Giuseppe Berio, Michel Dubois

INF2204

« Systèmes d'information décisionnels et entrepôts de données »
Version préliminaire (des ajustements sont encore possibles)

Sujet d'étude

Le travail demandé porte sur un cas pratique utilisant en support les outils suivants : (Talend Open Studio for Data Integration v 8.0 – TALEND par la suite, ou Pentaho Data Integration – PDI par la suite) / MySQL / Serveur ROLAP Mondrian / Power BI -PBI par la suite. Deux architectures distinctes sont à réaliser : une « simple » où le composant de reporting accède directement à un entrepôt de données réalisé sous MySQL ; l'autre plus aboutie, utilisant un « serveur ROLAP » Independent (Mondrian) s'interposant entre le composant de reporting et l'entrepôt MySQL (cette dernière architecture est particulièrement utile car MySQL ne dispose pas de mécanismes d'optimisation pouvant réutiliser les précalcules des mesures et, donc, le « serveur ROLAP » pourra prendre le relais pour rendre disponible les mécanismes d'optimisation nécessaires).

Cas pratique

La PME MoreMovies a récemment décidé d'acquérir trois magasins spécialisés dans la vente et la location de films et de produits dérivés. Auparavant, ces trois magasins avaient trois propriétaires différents et s'appelaient respectivement MovieMegaMart, BuckBoaster et MetroStarlet. Dans chacun de ces magasins, les clients sont identifiés par un code enregistré sur une carte magnétique personnelle. Seuls les clients de ces magasins peuvent y acheter ou louer les produits. De plus, ces magasins gèrent les locations et les ventes à travers une identification électronique des produits. Pour MoreMovies, il devient difficile de traiter les données provenant de chaque magasin car chacun de ceux-ci dispose de son propre système d'information, qui lui est spécifique. MoreMovies souhaite intégrer les données provenant des 3 magasins dans un système unique (un entrepôt de données), de manière à effectuer des études des ventes ou locations suivant les produits proposés, les magasins, les clients (homme, femme, tranche d'âge, ...). MoreMovies n'arrive pas à faire des prévisions fiables sur la base de ces données comme, par exemple, évaluer la possibilité d'étendre à l'ensemble des magasins la vente de gadgets ou encore proposer des livres concernant les films ou les secteurs de cinéma. On aimerait par exemple pouvoir faire des analyses fiables concernant les ventes et locations de films, ce qui n'est pas facilement réalisable dans l'état actuel du système.
Il vous est demandé de réaliser les étapes pour la construction d'un entrepôt pour MoreMovies et pour réaliser du reporting. Les sources de données sont des bases de données relationnelles ACCESS représentant les **états observés** de chaque système d'information à un instant. Aucun document expliquant ces bases n'est

disponible (dans la réalité les documents sont peut-être disponibles mais pas fiables). Il est à préciser que les objectifs de MoreMovies ne sont pas forcément atteignables mais ils peuvent l'être partiellement.

Étapes du travail à réaliser

Vous analyserez les schémas logiques de chaque source (comprenant seulement des tables) et les données mémorisées pour :

1. Produire **pour chaque source un schéma conceptuel** (par exemple un diagramme de classes UML ou un diagramme type ER) faisant des transformations de schéma d'enrichissement, de normalisation, d'abstraction, de spécialisation, de généralisation des schémas sources ; vérifier que le schéma conceptuel proposé est « faisable » par rapport aux données disponibles, utilisant des **requêtes appropriées** ; indiquer des **mesures de la qualité des données** telles que : pourcentage de données manquantes, pourcentage de données aberrantes, etc.... ; les mesures de qualité peuvent être utiles pour la reconceptualisation des sources ; ces mesures seront particulièrement utiles pour mieux conceptualiser et programmer l'échange de données (flux d'extraction) entre les sources et l'environnement ETL où les sources seront réimplantées.
2. Réaliser pour chaque schéma conceptuel correspondant à une source, un **schéma logique** (relationnel) ; en fonction de l'outil choisi au point 1, cela peut se faire d'une manière semi-automatique.
3. Conceptualiser les **flux d'échanges**, précisant des **mappings** entres chaque schéma « tel quel » d'une source et le schéma de la source reconceptualisé (produit au points 1 et 2). Ces mappings, de principe, fournissent un cadre pour réaliser les flux d'extraction et peuvent prendre la forme de tableau à 2 colonnes : intitulé de la colonne dans la source, intitulé de la colonne dans le schéma (logique) de la reconceptualisation.
4. Produire **un schéma conceptuel intégré** (visualisé comme diagramme de classes UML ou type ER) à l'aide de **correspondances inter-schémas** (vous pouvez aussi explorer les capacités de SQL Developer pour outiller l'intégration de schémas via un dictionnaire).
5. Produire un **schéma intégré logique** (relationnel) à partir du schéma conceptuel intégré défini au point 1 ; en fonction de l'outil choisi au point 1, cela peut se faire d'une manière semi-automatique.
6. Conceptualiser les **flux d'intégration**, précisant des **mappings** entres chaque schéma reconceptualisé des sources et le schéma intégré (produit au point 4 et 5).
7. Proposer un **schéma multidimensionnel conceptuel basé sur le schéma conceptuel intégré**, avec les dimensions {*produit, magasin, période, client*} et indicateurs d'intérêt (*CA, nombredeclients, nombreventes, nombredelocations, nombremoyenlocation, nombremoyenvente, CAmoyen*) sur les ventes et les locations réalisées ; les hiérarchies, doivent s'organiser en 2 ou 3 niveaux ; la dimension *produit* devra nécessairement distinguer entre *type* de produit (film, gadget etc .) ; la dimension *client* devra nécessairement comporter l'information sur l'*âge* des clients ; suivez

l'approche « supply driven » pour le concevoir, vous basant sur le schéma intégré produit au point 3.

   a. **Rappel.** Un schéma dimensionnel conceptuel décrit un ensemble de mesures (réunies dans un – ou plusieurs - « cube » -s- en terminologie SQL Developer), et chaque niveau des hiérarchies portées par chaque dimension selon laquelle ces mesures peuvent être obtenues (faits inférés),
   b. Vous pouvez dessiner ce schéma sans utiliser des outils dédiés ou avec des outils dédiés (par exemple, SQLDeveloper) sachant que ces outils dédiés ne serviront qu'à garantir la cohérence entre le schéma dimensionnel et le schéma rolap mais il sera nécessaire de « reporter manuellement » ces schémas suivant les consignes propres aux environnements technologiques. Soyez par contre assez précis sur l'additivité, et les 2 classes de mesure (calculée, dérivée) car fort probablement utile par la suite du projet.

8. Produire un **schéma logique rolap** suivant le schéma dimensionnel conceptuel ; compléter le avec des choix de **modélisation physique tels que des index, de partitions** (en fonction de ce qui est disponible dans les environnements technologiques utilisés) et 3 tables d'agrégats que vous aurez retenues. Pour les tables d'agrégats (stockant donc certains pré-calculs des mesures du schéma multidimensionnel), vous pouvez juste les indiquer (comme annotation au schéma rolap) ou utiliser des vues matérialisées au sens ORACLE, si le schéma rolap est disponible en SQLDeveloper (sachant que ces vues ne seront pas directement utilisables mais vous permettrons une définition précise des agrégats à utiliser ailleurs dans ce projet).
   a. **Rappel.** Un schéma logique rolap est en effet le schéma proposant une (ou plusieurs) table(s) de faits liée(s) aux tables de dimension via le mécanisme classique de la clé étrangère, et
   b. Le schéma logique rolap peut avoir 2 formes de base : **étoile et flocon**. Il faudra donc décider si le schéma proposé sera étoile ou flocon. Bien entendu, le schéma peut assumer la forme en **constellation** en cas de plusieurs tables de faits.
   c. La clé de la table de faits (ou des tables de faits) doit avoir été justifiée dans le livrable du projet.

Précisions sur les outils de modélisation utilisables : Vous pouvez utiliser tout outil permettant la modélisation (par exemple, Visual Paradigm pour une modélisation centrée UML) ou utiliser des représentations visuelles, sachant qu'en fonction de l'outil choisi, certains choses seront à faire manuellement. Vous pouvez aussi faire communiquer plusieurs outils si vous pensez cela utile. Par exemple, on peut créer des échanges entre Visual Paradigm et SQL Developer :

- Création d'un diagramme de classes (conceptuel), génération automatique du code pour Oracle, importation du code Oracle dans SQL Developer, visualisation des tables et visualisation du modèle type ER pour concevoir un modèle multidimensionnel par exemple ;
- Vice-versa, une fois généré le code ORACLE à partir de SQL Developer, il est possible de le visualiser en diagramme type ER au sein de Visual Paradigm (onglet *Outils* puis « *DB/Revenir à la DDL précédente* ») pour

ensuite générer automatiquement le code pour MySQL (après avoir rajouté par exemple des index)

Pour rappel, la liaison entre SQL Developer et MySQL pour la partie dimensionnelle n'est pas fonctionnelle car MySQL ne possède pas l'instruction « create dimension » comme ORACLE ; par conséquent, un schéma multidimensionnel conçu en SQL Developer ne sera utile que pour réaliser le schéma logique rolap contenant les tables (et à créer du code à partir de ce schéma logique rolap pour ensuite retirer/adapter les parties inutilisables au sein de MySQL si vous n'avez pas utilisé l'échange avec Visual Paradigm). Par ailleurs, la modèle physique representable en SQL Developer pour le schéma ROLAP n'est pas non plus utilisable tel quel pour coder en MySQL car les index, les partitions ne s'écrivent pas de la même manière qu'en ORACLE et les vues matérialisées ne sont pas disponibles.

Sous MySQL/Mondrian/PBI/ETL :

9. MySQL : Sous MySQL, réaliser toutes les bases de données correspondantes aux schémas logiques conçus aux points 2 et 5 suivant les consignés appropriées. Pour cela vous pouvez vous aider par le code automatiquement généré par les outils de modélisation.
10. ETL : Mettre en œuvre les flux d'extraction suivant les mappings conceptualisés au point 3 prenant en compte les nettoyages et standardisations nécessaires à l'aide de l'ETL choisi (TALEND, PDI). Les flux d'extraction devront prendre en compte les doublons éventuels existants dans certaines tables de chaque source (attention : la suppression des doublons demande la génération d'une nouvelle clé pour les données extraites). Alimenter ensuite les bases de données cibles des flux d'extraction. Après avoir toutes les erreurs éventuelles, évaluer le résultat faisant une analyse rapide du contenu des bases cibles. Si, suite à cette analyse, vous n'êtes par satisfaits du contenu, modifiez les flux. Faites en sorte que vos flux soient « re-exécutables » sans fin.
11. ETL : Ensuite, à l'aide de l'ETL choisi, réaliser les flux d'intégration de données, conceptualisés au point 6, entre les bases contenant les données extraites de chaque source et la base mise en œuvre au point 9 et correspondante au schéma intégré. Alimenter cette base exécutant les flux d'intégration. Procéder comme pour le point 10 pour accepter ou modifier les flux d'intégration.
12. MySQL : Réaliser le schéma logique rolap et le modèle physique conçus au point 8 suivant les consignes appropriées (pour le modèle physique, il faudra retrouver au sein de MySQL si possible, les mêmes mécanismes d'optimisation indiqués dans la première partie de ce document (partitions + index) ; si des mécanismes ne seraient pas utiles ou disponibles il faudra reporter les justifications techniques dans le rapport final).
13. ETL : Mettre en œuvre les flux de chargement de l'entrepôt (réalisé au point 9) utilisant les données intégrées (base de données correspondante au schéma intégré). Alimenter ensuite les tables de l'entrepôt à partir de cette base à l'aide de l'ETL choisi. L'entrepôt sera ainsi alimenté.

14. Mondrian : Créer une solution Mondrian ROLAP la plus aboutie, prenant en compte des tables agrégées (la distribution du serveur Mondrian, éventuellement compatible avec l'environnement serveur de la DSI de l'UBS, utilisée sera xmondrian). Pour la définition la plus étendue du cube ou des cubes dans le schéma Mondrian, vous pouvez vous aider de Mondrian Schema Workbench (voire de l'éditeur de schéma Mondrian de SQL Power Architect) et de Pentaho Aggregation Designer, s'inspirant du schéma multidimensionnel de l'entrepôt conçu au point 7. Pour rappel, les tables des agrégats permettent l'optimisation des requêtes via la réécriture (similaire aux mécanismes des vues matérialisées d'ORACLE).
15. PBI ou Mondrian/Excel : Se basant sur l'entrepôt alimenté (point 13), réaliser un **rapport** incluant les items suivants :

    Le 5 films représentant les plus forts montants de ventes (CA) mensuelles chaque année ?

    Par produit, le montant mensuel des ventes depuis toujours
    L'âge moyen des clients (femmes, hommes) qui louent des films

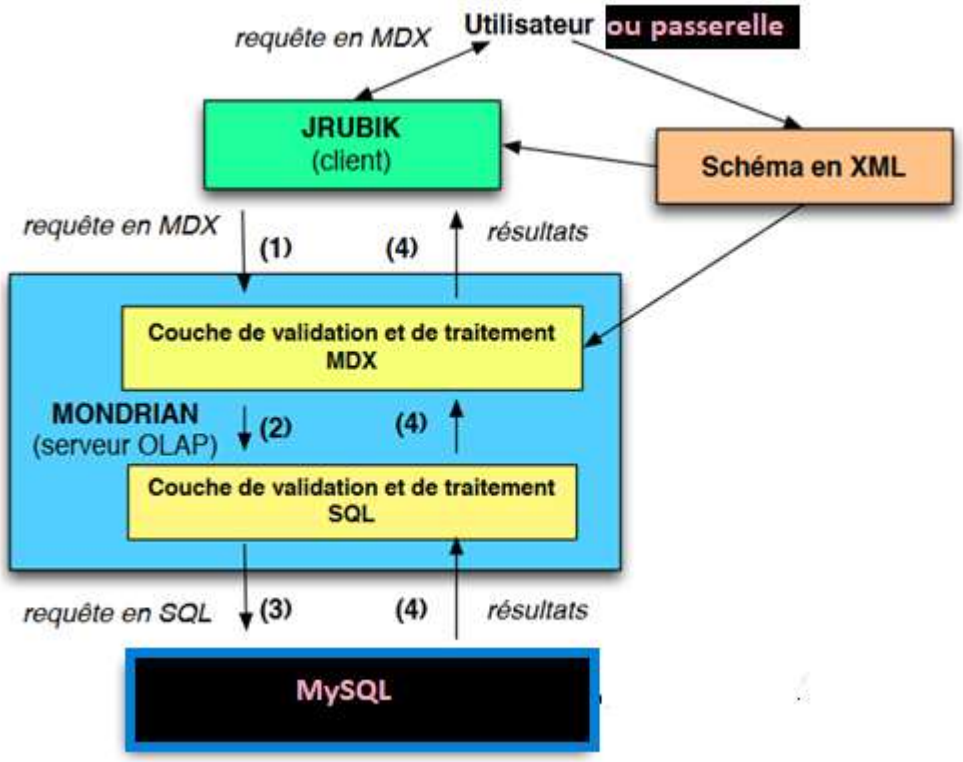
    Par magasin, les films les plus loués par mois et par année de location ?

    En nombre de locations d'un film par mois, les mois pour lesquels une baisse de ce nombre de plus de 15% par rapport au mois précédent est constatée

Les choix visuels (couleurs, formes, légendes, annotations) sont libres mais doivent être justifiés.

Pour mettre en place ce rapport, vous pouvez spécifier des requêtes choisissant parmi 2 différentes approches :

a. Architecture 1 (via serveur ROLAP) : Un client lourd graphique XML/A (MS Excel+XMLA Connect, JRubik en mode XMLA, SQL Power wabit connecté au serveur externe Mondrian) ou un client léger graphique XML/A (xavier) ou un client léger console XML/A (xmla4js, XMLAsh) permettant de spécifier des requêtes XMLA ou MDX (JRubik en mode Mondrian, la fonctionnalité MDX Query de Mondrian Schema Workbench, une fois connecté à MySQL et la définition du modèle Mondrian complète, SQL Power Wabit connecté à son serveur Mondrian, le schema Mondrian étant importé). Ces requêteurs XML/A ou MDX offrent au moins une représentation en tableau voire des graphiques. Vous utiliserez les fonctionnalités offertes par ces outils pour répondre aux questions.



Architecture avec Mondrian comme serveur ROLAP

b. Architecture 2 (connexion directe du composant de reporting à l'entrepôt) : PowerBI Desktop connecté à MySQL via ODBC (la connexion directe ne marche pas), permettant d'interroger les tables constituant l'entrepôt réalisé au point 13. Lors de l'utilisation de PBI, il faudra mettre en œuvre toute optimisation (coté PBI) permettant la réutilisation d'agrégats car dans la solution MySQL cette optimisation ne peut pas être réalisée au sein de l'entrepôt. La connexion PowerBI au serveur XML/A autre que SSAS multi-dimensionnel (solution propriétaire de Microsoft) nécessite une licence premium et donc ne peut pas être expérimentée actuellement. Vous confectionnerez un tableau de bord pour répondre aux questions.

Outils et consignes

Les étapes 1, 2, 3, 4, 5, 6, 7 et 8 doivent être décrites précisément et complètement (diagrammes UML/ER, autres visualisations utilisées, correspondances éventuelles, démarche suivie pour intégrer et nettoyer les données, conceptualisation des flux). Ces points doivent être abordés en séquence stricte. Les points 9,10, 11, 12, 13, 14 et 15 **doivent se baser exclusivement** sur MySQL / (TALEND ou PDI)/Mondrian/PBI. Toute autre solution technologique ne sera pas acceptée (même si répondant aux besoins). Ces points doivent être abordés en séquence stricte pour éviter tout problème.

Livrables du projet

1. **Document d'étape (Livrable 1) :**

   Les points 1, 2, 3, 4, 5, 6, 7 et 8 feront notamment l'objet d'une étude décrite dans un **document d'étape techniquement précis, complet et détaillé.** Ce document est à déposer en format pdf sur l'espace de cours, **pour le 15/4/2024 (**au plus tard).

2. **Rapport final et présentation (Livrable 2) :**
   a. L'ensemble du travail réalisé fera l'objet d'un **rapport final complet** (reprenant le contenu du document d'étape, éventuellement amélioré) techniquement précis et détaillé, à remettre **pour le 23/5/2024 à 23h** au plus tard, en document pdf à déposer sur l'espace de cours ;
   b. D'une présentation orale supportée par un **document de présentation** dans la semaine des contrôles. La présentation sera faite lors de la soutenance qui aura lieu le **29 ou 30 mai (matin jusqu'à 13h30) 2024. Attention : tout créneau publié est donné à titre indicatif et votre disponibilité sur la journée entière ou demi-journée est implicite.**
   Une démo d'un flux complet/reporting pourra/devra être réalisée.

Accès à MySQL

Un serveur local vous est fourni dans une archive à déployer sur les postes des salles informatique. Vous pouvez également installer MySQL sur votre ordinateur pour pouvoir restaurer le serveur local.

Accès à PowerBI

Chaque poste des salles informatique du campus de Tohannic (ENSiBS et Yves Coppens) dispose d'une distribution récente de Microsoft Power BI Desktop. PowerBI peut être obtenu gratuitement.

Accès à TALEND

Talend Open Studio for Data Integration 8.0 (les versions précédentes sont utilisables mais moins stables) est disponible sur les postes UBS. Étant un outil gratuit il peut être installé sur vos propres ordinateurs. La seule contrainte est liée à la disponibilité de JavaRE>=11.

Accès à Pentaho Data Integration

PDI 9.4 vous est fourni dans une archive à déployer sur les postes des salles informatique. La même archive peut être déployée sur votre ordinateur.

Formation des groupes de projet

Le travail à accomplir doit être fait un groupe. Il est conseillé d'utiliser TEAMS comme outil de communication et de coordination projet. Pour des questions d'organisation, il est obligatoire de composer **des trinômes (un maximum de 2 binômes est accepté, 1 binôme par promotion mais vous pouvez aussi former des trinômes mixant les 2 promotions). Il ne sera pas possible de conduire**

un travail individuel dans tous les cas. En cas de dépassement des seuils indiqués ci-avant, les responsables choisiront au hasard les binômes et des trinômes seront reformés automatiquement. Il est donc vivement conseillé, d'établir une liste binômes/trinômes pour tous les étudiants et faire des choix par vous-mêmes. Les étudiants n'appartenant à aucun trinôme (binôme) seront automatiquement affectés à un binôme existant ou seront inclus dans un nouveau trinôme/binôme.

La date limite pour former vous-mêmes les groupes de projet est le **11/3/2024.** Ensuite, les enseignants formeront les groupes projet. Pour former un trinôme/binôme il est suffisant d'envoyer un mail au plus tard le 11 mars 2023 à giuseppe.berio@univ-ubs.fr.

Il est demandé de produire un **diagramme de GANTT** pour montrer le planning de projet et le travail individuel ; ce diagramme doit être inclut dans le Livrable 1 en prévisionnel et Livrable 2 en réalisé.

Calcul de la note de projet

La note de projet est une note individuelle calculée sur la base de plusieurs notes. Ce calcul est basé sur le principe que si vous n'obtenez pas des résultats satisfaisants aux épreuves individuelles, votre contribution au projet est insatisfaisante, peu importe les livrables rendus, étant donné une faible maitrise de concepts et techniques nécessaires au projet.

NI1 : note individuelle (GBerio) /20 (si absent NI1=0)

NI2 : note individuelle (MDubois) /20 (si absent NI2=0)

NI : note individuelle intermédiaire /10 = (NI1+NI2)/4

Note : note individuelle calculée /20

NP : note des 2 livrables / 20 (si non déposé(s) NP=0) – Veuillez noter que cette note est une note par groupe de projet, sauf si contrairement indiqué par les membres de groupe ou par évidence contraire constatée lors de la soutenance.

NS : évaluation soutenance ([0%,100%]) (si absent NS=0%)

Calcul de N (les seuils s1,s2,s3,s4 sont recalculés chaque année ; à titre d'exemple, une année les seuils étaient ainsi s1=4,43,s2=3,8,s3=2,8,s4=0) :

N=NP*NS si s1<=NI<=10

N=NP*NS*0,9 si s2<=NI<s1

N=NP*NS*0,8 si s3<=NI<s2

N=NP*NS*0,7 si s4<=NI<s3

N=NP*NS*0=0 si NI<s4.