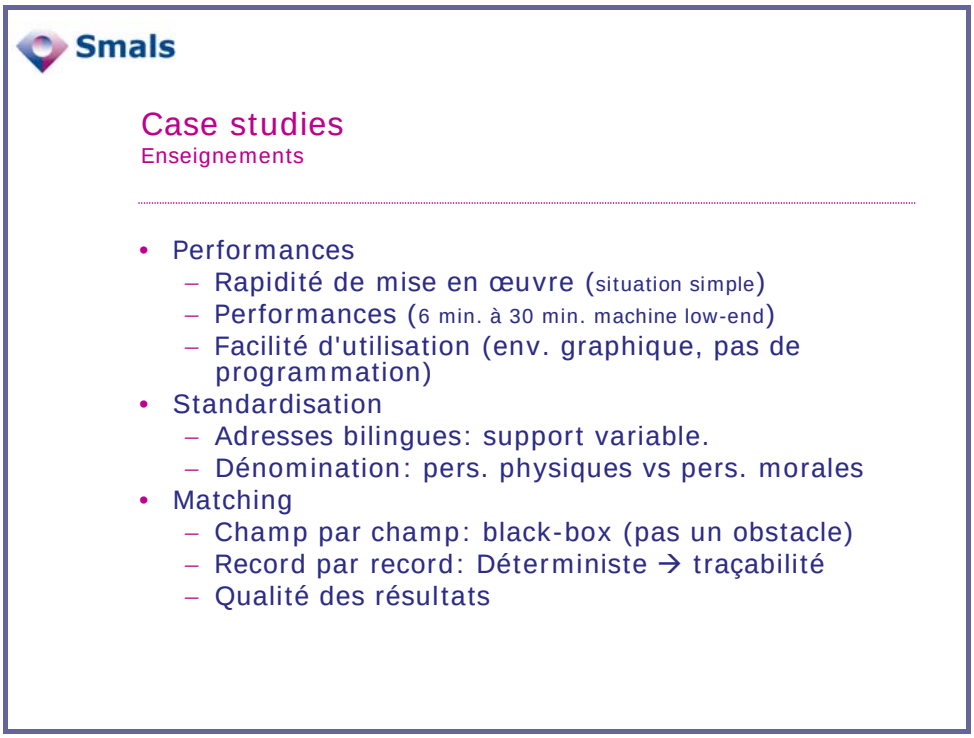
L'ensemble fait penser aux spécifications de processus des outils ETL (Extract-Transform-Load).



Démonstration du processus de matching, mis au point durant le workshop

On note une spécification des processus similaire, basée sur des sous-modules successifs accomplissant des tâches diverses. Un utilisateur expérimenté peut adapter les paramètres de certains modules à ses besoins.

Dans cette variante également, la similitude avec les outils ETL (Extract-Transform-Load) est stupéfiante.



Enseignements

Le case study, mené ici dans le cadre d'une *consultation informelle du marché* en collaboration avec quatre fournisseurs de solutions Data Quality, permet de tirer des enseignements sur quelques points d'attention capitaux. Ces points d'attention doivent certainement être pris en considération pour se prononcer sur le déploiement de solutions Data Quality commerciales et choisir judicieusement le fournisseur.

- **Performance**

  Les workshops ont démontré que pour cette problématique, certes relativement simple, la mise en œuvre du logiciel - en ce compris la spécification des processus et l'adaptation des paramètres à la problématique concrète - s'est achevée en un temps record (deux à trois heures).

  On estime que pour un projet « grandeur nature », suite à des discussions avec des analystes et informaticiens expérimentés, qu'un développement de ce type prendrait cinq fois plus de temps (avec des fonctionnalités moindres) dans le cadre d'un développement *in house*.

  Une fois les paramètres définis, l'opération de détection des doublons a duré de 6 à 30 minutes sur une machine *low-end* (ordinateur portable), ce qui suggère tout de même des différences de performance entre les produits des différents fournisseurs.

  La convivialité laisse une bonne impression : l'environnement graphique s'est montré intuitif et aucune programmation n'a été nécessaire pour accomplir les tâches.

- **Standardisation**

  En ce qui concerne le traitement du problème (belge) des adresses bilingues (autour de Bruxelles), le support offert n'était pas toujours identique. Ceci dit, la majorité des fournisseurs disposent de *knowledge bases* régionales spécialisées (voir point 2.2).

La standardisation pour le champ « Dénomination » devrait idéalement être réglée différemment pour les personnes physiques et les personnes morales. Ceci n'a pas été possible dans le timing des workhops. Pour cette raison entre autres, nous pouvons dire que la qualité des résultats atteinte lors des workshops n'est certainement pas la qualité maximale que peuvent offrir les Data Quality Tools.

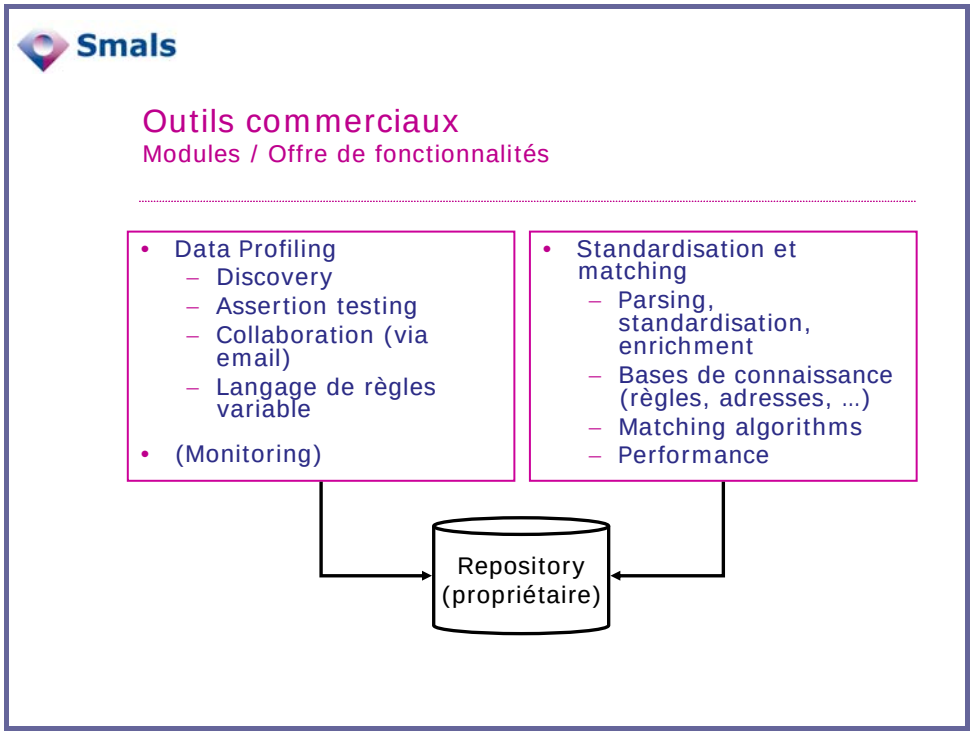
- **Matching**

  Les comparaisons champ par champ sont souvent cachées pour l'utilisateur (*black box)* : ce dernier ignore comment, avec quel algorithme, ces comparaisons sont opérées. Cela ne pose toutefois pas de problème pour la traçabilité des résultats du matching.

  Les comparaisons record par record n'ont pu être obtenues dans tous les cas avec l'approche déterministe, de sorte que la traçabilité se voit compromise : un logiciel qui implémente uniquement des approches probabilistes ne peut indiquer exactement pour quelle raison il a ou n'a pas décidé de juger une paire de *records* comme *match*.

  La qualité des résultats n'était pas homogène. Dans trois des quatre cas, les résultats semblaient bons à très bons (le degré de qualité était proportionnel au temps disponible pour les itérations ou ajustements de processus et paramètres).

  Force est de constater que les Data Quality Tools ont détecté bien plus de doublons que ceux connus jusqu'alors. La majeure partie des doublons connus a donc été découverte (plus de 90%). Un certain nombre (moins de 500 sur un total de 116 doublons réintroduits dans un ensemble de 239.651 records) de doublons sont apparus comme étant des *false positives* en raison d'un problème de longueur maximale de champ de la source de données utilisée (dénominations dont une grande partie des premiers caractères était identique) : par conséquent, les informations discriminantes ont été exclues de la partie analysée.



Logiciels commerciaux de Data Quality - Modules et fonctionnalités

Globalement, les fonctionnalités offertes par les logiciels de Data Quality peuvent être subdivisées en divers ensembles ou modules logiques. Du point de vue de l'utilisateur final, on trouve ainsi les fonctions suivantes :

1. **Data Profiling** :

   Comme expliqué au § 2.1, on trouve ici les fonctionnalités et programmes nécessaires pour :

   - **Discovery** : à partir des données réellement présentes, découverte bottom-up *de faits sur les données* (à la fois attendus et inattendus).
   - **Assertion testing** : il est ici vérifié si tout ce que l'on *admet* comme vrai au sujet des données (*rules*, relations et limites « connues » - soit les *faits supposés*) est effectivement corroboré par toutes les données observées. En résultat, soit on doit mettre à jour les méta-données, soit on a découvert les records dont les données sont en infraction.
   - **Collaboration** entre plusieurs utilisateurs ayant un rôle distinct, par exemple la gestion du workflow nécessaire est supportée par e-mail.
   - **Business rules** : il existe divers « rules languages », lesquels ont un potentiel distinct pour pouvoir exprimer ou non des règles business existantes : le plus fort, le mieux.

2. **Standardisation et Matching** :

   Eléments importants :

   - **Standardisation** (point 2.2) pour l'uniformité de représentation et l'élimination de l'impact de divers modes de représentation sur les résultats du Data Matching.
   - **Knowledge bases** (point 2.2): externes ou non, avec une profusion de règles et de données de référence pour le parsing, la standardisation et l'enrichment.

- **Matching** (point 2.3): plusieurs algorithmes pour le matching champ par champ (scores de similarité booléens et non booléens) et le matching de paires de records, dans une approche probabiliste ou déterministe.
- **Performance** : Voir p.45, point 2.3.

3. **Monitoring** :

   Voir point 2.4. Cette fonctionnalité n'a pas le même niveau de développement dans tous les logiciels ou n'est pas (encore) présent sous forme de module distinct. En effet, il est possible de réaliser un noyau d'activités de *monitoring* sur la base d'activités de *profiling* régulièrement réitérées, le cas échéant de façon moins automatisée.

Un **répertoire central** stocke des extraits de banques de données, des profils d'analyse, des résultats, des templates de projets, des infractions aux règles business, des indications d'anomalies et de records qui doivent être corrigés ou validés. Ceci pour soutenir et centralement gérer les activités susmentionnées, et guider la collaboration nécessaire pour le Data Quality.

Bien sûr, il y a d'autres outils, que l'utilisateur ne voit pas : 1) les modules nécessaires qui établissent la connexion entre les divers modules (susmentionnés et externes), les DBMS (Database Management Systems) et les knowledge bases, et 2) les API pour l'intégration avec des applications (en ligne).

Logiciels commerciaux de Data Quality - Prix

A première vue, les prix des licences des logiciels de Data Quality sont relativement similaires. Vous en trouvez ci-dessous l'**ordre de grandeur** :

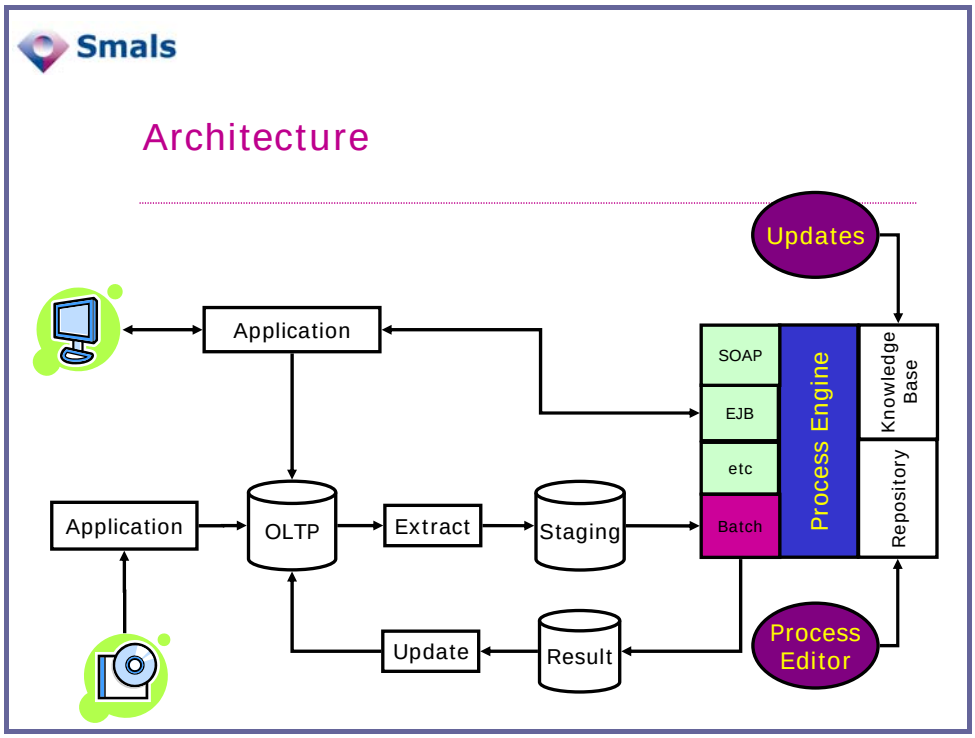
- **Profiling** :
  - +/- 25.000 € pour le premier *named user* ;
  - +/- 10.000 € par *named user* supplémentaire ;
  - normalement, seul un nombre limité de ces licences est nécessaire par entreprise.
- **Standardisation & Matching** :
  - +/- 100.000 € selon le système qui accueille l'installation (+/- 115.000 pour un système Unix, +/- 95.000 pour un système Windows/Linux) ;
  - un nombre variable de clients (+/- 10) est inclus dans le prix ;

  Certains fournisseurs ont des prix de licences moins chères, mais ajoutent +/- 30.000 € pour le module nécessaire pour une application en ligne ; d'autres fournisseurs donnent leur prix en y incluant les modes d'exécution batch et online ; pour une application « pur batch » on peut déduire +/- 25% des prix (licences).

  Les prix indiqués pour le Profiling, Standardisation & Matching sont des FYF (First Year Fees) ; il faut compter +/- 20% du prix total des licences pour le RF (Returning Fee), payable annuellement pour la maintenance et le support.
- **Knowledge base update**
  - +/- 8.000 € par an pour les bases de connaissance nécessaires pour le standardisation des adresses internationaux ;

Si l'on ajoute aux informations ci-dessus un coût de +/- 12.000 € pour l'achat du hardware, et un coût de +/- 10.000 € par an pour les mises-à-jour, l'infrastructure et l'organisationel, on arrive à un coût total d'environ 235.000 € pour la première année, d'environ 286.000 € après deux ans, et d'environ 336.800 € après trois ans.

Selon une étude coûts-bénéfices menée en 2007, à partir de plusieurs applications de l'ONSS, on estime que le « break even » (« Retour sur Investissement ») se produit déjà après un an.



Logiciels commerciaux de Data Quality - Architecture typique

La figure ci-dessus présente l'architecture et le workflow typiques d'application lorsque l'on relie un système de Data Quality (noyau : Process Engine) à une banque de données dans un environnement de production (OLTP), tant dans une opération **batch** que dans le cadre d'une application **en ligne**.

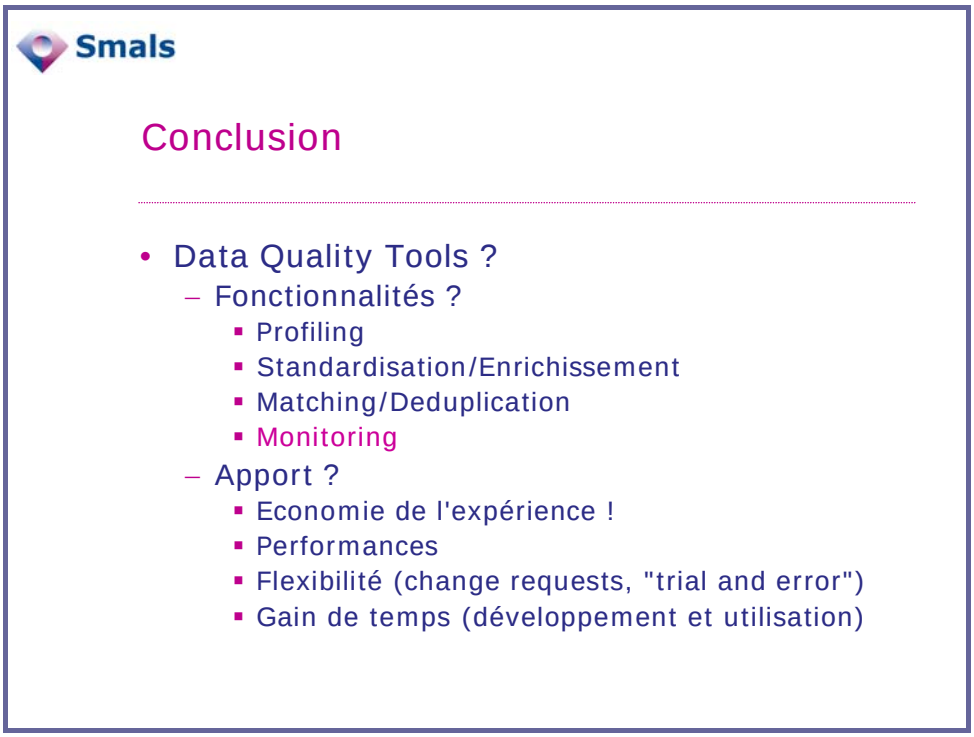
Etant donné que pour des raisons de performance et de sécurité, on ne peut travailler directement dans la banque de données en production, des procédures rigoureuses d'extraction et de mise à jour doivent être élaborées.

Une application en ligne peut demander des données à la banque de données en production, mais dans le souci de surveiller la qualité des données, il est préférable d'invoquer le système de Data Quality pour chaque mise à jour ou création d'un nouveau record, par exemple pour s'assurer de ne pas introduire de doublons dans la banque de données. L'application en ligne est reliée au Data Quality Server Process Engine par des connecteurs (par exemple SOAP, EJB, …).

Le Process Engine sollicite la *knowledge base* pour des informations régionales, une standardisation de noms et d'adresses, etc. Ces knowledge bases doivent être tenues à jour sur une base régulière.

Un Data Quality Repository central conserve les résultats, l'historique, les spécifications de processus, les configurations, les profils d'analyse Data Profiling, … et les met à la disposition d'experts pour consultation et modification via des Process Editor Clients.

4. Conclusion



L'étude Data Quality Tools nous éclaire sur les ***fonctionnalités*** offertes par les logiciels commerciaux. Globalement, celles-ci peuvent être regroupées autour de quatre concepts majeurs, qui forment ensemble un cycle d'amélioration continue de la qualité des données : **Profiling**, **Standardisation** & Enrichment, **Matching** (détection des doublons et des incohérences) et **Monitoring**.

La ***plus-value*** générée par les Data Quality Tools se manifeste essentiellement à deux niveaux :

- **Performance** :

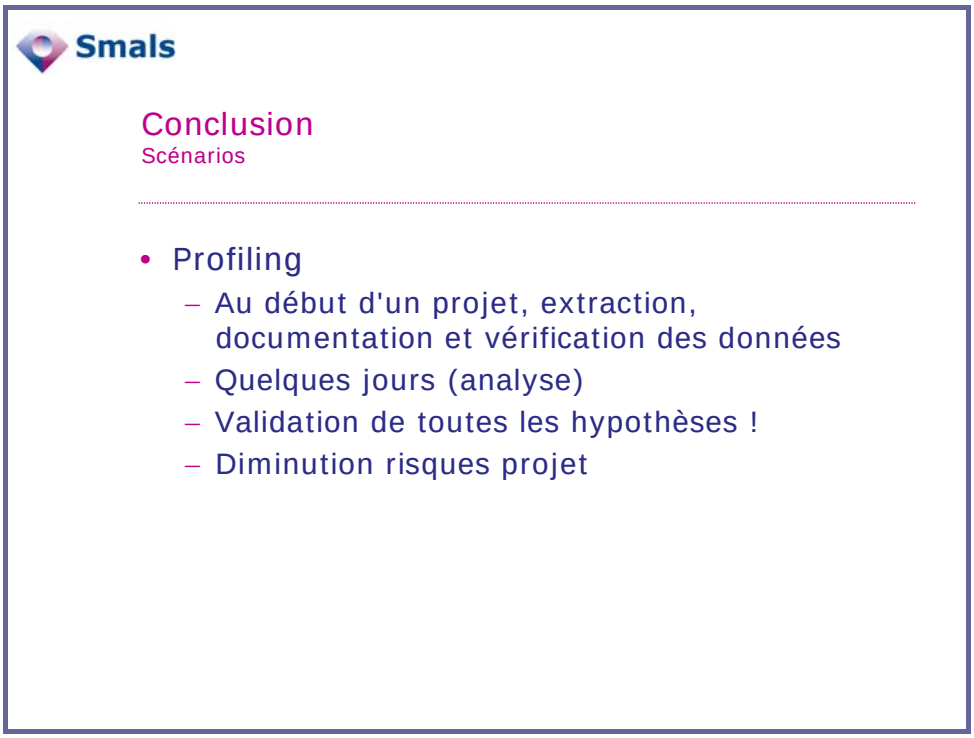
  Les algorithmes intégrés visent toujours à pouvoir traiter des grandes banques de données modernes - moyennant un hardware *high-end* - avec la plus haute performance possible. Les implémentations des algorithmes state-of-the-art contiennent de nombreuses optimisations que l'on ne peut développer soi-même, à moins de disposer de connaissances mathématiques (numériques) approfondies et d'un grand nombre d'années/homme pour le développement.

  On estime, suite à des discussions avec des analystes et informaticiens expérimentés, que, pour un projet « grandeur nature, un développement de ce type prendrait cinq fois plus de temps (avec des fonctionnalités moindres) dans le cadre d'un développement *in house*.

- **User Experience** :

  Plutôt que de devoir péniblement, manuellement résoudre les problèmes de qualité détectés de manière fortuite, les utilisateurs de Data Quality Tools

peuvent s'acquitter de leurs tâches de manière systématique et plus complète grâce à une détection automatisée, une interface utilisateurs intuitive riche en fonctionnalités et un support workflow.

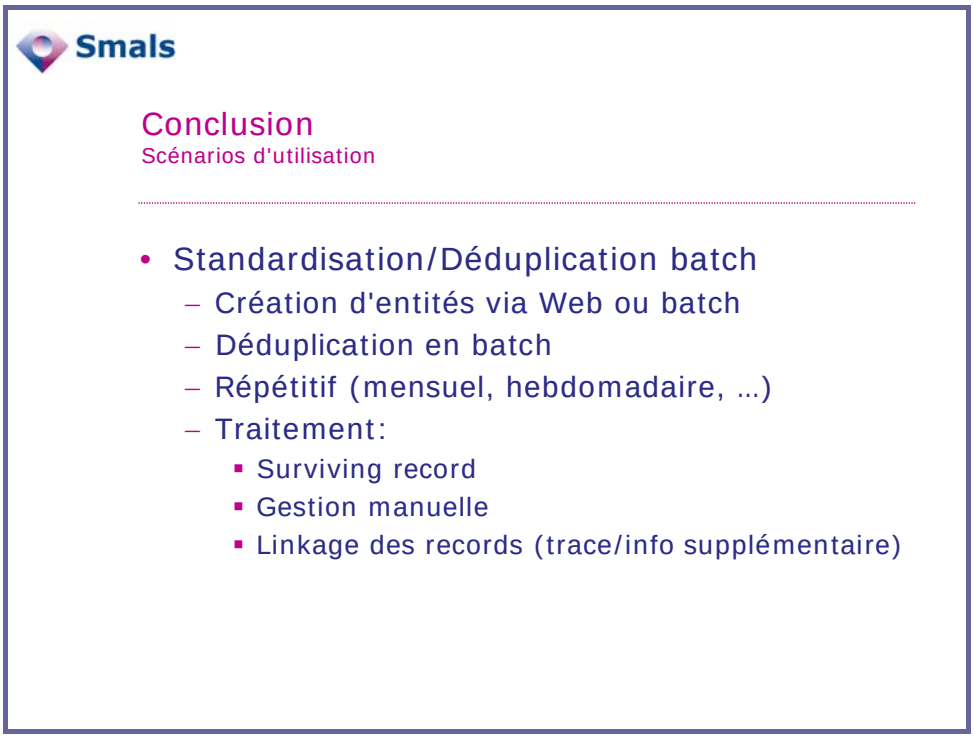


Plus-value des Data Quality Tools : Profiling

Au début de tout projet lié à une application de banque de données, il est utile de procéder à un Data Profiling : on extrait les données et la documentation, formalise les méta-données, vérifie les données par rapport aux méta-données et *règles business*, et on est à même de valider toutes les « hypothèses » formelles existantes, à l'aide des données présentes. Une telle analyse peut être clôturée en l'espace de quelques jours - en fonction du nombre de champs traitées et de leur complexité, à condition de disposer de Data Quality Tools. Il n'est pas impossible d'opérer un Data Profiling sans Data Quality Tools, mais si l'on ne possède pas un logiciel spécialisé, on ne pourra pas réaliser l'analyse de manière aussi complète et approfondie et on devra développer ses propres programmes, ce qui peut prendre du temps.

Ainsi, non seulement on trouve une liste des records problématiques, mais en plus on voit où se situent les problèmes (quelles colonnes) et pour quelle raison (grâce à des schémas types indiquant des problèmes systématiques). Une partie des problèmes peut être corrigée dans l'immédiat, tandis qu'une autre peut contribuer à la définition du scope d'un projet Data Quality.

Si l'on a réalisé un Profiling préparatoire, on peut espérer des risques nettement réduits dans le projet.



Plus-value des Data Quality Tools : Standardisation & Matching en batch

Des applications web ou batch apportent continuellement des nouvelles entités (records) et mises à jour. Par expérience, on sait que des doublons s'introduiront inexorablement par la même occasion. A intervalles réguliers (donc en batch, mensuellement, hebdomadairement, ...), on peut entreprendre des actions de **déduplication**. Il s'agit là d'une tâche laborieuse et répétitive, si elle n'est pas automatisée, comprenant d'une part :

- **la détection** :

  Les algorithmes de Data Matching incorporés dans les solutions Data Quality sont spécialement axés sur la détection des doublons dans une banque de données. Ils ont été mis au point pour autoriser la performance la plus élevée possible dans les banques de données de grande échelle, grâce à de nombreuses années/homme d'expérience et des connaissances mathématiques (numériques). Sans l'approche systématique des Data Quality Tools, la détection des doublons se limite bien souvent à une détection fortuite, *ad hoc* lorsque des problèmes se produisent dans certains dossiers, à moins qu'un propre logiciel n'ait été développé pour la détection des doublons. Il est toutefois difficile d'atteindre le même niveau de qualité et d'exhaustivité de détection, même si l'on dispose d'un grand nombre d'années/homme de développement et des connaissances nécessaires.

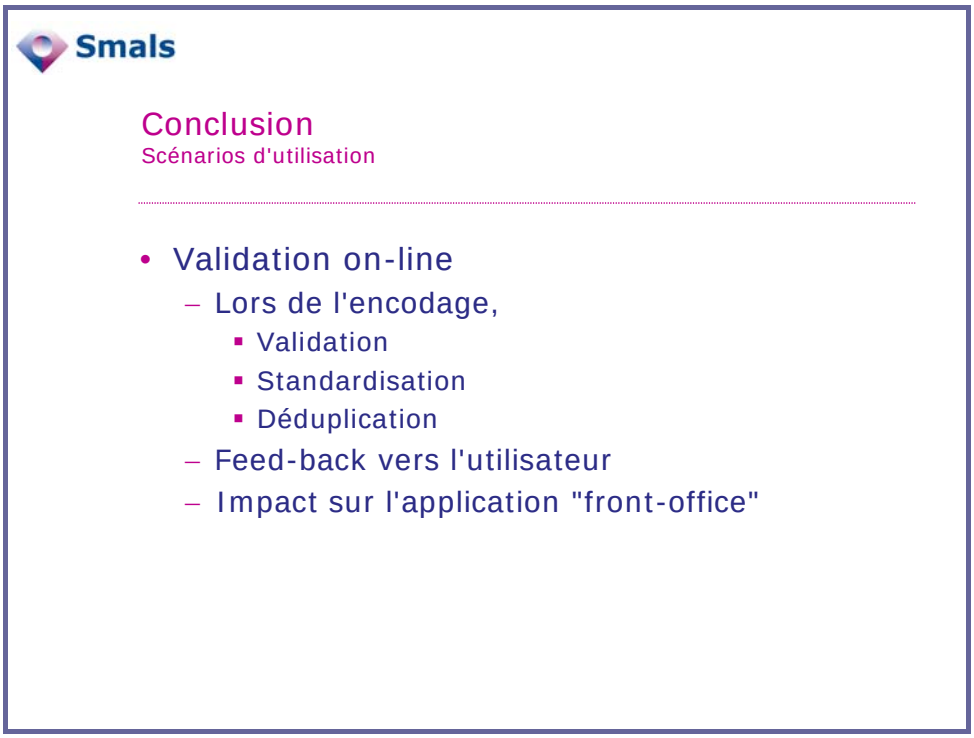
et d'autre part :

- **le traitement** :

  Si les doublons sont localisés, ceux-ci doivent encore être traités. Il faut déterminer quel sera le « *surviving record* » (est-ce un record existant ou l'information la plus exacte est-elle répartie entre les colonnes de plusieurs records ?). Cette mission requiert souvent une intervention manuelle : un expert du domaine doit se prononcer. En outre, la correction d'un record ou l'élimination de records redondants peuvent avoir des répercussions sur d'autres records (par exemple via des références ou des *foreign keys*, …).

Une automatisation est ici difficile. Les Data Quality Tools peuvent cependant soutenir le workflow et la collaboration nécessaires à la résolution de ces problèmes de qualité.

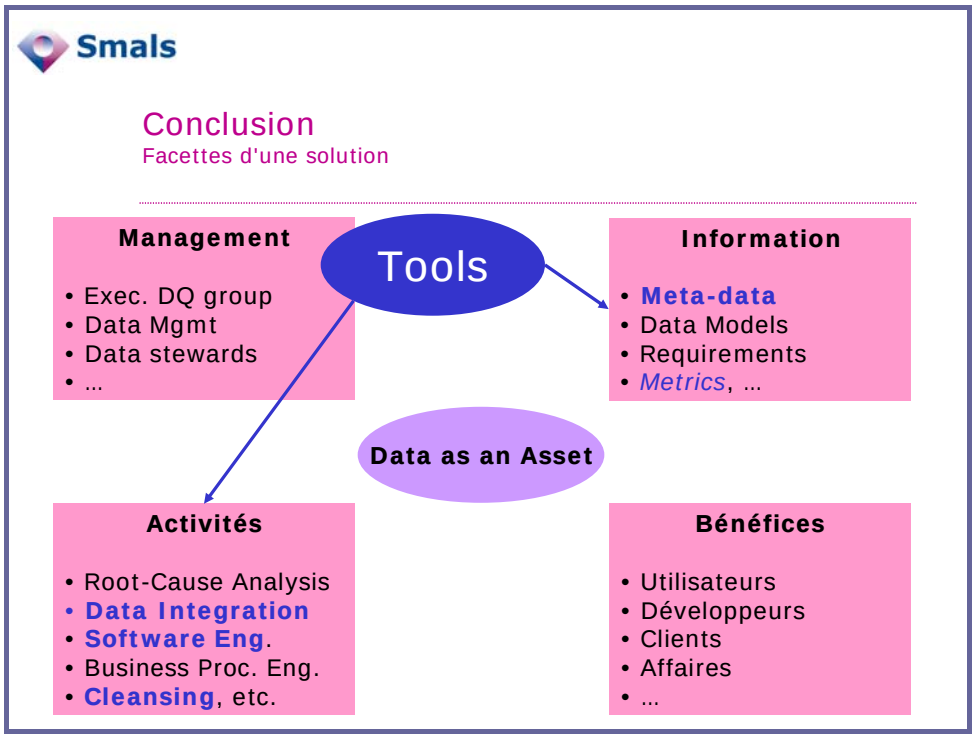
Ce qui vaut pour la détection des doublons dans une banque de données vaut également pour la détection des incohérences entre banques de données. La **standardisation** est une manipulation préalable de records qui peut largement être automatisée via des Data Quality Tools à l'aide des *knowledge bases.* Voir point 2.2.



Plus-value des Data Quality Tools : Standardisation & Matching *en ligne*

L'alternative aux opérations batch régulières consiste à invoquer les fonctionnalités Data Quality depuis l'application *en ligne* responsable de l'introduction de nouveaux records ou mises à jour. On peut alors, pendant l'encodage, standardiser et valider les nouvelles données et déterminer si elles sont susceptibles d'être redondantes (détection des doublons) ou incohérentes. L'application peut ainsi immédiatement procurer un *feed-back* à l'encodeur, lequel se prononce et rectifie son encodage si nécessaire. L'introduction de doublons peut ainsi être évitée et/ou les nouveaux records peuvent être marqués d'un drapeau pour vérification.

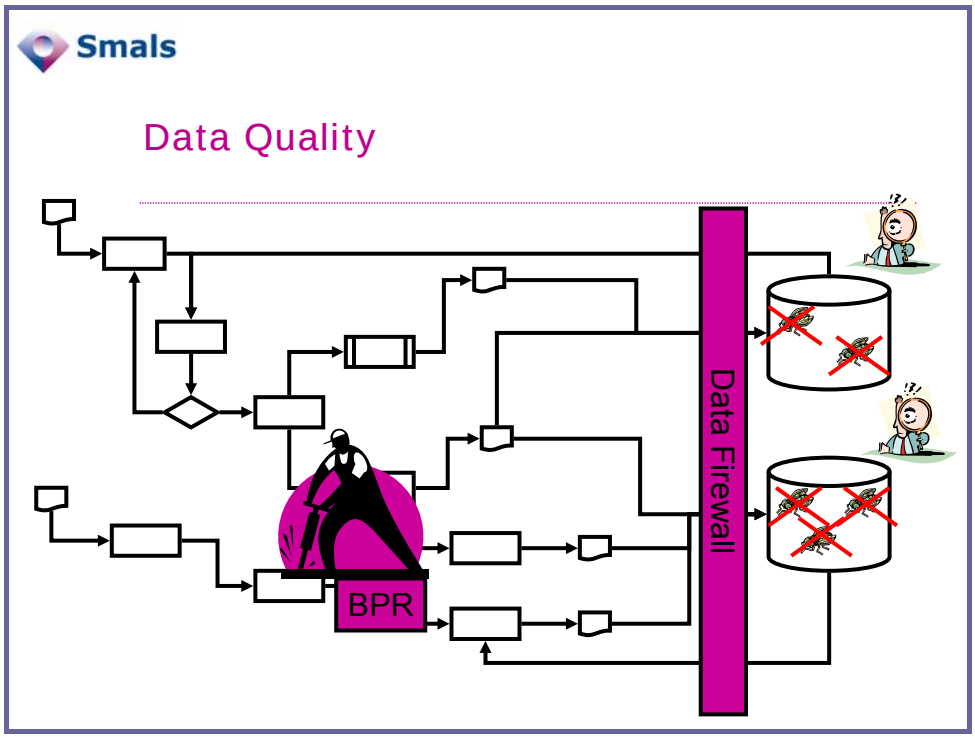
La bonne implémentation de cette possibilité requiert un effort supplémentaire durant le développement du « front-office » de l'application, mais augmente l'utilisabilité et la qualité de la (des) banque(s) de données sous-jacente(s). Nous en recommandons l'utilisation lorsque les données traitées sont stratégiques pour l'administration et que leur qualité soulève des enjeux sociaux et en termes de coûts-bénéfices importants.



Si dans la pratique, on veut mettre en place un cycle de Data Quality dans un environnement d'applications, il ne s'agit bien sûr pas de se limiter à installer et à utiliser des Data Quality Tools.

Il va de soi que lorsque l'on considère les données comme une ressource cruciale de l'entreprise et la garantie de la qualité comme stratégique, on ne pourra se passer d'une **approche systématique** où un nombre maximal de tâches sont **automatisées**, ceci en raison de l'ampleur des banques de données actuelles. Les Data Quality Tools semblent offrir une réponse complète, sinon très large. Un tel projet ne peut aboutir que si les conditions suivantes sont réunies:

- le top management procure un support suffisant ;
- il est désigné un Executive DQ Group suffisamment puissant (budget et main d'oeuvre) ;
- quelqu'un assume la responsabilité de la qualité des données dans chaque application (« data stewards ») ;
- une politique de data management est conduite à l'échelle de l'entreprise.

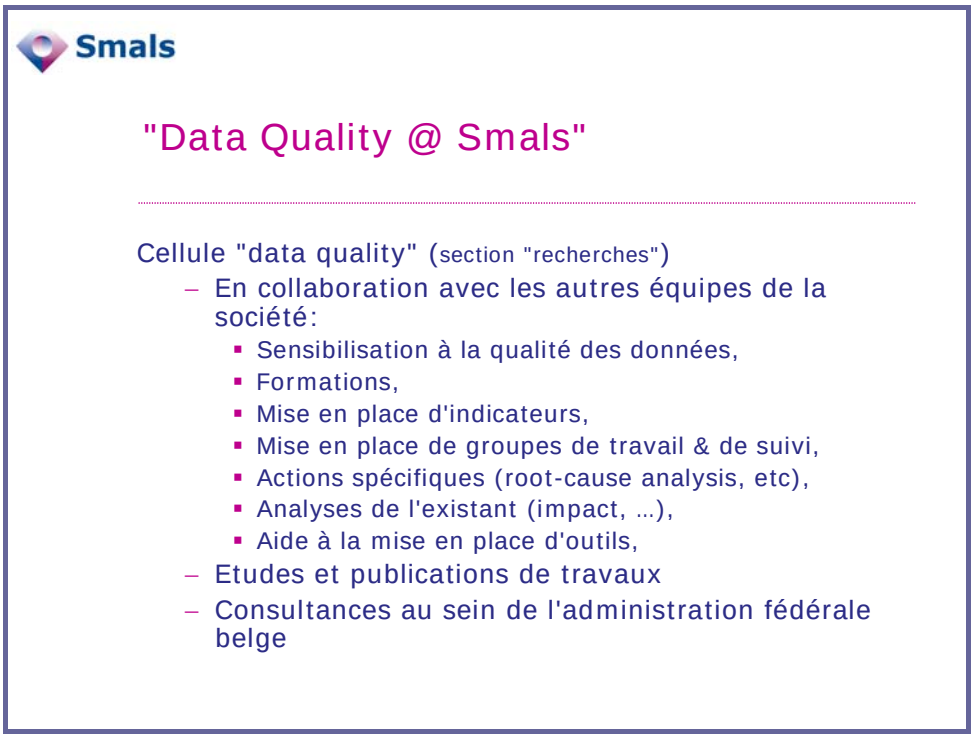


Data Quality Tools - Limites

Attention : si les Data Quality Tools permettent de détecter et corriger une grande partie des données inexactes dans un environnement, ils ne nous donnent pas la clé pour éradiquer la cause des erreurs.

Les Data Quality Tools peuvent être mis en place online en guise de Data Firewall, afin d'éviter dans une large mesure l'introduction de nouveaux doublons, incohérences et anomalies. L'origine des données inexactes, quant à elle, doit être cherchée dans les processus et le workflow qui alimentent les banques de données. Pour ce faire, il convient de se tourner, entre autres, vers le Business Process Reeningeering. Ce point a été analysé dans la première partie méthodologique de cette thématique ("data quality : best practices")[9]. Dans certains cas, une analyse avec des Data Quality Tools peut aider à déterminer les canaux et/ou processus responsables des erreurs récurrentes (exemple : Data Profiling d'*input streams*, comme les déclarations et modifications DIMONA et LIMOSA).

[9] Boydens I., *Data Quality : Best Pratices*. Deliverable 2006/trim2/01. Bruxelles : Smals, 2006.



Un centre de compétences en Data Quality a été créé chez Smals. Il offre de la consultance interne et externe en matière de gestion des flux d'information, de « best practices » et de Data Quality Tools. Cette consultance est étayée par de nombreuses études, publications nationales et internationales et travaux en la matière. Des formations et un coaching en Data Quality peuvent également être dispensés.

5. Références

Boydens I., Informatique, normes et temps. Bruxelles : Bruylant, 1999.

Boydens I., Qualité de l'information et administration électronique : enjeux et perspectives. In ASSAR S. et BOUGHAZALA I., éds., Administration électronique. Constats et perspectives. Paris : Lavoisier - Hermès Sciences, 2007, p. 103-120 (chapitre 5).

Boydens I., *Data Quality : Best Pratices. Deliverable 2006/trim2/01*. Bruxelles : Smals, 2006.

Elmasril R. et Navathe S. B., Fundamentals of Database Systems. New York : Pearson Addison Wesley, 2007 (5ème édition).

Friedman T., Bitterer A., *Magic Quadrant for Data Quality Toos, 2007*. Gartner Research Note, 29 juin 2007, n°G00149359.

Friedman T., *Gartner Study on Data Quality Shows That IT Still Bears the Burden*. Gartner Research Note, 23 février 2006, n°G00137680.

Friedman T., Bitterer A. et Hostmann B., *Focus on Data Quality in BI Motivates Business Objects Buy*. Gartner Research Note, 14 février 2006, n°G00137867.

Friedman T., *Strategic Focus on Data quality Yields Big Benefits ofr BT*. Gartner Research Note, 24 mars 2006, n°G00138085.

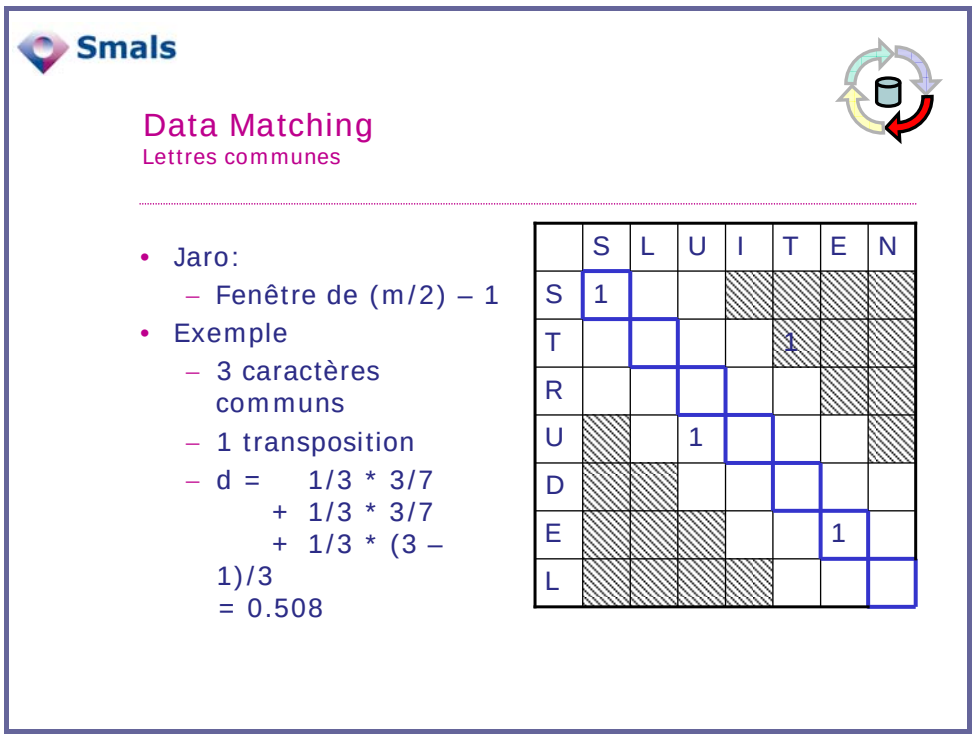
Friedman T., *Key Issues for Data Quality, 2007*. Gartner Research Note, 22 mars 2007, n°G00147383.

Olson J., Data Quality : the Accuracy Dimension. Elsevier : The Morgan-Kaufmann Series in Database Management, 2002.

Redman T., *Data Quality for the Information Age*. Boston, Artech House, 1996.

Redman T. , *Data Quality. The Field Guide*. Boston : Digital Press, 2001.

Annexe : Jaro Distance



Méthodes de similarité word-based : « Jaro Distance »

La « Jaro Distance » entre deux mots *s* et *t* se définit comme suit :

$$Jaro(s,t) = \frac{1}{3} \cdot \left(\frac{|s'|}{|s|} + \frac{|t'|}{|t|} + \frac{|s'| - T_{s',t'}}{2|s'|} \right)$$

où

|*s'*| représente le nombre de caractères que *s* a en commun avec *t* ;

|*t'*| représente le nombre de caractères que *t* a en commun avec *s* ;

|*s*|, |*t*| représentent respectivement la longueur en caractères des mots *s* et *t* ;

$T_{s',t'}$ représente le nombre de transpositions de caractères de *s'* par rapport à *t'*.

Attention : les nombres ne sont considérés que dans une « window » d'une longueur |s|/2 - 1 (voir illustration).

Il existe de nombreuses variantes et alternatives[10].

[10] http://www.dcs.shef.ac.uk/~sam/stringmetrics.html