Engineer

Informatique décisionnelle

convertir les données d'entreprise en rapports pour décider

Ce que c’est

L’informatique décisionnelle, généralement abrégée BI d’après l’anglais business intelligence, consiste à transformer les données qu’une organisation consigne elle-même en rapports, tableaux de bord et analyses que les personnes chargées de décider peuvent utiliser pour agir. Lorsqu’elle est employée comme adjectif en anglais, l’expression prend souvent un trait d’union, comme dans business-intelligence software, logiciel d’informatique décisionnelle.

L’expression est plus ancienne que les logiciels. En octobre 1958, l’IBM Journal of Research and Development a publié un article de H. P. Luhn intitulé « A Business Intelligence System ». Luhn décrivait un système automatisé qui prendrait les documents reçus ou rédigés par une organisation, en produirait automatiquement un résumé, les décrirait par une configuration de leurs mots et les enverrait aux « points d’action » dont le profil d’intérêts correspondait. Un point d’action pouvait être une personne, un groupe, un service ou une division entière. Il donnait à business un sens délibérément large : toute activité organisée, des sciences et du commerce au droit, aux administrations publiques et à la défense. Par intelligence, il entendait une compréhension des relations entre les faits disponibles assez bonne pour orienter l’action vers un objectif. Après avoir énuméré les techniques, il nommait l’ensemble ainsi, dans une traduction française : « Nous appelons cela un système d’informatique décisionnelle ».

La matière de Luhn était le texte. Aujourd’hui, la BI travaille surtout sur des nombres dans des tables, et la présentation d’un système d’entrepôt de données et de BI par Kimball Group en montre l’organisation habituelle. Une arrière-boutique extrait les données des systèmes sources, les nettoie, les rend cohérentes et les charge. Une partie ouverte aux utilisateurs les présente, organisées par processus métier et conservées au niveau le plus détaillé, avec des synthèses agrégées ajoutées pour la rapidité. Au-dessus se trouvent les

applications BI : requêtes ponctuelles, rapports standard, tableaux de bord et tableaux de suivi des objectifs, puis applications plus lourdes d'analyse et de fouille de données.

Ce qui relie les deux sens est la question de Luhn : qui a besoin de savoir cela, et comment le lui faire parvenir sous une forme qu'il lira effectivement?

Utilisez le concept pour prendre une décision

Utilisez « Qu'est-ce que l'informatique décisionnelle ? » pour éclairer une décision réelle. Nous pouvons vous aider à relier ce concept au système que vous construisez ou achetez. Des tableaux de bord et des rapports présentant les quelques chiffres qui déterminent une décision, actualisés automatiquement.

Découvrir nos prestations Contactez-nous →

Vérifiez les hypothèses derrière le terme

Appuyez-vous sur l'explication et les réalisations liées pour formuler les questions de votre projet. Nous pouvons examiner les hypothèses avec vous avant le choix d'un outil. Une communication de marque éclairée par le public — une communication de marque éclairée par les réactions du public

Une communication de marque éclairée par le public Contactez-nous →

Pourquoi c'est important

Toute organisation fonctionne déjà avec des chiffres. La question est de savoir si ces chiffres concordent. Lorsque les ventes comptent une commande le jour de sa signature et la finance le jour de sa facturation, les deux services apportent deux chiffres d'affaires à la même réunion. Celle-ci se passe alors à discuter de l'écart plutôt qu'à prendre la décision. Un système BI justifie son coût en fixant une définition par mesure, en calculant cette mesure en un lieu unique et en montrant le même chiffre à quiconque le demande.

Le diagnostic de Luhn de 1958 convient encore à la plupart des dossiers de rapports partagés. Il voyait des organisations diffuser largement l'information, au cas où, et constatait que les destinataires se noyaient dans le papier, si bien qu'il devenait probable qu'aucune information n'atteigne réellement sa cible. Un tableau de bord avec quarante graphiques fait la même chose sur un écran. Son remède était la sélection. Chaque point d'action recevait d'abord une courte liste identifiant les documents retenus par leur numéro, leur nom et leur auteur. Il pouvait ensuite composer ces numéros pour obtenir les résumés et accepter les documents qu'il souhaitait en version intégrale. Chaque acceptation mettait son profil à jour : la sélection s'améliorait donc avec l'usage. Un bon rapport actuel fonctionne de la même manière : il part d'une personne et d'une décision, puis présente les quelques chiffres dont cette décision dépend.

Il existe aussi une dimension pratique. La base derrière une application est conçue pour ses nombreuses petites lectures et écritures. Une requête lisant une année de commandes entre en concurrence avec ces écritures si elle s'exécute au même endroit. C'est une raison pour laquelle les données BI résident généralement dans un stockage séparé, chargé selon un calendrier. Conserver ce stockage au niveau le plus détaillé, comme dans la présentation de Kimball, détermine quelles questions pourront encore recevoir une réponse l'année suivante sans retour aux sources.

À quoi cela ressemble en pratique

Une question BI typique : le chiffre d'affaires du troisième trimestre 2026, par région et produit, avec un sous-total par région et un total général. Avec une table `sales` contenant une ligne par vente et les colonnes `sold_on`, `region`, `product` et `amount`, le SQL standard y répond en une requête :

```
SELECT region,
       product,
       sum(amount)              AS revenue,
       GROUPING(region, product) AS level
FROM   sales
WHERE  sold_on >= DATE '2026-07-01'
AND    sold_on <  DATE '2026-10-01'
GROUP  BY ROLLUP (region, product)
ORDER  BY GROUPING(region), region, GROUPING(product), product;
```

Avec quelques lignes d'exemple, elle renvoie :

```
 region | product | revenue | level
--------+---------+---------+-------
 North  | Boots   |    1200 |     0
 North  | Jackets |     800 |     0
 North  |         |    2000 |     1
 South  | Boots   |     500 |     0
 South  | Jackets |    1500 |     0
 South  |         |    2000 |     1
        |         |    4000 |     3
```

`ROLLUP (region, product)` est une forme abrégée de `GROUPING SETS ((region, product), (region), ())` : la liste complète, chacun de ses préfixes plus courts et enfin l'ensemble vide, qui additionne toutes les lignes. La documentation

PostgreSQL le présente comme l'outil habituel pour les données hiérarchiques, par exemple les salaires par service, par division et pour toute l'entreprise. `CUBE (region, product)` regrouperait au contraire selon chaque sous-ensemble et ajouterait donc un total pour chaque produit, toutes régions confondues.

Dans l'ensemble de regroupement d'une ligne, une colonne qui ne participe pas au regroupement revient sous forme de NULL. C'est pourquoi les lignes de sous-total ont des cellules vides. `GROUPING(region, product)` indique de quelles colonnes il s'agit. Il renvoie un masque de bits avec un bit par argument, l'argument le plus à droite occupant le bit de poids faible. Un bit vaut 1 lorsque cette colonne n'a pas participé au regroupement pour la ligne. Ainsi, 0 marque une ligne de détail, 1 un sous-total régional (sans regroupement par produit) et 3 le total général (sans regroupement par l'une ou l'autre colonne). Un rapport lit `level` pour décider quelles lignes imprimer en gras.

La prestation d'analyse de données et de tableaux de bord BI de l'entreprise vise des tableaux de bord et des rapports qui montrent les quelques chiffres dont dépend réellement une décision, actualisés sans que personne ait à le demander. Une requête comme celle-ci constitue la couche sous-jacente à un tel rapport : les totaux sont calculés une fois, dans la base, et le tableau de bord se contente de les afficher.

Où les problèmes surviennent

Deux chiffres sous le même nom. Chiffre d'affaires, client actif et attrition ont chacun plusieurs définitions raisonnables, et tout rapport qui les calcule lui-même en choisit une. Consignez chaque mesure une fois, avec ce qu'elle compte, sa source et sa date de référence. Calculez-la dans la base, par exemple sous forme de vue, plutôt que dans les formules de chaque rapport. Kimball appelle la partie en langage courant les métadonnées métier : quelles données sont disponibles, d'où elles viennent et ce qu'elles signifient.

Tout est envoyé à tout le monde. Un tableau de bord comptant des dizaines de graphiques et une liste de diffusion de tout le service reproduit le déluge que Luhn voulait arrêter. Partez d'une personne désignée et de la décision qu'elle prend, montrez les chiffres nécessaires à cette décision et retirez les rapports que personne n'a ouverts depuis un trimestre.

Des rapports ralentissent l'application. Une requête lourde sur la base principale d'un système en service entre en concurrence avec les clients qui l'utilisent. Exécutez les rapports sur une réplique de lecture ou dans un entrepôt séparé chargé selon un calendrier. Les agrégats coûteux et souvent demandés peuvent résider dans une vue matérialisée actualisée après chaque chargement.

Des totaux construits à partir de totaux. La moyenne des montants moyens de commande de chaque région n'est pas le montant moyen des commandes de l'entreprise, sauf si chaque région compte le même nombre de commandes. On ne peut pas additionner les nombres mensuels de clients distincts pour obtenir celui d'un trimestre : un client ayant acheté en juillet et en août compterait deux fois. Conservez les lignes détaillées et calculez chaque niveau d'agrégation à partir d'elles. C'est exactement ce que font les ensembles de regroupement : chaque niveau de la requête ci-dessus est calculé à partir des lignes de ventes, pas du niveau inférieur.

Une cellule vide a deux sens. Ajoutez une vente dont la région est inconnue : la requête ci-dessus renvoie une ligne de détail avec une région vide, son sous-total avec les deux cellules vides et le total général avec lui aussi les deux cellules vides. Avec `CUBE`, c'est pire : deux lignes portent une région vide et Boots, l'une pour les bottes vendues dans une région inconnue, l'autre pour les bottes toutes régions confondues. La documentation PostgreSQL avertit qu'un NULL présent dans les données ne peut être distingué du NULL d'une colonne écartée du regroupement. Testez `GROUPING()`, jamais `IS NULL`, pour repérer les lignes de sous-total et donnez aux valeurs manquantes une étiquette visible comme « inconnu » au chargement des données.

Les chiffres d'hier sont pris pour ceux d'aujourd'hui. Lorsqu'un chargement nocturne échoue, le tableau de bord continue à montrer les derniers chiffres valides, sans indication de leur ancienneté à l'écran. Affichez l'heure du dernier chargement réussi sur chaque rapport et faites déclencher par la supervision une alerte destinée à une personne désignée lorsqu'un chargement planifié ne se termine pas.

Pour aller plus loin

- A Business Intelligence System — l'article de H. P. Luhn dans l'IBM Journal of Research and Development, en octobre 1958, d'où vient l'expression.

- Architecture technique d'un système DW/BI selon Kimball — la présentation par Kimball Group de l'arrière-boutique, de l'espace de présentation, des applications BI et des métadonnées.
- Expressions de table — la référence PostgreSQL pour `GROUPING SETS`, `ROLLUP` et `CUBE`, et l'emploi de NULL pour une colonne sans regroupement.
- Fonctions d'agrégation — la fonction `GROUPING` et son masque de bits, avec un exemple détaillé.

Glossaire

Choisissez selon vos contraintes réelles

Présentez un jeu de données représentatif, les questions auxquelles il doit répondre et le goulot d'étranglement actuel. Nous pouvons convenir de la manière de mesurer la qualité, le débit et la reprise.

Consulter les tarifs Contactez-nous →

Rendez la terminologie concrète

Demandez comment « Qu'est-ce que l'informatique décisionnelle ? » s'applique à votre organisation. Un exemple concret tiré de votre système actuel suffit pour engager la discussion.

Découvrir nos prestations

welcome@engineer.company +45 50 10 61 84 Copenhague, Danemark

Emportez cette page avec vous

Fiche de contact

Créé avec soin et amour par D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Droit d'auteur © 2025 – aujourd'hui · Engineer ApS · Tous droits réservés.

Imprimé depuis https://engineer.company/fr/glossary/business-intelligence/