ARCHIVAGE DES BASES DE DONNEES

ARNAUD HULSTAERT ET GRÉGORY OGONOWSKI

1. Introduction

La croissance continue des volumes de données stockés dans les bases de données n'est pas sans poser des problèmes : temps de réponse plus longs, difficultés à maintenir les performances, allongement du temps nécessaire pour réaliser les opérations de migration, de sauvegarde, de plan de continuité d'entreprise… Cela alors qu'une grande majorité des données présentes dans les bases de données en production ne sont que peu, voire plus du tout, utilisées (Figure 1).

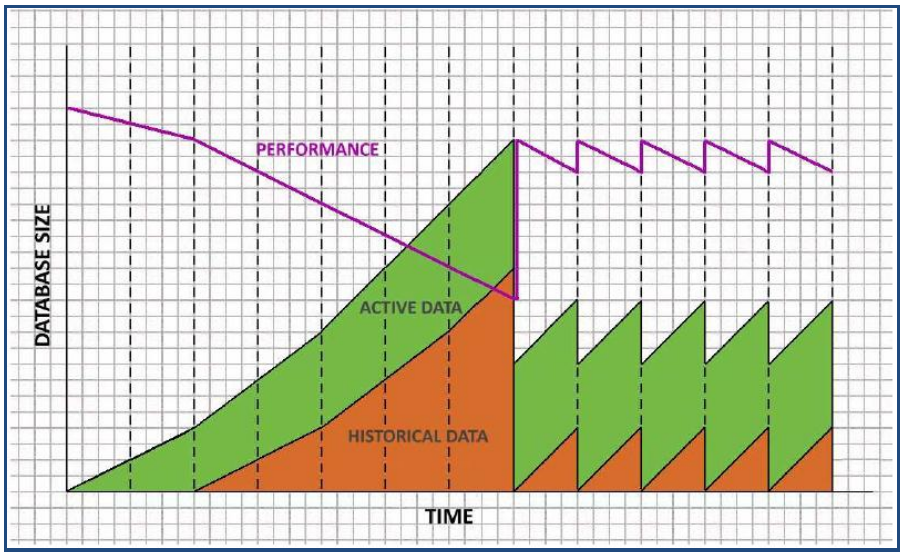


Figure 1 : Évolution dans le temps du volume des données dans une base de données et des performances de celle-ci (source : IBM)

De plus, il n'est pas rare de devoir maintenir des applications obsolètes en production uniquement pour pouvoir accéder aux données, ce qui engendre des coûts importants en termes de maintenance (licence, compétences requises…).[1]

[1] « Conserver et consolider des applications obsolètes peut s'avérer très compliqué. Dans les fais, des coûts importants sont souvent nécessaires pour maintenir des applications non utilisées simplement pour s'assurer un accès aux données qui pourraient être ou ne

Enfin, de nombreuses données doivent être conservées sur des périodes plus ou moins longues pour des raisons légales et réglementaires ou pour se protéger contre tout risque juridique, ce qui implique de pouvoir démontrer leur intégrité et leur authenticité. À cet égard, conserver les données dans une base de données en production n'apporte aucune garantie.

Face à cette situation, l'archivage des données contenues dans les bases de données apporte une réelle valeur ajoutée et des éléments de réponse à ces différents défis. Il s'agit d'exporter périodiquement ou ponctuellement les données qui ne sont plus régulièrement utilisées en vue de les mettre en sécurité dans un système d'archivage électronique. La base de données de production est ensuite purgée des éléments archivés.

Dans le cadre de cette étude, nous retiendrons la définition suivante :

« **Archiver** consiste à prendre un objet et à le transférer sous certaines conditions dans un système qui permettra d'en assurer la préservation pendant un certain laps de temps avec toute la sécurité requise ».[2] Ce qui implique les actions suivantes :

- sélection de l'information ;
- transfert dans un autre système pour en assurer la sécurité (gestion de l'intégrité et de l'authenticité) ;
- préservation de l'information, c'est-à-dire tant la couche physique que la couche logique et sémantique ;
- gestion de la durée de conservation de l'information.

Il s'agit donc d'une définition et d'une acceptation plus larges (issue du domaine du « records management ») que le transfert de vieilles données « à la cave », notion encore largement répandue et reprise sous le terme anglais « archiving ». Nous verrons que cette définition aura son importance lors de la confrontation de notre démarche aux solutions commerciales existantes.

Les risques de ce type de projet sont aujourd'hui bien identifiés : l'obsolescence du matériel, la disparition des logiciels de lecture, la disparition des formats de fichier et la perte de la signification de l'information. Face à cela, les bonnes pratiques sont connues : copies multiples sur différents types de supports, veille sur les technologies existantes, sélection des formats de fichiers avant archivage, utilisation de métadonnées pour documenter les informations archivées... Le défi actuel est de les mettre en œuvre dans une approche cohérente et intégrée.

---

jamais être nécessaires. » S. CHILDS, *Use Database Archiving to Preserve Data When Retiring Applications*, Gartner Research, 2 mars 2010.

[2] M.-A. CHABIN, *Moreq2 et archivage sécurisé*, Fédération Nationale des Tiers de Confiance, 2009, p. 6.

2. Normes et recommandations

De nombreuses normes et recommandations ont été publiées ces dernières années, de profil, de précision et de qualité variables. La plupart d'entre elles sont cependant d'une grande utilité et apportent une réelle valeur ajoutée pour toute personne impliquée dans un projet d'archivage en ce qu'elles exposent, encadrent et balisent la problématique et apportent des solutions concrètes.

Cette abondance entraîne toutefois une certaine confusion qui les rend difficiles à appréhender. Par souci de facilité, ces normes et recommandations peuvent être classées en quatre grandes catégories comme l'illustre la Figure 2.

À l'heure actuelle, les trois normes et recommandations principales sont :

- **ISO 15489** : il s'agit de la norme de référence en matière de records management dans le monde. Elle expose la démarche d'archivage et les outils à mettre en place pour ce faire. Elle fixe le cadre général.
- **ISO 14721** : plus connue sous le nom de modèle OAIS (Open Archival Information System), il s'agit d'un modèle conceptuel qui expose les problèmes de l'archivage électronique, les fonctionnalités attendues d'un tel système et les informations complémentaires (métadonnées) à attacher à un objet archivé en vue d'assurer sa pérennité. Elle propose deux référents (un modèle fonctionnel et un modèle d'information), mais ne spécifie aucune implémentation.
- **MoReq2010** : recommandation très intéressante publiée en 2011 sous le sponsor de la Commission européenne, elle définit les fonctionnalités d'un système d'archivage électronique et un modèle de données associé.

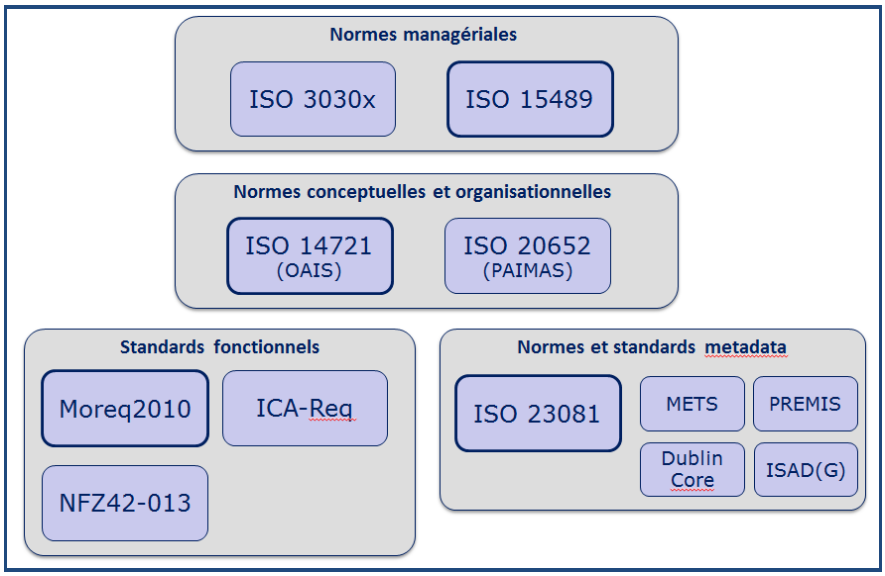


Figure 2 : Classification des normes et des recommandations dans le domaine de l'archivage électronique

3. Archivage d'une base de données

Enjeux

Une **donnée** est un triplet composé d'un intitulé renvoyant à un concept (un salaire brut, par exemple), d'un domaine de définition (spécifiant l'ensemble des valeurs admises pour ce concept) et d'une valeur à un instant *t*.

Une **base de données** est « *a collection of related data with a logically coherent structure, some inherent meaning, a specific purpose, a largely varying size, a scope or content of varying breadth, a physical organization of varying complexity and a persistence over a long period of time* ».[3]

Il existe de nombreux types de bases de données : relationnel, hiérarchique, objet, XML, NoSQL… La démarche et les solutions conceptuelles proposées dans l'étude sont indépendantes du type de base de données. Toutefois, les solutions sur le marché sont exclusivement destinées au modèle relationnel et XML (dans une moindre mesure).

L'enjeu est donc d'archiver les données issues d'une base de données en prenant soin d'en préserver la signification, d'en capturer tous les éléments pour être certain que les données seront, après archivage, toujours utilisables. Ceci est d'autant plus difficile que les données à archiver ont une structure complexe (certaines données interdépendantes pouvant être réparties sur plusieurs tables, voire plusieurs bases de données) et que le volume des données peut être très important. Enfin, l'extraction des données ne doit pas mettre en péril la consistance de la base de données de production.

Dans les bases de données administratives, les données sont parfois soumises à une valeur probante, puisqu'elles sont la preuve d'un fait correspondant du réel à un moment *t*. Par conséquent, les valeurs correspondantes doivent être conservées, y compris en cas de modifications pour des raisons administratives ou de contrôle. Afin de capturer ces modifications, il est nécessaire de mettre en place un historique des versions via un design adéquat du schéma de la base. À cet égard, nous renvoyons aux travaux d'Isabelle Boydens[4], responsable du centre de compétence Data Quality au sein de la section Recherche de Smals, qui représentent à l'heure actuelle les réflexions les plus avancées sur le sujet. Dans tous les cas, ce schéma doit être prévu en amont de

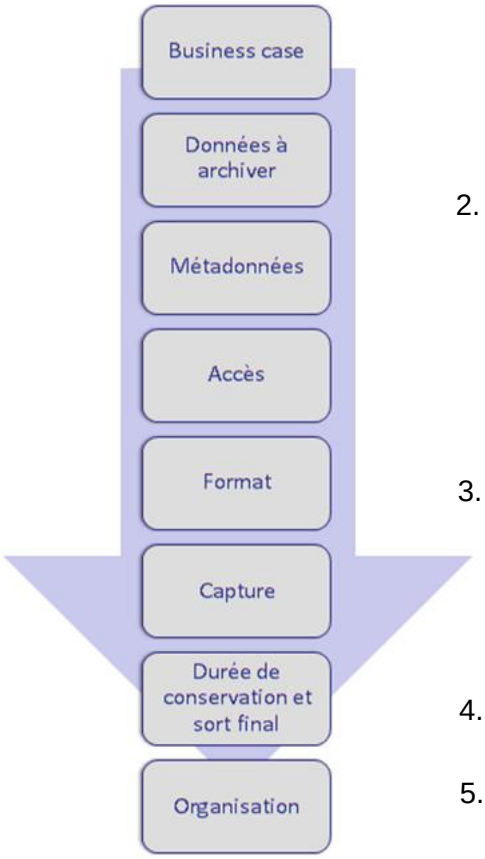
[3] Basé sur R. ELMASRI, S. NAVATHE, *Fundamentals of database systems*, 6e édition, Adison-Wesley, Boston, 2007.

[4] Voir e.a. I. BOYDENS, A. HULSTAERT et D. VAN DROMME, *Gestion intégrée des anomalies. Evaluer et améliorer la qualité des données*, Section Recherche, Smals, 2011 (lien).

l'application, dès la conception de la base de données. On ne le répétera donc jamais assez : l'archivage doit se penser dès le début d'un projet.

Méthodologie

L'archivage des données nécessite de suivre une méthodologie rigoureuse :



1. **Définir le business case** : pour quelles raisons souhaite-t-on archiver les données ? Pour des raisons légales ? Pour information ? Pour augmenter les performances des applications en production ? Les réponses à ces questions auront naturellement une influence sur la solution à mettre en œuvre.
2. **Sélectionner les données à archiver** sur la base des obligations légales, des besoins en consultation et en utilisation des données. Cette étape consiste à définir des objets métier autonomes, par la définition d'une table racine et le suivi des relations. C'est un travail important et délicat. On prendra également soin de déterminer la valeur des données (probantes vs informatives) ainsi que les risques liés à leur non-conservation.
3. **Sélectionner les métadonnées**, c'est-à-dire les informations complémentaires de description, de structure et de préservation qu'il faut archiver avec les données en vue d'assurer leur utilisation logique, syntaxique et sémantique dans le futur. Conserver des données inutilisables à terme est coûteux et inutile.
4. **Définir les accès**, à savoir qui peut accéder aux données, selon quelle fréquence, à quelles données.
5. **Définir un format d'archivage** : la durée de vie des données est largement supérieure à la durée de vie des applications, en ce compris les systèmes d'archivage. Il faut donc sélectionner un format dit pérenne, afin de réduire autant que possible les migrations de format (qui seront de toute façon un jour inévitable). Dans la grande majorité des cas, il s'agira de XML, CSV ou flat file, plus rarement de dump SQL.
6. **Élaborer la capture des données**, ce qui inclut aussi bien la définition d'une périodicité d'extraction que le processus et les critères de sélection.
7. **Définir les durées de conservation et le sort final** au terme de celles-ci : on n'archive que rarement *ad vitam aeternam*. Une durée temporelle ainsi qu'un évènement déclencheur de cette durée doivent être définis.
8. Enfin, **mettre en place une organisation** permettant de réaliser les travaux d'analyse et le fonctionnement du système d'archivage électronique. Il est bon de rappeler que toute destruction d'une archive publique (ce qui inclut aussi bien les documents que les

données) est soumise à l'approbation de l'Archiviste général du royaume ou de ses délégués conformément à la loi du 24 juin 1955 et du 6 mai 2009. Il est donc recommandé de les impliquer dès le départ.

4. Système d'archivage électronique

Un **système d'archivage électronique** est un « *ensemble des matériels, logiciels et procédures qui organise et contrôle la capture, la conservation et la destruction (…) des objets* »[5].

Familles

Après examen des systèmes d'archivage existant actuellement sur le marché, il est possible de distinguer différentes familles :

- Au niveau de la **forme d'archivage** : certains systèmes vont archiver les données sous la forme d'une autre base de données, de même type que la base de données de production ou non, tandis que d'autres vont archiver les données sous la forme de fichiers, eux-mêmes référencés à l'aide de métadonnées. Si la forme « base de données » peut présenter des avantages, notamment en termes d'accès et de consultation, elle ne représente pas une solution à long terme. La conservation des données sous la forme de fichiers est donc vivement recommandée à long terme.
- Au niveau du **processus de capture** : certaines solutions (dites PULL) se connectent à la base de données source en vue d'extraire elles-mêmes les données sur la base des paramètres introduits. Ces solutions proposent des fonctionnalités avancées de data profiling, d'extraction… Les autres systèmes (dits PUSH) sont plus passifs : les données sont extraites de la base de données de production et poussées vers la solution d'archivage (on parle de versement). Les systèmes PULL offrent une aide non négligeable aux utilisateurs mais posent néanmoins des questions de sécurité (puisque la solution d'archivage doit disposer de droits de suppression dans la base de données de production) et gèrent rarement les questions d'intégrité (cf. point suivant).
- Au niveau de **la gestion de l'intégrité** : certains systèmes font de la gestion et du contrôle de l'intégrité des données une fonctionnalité au cœur de la solution, tandis que d'autres laissent ce soin à un outil tiers (que ce soit au niveau software ou hardware).

[5] M.-A. CHABIN, *Nouveau glossaire de l'archivage*, février 2010.

Sécurité et architecture

Comme nous l'avons déjà évoqué, un projet d'archivage est avant tout un projet de sécurité. Reste à définir le bon niveau de sécurité eu égard à la valeur des données et des risques en cas de mauvaise conservation.

Et sur cette base l'architecture adéquate. Le risque zéro n'existant pas, il s'agira de scinder les fonctionnalités entre différents modules fonctionnels dont la responsabilité organisationnelle est confiée à des acteurs différents, comme la gestion des supports de stockage, l'administration de la base de données, l'administration des logs…

Accès

L'accès à des données archivées est évidemment plus facile dans le cas d'un archivage sous la forme de base de données mais, comme nous l'avons déjà mentionné, il ne s'agit pas d'une solution à long terme. Accéder à des données archivées sous la forme de fichiers est toutefois plus difficile. Soit l'outil fournit des IHM plus ou moins avancées, soit, si un accès à partir d'une application est requis, une couche intermédiaire dite « Unified Data Access » est nécessaire. Celle-ci prend en charge les requêtes (SQL) en provenance de l'application d'origine ou d'un outil tiers et gère l'interrogation des données archivées. Il s'agit souvent de transformer des requêtes SQL en provenance de l'application en requêtes XQuery.

5. Market overview

Dans le cadre de cette étude, quatre solutions ont été examinées et confrontées aux différentes caractéristiques identifiées comme importantes pour ce type de solution (Figure 3) :

- IBM Optim Data Growth (logiciel de type PULL)
- Arcsys Software (logiciel de type PUSH essentiellement)
- Une combinaison d'HP AIO et HP TRIM
- Informatica Data Archive (logiciel de type PULL)

	IBM Optim	Arcsys	HP AIO-TRIM	Informatica
Scope	Performance	Intégrité	AIO = performance TRIM = intégrité	Performance
Type de capture	PULL	PUSH-PULL	AIO = PULL et PUSH vers TRIM	PULL
Extraction	+++	+	+++ (AIO)	+++
Forme d'archivage	File	File	DB / File	DB / File
Intégrité	Hors scope	+++	Scope TRIM	Hors scope
Gestion cycle de vie	Hors scope	+++	Scope TRIM	Hors scope
Consultation	Via Optim et ODBC/JDBC	Via Arcsys ou outil tiers	Via HP et ODBC/JDBC	Via Informatica et ODBC/JDBC
Format des archives	Propriétaire et fermé	Format ouvert (TAR.GZ + XML)	Format ouvert (XML)	Propriétaire et fermé

Figure 3 : Évaluation des solutions d'archivage analysées

L'examen du marché et des quatre solutions permet de mettre en évidence les points suivants :

- Les solutions positionnées sur le marché du « Database Archiving » sont toutes de type PULL. Ce positionnement est en cohérence avec la définition du terme anglais « **archiving** » qui consiste à déplacer des données moins utilisées ou « obsolètes » vers un espace tiers afin d'optimiser les applications en production. Ces solutions ne traitent quasiment jamais de l'intégrité des données archivées. Par conséquent ces solutions ne résolvent pas totalement la problématique de l'archivage selon la définition que nous y avons donnée.

  Ces solutions présentent toutefois des fonctionnalités avancées pour l'extraction des données.

- Les solutions de type PUSH se situent davantage sur le marché du **records management**. Dans ce cas, l'intégrité fait partie intégrante des solutions, mais elles disposent de fonctionnalités de gestion du cycle de vie.

- Pour une couverture fonctionnelle complète de la définition de l'archivage que nous avons proposée, deux outils seront donc nécessaires, quoique la partie extraction puisse être exécutée manuellement, c'est-à-dire par des administrateurs de la base à l'aide de requêtes SQL.

- La méthode PULL est transactionnelle, ce qui correspond davantage à la manière de travailler dans le monde des bases de données. La transaction est terminée quand les données sont archivées, alors que dans le cas de la méthode PUSH, la transaction se terminerait par le dépôt des données sur un file system où la solution d'archivage les capture. Par conséquent, la méthode PUSH ne permet pas une transaction unique positionnant

d'abord l'archivage effectif des données et ensuite la suppression desdites données.

- Les fonctionnalités d'extraction sont uniquement disponibles pour les bases de données relationnelles. Aucun fournisseur n'a de connecteurs vers des bases de données non relationnelles, même ceux qui ont des relations historiques avec ce type de base de données.
- La consultation des données archivées issues d'un DBMS est plus mûre dans le cas des outils PULL que PUSH. Ces solutions proposent généralement des fonctionnalités d'accès soit via IHM, soit via des connecteurs ODBC/JDBC (ce qui autorise les accès applicatifs). Dans le cas des outils PUSH, des fonctionnalités de consultation et d'accès sont possibles, mais elles sont plus génériques et tiennent donc moins compte des spécificités des données issues d'un DBMS.
- Enfin, plusieurs solutions archivent les données dans un format ouvert et documenté (CSV, XML, containeur tar.gz), ce qui est un atout pour un archivage pérenne. Les solutions proposant des formats propriétaires ne sont donc pas à privilégier.