

➤ suivants qui feront le *reduce*. Les opérations *map* et *reduce* sont dépendantes de l'application et donc programmées de façon appropriée, alors que le *shuffle* est une opération générique. Ce modèle de calcul permet un gain de temps significatif grâce à sa capacité de mobilisation de ressources à chacune de ses phases d'exécution, mais le prix à payer est au niveau du développement qui nécessite une compétence pointue et au niveau de la mise en œuvre qui dépend d'un grand nombre de paramètres techniques.

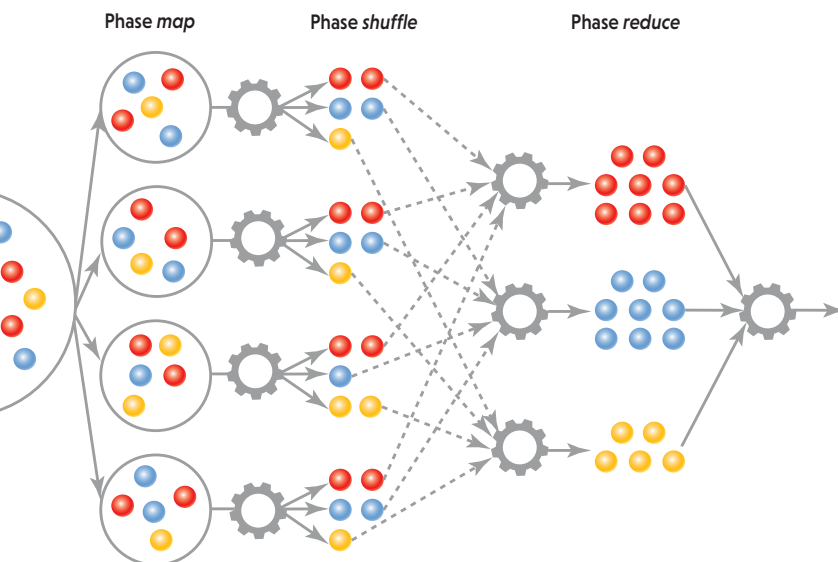
Ce modèle est la pierre angulaire de nombreux systèmes de traitement massif des données. Hadoop est l'un des représentants de ces systèmes, combiné avec un système de fichier massivement distribué, de nombreuses extensions ont été réalisées soit au cœur du système pour améliorer ses performances soit au-dessus du système (on parle de surcouches) pour offrir des interfaces de développement plus riches et plus simples.

AU-DELÀ DU CALCUL

Si les opérations de calcul sur les données sont importantes et cristallisent à juste titre l'attention des spécialistes et des programmeurs, la gestion des données ne se limite pas à ces opérations. L'indexation des données et les techniques de hachage, c'est-à-dire la répartition des données sur divers disques, sont cruciales pour la gestion des données, quel que soit le modèle de calcul choisi. Indexer des tables comportant des milliards de lignes ou des documents de plusieurs centaines de millions de pages est une tâche coûteuse. Les techniques mises en œuvre rivalisent d'ingéniosité à la fois dans la façon de construire l'index pour accélérer les accès aux données et dans les performances de cette construction.

Outre l'indexation et le hachage, la répartition des données est aussi un bon moyen d'augmenter la disponibilité de l'information en multipliant les exemplaires et les points d'accès. La répartition est très importante dans un système distribué afin de réduire les transferts de données, souvent coûteux, sur le réseau. Cependant, les avantages de l'indexation et de la répartition sont vite anéantis quand les données sont fréquemment mises à jour, ce qui n'est heureusement pas le cas dans de nombreuses applications des *big data*.

La protection des données reste un sujet sensible lorsqu'il s'agit de données personnelles, de santé ou concernant des processus industriels. De nombreux systèmes intégrés de gestion de données sont dotés de mécanismes d'anonymisation (voir *L'art de préserver l'anonymat*, par T. Allard, page 98), de cryptage, ou de restriction d'accès à certaines



données. Des hiérarchies de droits sont souvent associées à des hiérarchies d'utilisateurs contrôlant ainsi l'accès aux données sensibles. Les algorithmes associés à ces garde-fous sont souvent complexes.

Dans le registre des risques, ajoutons que la complexité des algorithmes mis en œuvre dans l'exploitation des *big data* peut entraîner des effets de bord qui, si l'on n'y prend pas garde, peuvent poser des problèmes d'éthique, violer la vie privée, réduire la liberté et avoir un impact fortement intrusif dans la vie des personnes. Ce sont des questions essentielles mises en débat récemment (voir l'entretien avec N. Boujemaa, page 104).

La sensibilité des données n'est pas seulement liée à leur usage, elle peut être liée à la technologie. Les erreurs de matériel ou logicielles ne sont pas rares. Ces incidents ne doivent pas mettre le système de données dans un état incohérent, avec des risques de perte d'informations. Des mécanismes de fiabilisation sont développés, notamment au niveau des systèmes de gestion de base de données pour protéger ces systèmes et permettre un redémarrage à chaud ou à froid, sans perte d'informations et sans incohérence sur les données globales. Ces algorithmes de protection des données à tous les niveaux occupent une place stratégique. Ils sont complexes et leur mise en œuvre n'est pas sans incidence sur les coûts de calcul.

En fin de compte, les algorithmes destinés à la gestion des *big data* constituent une vaste faune, toujours plus diversifiée et riche à mesure des développements. C'est que, pour filer la métaphore, leur monde, celui des *big data*, ne cesse de croître. D'ailleurs, la comparaison avec le pétrole trouve ici ses limites. Les ressources en cette énergie fossile sont inéluctablement appelées à se tarir, alors que celles en données sont en augmentation permanente. Et une donnée peut être utilisée autant de fois que nécessaire sans se consumer. ■

MapReduce accélère le processus de traitement des données grâce au parallélisme, c'est-à-dire à une distribution des tâches. Ici, elle consiste en une répartition des données pour les classer. D'abord, le paquet initial est divisé en quatre parties (à gauche), chacune étant ensuite classée (au centre). Les résultats partiels sont enfin réunis en un seul (à droite).

BIBLIOGRAPHIE

M. BOUZEGHOUB ET R. MOSSERI (DIR.), *Les Big Data à découvert*, CNRS Éditions, 2017.

S. ABITEBOUL, « Sciences des données : de la logique du premier ordre à la Toile », Leçon inaugurale de la Chaire d'Informatique et sciences numériques au Collège de France, 2012.