SQL Server

Matthias Colin

SGBDR/SQL History

- 1972-1974: Algèbre Relationnelle (Recherche IBM)
- **1974** : SEQUEL => **SQL** (Structured Query Language), IBM
- 1979 : Oracle Database, 1er SGBDR (RDBMS) commercial
    - Système de Gestion de Base de Données Relationnelle

Autres SGBDR:

- IBM DB2 (1983: presque disparu)
- **Microsoft SQL Server (1989)**
- MySQL (1995, racheté par Oracle) / MariaDB (2009: OpenSource)
- PostgreSQL (1989/1995: OpenSource)
- SQLite (2000: OpenSource)

Standard SQL: 1986

Standard SQL

- 1986 1ere version du standard
- …
- 1999 code stocké, tableaux, objets
- 2003 fonctions analytiques (window functions) + XML
- …
- 2016 JSON
- 2023 dernière version du standard

Les éditeurs suivent +/- ce standard (optional features)

https://en.wikipedia.org/wiki/SQL

Microsoft SQL Server

Associé à Sybase à la création

- 1989 version 1.0
- …
- 2012 version 11.0 : SQL Server 2012
- 2014 version 12.0 : SQL Server 2014
- 2016 version 13.0 : SQL Server 2016
- 2017 version 14.0 : SQL Server 2017
- 2019 version 15.0 : SQL Server 2019
- 2022 version 16.0 : SQL Server 2022

Transact SQL

SQL de MS SQL Server = Transact SQL (TSQL)

- SQL
- Types de données
- Fonctions SQL
- Code stocké (fonctions, procédures, triggers)

Tables

https://github.com/matthcol/dbmovie/archive/refs/heads/sqlserver-nodocker.zip

Langage SQL

Découpé en plusieurs parties

- DDL: Data Definition Language
  - **CREATE** TABLE Person (….)
  - **ALTER** TABLE Person ..
  - **DROP** TABLE Person
  - gérer tables, vues, indexes, users, …
- DML: Data Manipulation Language
  - **SELECT** …. pour extraire des données (requête / query)
  - **INSERT** … pour ajouter des données
  - **UPDATE** … pour modifier la donnée
  - **DELETE** … pour supprimer la donnée (toute une ligne)
- Transactions: COMMIT, ROLLBACK
- Privilèges: GRANT, REVOKE

Data types

Data type category	Data type
Exact numerics	bigint, numeric, bit, smallint, decimal, smallmoney, int, tinyint, money
Approximate numerics	float, real
Date and time	date, datetimeoffset, datetime2, smalldatetime, datetime, time
Character strings	char, varchar, text
Unicode character strings	nchar, nvarchar, ntext
Binary strings	binary, varbinary, image
Other data types	cursor, rowversion, hierarchyid, uniqueidentifier, sql_variant, table

Opérateurs

- Arithmétiques: + - * / %
- Concaténation: + (autre éditeurs ||)
- Liste de valeurs: IN, NOT IN
- Conditions: = <> != < <= > >= BETWEEN, IS NULL, IS NOT NULL
- Combinaisons de conditions: AND, OR, NOT

Fonctions (en ligne)

- Nombres: ROUND, CEILING, FLOOR, ABS
- Texte: LOWER, UPPER, SUBSTRING, LEN, CONCAT, TRIM, STUFF
- Date et heures:
  - DATEPART, YEAR, MONTH, DAY
  - DATEDIFF, DATEADD
  - EOMONTH (fin du mois)
  - CURRENT_TIMESTAMP (pas de CURRENT_DATE ou CURRENT_TIME),
  - DATETRUNC (SQL Server 2022)
- NULL : COALESCE, NULLIF

Fonctions de groupe

- COUNT : lignes, valeurs (distinctes) sur une colonne
- SUM, MIN,MAX, AVG, STD : nombres
- RANK : classement (+ fenêtrage)
- STRING_AGG : concaténation avec séparateur

Jointures

- Interne
  - INNER JOIN
- Externe
  - LEFT OUTER JOIN
  - RIGHT OUTER JOIN
  - FULL OUTER JOIN

Indexes

1 index est **trié**

1er critère

- Clustered: 1 by table (dans la table, définit l'ordre de stockage)
- Non Clustered: plusieurs par table, stocké à côté

2e critère: unique (ou pas)

avantage: gain de temps pour lecture (requête SELECT)

inconvénients:

- maj (1 modif data => modif tous les index liés)
- stockage supplémentaire

Indexes

coût de recherche dans un index: n => log(n) (base 2)

- 1K = 1024 => c = 10
- 1M = 2*20 => c = 20
- 1G = 2*30 => c = 30
- …
- 1E = 2*60 => c = 60

Tri

Coût: n => n.log(n)

- 1K => 10K
- 1M => 20M
- 1G => 30G

Index implicite

Créés automatiquement

Liés à une contrainte:

- primary key (unique, not null) => clustered
- unique => not clustered

Type de tables / index

Table avec clustered index (pk en général)

Table sans clustered index: Heap Table => organisation random, balayage total

(à suivre)

Optimisation de requête

- dénormalisation (cons: maintenance, intégrité de la donnée)
  - moins de jointures
- réécriture logique (dev ou sgbd)
  - inversion de clause
  - sous-requête
- index
- + CPU
- + mémoire
- + espace disque temporaire