



S. Q. L. S p o t

www.sqlspot.com

L'expertise des données SQL

audit, conseil, expertise, tuning, optimisation, modélisation, qualité de données, formation MS SQL Server
siège social : chemin du Riou 83670 BARJOLS — 06 11 86 40 66 — sqlpro@sqlspot.com



COMPARAISON DES FONCTIONNALITÉS de MS SQL Server 2012 et PostGreSQL 9.2

Légende :

- DEVELOPPEMENT
- PERFORMANCE
- ADMINISTRATION
- BUSINESS INTELLIGENCE

- en rouge fonctionnalité non supportée
- en bleu fonctionnalité moins avancées que son concurrent

MARS 2013
PAR FREDERIC BROUARD - ARCHITECTE DES DONNÉES

Fonctionnalité	SQL Server 2012	PostGreSQL 9.2
Serveur multi instance ¹	oui	non
Serveur multibase	oui	oui (mais cloisonnement)
Base multi schéma	oui	oui
Cliché de bases de données ²	oui	non
Parallélisme des requêtes ³	oui	non
Parallélisme d'accès disque ⁴	oui	non
Langage SQL (core)	norme SQL:2008	norme SQL:2008
Collations ⁵	3 885 collations différentes	quelques dizaines dépendantes de l'OS [A]
Requêtes de mise à jour ensembliste	oui	non (possible via contraintes différées)
Stockage de fichier (DataLink) ⁶	oui (FileStream)	non
PSM (routines SQL)	oui (Transact SQL)	oui (PG/PLSQL) [B]
Contraintes déferables ⁷	non	oui
Transaction explicite ⁸	oui	non [C]
Transaction imbriquées ⁹	oui	non

¹ plusieurs serveurs avec un réglage différents pour chacun. Par exemple un serveur pour le développement, un pour les tests

² une photographie des données de la base à un instant déterminé et en lecture seule

³ la possibilité pour une même requête d'être parallélisé sur plusieurs CPU simultanément

⁴ la possibilité pour les écritures/lecture physique de données d'être parallélisé sur plusieurs CPU simultanément

⁵ la façon dont sont gérés les caractères diacritiques telles qu'accents ou ligature, la casse et langue

⁶ chemin vers un stockage de fichier dans le "file système" transactionné et géré intégralement par le SGBDR de façon à conserver les données intègre y compris lors des sauvegardes

⁷ La possibilité pour une contrainte de voir son application repoussé en fin de transaction et non au moment de la commande

⁸ à l'aide des ordre SQL BEGIN WORK/TRANSACTION, COMMIT et ROLLBACK

⁹ la possibilité dans une transaction de démarrer une autre transaction

Fonctionnalité	SQL Server 2012	PostGreSQL 9.2
Transaction distribuées ¹⁰	oui via MS DTC	oui, via outils tiers dépendant des os
Transaction DDL ¹¹	oui	non
Niveau d'isolation des transactions	4 niveaux normatifs + snapshot, dynamique	3 niveaux normatifs + snapshot, statique
Système d'Information Géographique (SIG)	standard OGC	standard OGC
Support de XML	oui (XQuery + Xpath) + mise à jour (spécifique)	oui (sans mise à jour), limité à UTF8 [D]
Vues partitionnées distribuées ¹²	oui	non
Vue matérialisées ¹³	oui, synchrone, optimisation automatique	non
Déclencheurs DML	AFTER (ensemblistes)	BEFORE, AFTER (ligne par ligne) [E]
Déclencheurs INSTEAD OF ¹⁴	oui	Non, sauf vues et avec certaines limitations
Déclencheurs DDL ¹⁵	oui (niveau base et serveur)	non
Fonctions scalaires	oui	oui
Fonctions table	oui	oui
Fonctions d'agrégation	oui (via SQL CLR)	oui
Procédure	oui	non (fonctions) [F]
Opérateur d'intra jointure ¹⁶	oui APPLY	non
Requête pilotées par événement ¹⁷	Oui (query notification et streaminsight)	non

¹⁰ la possibilité de faire des transactions multi serveur (nécessite un superviseur comme MS DTC)

¹¹ la possibilité de gérer des transactions sur les ordres de définition des objets de la base comme CREATE ALTER ou DROP...

¹² vues transverses concaténant des données sur différents serveurs et permettant de les mettre à jour de manière transparent

¹³ dénormalisation par l'intermédiaire de vue stockant des données de manière synchrone et optimisée

¹⁴ déclencheurs de substitution permettant de mettre à jour des tables ou des vues complexes

¹⁵ routines déclenchées sur des commandes de définition des données comme CREATE, ALTER ou DROP...

¹⁶ permet de joindre une fonction table ou un nœud XML extrait d'une colonne d'une table de la requête

¹⁷ Le lancement d'une requête lié à l'abonnement à un événement extérieur à la base

Fonctionnalité	SQL Server 2012	PostgreSQL 9.2
Codage de routines externe	oui (tous langage .net via SQL CLR)	oui (Tcl, Perl, Python ...)
DDL cascade ¹⁸	non	oui
Authentification par certificat	oui	non
Chiffrement de données	oui, 4 méthodes (passe, certificat, clefs symétrique et asymétrique), clefs stockées dans la base	oui, 2 méthodes via pgcrypto (clef symétriques et asymétriques), clefs stockées en dehors de la base [G]
Signature de données	oui	oui [H]
Signature de routines	oui	non
Gestion des clefs par HSM ¹⁹	oui	non
Chiffrement du stockage ²⁰	oui (TDE)	non
Chiffrement des sauvegardes	oui (via TDE)	non
Réplication de données ²¹	4 méthodes : transactionnelle, par cliché, de fusion et point à point	non [I]
Réplication d'information ²²	oui : service broker	non
DDL "on line" ²³	oui	non
Index Btree	oui	oui
Index hash	possible via colonnes calculées CHECKSUM	oui
Index filtrés	oui	oui
Index calculés	oui sur colonne calculées persistante	oui sur expression
Index couvrants ²⁴	oui	non

¹⁸ la propagation de l'ordre DML aux objets fils (par exemple DROP TABLE... CASCADE).

¹⁹ HSM : Hardware Security Module : boitiers de sécurité externe ou sont stockées les clefs de cryptage

²⁰ chiffrement des fichiers de la base (données et transactions)

²¹ copie d'un serveur à l'autre de quelques lignes de quelques tables d'une base dans l'autre sans que la structure de la base soit strictement la même

²² envoi asynchrone d'informations sérialisées et transactionnées d'un serveur à l'autre avec acquittement des messages

²³ la possibilité d'envoyer une commande DDL comme ALTER TABLE sans blocage

Fonctionnalité	SQL Server 2012	PostgreSQL 9.2
Index XML	oui (4 méthodes)	non
Index SIG	oui BTree quadratique	oui (GIST)
Index vertical	oui, (columnStore)	oui (GIN, GIST)
Indexation textuelle	norme SQL + spécifique	spécifique (ts_vector) [J]
Indexation textuelle XML	oui	oui
Indexation textuelle doc. électronique	oui	non
Indexation textuelle sémantique ²⁵	oui	non
Partitionnement de données ²⁶	oui	non [K]
Compression des données	oui (page ou ligne)	non (sauf LOBs systématiquement compressés)
Optimisation statistique	oui (automatique)	oui (réglage manuel) [L]
Optimisation sémantique ²⁷	oui	non [M]
Indicateurs de requête ²⁸	oui	non
Réinjection de plan de requête ²⁹	oui (2 méthodes : dans le code SQL de la requête ou à la volée)	non
Mise à jour des statistiques d'optimisation	oui (4 méthodes : induite, manuelle, automatique, asynchrone)	oui (2 méthodes : induite, manuelle)
Création des statistiques d'optimisation	oui	non
Statistiques d'exécution ³⁰	oui	non

²⁴ colonnes ajoutées dans une clause INCLUDE pour éviter les doubles lectures (index + table) et assurer la couverture de la requête par le seul index

²⁵ Concerne les méta tags des documents électroniques

²⁶ Consiste à ce qu'une même table soit stockée dans différentes espaces de stockage en fonction d'un critère de données.

²⁷ La possibilité de tirer parti des contraintes SQL pour simplifier et optimiser les plans de requêtes

²⁸ La possibilité de contraindre l'optimiseur en posant des indicateurs de requête lui intimant une façon statique de créer son plan de requête

²⁹ La possibilité de récrire l'intégralité du plan de requête pour une requête précise

Fonctionnalité	SQL Server 2012	PostGreSQL 9.2
Statistiques prédictives ³¹	oui (index manquants)	non
Audit des requêtes pour tuning	oui (SQL profiler)	oui (PGFouine)
Assistant de tuning	oui (DTA)	non
Limitation de coût de requête	oui	non
Gouverneur de ressources ³²	oui	non
Audit légaux	C2, SOX	non
Audit de sécurité des données ³³	oui (Database Audit)	non
Audit de changement ³⁴	oui (2 méthodes : change tracking et CDC)	non
Audit d'exécution	oui (4 méthodes : SQL profiler, eXtended Events, Moniteur de performances, data collector)	non [N]
Gestion des espaces de stockage	oui	oui [O]
Espace de stockage "Read Only"	oui	non
Espace de stockage pré dimensionnés	oui	non
Journaux de transaction pré dimensionnés	oui	non
Un journal de transaction par base	oui	non
Un journal de transaction pour les tables temporaires	oui	non
Analyse de fragmentation d'index	oui	oui (très difficile) [P]
Défragmentation d'index	oui	oui
Reconstruction d'index "on line" ³⁵	oui	non

³⁰ telle que les n requêtes, procédures, triggers les plus couteux en temps CPU, pages lues...

³¹ Des statistiques d'action prédictives à réaliser, afin d'améliorer les performances, basé sur les requêtes passées

³² pour limiter les ressources CPU ou RAM de certains utilisateurs

³³ Permet par exemple de savoir qui, quand et sur quel objet a été fait un SELECT, INSERT, UPDATE DELETE, MERGE, etc.

³⁴ pour pouvoir alimenter un datawarehouse par différence et non plus par reconstruction

Fonctionnalité	SQL Server 2012	PostgreSQL 9.2
Sauvegardes complètes	oui	Oui
Sauvegardes partielles	oui (fichiers ou groupes de fichier)	non [Q]
Sauvegardes différentielles ³⁶	oui	non
Sauvegardes transactionnelles	oui	oui
Sauvegardes compressées	oui	non
Sauvegardes multiples ³⁷	oui	non
Sauvegardes multi-fichiers ³⁸	oui	non
Restauration complète	oui	oui
Restauration avec arrêt dans le temps	oui	oui
Restauration avec arrêt à transaction	oui	non
Restauration de pages endommagées	oui	non
Restauration en ligne ³⁹	oui (page fichier)	non
Vérification d'intégrité physique	oui	non
Vérification d'intégrité logique	oui	non
Log shipping	oui	oui
Mirroring de base de données	oui synchrone avec ou sans basculement auto, asynchrone	oui, synchrone, asynchrone
Clustering ⁴⁰	oui (via Windows Cluster Services)	non
Intervention hardware à chaud	oui (RAM et CPU)	non
Gestion qualité des données	oui (Master Data Services)	non [R]

³⁵ reconstruction des index sans blocage des tables

³⁶ sauvegarde de certains espaces de stockage et pas d'autres

³⁷ la possibilité de livrer une même sauvegarde à différents emplacement. par exemple localement et sur un serveur distant simultanément

³⁸ la possibilité d'étendre la sauvegarde sur plusieurs fichiers afin de bénéficier du parallélisme d'écriture sur différents disques par exemple

³⁹ La possibilité de restaurer alors que la base est encore en ligne

⁴⁰ Deux serveurs se partageant les fichiers des bases de données l'un reprenant à chaud le service en cas de panne de l'autre

Fonctionnalité	SQL Server 2012	PostgreSQL 9.2
Datawarehouse	oui	oui [R]
ETL intégré	oui (SSIS)	non [R]
Moteur décisionnel	oui	non [R]
langage de requête MDX	oui	non [R]
Cubes et datamart	oui	non [R]
Stockage MOLAP	oui	non [R]
Stockage ROLAP	oui	non [R]
Stockage HOLAP	oui	non [R]
Reporting intégré	oui	non [R]
Reporting interactif (web)	oui	non [R]
Desktop Olap	oui (PowerPivot)	non [R]
Publication de KPI	oui (SharePoint)	non [R]

Ressources disponibles

		PostgreSQL	MS SQL Server	ratio	
Ressources documentaires	WEB	Google	30 400 000	113 000 000	3,7
		Forums (Q/R)	4 385 / 19 336	18 210 / 92 006	4,1 / 4,8
	Livres	AMAZON.com	332	4 290	13
Ressources humaines	Offre d'emploi, membres	APEC	118	666	5,6
		Viadeo	5 022	20 403	4,1
		Linkedin	5 697	22 134	3,9

Ressources d'entreprises :

- PostgreSQL** : Dalibo, 2ndQuadrant, OpenWide, Linagora, Smile...
 NOTA : peu d'entreprises spécialisées sous PostgreSQL (en dehors des deux premières citées), pas de circuit de validation des connaissances. En 2008 on comptait 70 entreprise SSII dans l'Open Source (étude OPIIEC / Pierre Audoin)
- SQL Server** : 658 entreprises agréées "SQL Server" par Microsoft en france
 NOTA : certifications MS par spécialités sur SQL Server (MCSA, MCSE, MCSM, MCTS, MCITP, MCM, MCA) référencement des experts MVP

Au niveau documentaire, on note un ratio au mieux de 4 au pire de 12 et en matière d'entreprises prestataires, même en comptant les 5 intervenants mentionnés on est à un ratio de 100. Bref, même si PostgreSQL semble avoir le vent en poupe (les demandes de recrutement le prouve), il reste plus un outil confidentiel qu'industriel.

Comparaisons complémentaires :

Étude comparative du TCO SQL Server / PostGreSQL :

<http://bornagainagilist.wordpress.com/2012/06/22/sql-server-2012-cheaper-than-open-source-database-options/>

Comparaison détaillée des approches et fonctionnalités :

http://download.microsoft.com/download/F/A/0/FA04BB79-2EC0-4A3B-A8C6-58679D43B582/SQL_Server_2012_Compared_With_PostgreSQL_9_White_Paper_Apr2012.pdf

Ten Reasons PostgreSQL is Better Than SQL Server :

<http://facility9.com/2011/12/ten-reasons-postgresql-is-better-than-sql-server/>

L'article est farci d'approximations et d'erreurs notamment sur :

- les niveaux d'isolation des transactions (confusion faites sur le versionning des lignes, le niveau d'isolation SNAPSHOT et le niveau d'isolation SERIALISZABLE) ;
- la réplication (en fait mirroring) plus avancé chez SQL Server que PostGreSQL (notamment du fait de l'automatisation du basculement) ;
- le géo spatial (les deux systèmes étant à même niveau et il existe de nombreuses contributions pour en étendre les possibilités via SQL CLR pour SQL Server) ;
- les extensions (elles sont possible via SQL CLR, nettement plus sécurisé, en sus d'être nombreuses et riches...) ;
- les CTE "mise à jourables" (non prévue par la norme SQL et équivalent de la clause OUTPUT de SQL Server) ;
- quant aux types temporels c'était vrai avec les versions antérieures à la 2008 cela ne l'est plus du tout aujourd'hui.

Bref, l'auteur ne maîtrise visiblement pas son sujet...

Notons quand même les seules avancées réelles de postGreSQL pointées par le document, qui sont :

- les "unlogged tables" (mais que se passe-t-il dans la transaction en cas de crash ?)
- les contraintes d'exclusion, dont l'équivalent est facile à réaliser sous SQL Server via contraintes CHECK + UDF ou bien par des déclencheurs.

Explication des différences spécifiques à PostgreSQL

[A] collations : sous PostgreSQL, la gestion des collations est très dépendante des jeux de caractères, eux-mêmes dépendant des OS. Les collations peuvent être posées au niveau colonne et dans les comparaisons et les tris, mais pas dans les fonctions ni les opérateurs SQL. Il n'y a pas de possibilité d'obtenir des collations sensibles ou insensibles à la casse ou aux caractères diacritiques pour une même langue et donc d'effectuer des recherches en paramétrant la casse ou les accents. Sous Windows, la liste des collations PostgreSQL est limitée aux langages locaux supportés par l'OS : <http://msdn.microsoft.com/en-us/library/39cwe7zf>

[B] PSM - Routines : PostgreSQL dispose d'un langage (PG PL/SQL) dont la structure est proche du PL/SQL d'Oracle.

[C] Transaction explicites : PostgreSQL ne permet pas de démarrer des transactions explicitement par BEGIN TRANSACTION (ou BEGIN WORK) ni des les terminer (COMMIT, ROLLBACK) dans le code des routines.

[D] XML : PostgreSQL s'est inspiré de la norme SQL pour la manipulation des données sous XML, tandis que la concurrence (IBM DB2, Oracle et SQL Server) utilise les standards spécifique à XML que sont XQuery et XPath.

[E] trigger DML : les triggers PostgreSQL sont compliqués à écrire et limités dans l'étendue des possibilités (par de DML par exemple). Il faut créer une fonction pour gérer le comportement du trigger puis une routine pour capturer l'événement renvoyant vers cette fonction. Il n'est pas possible de gérer la transaction dans le trigger (COMMIT ou ROLLBACK par exemple), mais on peut coder des triggers particulier appelés "CONSTRAINT TRIGGER" par PostgreSQL.

[F] procédure : en fait la notion de procédure avec gestion de la transaction n'existe pas dans PostgreSQL. PostgreSQL n'implémente que des "fonctions" qui sont toujours transactionnellement atomiques.

[G] cryptage : sous PostgreSQL avec l'extension pg_crypto, la sécurité du cryptage est faible puisqu'il n'est pas possible de stocker les clés dans la base ni d'assurer une ouverture/fermeture automatique de ces dernières. La présence de clés ou certificats sous forme de fichiers dans le système d'exploitation constitue une pratique inacceptable dans certains milieux comme le bancaire ou la santé.

[H] signature des données : possible via la contribution externe PG_crypto

[I] réplication de données : il est cependant possible de faire de la réplication de données sous PostgreSQL en codant manuellement et pour chaque table des déclencheurs, mais cela reste une solution synchrone sujette à "l'effet convoi".

[J] indexation textuelle : PostgreSQL ne propose pas de mécanisme d'automatisation des tâches d'alimentation des index textuels, notamment au fil de l'eau de manière asynchrone ou bien par batch différentiel. Il faut le faire à la main notamment via des triggers synchrones qui pèsent sur les performances. Les possibilités de recherche sont faibles et le paramétrage statique. Il n'est pas possible de changer la langue ou la liste des mots noirs dynamiquement dans la requête de recherche. Les recherches avec paramétrage des caractères diacritiques sont impossible (accents ligature).

[K] partitionnement : PostgreSQL propose une solution intitulé partitionnement des données... qui n'en est pas une ! Le partitionnement version PG nécessite la création de plusieurs tables et quantité de triggers. À lire : http://blog.developpez.com/sqlpro/p10378/langage-sql-norme/partitionner_une_table_comparaison_postg

[L] optimisation statistique : l'optimiseur de PostgreSQL est basé sur deux algorithmes : l'un de backtracking l'autre génétique se déclenchant au delà de 12 jointures (paramétrable). Mais l'un comme l'autre posent des problèmes sur des requêtes complexes : durée de calcul du plan pour le premier et qualité du plan produit pour le second. En pratique, se limiter à une vingtaine de jointures. A noter aussi la complexité de paramétrage de l'optimiseur dont la documentation indique explicitement qu'il n'existe pas de méthode déterminé pour procéder au réglage et que se hasarder à le régler "is very risky" !

[M] optimisation sémantique : PostgreSQL ne la pratique pas, sauf quelques exceptions, comme pour le « partitionnement » (voir [K])

[N] audit d'exécution : mis à part quelques vues PostgreSQL ne dispose d'aucun outil pour investiguer ce qui se passe au niveau physique dans le moteur SQL, par exemple la RAM utilisée, le nombre de CPU actif ou la quantité de données écrites sur les disques.

[O] gestion des espaces de stockage : PostgreSQL ne dispose d'aucun moyen autre que d'indiquer un répertoire pour gérer le stockage des données de la base. À lire : <http://blog.developpez.com/sqlpro/files/2013/03/Ou-sont-stock%C3%A9s-mes-donn%C3%A9es-avec-PostgreSQL.pdf>

[P] analyse de fragmentation d'index : il n'y a pas d'outil (fonction ou vue) donnant directement le taux de fragmentation de chaque index ou table dans PostgreSQL. La requête est très complexe à mettre au point. Un exemple d'une telle requête figure dans l'ouvrage "SQL" 4e édition, collection Synthex, Pearson 2012 au chapitre "Administration", disponible en ligne : http://compagnons.pearson.fr/synthex_sql_4/

[Q] sauvegarde partielle : il reste toujours la possibilité d'exporter des données table par table, mais au risque d'un défaut de cohérence ou d'intégrité !

[R] Business Intelligence : PostGreSQL n'offre aucune solution de BI intégré. Pas de moteur de stockage de base décisionnelles et datawarehouse, ni de langage de fouille de données comme MDX. Pas d'outil de migration de données (ETL), pas d'outil de reporting interactif (drill down, drill through, tableaux croisés interactifs...), pas de solution client coté utilisateur.

En comparaison, les éditions BI, standard et Enterprise de SQL Server offrent toutes ces fonctionnalités à travers SSAS (SQL Server Analysis Services), SSRS (SQL Server Reporting Services), SSIS (SQL Server Integration Services) et pour les outils clients Excel associé à PowerPivot pour l'analyse client en ligne et SharePoint pour la publication des tableaux de bord et des indicateurs clefs.