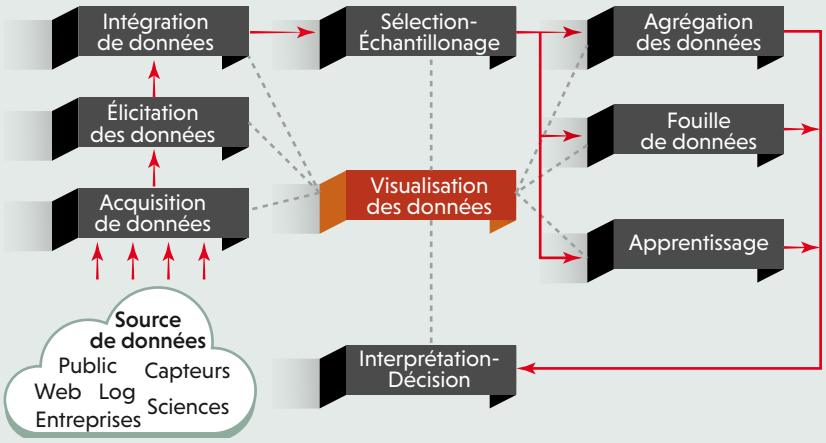
## LES MILLE ET UNE VIES DES DONNÉES

Le traitement des *big data* implique des activités que l'on retrouve quelle que soit l'application. Passons-les en revue. Le stockage pose la question du choix des données brutes à enregistrer, de leur acquisition et des données à résumer, ainsi que les méthodes d'indexation pour accélérer leur recherche ultérieure. La validation des données, c'est-à-dire leur nettoyage et leur annotation (on parle d'élicitation), nécessite souvent une interaction avec un expert, ce qui est hors de portée d'un humain si le volume de données est considérable. L'intégration de données est une phase très complexe car elle concerne des sources de données distribuées à grande échelle où l'hétérogénéité est très élevée. Le modèle d'intégration dépendra des objectifs assignés au système : la recherche d'information, la prise de décision, l'extraction de connaissances… C'est là que se situent les principaux verrous liés aux performances des systèmes et à la pertinence des données produites. La sélection et l'échantillonnage constituent des vues sur les données intégrées, c'est-à-dire une façon de réduire les données aux sous-ensembles nécessaires à un type de requête ou un type d'analyse. Un mauvais choix de ce sous-ensemble hypothèque la qualité de la tâche ultérieure. Quant à l'analyse elle-même, elle peut être réalisée par différentes méthodes : l'agrégation pour générer des indicateurs statistiques (somme, moyenne, tendance…), la fouille de données pour identifier des règles d'association ou des motifs fréquents, l'apprentissage automatique (profond ou par renforcement) pour définir un modèle de prédiction permettant de décider ultérieurement si un nouvel objet peut être classé de façon pertinente. Enfin, la visualisation des données est utile à toutes les étapes. Elle offre souvent une vue synthétique des grandes données et aide à leur compréhension.




Le cycle de vie des données au terme duquel elles peuvent livrer des informations.

les tables ; la recherche d'éléments ou le parcours de chemin sur un arbre ; l'appariement ou l'extraction de sous-graphes ; une copie instantanée d'une séquence. Ces structures et ces opérations ont été étudiées depuis des décennies pour formaliser leurs sémantiques, expliciter leurs propriétés et trouver les meilleurs programmes qui leur correspondent dans différents environnements.

Plusieurs facteurs interviennent dans leur mise en œuvre : la hiérarchie et la taille des mémoires (mémoire principale, mémoire cache, disque…), la localisation des données (un site centralisé, de multiples serveurs…), les ressources de calcul mobilisables (une ou plusieurs machines…) et les caractéristiques techniques de ces ressources (puissance de calcul, partage de mémoire…).

### DES TABLEAUX MALLÉABLES

L'algèbre relationnelle a représenté un saut technologique et qualitatif dans la gestion des données. En quoi consiste-t-elle ? Rappelons d'abord qu'une table relationnelle est décrite par un ensemble de variables (ou attributs) désignant les colonnes et un ensemble de valeurs à ces variables organisées en lignes, chacune représentant un objet du réel. En d'autres termes, il s'agit de tableaux à deux dimensions, on parle aussi de tableau à double entrée. Or avec cinq opérateurs de base (*voir la figure page suivante*), on sait aujourd'hui transformer ces tableaux et en dériver d'autres.

C'est d'une simplicité confondante, et pourtant, la combinaison de ces opérations permet d'exprimer une large palette de requêtes, sans programmer une seule ligne de code, puisque le code correspondant à chacune des opérations est défini une fois pour toutes et pour toutes les applications. Cette façon de représenter le réel par un ensemble de tableaux indépendants et d'exprimer une requête sur ces tableaux au moyen d'une expression algébrique a notablement modifié notre rapport à la programmation.

Parmi ces opérateurs, certains sont plus sensibles que d'autres au volume des données. Ainsi la jointure de deux tables comportant des milliards de lignes pose de réels problèmes de calcul pour comparer ces milliards de lignes. De nombreux travaux de recherche proposent des algorithmes de jointure tenant compte ou non du tri des données, de la taille de la mémoire centrale, de l'indexation des données, de la distribution des données sur différents disques, du parallélisme des machines et du débit sur les réseaux.

Ce coût dépend aussi du choix initial de la représentation d'une table selon ses lignes ou ses colonnes. Par exemple, pour les requêtes >