

# Qu'est-ce qu'un SGBDR



par fadace ([Fabien Celaia](#))

Date de publication : 6.1.2007

Dernière mise à jour : 6.1.2007

Définition d'un système de gestion de base de données relationnel, selon Codd



Sur developpez.com, et bien sûr bien souvent ailleurs, le débat des fervents et des détracteurs d'une base de données a souvent tourné autour d'une question : la base en question pouvait-elle être considérée à proprement parlé comme un SGBDR.

Foin de débat : le plus sûr pour répondre à cette épineuse question est de se rabattre sur la définition d'un SGBDR (RDBMS en anglais). Et qui de plus à même d'amener une définition que le celui même qui est à l'origine de la théorie du modèle relationnel, à savoir ce cher Edgar F. Codd (Ted pour les intimes).

Les 12 règles de Codd sont un ensemble de treize règles (eh oui, comme informaticien, on commence à compter à 0) proposées par Edgar F. "Ted" Codd, conçu pour définir ce qui est exigé d'un système de gestion de base de données afin il pour être considéré relationnel, c.-à-d., un SGBDR.

Au début des années 80, Codd s'est brouillé avec les éditeurs (et plus particulièrement son employeur IBM) qu'il accusait de pervertir son modèle dans leurs implémentations. Il a donc produit ces règles afin d'empêcher ces derniers de s'éloigner de sa vision du SGBDR. Ce faisant, il les obligea à suivre sa ligne.

N°	Règle	Description / explication
0	Le système doit se qualifier comme relationnel, comme base de données, et comme système de gestion	Pour qu'un système soit qualifié de système de gestion de base de données relationnelles (SGBDR), ce système doit employer (exclusivement) le modèle relationnel pour contrôler la base de données.
1	Information	Toute l'information dans la base de données est représentée d'une et une seule manière, à savoir par des valeurs dans des champs de colonnes de tables.
2	Garantie d'accès	Toutes les données doivent être accessibles sans ambiguïté. Cette règle est essentiellement un ajustement de la condition fondamentale pour des clés primaires. Elle indique que chaque valeur scalaire individuelle dans la base de données doit être logiquement accessible en indiquant le nom de la table contenant, le nom de la colonne contenant et la valeur principale primaire de la rangée contenant.
3	Traitement systématique des valeurs nulles	Le système de gestion de bases de données doit permettre à chaque champ de demeurer nul (ou vide). Spécifiquement, il doit soutenir une représentation "d'information manquante et d'information inapplicable" qui est systématique, distincte de toutes les valeurs régulières (par exemple, "distincte de zéro ou tous autres nombres," dans le cas des valeurs numériques), et ce indépendamment du type de données. Cela implique également que de telles représentations doivent

		être gérées par le système de gestion de bases de données d'une manière systématique.
4	Catalogue en ligne actif basé sur le modèle relationnel	Le système doit supporter un catalogue en ligne, intégré, relationnel, accessible aux utilisateurs autorisés au moyen de leur langage d'interrogation régulier. Les utilisateurs doivent donc pouvoir accéder à la structure de la base de données (catalogue) employant le même langage d'interrogation qu'ils emploient pour accéder aux données de la base de données.
5	Sous-langage de données complet	Le système doit soutenir au moins un langage relationnel qui <ul style="list-style-type: none"> <li>a a une syntaxe linéaire</li> <li>b peut être employé interactivement et dans des programmes d'application,</li> <li>c supporte des opération de définition d'informations supplémentaires (incluant des définitions de vues), de manipulation de données (mise à jour aussi bien que la récupération), de contraintes de sécurité et d'intégrité, et des opérations de gestion de transaction (commencer, valider et annuler une transaction).</li> </ul>
6	Mise à jour des vues	Toutes les vues pouvant théoriquement être mises à jour doit pouvoir l'être par le système.
7	Insertion, mise à jour, et effacement de haut niveau	Le système doit supporter les opération par lot d'insertion, de mise à jour et de suppression. Ceci signifie que des données peuvent être extraites d'une base de données relationnelle dans des ensembles constitués par des données issues de plusieurs tuples et/ou de multiples table. Cette règle explique que l'insertion, la mise à jour, et les opérations d'effacement devraient être supportées aussi bien pour des lots de tuples issues de plusieurs tables que juste pour un tuple unique issu d'une table unique.
8	Indépendance de donnée physique	Les modifications au niveau physique (comment les données sont stockées, si dans les rangées ou les listes liées etc...) ne nécessitent pas un changement

		d'une application basée sur les structures.
9	Indépendance de donnée logique	Les changements au niveau logique (tables, colonnes, rangées, etc) ne doivent pas exiger un changement dans l'application basée sur les structures. L'indépendance de données logiques et plus difficile a atteindre que l'indépendance de donnée physique.
10	L'indépendance d'intégrité	Des contraintes d'intégrité doivent être indiquées séparément des programmes d'application et être stockées dans le catalogue. Il doit être possible de changer de telles contraintes au fur et à mesure sans affecter inutilement les applications existantes.
11	L'indépendance de distribution	La distribution des parties de la base de données à de diverses localisations devrait être invisible aux utilisateurs de la base de données. Les applications existantes devraient continuer à fonctionner avec succès a quand une version distribuée du système de gestion de bases de données est d'abord présentée, et b quand des données existantes sont redistribués dans le système
12	Nonsubversion	Si le système fournit une interface de bas niveau , alors cette interface ne peut pas être employée pour contourner le système, par exemple, pour contourner une contrainte relationnelle de sécurité ou d'intégrité.

Source : selon traduction de la version anglaise sous Wikipédia