Langages d'interrogation et de manipulation

1. Algèbre relationnelle

Ensemble d'opérations qui permettent de manipuler des relations considérées comme des ensembles de tuples. Chaque opération prend une ou deux relations en entrée et produit une relation en sortie. On peut composer des opérations. On construit des expressions algébriques complexes qui permettent d'exprimer des manipulations sur un grand nombre de relations.

Une requête est une expression algébrique qui s'applique à un ensemble de relations et produit une relation finale.

Les opérateurs

5 opérateurs sont nécessaires et suffisants

1. la sélection σ
2. la projection π
3. le produit cartésien $\times$
4. l'union $\cup$
5. la différence -

Les deux premiers sont des opérateurs unaires et les autres sont binaires. A partir de ces opérateurs, il est possible d'en définir d'autres comme la jointure ∞ qui est la composition d'un produit cartésien et d'une sélection.

La sélection σ

La sélection $\sigma_F(R)$ extrait de la relation R les tuples qui satisfont un critère de sélection F (=, <, >, ≥, ≤). La sélection a pour effet de supprimer des lignes, mais chaque ligne garde l'ensemble de ses attributs.

La projection π

La projection $\pi_{A1,\ldots,\ Ak}(R)$ ne garde que les attributs $A_1,\ldots,\ A_k$ de R. Contrairement à la sélection, on supprime des colonnes.
En principe, le nombre de lignes dans le résultat est le même que dans la relation initiale. Mais comme le résultat est une relation, il ne peut pas y avoir deux lignes identiques. Le fait de supprimer des colonnes permet ce cas. On ne conserve alors qu'une seule des deux lignes identiques.

Le produit cartésien $\times$

$R\times S$ permet de créer une nouvelle relation où chaque tuple de R est associé à chaque tuple de S. Le nombre de ligne est $|R|*|S|$.

Conflit de nom d'attribut : il peut arriver que les deux relations aient des attributs qui ont le même nom. On doit alors préfixer les attributs par le nom de la table d'où il provient. On peut aussi faire un renommage noté ρ. Par exemple $\rho_{A\rightarrow C,\ B\rightarrow D}(T)$, permet de faire le renommage dans la relation T de A en C et de B en D.

Il est possible de faire le produit cartésien d'une relation avec elle-même. Le renommage est alors indispensable.

L'union $\cup$

$R\cup S$ crée une relation comprenant tous les tuples existants dans l'une ou l'autre des relations R et S. Les deux relations doivent avoir le même schéma (même nombre d'attributs, mêmes noms et mêmes types). Comme pour la projection, les doublons sont supprimés.

La différence –

R-S a pour résultat tous les tuples de R qui ne sont pas dans S. R et S doivent avoir le même schéma.

La jointure ∞

Elle consiste à rapprocher les lignes de deux relations pour lesquelles les valeurs d'un (ou plusieurs) attributs sont identiques. Dans 90% des cas, ces attributs communs sont la clé primaire d'une des relations et la clé étrangère dans l'autre relation.
$R\infty_F S = \sigma_F(R\times S)$, F étant une opération de comparaison liant un attribut de R à un attribut de S.
Le renommage peut intervenir dans la jointure au même titre que pour le produit cartésien.

La division $\div$

Elle permet de déterminer les occurrences de la première relation qui sont associées à toutes les occurrences de la seconde.

X : ensemble des attributs de R
Y : ensemble des attributs de S
Z : ensemble des attributs communs à X et à Y
R÷S renvoie les tuples r de R sur les attributs X-Z qui vérifient que pour tout tuple s de S sur les attributs de Z, le tuple rs (sur les attributs de X) appartient à R.
$R \div S = \pi_{X-Z}(R) - \pi_{X-Z}((\pi_{X-Z}(R) \times \pi_{Z}(S)) - R)$

Expression de requêtes avec l'algèbre

Application à la base « organisation de voyage » dont le schéma est

Station(**nomStation**, capacité, lieu, région, tarif)
Activité(*nomStation*, **libellé**, prix)
Client(**idClient**, nom, prénom, ville, région, solde)
Séjour(*idClient*, **nomStation, début**, nbPlaces)

On utilise la composition des opérations possibles par le fait que tout opérateur produit en sortie une relation sur laquelle on peut appliquer à nouveau des opérateurs.

Sélection généralisée
On peut généraliser les critères de sélection de σ. La composition de plusieurs sélections permet d'exprimer une *conjonction* de critères de recherche.
$\sigma_{\text{capacité}<200}(\sigma_{\text{région}=\text{'Antilles'}}(\text{Station})) \Leftrightarrow \sigma_{\text{capacité}<200 \wedge \text{région}=\text{'Antilles'}}(\text{Station})$
De même, la composition de la sélection et de l'union permet de d'exprimer la *disjonction*.
$\sigma_{\text{capacité}<200}(\text{Station}) \cup \sigma_{\text{région}=\text{'Antilles'}}(\text{Station}) \Leftrightarrow \sigma_{\text{capacité}<200 \vee \text{région}=\text{'Antilles'}}(\text{Station})$
Enfin, la différence permet d'exprimer la *négation* et d'éliminer des lignes.
$\sigma_{\text{capacité}<200}(\text{Station}) - \sigma_{\text{région}=\text{'Antilles'}}(\text{Station}) \Leftrightarrow \sigma_{\text{capacité}<200 \wedge \text{région}\neq\text{'Antilles'}}(\text{Station})$
Les opérateurs d'union et de différence permettent de définir une sélection σ_F où le critère F est une expression booléenne.

Requêtes conjonctives
Elles constituent l'essentiel des requêtes courantes. Ce sont celles avec des « et » (par opposition au « ou » et « non »). Elles s'écrivent avec des π, des σ et des × (et donc indirectement avec des ∞). On utilise la jointure dès que les attributs nécessaires pour évaluer une requête sont répartis dans plusieurs tables.
« Nom et région des stations où on pratique la voile ? » :
$\pi_{\text{nomStation,région}}(\text{Station} \infty_{\text{nomStation= nomStation}} \sigma_{\text{libellé='Voile'}}(\text{Activité}))$

« Nom des clients qui sont partis en vacances dans leur région ainsi que le nom de cette région ? » :
$\pi_{\text{nom,client.région}}(\text{Client} \infty_{\text{idClient=idClient} \wedge \text{région=région}}(\text{Séjour} \infty_{\text{station=nomStation}} \text{Station}))$

Requêtes avec -
Elles permettent d'exprimer « tous les O qui ne satisfont pas p ». On construit alors la requête A qui sélectionne tous les O, ensuite la requête B qui sélectionne tous les O qui satisfont p et finalement on fait A-B.
« IdClient des clients qui ne sont pas allés aux Antilles ? » :
$\pi_{\text{idClient}}(\text{Client}) - \pi_{\text{idClient}}(\sigma_{\text{région='Antilles'}}(\text{Station}) \infty_{\text{nomStation= station}} \text{Séjour})$
La différence peut être employée pour calculer le complément d'un ensemble.
« IdClient des clients et les stations où ils ne sont pas allés ? »
$(\pi_{\text{idClient}}(\text{Client}) \times \pi_{\text{nomStation}}(\text{Station})) - \pi_{\text{idClient,station}}(\text{Séjour})$

Quantification universelle
Une propriété est vraie pour tous les éléments d'un ensemble ssi il n'existe pas un élément de cet ensemble pour lequel la propriété est fausse. On emploie alors la négation et la quantification existentielle.
« Quelles sont les stations dont toutes les activités ont un prix supérieur à 100 ? »
$\pi_{\text{nomStation}}(\text{Station}) - \pi_{\text{nomStation}}(\sigma_{\text{prix}<100}(\text{Activité}))$

Les requêtes les plus complexes sont celles qui utilisent la division.
« Quelles sont les activités qui sont dans toutes les stations ? »
Activité ÷ Station
« IdClient des clients qui sont allés dans toutes les stations ? » (= « idClient des clients tels qu'il n'existe pas de station où ils ne soient pas allés ») :
$\pi_{\text{idClient}}(\text{Client}) - \pi_{\text{idClient}}((\pi_{\text{idClient}}(\text{Client}) \times \pi_{\text{nomStation}}(\text{Station})) - \pi_{\text{idClient,station}}(\text{Séjour}))$