L'Audit des Bases de Données Relationnelles

par Frédéric Brouard, alias SQLpro
MVP SQL Server
Expert langage SQL, SGBDR, modélisation de données

Auteur de :

- SQLpro http://sqlpro.developpez.com/
- "SQL", coll. Synthex, avec C. Soutou, Pearson Education 2005
- "SQL" coll. Développement, Campus Press 2001

Enseignant aux Arts & Métiers et à l'ISEN Toulon

Il n'y a pas qu'en matière financière ou comptable que l'audit se pratique. Si en matière informatique l'audit le plus connu est celui qui concerne la sécurité des systèmes, l'audit de bases de données constitue une demande de plus en plus croissante. N'oublions pas que les données constituent la matière même de l'informatique et que la croissance du volume des données stockées nécessites quelques exigences de sérieux que certaines officines informatiques ont tendance à oublier au profit d'application poudre aux yeux.
Voici un article qui fait le point de ce que l'on est en droit d'attendre d'un audit de bases de données relationnelles.

1 - Définition de l'audit

Audit vient du latin *audire* qui signifie écouter. A cette époque ce terme désignait un contrôle effectué au nom de l'empereur sur la gestion des provinces.

Extrait du site web Wikipedia :

> « *L'audit, exercé par un auditeur, est un processus systématique, indépendant et documenté permettant de recueillir des informations objectives pour déterminer dans quelle mesure les éléments du système cible satisfont aux exigences des référentiels du domaine concerné. Il s'attache notamment à détecter les anomalies et les risques dans les organismes et secteurs d'activité qu'il examine. Auditer une entreprise, un service, c'est écouter les différents acteurs pour comprendre et faire comprendre le système en place ou à mettre en place.* »

1.1 - Objectifs de l'audit

Un audit de qualité doit être conçu pour s'approcher des objectifs suivants :

- déterminer la conformité des éléments du système aux exigences spécifiées;
- déterminer l'aptitude du système à atteindre les objectifs spécifiés;
- donner à l'audité la possibilité d'améliorer son système et son efficacité.

Les apports d'un audit sont constitués par :

- une mesure d'écart,
- une source de progrès,
- un facteur d'économie,
- une occasion de considérer son système de façon objective,
- une approche à la compréhension de son système,
- une implication concrète des acteurs dans le système.

1.2 - Audit informatique :

L'audit informatique est l'évaluation des risques et performances liée aux activités informatiques, dans le but d'apporter une diminution des premiers, une amélioration des seconds et globalement de mieux maîtriser le système.

2 - Audit de base de données relationnelles :

Il consiste à analyser l'existant d'une base de données (ou d'un ensemble cohérent de bases de données) plus ou moins profondément afin d'en diagnostiquer l'état et de préconiser des améliorations, essentiellement sur le plan de la conformité et des performances.
Les différents niveaux de l'audit de base de données

2.1 - Audit de structure :

Il s'agit de montrer si la structure de la base est en adéquation avec les exigences fonctionnelles et particulièrement adaptée à l'usage qui en est fait (requêtes).
On vérifiera en particulier que le modèle de données a été respectueux des règles de l'art : modélisation relationnelle (MCD, MLD), respect des formes normales, contraintes de domaine, schéma externe... et correspond à la nécessité de service.

2.2 - Audit de qualité des données :

Il s'agit de vérifier que la base n'est pas polluée par de nombreuses données inutiles ou erronées. En particulier on vérifiera l'existence de contraintes telles que : contraintes de

domaine, intégrité référentielles, unicité, validation, format (notamment les formats normalisés de données)...
Dans le cas d'absence de telles contraintes, des mesures par sondage devront être entreprises afin de remonter les anomalies.
Dans une base de données comptable on pourra par exemple utiliser un algorithme basé sur la Loi de Benford afin de déterminer si des anomalies d'écriture n'ont pas été commises en masse.

2.3 - Audit de configuration et de performances :

Il s'agit de vérifier si la configuration du serveur logique (SGBDR) et du serveur physique (hardware) est conforme aux exigences du service des données : en particulier RAM, disques, processeurs, paramétrages à tous niveau.
Cela nécessite de tracer l'activité du serveur sur divers plans techniques puis d'analyser les données recueillies à l'aide de différentes techniques et moyens qui peuvent faire l'objet de plusieurs passes successives pour affinement.

2.4 -Audit des requêtes clientes :

Il s'agit de vérifier le style de développement adopté (requêtes *adhoc*, emploi de procédures stockées, *mapping* relationnel objet...), la qualité de l'écriture des requêtes et l'indexation des tables. On procède à l'aide de différentes techniques en fonction de la façon dont est écrit le programme applicatif, techniques qui peuvent être combinées (analyse d'échantillon, traçage de l'activité du moteur SQL, revue de code...). Cela nécessite une bonne connaissance de l'optimisation et du fonctionnement du moteur de requête et du moteur de stockage.
Par exemple, on débusquera par "écrémage" les 20% de requêtes qui consomment 80% des ressources (loi de Pareto) et on s'appliquera à en diminuer drastiquement le coût, par exemple par récriture, indexation, voir *refactoring* du modèle.
A ce stade, ou pourra aussi rechercher les problèmes potentiels liés à la sécurité : configuration des comptes d'accès, mise en place des privilèges sur les objets, utilisation de procédure accédant à des ressources externes, injection de code... et en donner les remèdes.

2.5 - Audit d'infrastructure réseau :

Il s'agit de vérifier ce qui se passe entre les serveurs SQL et les "clients". Ces clients pouvant être d'autres serveurs (Web, objet...) ou des clients applicatifs finaux. Il faut mesurer les temps de réponse effectifs (trames) et ressentis (utilisateur). La technique pour ce faire consiste à analyser les trames réseau pour en connaître la volumétrie et chronométrer les allers-retours des IHM. Pour une volumétrie anormale de trame, on en déduira quelles sont les commandes à l'origine et comment on peut agir dessus. Pour un temps de réponse IHM trop important, il faudra déterminer quel élément dans la boucle est à l'origine de la consommation anormale des temps de réponse.
L'inconvénient de cet audit, somme toute assez rare, est qu'il nécessite des moyens matériels et logiciels couteux et une analyse qui, compte tenu de la volumétrie et de la complexité des données recueillies, peut s'avérer assez chère.

3 - Responsabilité juridique

En matière de SGBD il existe différents textes de Loi sur lequel s'appuyer, mais aussi une importante jurisprudence. Ainsi les règles de l'art imposent-elles une manière de concevoir des applications utilisant des bases de données relationnelles.

Citons quelques textes...

3.1 - Lois :

L'action en non-conformité (fondée sur l'**article 1184 du Code civil**) peut être invoquée en matière de développement d'application informatique, si le donneur d'ordre a bien renseigné son maître d'œuvre sur les métriques attendues à terme (en concurrence; en volume de données, en nombre de transactions par minutes...) et qu'il résulte de l'exploitation normale, au bout de quelques années que le système connait des temps de réponse anormalement longs. Il est en revanche plus difficile d'utiliser le défaut pour vice caché introduit par l'article 1641 du code civil, car ce dernier introduit des délais pour agir souvent incompatibles avec la découverte du problème.

La **Loi 98-356 du 1er juillet 1998** protège les structures des bases de données au titre du droit d'auteur (cette façon de protéger les auteurs de modèle de données n'est pas nouvelle, et est héritée d'une jurisprudence dite *affaire Didot Bottin* - 18 décembre 1924).

3.2 - Jurisprudence :

L'obligation d'information et de conseil du professionnel envers son client impose que le savoir faire du maître d'œuvre soit mis en avant afin d'apporter tous conseils et informations afin de réaliser au mieux l'application dans les limites techniques connues.
Ce que l'on appelle les "*règles de l'art*", sont pour chaque discipline l'ensemble des éléments techniques sur lesquels les professionnels s'accordent afin de satisfaire le client. Par exemple pour un maçon qui devrait ériger un mur, on serait en droit de demander réparation du préjudice s'il n'était pas d'aplomb ou bien que son faîtage ne soit pas droit, sans qu'il soit besoin de produire le moindre contrat écrit.

3.3 - Règles de l'art :

Il est certain que le viol des règles de l'art par une entreprise professionnelle de l'informatique constitue le fondement juridique d'un préjudice constatable pour lequel on est en droit de réclamer réparation.
Voici une liste de textes fondamentaux applicables aux règles de l'art en matière de développement d'applications informatique utilisant des bases de données relationnelles et du langage SQL :

- Théorie de l'algèbre relationnelle appliquée aux bases de données (travaux de Franck Edgar Codd).
- Articles de vulgarisation de Franck Edgar Codd sur ce qu'est un SGBD relationnel.
- Modélisation de données, notamment travaux de Peter Chen (modèle entité association) et Hubert Tardieu (notation MERISE). Plus récemment notation UML (*voir les ouvrages de Christian Soutou sur l'utilisation d'UML pour modéliser les bases de données*).
- Réalisation des modèles de données : conceptuel (les entités et associations — *le plus important car il donne la sémantique des données*), logique (les relations), physique (les tables), externe (les vues).
- Normes et standards nationales, européennes et internationales. Quelques exemples :
    - ISO /IEC 9075 (1999 à 2008) : bases des données relationnelles et langage SQL.
    - Formatage des adresses postale : norme Européenne N° 14142-1 (2003) : « Services postaux - Bases de données d'adresses - Partie 1 : Composants des adresses postales » Disponible auprès de l'AFNOR.
    - Entités géographiques : Code Officiel Géographique de l'INSEE
    - ...

Ont trouvera quelques éléments d'étude juridique et la jurisprudence à ce sujet dans la webographie en figurant en fin d'articles.

4 - Quelques *success* stories :

Tel éditeur informatique avait raté l'introduction de son logiciel dans l'univers des gros clients par le fait de ne pas avoir étudié les effets d'une volumétrie importante des données sur son système, à cause d'un modèle de données hérité des systèmes à fichiers. Après avoir tenté en vain de contourner le problème et perdu quelques années, cette société nous a confié un audit et a pu rapidement mettre en œuvre les solutions préconisées et réattaquer ce marché.

Telle autre entreprise du web, devant l'accroissement de son volume d'activité, avait prévu la mise en œuvre de trois nouveaux serveurs de bases de données afin d'assurer le fonctionnement pérenne de ses sites web. Ayant senti que quelque chose n'allait pas dans ce couteux scénario, le chef de projet a demandé qu'un audit soit entrepris. Cet audit démontra qu'avec un peu de refactoring de bases de données, la récriture de certaines procédures et une bonne organisation matérielle et administrative du serveur, il n'était pas nécessaire de mettre en œuvre trois nouveaux serveurs, mais que deux serveurs en tout et pour tout suffisaient amplement pour assurer quelques années de fonctionnement ce qui permit une économie drastique en terme d'acquisition de matériel et de licences.

Un éditeur avait été mis en cause par son client par ce que le logiciel répondait de manière aléatoire aux sollicitations des utilisateurs. Un premier audit avait conclus que ni le serveur, ni l'application n'était en cause. Cependant les problèmes persistants, l'éditeur demanda qu'un audit à tous les niveaux soit entrepris. C'est à l'analyse des trames réseau que nous découvrîmes le problème. Une surveillance des bâtiments qui s'effectuait par vidéo sur IP, passait dans les mêmes "tuyaux" que ceux de l'application. Mais la vidéo étant prioritaire (temps réel), elle perturbait parfois fortement les autres communications. Le client de cet éditeur nous demanda par la suite comment résoudre au mieux ce conflit.

Une entreprise française ayant un important site web de service en ligne avait l'intention de se vendre à son concurrent américain. Mais la notoriété de cet acheteur en terme technique, faisait que le vendeur craignait que la qualité technique de la base de données et la faiblesse de l'écriture des requêtes ne soit la cause d'âpres discussions pour en faire baisser le prix. Un audit de base de données fût entrepris, qui permit de démontrer que cette application n'était pas si mal écrite, mais que certaines améliorations pouvaient être entreprises très rapidement avant la vente.

Une petite entreprise de fabrication d'ascenseurs et de portes automatiques avait confié le développement de son "ERP" à l'extérieur mais la montée en charge produisit une application de plus en plus lente. L'audit montra que le style de développement utilisé, particulièrement archaïque, en était la cause. Il s'avérait difficile à remédier sans une refonte complète de la base et du logiciel applicatif. L'entreprise ayant l'intention de changer ce logiciel, on précipita l'introduction du nouvel outil, et pendant la phase intermédiaire on mit en place quelques remèdes comme le redimensionnement du réseau, l'utilisation de serveurs de terminaux et l'achat d'un nouveau serveur de base de données devant servir pour le futur à la nouvelle application.

Un grand compte suisse ayant racheté une entreprise d'un secteur concurrent avait hérité de son informatique. L'ajout d'une nouvelle fonctionnalité à cet applicatif provoqua un basculement global des temps de réponse vers le bas. L'audit révéla que malgré une bonne qualité d'écriture des requêtes et une indexation drastique, quelques oublis importants et des techniques particulières avaient conduit le système à recompiler systématiquement toutes les procédures stockées et par conséquent tous les plans de requêtes et de ce fait, fait perdre 75% de temps en calculs inutiles.

Une compagnie d'aviation avait fait développer un logiciel de planification des équipages mais des temps de réponse erratiques se produisaient de manière aléatoire. L'audit des requêtes clientes montra que se produisait parfois des salves de près d'une centaine de requêtes qui étaient la cause de ces temps de réponses importants. Un passage sur le poste de travail pour corréler ces temps anormaux aux actions entreprises nous montra que ces

derniers se produisaient lorsque l'utilisateur passait sa souris sur l'écran principal de l'application. Dans cet écran (24 pouces) constitué d'une immense grille, chaque cellule permettait d'afficher la composition de l'équipage dans une bulle. Cependant les développeurs ayant oublié de spécifier un délai avant affichage, le parcours de la souris sur l'écran provoquait la rafale de requête constatée. Bien entendu l'éditeur nia cet effet avant de le corriger discrètement dans la version suivante de son logiciel qui arriva peut de temps après que notre audit lui fut envoyé !

Webographie sélective :

Juridique :

- http://www.fit-europe.org/vault/barcelone/StRobert.pdf
- http://www.village-justice.com/articles/devoir-conseil-contrats,3715.html
- http://www.brmavocats.com/fr/gui/guidejur_model.asp?article_id=43
- http://jurisfac.chez.com/prive/civil/p1civil3.htm

Algèbre relationnelle :

- http://www.scribd.com/doc/7034329/2-ALGEBRE-RELATIONNELLE-GRAPHIQUE
- http://cuiwww.unige.ch/~guyot/TPBD1/COURS/D_algebre.pdf
- http://ceria.dauphine.fr/cours98/CoursBD/ALG-REL-97.ppt

Règles de Codd :

- http://www.sqlspot.com/Les-regles-de-CODD-pour-un-SGBD-relationnel.html
- http://fr.wikipedia.org/wiki/Edgar_Frank_Codd

Modèle entité association :

- http://www.informatik.uni-jena.de/dbis/lehre/ws2004/dbs1/Chen.pdf
- http://www.springerlink.com/content/p876324783908g41/

Méthode, Merise :

- http://fr.wikipedia.org/wiki/Merise_(informatique)
- http://www.alapage.com/-/Fiche/Livres/2708124730/la-methode-merise-hubert-tardieu.htm
- http://merise.developpez.com/faq/
- http://sqlpro.developpez.com/cours/modelisation/merise/

Modélisation des données

- http://philippe.guezelou.free.fr/mcd/mcd.htm
- http://books.google.fr/books?id=6ThJUVaps7kC&printsec=frontcover&dq=mod%C3%A9lisation+des+donn%C3%A9es#PPR5,M1

Normes :

- http://www.ncb.ernet.in/education/modules/dbms/SQL99/ansi-iso-9075-2-1999.pdf
- http://www.industrie.gouv.fr/poste/actu/normalisation2.htm
- http://www.insee.fr/fr/methodes/nomenclatures/cog/