



(12) FASCICULE DE BREVET

(11) N° de publication : **MA 33787 B1** (51) Cl. internationale : **G06F 1/00**

(43) Date de publication :
03.12.2012

(21) N° Dépôt :
33810

(22) Date de Dépôt :
04.05.2011

(71) Demandeur(s) :
**BEN EL KHATTAB ADIL, 3, RESIDENCE BELLE VUE, RUE ABOULMHACINE
ARROUAYNI, RACINE, 20000 CASABLANCA (MA)**

(72) Inventeur(s) :
BEN EL KHATTAB ADIL

(74) Mandataire :
ADIL BEN EL KHATTAB

(54) Titre : **TECHNIQUE DE MODÉLISATION D'UN SYSTÈME INFORMATIQUE MULTI-CLIENT**

(57) Abrégé : LA TECHNIQUE INVENTÉ ET FAISANT L'OBJET DE CE BREVET PERMET D'UTILISER UNE SEULE BASE DE DONNÉES RELATIONNELLE POUR STOCKER LES DONNÉES DE PLUSIEURS CLIENTS SANS DEVOIR AJOUTER L'IDENTIFIANT DU CLIENT À CHAQUE ENTRÉE DE LA BASE ET TOUT EN GARANTISSANT LA SÉPARATION TOTALE DES DONNÉES DES CLIENTS. CETTE TECHNIQUE FONCTIONNE GRÂCE À L'APPLICATION D'UNE NOTION DE RÉSEAUX D'INFORMATION (INFORMATION NETWORK) DANS LE CADRE D'UN LOGICIEL QUI GÈRE LES DONNÉES DE PLUSIEURS CLIENTS À LA FOIS.

Mémoire descriptif en vue de l'obtention d'un brevet d'invention

TECHNIQUE DE MODÉLISATION D'UN SYSTÈME INFORMATIQUE MULTI-CLIENT

Date de dépôt: Mai 2011

Déposant, Inventeur et Bénéficiaire: **Adil BEN EL KHATTAB**, citoyen Marocain titulaire de CIN N° BE746544 né le 18 Septembre 1981 à Casablanca demeurant au 3, Résidence Belle Vue, Rue Aboulmahacine Arrouyani, Racine Casablanca.

ABRÉGÉ

La technique inventé et faisant l'objet de ce brevet permet d'utiliser une seule base de données relationnelle pour stocker les données de plusieurs clients sans devoir ajouter l'identifiant du client à chaque entrée de la base et tout en garantissant la séparation totale des données des clients. Cette technique fonctionne grâce à l'application d'une notion de réseaux d'information (information network) dans le cadre d'un logiciel qui gère les données de plusieurs clients a la fois.

ETAT DE LA TECHNIQUE

Dans le monde informatique, les données manipulées par des ordinateurs sont le plus souvent stocké dans des bases de données sous forme de tables.

Un logiciel a souvent besoin de manipuler les données de plusieurs clients a la fois. Dans le cadre d'un logiciel d'entreprise, un client est typiquement une entreprise qui utilise le logiciel pour gérer ses activités. Afin de gérer ces données, un logiciel peut utiliser plusieurs techniques pour séparer les données des différents clients. La manière la plus simple, mais la moins performante, consiste à utiliser une base de données par client. Un logiciel a en effet la capacité de communiquer simultanément avec plusieurs bases. Cette méthode garanti la séparation des données des clients mais cause une redondance des données et génère des taches de maintenance importantes.

Une deuxième méthode consiste à utiliser une seule base de données pour tous les clients, mais stock les données dans des tables différentes pour chaque client. Dans ce cas l'identifiant de chaque table est précédé par l'identifiant du client. Bien que cette méthode allège les taches de maintenance, ca garde néanmoins le même inconvénient de la méthode précédente, à savoir, la redondance des données.

Une troisièmes méthode, encore plus optimisé, consiste à stocker les données des clients sur une seule base, sur les mêmes tables, en identifiant chaque entré de table par l'identifiant du client. Cette méthode permet d'éviter la redondance des données mais n'est pas applicable à une base de données existante, puisque pour appliquer cette méthode toutes les tables de la base doivent comporter un champ supplémentaire qui permet d'associé chaque entré à un client.

DESCRIPTION DE L'INVENTION

La méthode inventé et faisant l'objet de ce brevet consiste à utiliser une seule base de données pour tous les clients, en stockant les données des différents clients sur les mêmes tables, sans devoir ajouter l'identifiant du client aux tables existantes. Ce qui rend cette méthode unique est le fait qu'elle permette la migration d'une base de données existante, mono-client, vers une base de données multi-client, sans changer le modèle de données existant.

Ce procédé consiste à grouper les données des utilisateurs d'une base de données dans des réseaux d'information. Chaque utilisateur appartient à un seul réseau, et chaque client peut posséder un ou plusieurs réseaux dans la base, sans que les données d'un réseau ne puissent jamais être accessibles depuis un autre réseau. Un réseau doit être et rester totalement

hermétique. Seuls les utilisateurs appartenant à un réseau peuvent y avoir accès. Ce modèle d'appartenance à trois niveaux est schématisé sur le graphique 1.

Etape 1 : identifier les points d'appartenance

Les points d'appartenance sont des entités de la base qui définissent l'appartenance d'un objet à un réseau. A titre d'exemple : une base qui contient les données d'employés, de clients, de fournisseurs, et de produits, a besoins d'identifier l'appartenance de chacune des occurrences de ses quatre entités. Dans ce cas le point d'appartenance n'est pas l'entité employé ou fournisseur, mais une entité abstraite "Acteur" qui englobe les trois entités d'acteurs dans le system, ainsi qu'une entité "Produit" qui englobe tous les types de produit existant dans le système. Ces nouvelles entités contiennent tous les attribues communs aux sous entités. Bien entendu, chaque entité existe sur la base sous forme de table.

Le graphique 2 montre les points d'appartenance d'une base de données d'entreprise typique. On note que seul les entités **Acteur** et **Produit** représentent des points d'appartenance, et donc contiennent l'identifiant du réseau. Toutes les autres entités contiennent les données de tous les réseaux sans faire référence au réseau propriétaire des données. Ces entités qui ne font pas référence au réseau doivent impérativement avoir un ou plusieurs liens relationnels avec d'autres entités qui permettent de remonter jusqu'au point d'appartenance le plus proche. Il est ainsi possible d'identifier l'appartenance de chaque entré dans la base en naviguant à partir de cette entré jusqu'au point d'appartenance. Cette navigation est possible grâce à la nature relationnelle des bases de données

Etape 2 : Filtrer les données par point d'appartenance

Une fois les points d'appartenance définis, ces derniers serviront à sécuriser et filtrer l'accès aux données. Comme le montre le graphique 3, chaque requête est d'abord traitée par une routine qui vérifie si l'utilisateur à l'origine de la requête a bien accès au réseau correspondant aux points d'appartenance spécifié dans la requête. Si la requête ne fait référence à aucun point d'appartenance, les points d'appartenance par default de l'utilisateur sont utilisés.

Des exemples sont donnés ci-après afin d'illustrer le fonctionnement de la routine de vérification d'accès. La technique faisant objet de ce brevet ne peut en aucun cas être limité a ces exemples, elle reste applicable à tout autre type de données.

Exemple 1 :

Une requête est reçue provenant de l'utilisateur λ demandant la création d'une nouvelle commande du produit fini A de chez le fournisseur B.

1. Le réseau auquel appartient l'utilisateur λ est obtenu à partir de l'acteur qui correspond à l'utilisateur.

2. Le réseau auquel appartient le fournisseur B est obtenu à partir de l'acteur qui correspond au fournisseur.
3. Le réseau auquel appartient le produit fini A est obtenu a partir du produit qui correspond au produit fini.
4. Si le réseau de l'utilisateur est différent du réseau du fournisseur ou différent du réseau du produit fini, l'accès est refusé.
5. Si les réseaux correspondent, l'accès est autorisé.

Exemple 2 :

Une requête est reçue provenant de l'utilisateur λ demandant la lecture de la commande X.

1. Un fournisseur correspondant à la commande X est obtenu. Si la commande a plusieurs fournisseurs le premier trouvé est utilisé.
2. Les étapes (1, 2, 4, 5) de l'exemple 1 sont reprises en utilisant le fournisseur obtenu a l'étape 1 de cet exemple.

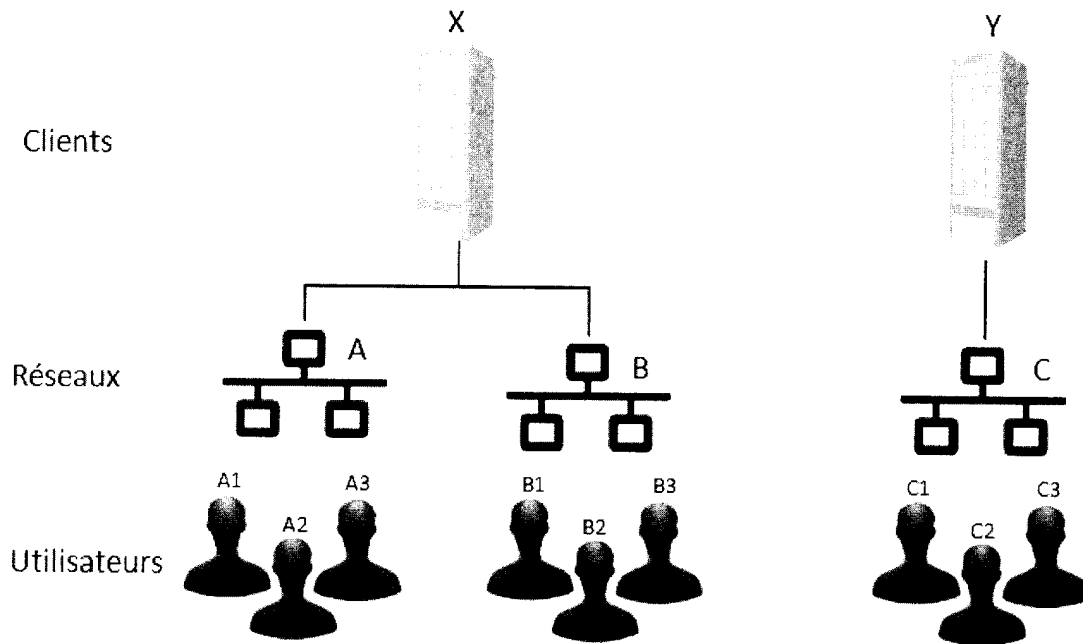
Cette vérification repose sur le fait que chaque paire d'objets reliés dans la base appartiennent automatiquement au même réseau. Aucune relation ne peut exister entre deux objets de deux réseaux différents.

En règle générale, la routine de vérification s'assure que tous les objets écrits suite à une requête sont écrits sur le réseau de l'utilisateur à l'origine de la requête. Et que tous les objets lus suite à une requête appartiennent bien au réseau de l'utilisateur à l'origine de la requête.

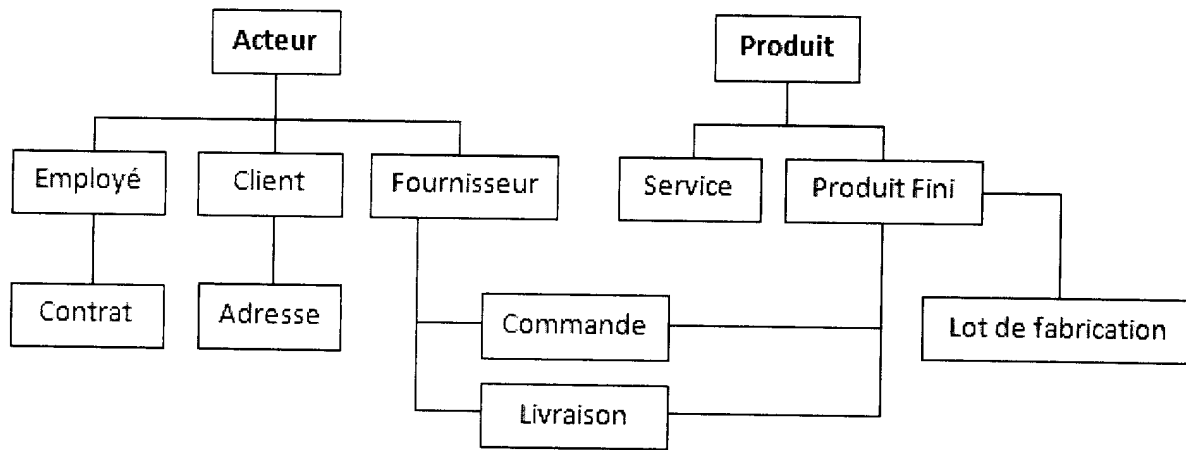
REVENDICATIONS

1. Procédé de modélisation et de construction de logiciel informatique utilisant une seule base de données pour le stockage des données de plusieurs clients tout en garantissant la séparation des données des différents clients. Ce procédé comporte le groupage des données dans des "réseaux" et l'identification de "points d'appartenances" qui définissent l'appartenance de chaque objet dans la base de données à un réseau.
2. Procédé selon la revendication 1, caractérisé par l'utilisation de la notion de réseaux d'information pour la séparation des données des clients et l'utilisation d'un champ "networkId" pour l'affectation des données aux réseaux.
3. Procédé selon la revendication 1, caractérisé par l'ajout à la base de données d'entités abstraites qui servent à grouper les attributs communs de sous-entités et qui sert comme point d'appartenance en ajoutant un champ d'identification du réseau.
4. Procédé selon la revendication 1, caractérisé par l'exécution systématique d'une routine de vérification avant chaque exécution de requête. Cette routine s'assure que tous les objets écrits suite à une requête sont écrits sur le réseau de l'utilisateur à l'origine de la requête. Et que tous les objets lus suite à une requête appartiennent bien au réseau de l'utilisateur à l'origine de la requête.

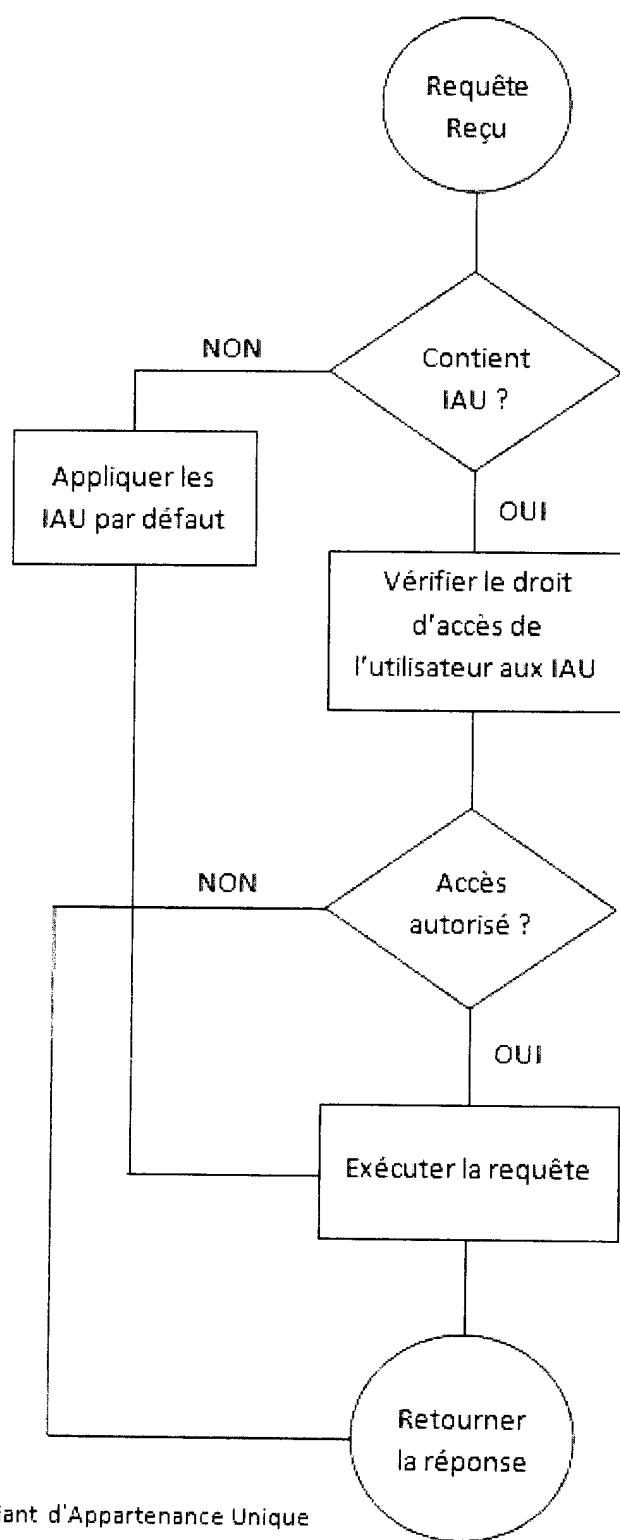
SCHÉMAS



Graphique 1



Graphique 2



Graphique 3