

FAQ Merise et modélisation de données

Date de publication :

Dernière mise à jour : 27/05/2007

La FAQ Merise, toutes les réponses à vos questions.

Ont contribué à cette FAQ :

Cian - Petrus - DrQ - fadace - Nanci - SQLpro
- skapull - ddams - sophnouille - Bouboubou -

1. Introduction (5)	4
2. Généralités (10)	6
3. MCD (13)	14
4. MLD (4)	26
5. Diagramme des flux (2)	29
6. MCT (2)	33
7. MOT (2)	34
8. Bibliographie (1)	36
9. Outils (8)	37
9.1. PowerDesigner/PowerAMC (6)	38
9.2. Win'design (2)	39
10. Exercices (8)	41

[Sommaire > Introduction](#)

Remerciements

Auteurs : [Cian](#) ,

Merci à l'ensemble des contributeurs de cette FAQ et tout particulièrement à Nanci pour la qualité de ses interventions sur nos forums. Egalement, un énorme merci à Anomaly, SheikYerbouti et Laurent Dardenne pour leur aide précieuse pour la relecture de cette FAQ et de nombreux articles de developpez.com. Grâce à eux il vous est plus agréable de nous lire.

Comment bien utiliser cette faq ?

Auteurs : [Cian](#) ,**Le but :**

Cette faq a été conçue pour être la plus simple possible d'utilisation. Elle tente d'apporter des réponses simples aux questions auxquelles sont confrontés tous les utilisateurs de MERISE.

L'organisation :

Les questions sont organisées par thème, les thèmes pouvant éventuellement contenir des sous-thèmes. Lorsqu'une question porte sur plusieurs thèmes, celle-ci est insérée dans chacun des thèmes, rendant la recherche plus facile.

Les réponses :

Les réponses contiennent des explications issues des multiples sujets postés sur nos forums. Les réponses peuvent également être complétées de liens vers d'autres réponses ou vers un autre site en rapport.

Nouveautés et mises à jour :

Lors de l'ajout ou de la modification d'une question/réponse, un indicateur est placé à côté du titre de la question. Cet indicateur reste visible pour une durée de 15 jours afin de vous permettre de voir rapidement les modifications apportées.

J'espère que cette faq pourra répondre à vos questions.

N'hésitez pas à contacter le responsable de la section Modélisation pour vos commentaires/remarques/critiques.

Comment participer à cette faq ?

Auteurs : [Cian](#) ,

Cette faq est ouverte à toute collaboration. Pour éviter la multiplication des versions, il serait préférable que les collaborations soient transmises directement aux administrateurs de la faq.

Plusieurs compétences sont actuellement recherchées pour améliorer cette faq :

- **Rédacteur :** Bien évidemment, toute nouvelle question/réponse est la bienvenue. N'hésitez pas à la faire connaître au travers du forum Modélisation ou en mailant le responsable de la section Modélisation/UML!
- **Web designer :** Toute personne capable de faire une meilleure mise en page, une feuille de style ou de belles images...
- **Programmeur :** Une personne capable de faire un petit moteur de recherche...

- **Correcteur :Malgré nos efforts des fautes d'orthographe ou de grammaire peuvent subsister. Merci de contacter les administrateurs si vous en débusquez une... Idem pour les liens erronés.**

Quels sont les droits de reproduction de cette faq ?

Auteurs : Cian ,

Dans l'attente d'un statut plus précis, aucune publication n'est autorisée sans l'accord écrit des auteurs.

Quel outil a été utilisé pour réaliser les illustrations de cette faq ?

Auteurs : Cian ,

La majeure partie des modèles servant d'illustrations dans cette faq ont été réalisées grâce à l'outil win'design.

lien :  [Plus d'infos sur Win'Design](#)

[Sommaire > Généralités](#)

MERISE, Qu'est-ce que c'est ?

Auteurs : Cian ,

Merise est une méthode d'analyse pour les projets informatiques

Merise est une méthode de conception de systèmes d'information de gestion.

Merise ne se limite pas à l'aspect informatique. Cette méthode a eu comme objectif premier de jeter un pont entre les besoins des utilisateurs et les solutions des informaticiens. Certes sa finalité est quand même de faciliter la conception des projets informatiques en permettant d'analyser et de formaliser très tôt les « besoins » des utilisateurs.

Qu'est qu'un système d'information La définition du système d'information est issue de la théorie des systèmes ou systémique. Bien que fortement « théorisante », elle fournit un éclairage assez solide sur le terrain (lorsqu'on ne voit plus de quoi on parle#)

Une « organisation » (entreprise, administration, collectivité, tout groupe social organisé exerçant une activité) peut être modélisé comme comportant trois sous systèmes :

- le système de pilotage (celui qui réfléchit, décide, oriente)
- le système opérant (celui qui produit, transforme, agit)
- le système d'information

Le système d'information est la représentation de l'activité du système opérant ,construite par et pour le système de pilotage pour en faciliter le fonctionnement. Le système d'information a deux grandes fonctions :

- recueillir, mémoriser et diffuser les informations
- assurer le traitement de ces informations

On reconnaît ici la fameuse dichotomie Données / Traitements.

Le système d'information, dans son principe, n'est pas né avec l'informatique ! Les pharaons disposaient de systèmes d'informations ; seuls les moyens différaient. C'est la capacité de l'outil informatique à amplifier la gestion des données et des traitements qui a conduit à distinguer deux niveaux de système d'information :

Une « organisation » (entreprise, administration, collectivité, tout groupe social organisé exerçant une activité) peut être modélisé comme comportant trois sous systèmes :

- le système d'information organisationnel (SIO), la partie visible, basée essentiellement sur des ressources humaines, de l'activité organisée
- le système d'information informatisé (SII) correspondant au contenu informatisé du SI (logiciel, base de données)

Ainsi, un projet informatique a pour objectif de construire une application informatique (logiciel et base de données), support d'un système d'information informatisé, inclus dans un système d'information organisationnel. Merise ne fait que rappeler que l'on ne peut pas construire un SII sans comprendre au préalable le SIO dans lequel on l'implante ! J'utilise parfois la métaphore de la « prothèse » que l'on greffe au sein de l'organisation SIO pour amplifier ses fonctions, qui doit être adaptée à son contexte, ne pas « blesser » le récepteur sous peine de rejet#


Qu'est ce qu'une méthode Une méthode comporte trois axes indispensables pour obtenir ce label « méthode » :

- une démarche, ensemble coordonné d'étapes, de phases et de tâches indiquant le chemin à suivre [Hodos, le chemin en grec, serait une des étymologies de méthode] pour conduire un projet, ici, la conception d'un SI,
- des raisonnements et des techniques nécessaires à la construction de l'objet projeté, traduits ici par des modélisations,
- des moyens de mise en #uvre, en l'occurrence une organisation de projet et des outils.

Seule la réunion de ces trois dimensions permet une utilisation opérationnelle. Dès le début, Merise s'est voulu une méthode. Avec un effet induit : dès que l'un des axes est remis en cause, l'ensemble en pâti#C'est en effet la lourdeur de la démarche, dénoncée dans les années 90, qui a « plombé » Merise.

Nota : On remarquera qu'UML, en se positionnant sur le deuxième axe, ne peut être une méthode tant que l'on ne lui adjoint pas une démarche et des moyens.

Pour conclure, Merise est donc volontairement une méthode pour la conception de systèmes d'information, en vue de leur informatisation.

lien :  MCD, MCT, MOD, MPD.... qu'est-ce que "les niveaux d'abstraction" dans Merise ?

Historique et origine de la méthode Merise - Complet

Auteurs : Nanci ,

Début des années 70

Les bases de données commencent se développer. Aux Etats-Unis, Codd (70-72) propose le formalisme relationnel. En France,

- **la recherche, essentiellement universitaire, est très active sur ce domaine,**
- **les chercheurs et les professionnels « éclairés » se rencontrent dans le cadre du Groupe Bases de données de l' AFCET (société savante qui jouera un rôle important pour la diffusion des connaissances, dont Merise)**
- **Hubert Tardieu, dans le cadre du Centre d'Etude Technique de l'Equipement d'Aix en Provence, développe un prototype de SGBD relationnel et réfléchit à une approche alternative. Il publie « Modèle individuel pour un base de données » H. Tardieu, C. Deheneffe, JL Hainaut (Namur # mai 74), qui préfigure le modèle entité-relation.**

La recherche sous contrat (74 # 81)

Courant 74, le CETE d'Aix en Provence et l'Université d'Aix-Marseille s'associent pour présenter un projet de recherche auprès de l'I(N)RIA intitulé : « Méthode, modèles et outils pour la conception de la base de données d'un système d'information ».

L'équipe, placée sous la direction scientifique du Prof J.L. Le Moigne (qui vient d'inventer en 73 la notion de système d'information), est pilotée par Hubert Tardieu. Elle est composée au départ d'Henri Heckenroth et de Dominique Nanci. Au cours du projet, d'autres personnes rejoindront l'équipe, plus ou moins longtemps, dont : Daniel Pascot, Bernard Espinasse, Mokrane Bouzgehoub.

Dans ce contexte, s'élaborent progressivement :

- **la modélisation conceptuelle des données (formalisme individu # relation qui deviendra ensuite entité # relation)**
- **la confrontation par les vues externes**
- **la transformation en modélisation logique puis la transformation en bases de données (Codasyl et relationnel)**
- **des prototypes d'outils d'aide à la conception**
- **plus tard, la modélisation des traitements selon ses différents niveaux(par adaptation des réseaux de Petri au contexte de la gestion)**

Régulièrement, des rapports de recherche sont remis à l'I(N)RIA, où le référent est Arnold Rochfeld, jalonnant ainsi les principales étapes de construction.

Parallèlement, l'équipe participe à de nombreux groupes de travaux (dont l' AFCET) et à des congrès scientifiques pour diffuser la modélisation conceptuelle entité-relation. Des sociétés de service s'intéressent au travaux (Sema avec R. Coletti, Gamma avec Ph Duché, CGI avec G. Vahee).

La première publication internationale s'inscrit dans le cadre de l'IFIP. (janv. 76)

« Conceptual model as a data base design tool » P. Moulin, J. Randon, S. Savoyski, S Spaccapietra, H. Tardieu, M. Teboul. Ces personnes constituaient le groupe Base de données de l' AFCET. L' AFCET étant le représentant français de l'IFIP, Tardieu avait usé de ce moyen pour obtenir un droit d'entrée dans un congrès international (où l'accès à de nouveaux venus est toujours très verrouillé). Les personnes sont citées par ordre alphabétique mais l'article avait été entièrement rédigé par H. Tardieu et son équipe#. Voilà comment Moulin et Teboul se retrouvent propulsés à l'origine de Merise (par l'auteur du mémo à l'origine de ma réaction)

En octobre 76, Chen publie dans la revue de l'ACM « The Entity-Relationship model, toward a unified view of data » où il présente son formalisme, très proche du formalisme de Tardieu ; Cette publication américaine lui donne un statut (évidemment) international. Le formalisme de Tardieu devient Entité # Relation.

L'équipe de recherche « aixoise » continue à jaloner ses travaux de publications et communications pour promouvoir les principes de sa méthode. Un premier ouvrage « Conception d'un système d'information # Construction de la base de données » paraît en 79 aux Editions d'Organisation, écrit par H. Tardieu, D. Nanci, D. Pascot. Il expose pour la première fois au « grand public » la conception d'une base de données avec la notion de MCD Entité-Relation, puis MLD et génération de base de données, ainsi que la nécessité d'outils.

Le projet Merise (77-81)

Dès 77, la Mission informatique du Ministère de l'Industrie (on est encore dans une volonté importante de l'Etat pour une politique informatique française) souhaite établir une méthode nationale (unifiée ?) dans le domaine de la conception des systèmes d'information. On constitue un groupe de travail qui réunit l'équipe de recherche « aixoise » et des représentants des principales sociétés de services avec comme objectif de : >

- intégrer les travaux de recherche proposés par l'équipe de H. Tardieu,
- mettre au point une démarche et des moyens de mise en œuvre, issus de l'expérience pratique des SSII
- être le vecteur de la diffusion de la synthèse méthodologique ainsi construite.

Merise voit officiellement le jour en 79, sous la forme d'un premier fascicule publié par Ministère de l'Industrie: « Méthode de définition d'un système d'information ».

Le nom de Merise a été trouvé comme la métaphore du merisier qui doit être greffé pour porter des fruits. En effet, dans l'introduction du premier fascicule, il est écrit : « Merise n'est pas une méthode, mais un tronc commun méthodologique sur lequel #. ». En effet, les différentes SSII veulent pouvoir ultérieurement apporter leur valeur ajoutée en personnalisant cette méthode. Le vers est dans le fruit. Quinze ans après, on observera des méthodes Merise, parfois avec des écarts notables (donnant naissance à des querelles dont seuls les « gaulois » ont le secret#). Par ailleurs, quasiment aucune structure (tel l'OMG avec UML) ne veillera au respect d'une certaine « normalisation » de Merise ; le merisier est redevenu un « sauvageon » ; plus aucun groupe pour promouvoir cette méthode au niveau international face au lobbying d'autres approches.

Le projet Merise se poursuit donc jusqu'en début 81 avec la publication de plusieurs documents de référence sur la méthode Merise. Le projet est alors interrompu suite au changement politique. Une tentative de bilan aura lieu en 84, mais il se heurtera à des conflits d'intérêt entre les grandes SSII.

Entre-temps, l'Administration consacre Merise comme la méthode de référence pour tous ses projets de conception de SI, assurant ainsi sa pérennité et son enracinement.

L'essor de Merise (81- 85)

A partir de 81, certaines grandes SSII qui avaient accompagné Merise, dont SEMA, CGI, GAMMA [devenu depuis MEGA= MEriseGAMma], entament la diffusion de la méthode auprès des grandes entreprises et de l'Administration. En 83, est publié le premier ouvrage sur Merise, ouvrage qui restera la référence.

« La méthode Merise # Tome I :Principes et outils »

H. Tardieu, A. Rochfeld, R. Coletti,

qui sera suivi en 85 par :

« La méthode Merise # Tome II : Démarches et pratiques »

H. Tardieu, A. Rochfeld, R. Coletti, G. Panet, G. Vahee

L'expansion de Merise (85-95)

Dès lors, Merise connaît un engouement. Toutes les SSII font désormais du Merise (avec des résultats plus ou moins probants#). De nombreux ouvrages paraissent. Merise est désormais enseigné dans les formations universitaires. Des outils apparaissent, certains issus des prototypes conçus lors de la recherche. La fin des années 80 dénombre plus de 15 outils français sur Merise. Quasiment toute grande SSII propose le sien. Certains sont encore sur le marché.

L'AFCET constitue un groupe de travail sur Merise dans lequel se retrouvent les promoteurs-défenseurs de cette méthode. Ils mettent en commun leurs expériences et améliorent progressivement Merise tant sur les modélisations que sur la démarche. De nombreuses publications et communications, associées avec le monde universitaire, jalonnent cette période et appuient les idées de la méthode Merise.

Cette promotion se traduit également par la participation, en 89-90, de représentants français (la plupart issus du groupe AFCET) au projet EuroMethod (l'équivalent du projet initial Merise au niveau européen). Comme on aurait pu s'en douter, les antagonismes nationaux attisés par des enjeux politiques, stérilisèrent complètement ce projet. Adieu donc

à une méthode européenne et à ses conséquences au niveau de l'industrie du logiciel, du moins de sa partie intellectuelle (au grand bonheur des américains# ?).

A partir de 90 Merise est devenu une figure imposée dans le cursus de formation de tout informaticien, du moins sur la partie de modélisation, plus particulièrement des données.

En 91, l'AFCET organise un congrès « Autour et alentour de Merise » qui réunira plus de 1000 personnes pendant 3 jours avec un grand nombre de communications. Ce congrès marque, à mon avis, l'apogée de Merise. C'est à cette occasion que sont présentées un certain nombre d'évolutions qui vont conduire à Merise / 2 (ou deuxième génération)

L'émergence de cette deuxième génération se concrétise par deux ouvrages :

« Merise / 2 : Modèles et techniques Merise avancées » G. Panet ; R. Letouche (94)

« Ingénierie des systèmes d'information : Merise deuxième génération » D. Nanci, B. Espinasse (94)

La concurrence et la remise en cause (95 # 2000)

Deux courants viennent contester Merise : le RAD et l'approche objet.

Il est indéniable que la démarche de Merise initialement proposée, enrichie (ou alourdie#) par les SSII (dont certaines y ont vu un moyen de s'incruster chez le client#), appliquée dogmatiquement (par ceux qui la maîtrisait le moins#) a conduit à une lourdeur certaine.

L'émergence des démarches rapides puis agiles, en opposition aux démarches traditionnelles, remet vivement en cause Merise (hélas sans préciser qu'il ne s'agissait que de la démarche#, on jetait le bébé avec l'eau du bain !). Le succès de ces démarches s'explique par un changement radical des conditions économiques et la nature des projets (fin des grands projets).

Le développement des langages objets et des méthodes associées bouleverse la conception du logiciel. Il faut admettre que Merise, méthode de conception de SI, contribuait plus à la partie SIO que SII et laissait les informaticiens sans réponse au niveau de la conception du logiciel (MLT, MPT). Ces méthodes et ces langages comblaient ce vide ; en particulier pour les développeurs.

Dans le camp merisien (déserté par ceux qui l'encensait, en particulier au niveau des médias#), deux réactions se distinguent :

- une proposition d'intégration # fusion de l'approche objet dans les modélisations. Merise-Objet ou Merise 3,
- une proposition de cohabitation en complémentarité (remplacer toute la partie Traitement du SII par l'approche objet)

Pour de nombreuses personnes, Merise est mort (titre de première page d'un grand hebdomadaire professionnel en 97). Toutefois, les modélisations de Merise continuent à avoir leur place dans les programmes de formation scolaires et universitaires.

Les outils (du moins ceux qui restent) sont de plus en plus performants, conviviaux et professionnels.

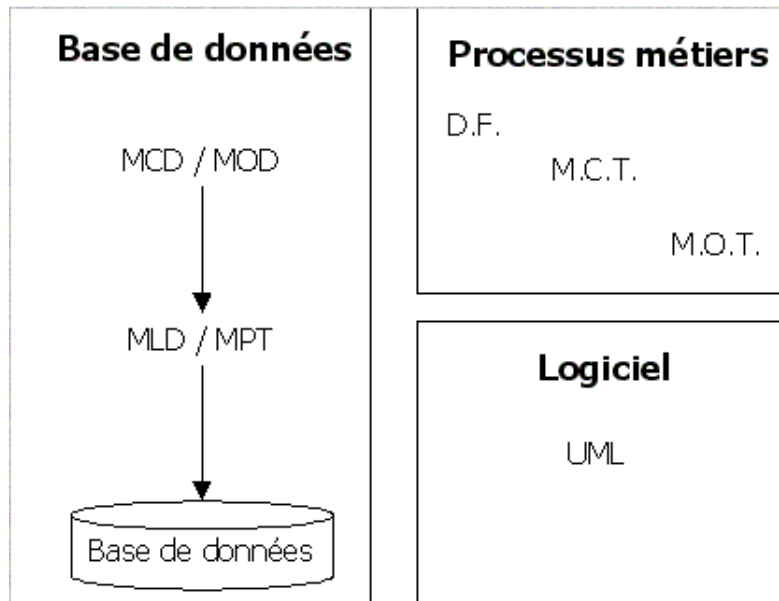
Un repositionnement possible ? (2000 ?)

UML a unifié les modélisations objet concurrentes et s'est imposé comme modélisation pour la conception de logiciel orienté objet. Les bases de données restent fortement relationnelles. La modélisation des processus métiers (BPM) s'affirme comme indispensable dans la conception des systèmes d'information. Par ailleurs, la conception des bases de données devient de plus en plus « indépendante » de la conception du logiciel.

Aujourd'hui trois pôles semblent se distinguer dans la conception de systèmes d'information (qui ne se réduit évidemment pas à la conception du logiciel !#)

- la conception des bases de données, où la séquence MCD ? MLD ? base reste toujours en bonne place,
- la conception du logiciel dans des environnements objets où UML et les méthodes associés remplissent incontestablement et efficacement leur tâche,
- la modélisation des processus métiers, indispensable pour la compréhension du fonctionnement de l'entreprise et pour l'interconnexion des applications.

Si l'on se réfère à schéma bien connu, on retrouvera l'articulation suivante :



D'ailleurs, dans ma pratique, je déduis très facilement les cas d'utilisation d'une « vrai » modélisation MCT ou MOT métier (et non pas de MCT/MOT représentant un fonctionnement informatique#). Les opérations ou les tâches à informatiser déterminent « naturellement » les cas d'utilisation du futur système projeté ; sinon, on se demande d'où viennent ces cas d'utilisation (hormis d'un interview des utilisateurs sur leurs besoins et l'on retrouve intact le questionnement initial de Merise : comment mieux appréhender et exprimer les besoins des utilisateurs#). Maintenant, le Merise monolithique est certainement défunt (et c'est par certains aspects excessifs, une bonne chose..). Mais son héritage reste très important et bénéfique, en particulier au niveau des modélisations et de l'approche générale SIO/SII ; il serait stupide de le rejeter (il est vrai que notre époque est propice aux intégrismes#). Et si le merisier avait laissé des noyaux qui ne demandent à leur tour qu'à germer et être supports de nouvelles greffes ? [on greffe bien des merisiers pour porter des mirabelles#]. Faisons confiance à la nouvelle génération de « jardiniers ».

Origine de la méthode Merise (résumé)

Auteurs : Nanci ,

La méthode Merise est issue d'une initiative de la Mission informatique du Ministère de l'Industrie qui, de 1977 à 1980, réunit dans un groupe de projet des chercheurs et des praticiens des SSII pour élaborer une méthode unifiée pour la conception de systèmes d'information, avec pour premier objectif de mettre cette méthode en #uvre dans les projets de l'Administration et d'inciter les grandes entreprises à y adhérer.

Ce groupe :

- reprend les travaux de recherche de l'équipe d'Hubert Tardieu développés depuis 1974 et qui proposaient un ensemble de modélisations pour la conception d'un système d'information (MCD/MOD/MLD/MPT, DF/MCT/MOT/MLT/MPT), ainsi que des prototypes d'outils,
- élabore une démarche de mise en #uvre issue de la pratique des SSI (Etude préalable, Etude détaillée, Etude Technique, Réalisation, #)
- propose une cadre d'organisation et de conduite de projet (maîtrise d'ouvrage / maîtrise d'#uvre, principaux rôles)
- définit les fonctions des outils associés (interface graphique, référentiel, règles et transformations, #)

Les travaux se concrétisent de 1979 à 1981 par des publications de fascicules du Ministère.

La parution en 1983 de l'ouvrage

« *La méthode Merise # Tome I :Principes et outils* »

H. Tardieu, A. Rochfeld, R. Coletti,

marque la diffusion publique de la méthode Merise ainsi que le début de son expansion.

Présentation des niveaux d'abstraction

Auteurs : Nanci ,

Données et traitements sont indissociablement liées dans un SI. Les données ne vivent que par les traitements, les traitements ne fonctionnent qu'avec des données. C'est le pile et le face de la même pièce.

Par contre, de natures différentes (les données expriment la statique, les traitements expriment la dynamique), ces deux facettes d'un même système d'information sont analysées séparément, chacune avec des raisonnements spécifiques. (c'est l'analyse qui est séparée, et non Données et Traitements). On retrouve d'ailleurs cette dichotomie dans l'approche objet.

En conséquence, il est hâtif de dire « si les traitements doivent changer ,les données ne sont pas impactées et inversement ». Tout au plus, on peut dire que la manière de construire les données n'est pas dictées par les traitements qui les utilisent, et inversement. Cette analyse séparée, avec des vues différentes, vise à apporter le plus d'autonomie donc d'évolutivité.

Les niveaux d'abstraction

Ces niveaux d'abstraction sont un artifice pour faciliter l'analyse, une bonne pratique. Cette dernière suggère d'aborder d'abord le « Quoi » et le « Pourquoi », avant de prendre en compte « Qui », « Où », « Comment » et « Quand », « Quels moyens » . [Cette recommandation fort ancienne est préconisée par les rhétoriciens latins, : Quis quid, ubi, quibus auxiliis, cur, quomodo, quando,# et a récemment été reprise par un centurion dans Astérix].

Ces niveaux (préconisés à l'époque par l'ANSI devenu depuis l'ISO) sont au nombre de quatre et se conjuguent avec les deux niveaux de SI.

Pour le Système d'Information Organisationnel (SIO)

- **niveau Conceptuel ? Quoi, Pourquoi**
- **niveau Organisationnel ? Qui, Où, Comment, Quand**

Pour le Système d'Information Informatisé (SII)

- **niveau Logique ? avec Quel moyens logiciel (général)**
- **niveau Physique ? avec quelle technique concrète de réalisation.**

Si l'on croise ces quatre niveaux avec le deux facettes (données / traitements), on obtient le tableau de positionnement des différentes modélisations :

- **Modèle Conceptuel de Données / Modèle Conceptuel de Traitements**
- **Modèle Organisationnel de Données / Modèle Organisationnel de Traitements**
- **Modèle Logique de Données / Modèle Logique de Traitements**
- **Modèle Physique de Données / Modèle Physique de Traitements**

Bien évidemment, ces différentes modélisations n'ont pas toutes la même importance, et n'ont pas eu le même succès.

Qu'est-ce qu'un processus ?

Auteurs : Nanci ,

Un processus est un ensemble structuré d'événements, d'activités et de résultats qui concourent à un même but. Il représente généralement un ensemble d'activités de l'entreprise dont les événements initiaux et les résultats finaux déterminent une situation stable du domaine.

Le processus s'utilise au niveau d'un MCT macroscopique (ou macro-MCT).

Qu'est-ce qu'une opération ?

Auteurs : Nanci ,

Une opération est un ensemble d'activités déclenché par la survenance d'un événement et produisant en retour des résultats. On regroupe dans une même opération l'ensemble des activités que le domaine peut effectuer sans autre information extérieure.

La segmentation en plusieurs opérations ne se justifie que si l'on attend des informations complémentaires en provenance d'événements nécessaires à la poursuite des activités. Les activités élémentaires constituant l'opération peuvent être conditionnelles.

On ne tient pas compte de leur organisation.

L'opération s'utilise au niveau d'un MCT standard.

Qu'est-ce qu'une opération élémentaire ?

Auteurs : Nanci ,

L'opération élémentaire représente un ensemble d'activités "homogène" en terme de finalités; cette homogénéité peut être évaluée de différentes façons (nature de l'activité, rupture conditionnelle, données utilisées, ..) .

L'opération s'utilise au niveau d'un MCT détaillé ou analytique (MCTA).

Qu'est-ce qu'une phase ?

Auteurs : Nanci ,

Phase (MOT) :

Une phase est un ensemble d'activités

- consécutives
- réalisées dans la même période de temps
- dans le même poste (ou unité organisationnelle)

L'opération s'utilise au niveau d'un MOT macroscopique (ou macro-MOT).

Qu'est-ce qu'une tâche ?

Auteurs : Nanci ,

Tâche (MOT):

Une tâche est un ensemble d'activités

- homogènes en terme de finalité
- réalisées dans un même poste
- d'un même degré d'automatisation (manuel, conversationnel, automatique)
- d'un même délai de réponse (immédiat, différé)

L'opération s'utilise au niveau d'un MOT standard.

MCD, MCT, MOD, MPD.... qu'est-ce que "les niveaux d'abstraction" dans Merise ?

Auteurs : Cian , Nanci ,

Les modèles de données et de traitements sont déclinés selon des niveaux d'abstraction :

- **Le niveau conceptuel (MCD, MCT) formalise les activités (que fait-on, pourquoi le fait-on) sans préoccupation de l'organisation (qui le fait, où, quand, avec quelles ressources humaines, informatique, données).**
- **Le niveau organisationnel (MOD, MOT) formalise les activités en prenant en compte l'organisation**
- **Le niveau logique (MLD, MLT) formalise fonctionnement du logiciel.**
- **Le niveau physique (MPD, MPT) est le niveau le plus proche de l'implémentation.**

Les niveaux d'abstraction se conjuguent avec le degré de détail.

Le degré de détail (de global à détaillé) exprime la "granularité" de l'activité modélisée (généralement par un rectangle) qui peut être composée d'activités élémentaires.

Le mécanisme de décomposition (ou "refinement") permet de passer d'une activité "macroscopique" à un ensemble d'activités plus détaillées et formalisées, un peu comme un effet de zoom bien connu dans les méthodes d'analyse structurées anglo-américaines.

La décomposition peut se faire à niveau d'abstraction constant, par exemple entre MCT et MCTA. Elle peut se faire également en changeant de niveau, par exemple du MCT au MOT. On peut avoir également un mécanisme de recombinaison.

Sommaire > MCD

Qu'est-ce qu'un MCD ?

Auteurs : Nanci ,

Un Modèle Conceptuel de Données est la formalisation de la structure et de la signification des informations décrivant des objets et des associations perçus d'intérêt dans le domaine étudié, en faisant abstraction des solutions et contraintes techniques informatiques d'implantation en base de données.

Un MCD est, dans la culture francophone, exprimé en entité-relation Merise qui comporte les concepts basiques suivants :

- **Entité** : modélisation d'un objet d'intérêt (en terme de gestion) pour l'utilisateur,
- **Relation** : modélisation d'une association entre deux ou plusieurs entités
- **Cardinalités** : modélisation des participations mini et maxi d'une entité à une relation
- **Propriétés** : modélisation des informations descriptives rattachées à une entité ou une relation
- **Identifiant** : modélisation des propriétés contribuant à la détermination unique d'une occurrence d'un entité.

Le MCD a pour objectif de modéliser le discours # métier. Il ne doit pas anticiper sur les solutions relationnelles de mise en #uvre. C'est un contresens (ou une confusion#) que parler de MCD en termes de tables, clés primaires, clés étrangères ; il vaut mieux se situer directement au niveau MLD (si l'on y est plus à l'aise#). Cependant, plus de trente années d'expérience ont démontré l'intérêt de passer par une telle étape avant de passer à une structure logique, apportant une meilleure résilience aux bases de données ainsi construites (surtout lorsqu'elles atteignent des tailles professionnelles).

Comment modéliser les notions de temps (date, heure) ?

Auteurs : Nanci ,

La modélisation du temps peut intervenir de trois manières.

1 - Des propriétés (ou attributs) temporelles

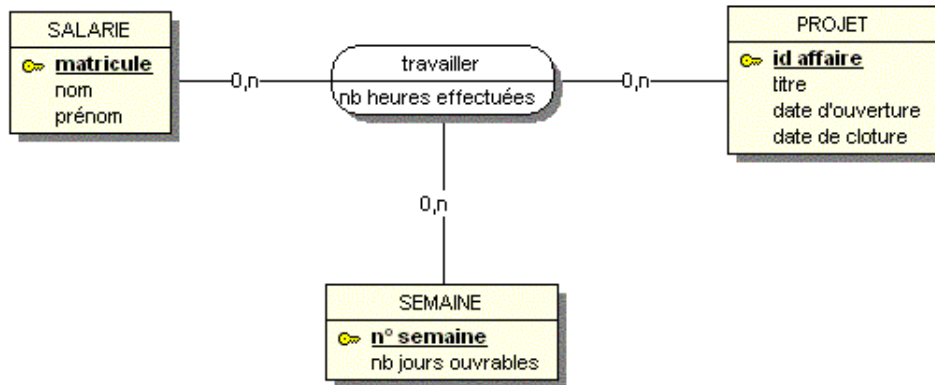
Il s'agit d'informations (modélisées comme des propriétés rattachées à des entités ou des relations) qui prennent leurs valeurs sur le domaine « temps ». Ces propriétés peuvent souvent se nommer date de#, ,mois de#, année de ..

Ainsi date de commande, date de naissance, date de survenance, #. sont certainement des propriétés des entités Commande, Personne, Sinistre. De telles propriétés peuvent également être affectées à des relations, au même titre et selon les mêmes règles que toute autre propriété.

2-La modélisation de phénomènes chronologiques Il s'agit alors de modéliser des situations où la dimension temps intervient explicitement, de manière régulière. Par exemple dans les planning, les emplois du temps, les suivis d'activités, les prévisions ou les statistiques. >

Par exemple, dans un suivi d'activité en gestion de projet, on s'intéresse aux nombre d'heures effectuées hebdomadairement par chaque salarié sur chaque affaire (projet, dossier ou activité nomenclaturée). >

Dans un tel cas, que l'on peut décliner à l'infini, le temps (ici chaque semaine) est une dimension à part et va donc être modélisé comme une entité : on gère des semaines ! La modélisation la plus simple est la suivante :>



L'unité de temps modélisée comme une entité peut être un Jour, une Semaine, un Mois ou toute autre période de temps à délimiter (par son identification).

3-L'historisation Cette notion permet de modéliser, de façon concise au niveau conceptuel, le fait que l'on souhaite conserver les valeurs antérieures prises par une propriété, pour une occurrence d'une entité (ou d'une relation). Il existe ainsi, la valeur actuelle de la propriété et les valeurs antérieures.

Par exemple, pour une entité Contrat, on a une propriété bonus. Conformément aux règles de modélisation, le bonus accueille la valeur actuelle. L'historisation du bonus indique que l'on conservera également les valeurs antérieures du bonus pour tout contrat.

Cette historisation se note, au niveau conceptuel par un (H) au niveau de la propriété concernée. On peut en plus préciser :

- la datation, c'est à dire l'instant auquel la valeur a été remplacée par une valeur plus récente (heure-jour, jour, mois, année),
- le nombre de valeurs antérieures à conserver.

L'historisation peut également s'appliquer au niveau global d'une entité ou d'une relation. Des règles de transformation existe pour le passage au MLD.

lien :  [Les attributs merise](#)

lien :  [Les entités](#)

lien :  [Où placer les attributs d'association ?](#)

Comment indiquer qu'une entité participe conditionnellement à plusieurs relations ?

Auteurs : Nanci , Cian ,

Cela met en oeuvre la notion de contrainte inter-relations.

De la même façon que la notion de cardinalité exprime une contrainte de participation mini et maxi d'une occurrence quelconque d'une entité aux occurrence d'une relation, la contrainte inter- relations exprime une contrainte de participation à des occurrences de plusieurs relations.

Ces contraintes ont été mises au point dans la deuxième génération de Merise (90's).

Leur représentation graphique est la suivante:

- un rond qui accueille la notation du type de contrainte
- un trait pointillé qui relie le rond à l'entité (ou aux entités) concernées
- un trait plein qui relie le rond aux relations concernées

Les types de contraintes recensées et les plus communément utilisés sont :

- l'exclusivité, notée X: la participation aux relations impliquées est mutuellement exclusive

- la totalité, notée T: la participation à au moins une des relations impliquées est obligatoire
- la partition, notée XT; combinaison des 2 précédentes
- la simultanée, notée S: la participation aux relations impliquées est simultanée
- l'inclusion, noté I et orientée: si R2 est incluse dans R1, la participation d'une occurrence de l'entité à R2 doit auparavant participer à R1

Au-delà de ces types standards, on peut associer à toute contrainte inter-relation, une règle qui explicite la modalité spécifique de la contrainte.

Exemple :

l'entité M est liée dans un premier cas à l'entité A (si la valeur de l'attribut x est 1), sinon à B (si la valeur de x est 2) . Alors on choisira une contrainte en X car les valeurs 1 et 2 sont (a priori) mutuellement exclusives, ou XT s'il n'y a que ces deux valeurs possibles. Préciser ensuite l'affectation à l'une ou l'autre par une règle. Par ailleurs, dans ce cas, les cardinalités mini des "pattes" des relations concernées devraient être à 0; peu importe les maxis.

Est-il possible d'avoir une association porteuse de données, alors que les cardinalités sont de part et d'autre 1,1 et 0,n?

Auteurs : Cian , Nanci ,

Depuis les débuts de la modélisation Entité-Relation de Merise, une relation binaire de cardinalité *,n - 1,1 ne doit pas être porteuse de propriétés. C'est considéré comme une "anomalie" et certains outils le diagnostiquent. Ces éventuelles propriétés doivent être définies du côté de l'entité de cardinalité 1,1 (c'est ce qui se passera de toute façon lors de la transformation). Mais attention, ce n'est plus vrai si la cardinalité est 0,1 !

Toutefois, on peut tolérer qu'une association 1,1 # 0,n soit porteuse de propriétés dans les premières phases d'élaboration du modèle (brouillon) ; mais cela devra être rectifié avant la finalisation du modèle.

Qu'est-ce qu'une contrainte de stabilité ?

Auteurs : Nanci ,

Les contraintes de stabilité indiquent des limitations sur les possibilités d'évolution du système modélisé.

Elles portent sur:

- les valeurs des propriétés (propriété stable)
- le rattachement d'occurrences d'entité aux occurrences de relations (patte définitive, patte verrouillée)

Ces contraintes de stabilité s'apprécient par rapport au fonctionnement normal du système (par rapport au métier) et ne prennent pas en compte les éventuelles corrections d'erreurs.

Propriété stable: Les valeurs de cette propriété n'évoluent pas dans le temps; on peut dire qu'elles ne sont pas modifiables (sauf correction d'erreur). Sur les diagrammes, elle est notée (S) sur la propriété. Par définition, tout identifiant est stable. Il n'existe pas (lacune?) de notation indiquant que l'ensemble des propriétés de l'entité est stable.

Dans l'exemple, on considère que nom de Client et date de commande sont stables.

Patte définitive ou rattachement définitif : Une fois l'occurrence de l'entité impliquée dans une occurrence d'une relation, elle ne peut plus se détacher de cette occurrence de relation pour se rattacher à une autre occurrence. Sur les diagrammes, la patte définitive est notée (D). Dans l'exemple, une commande n'est rattachée qu'à un et un seul client(cardinalité 1,1), et cela définitivement; une commande ne peut pas changer de client !

Patte verrouillée ou rattachement verrouillé : Cette notion ne concerne que les pattes de relation à cardinalité maxi n. Elle signifie que les n occurrences de la relations rattachées à l'occurrence d'une entité ont été créées en même temps. On ne peut, dans le temps, ni rajouter (n+1) ni supprimer (n-1) d'occurrences de la relation à l'occurrence de l'entité; cette patte est verrouillée. Dans l'exemple, on ne peut ni rajouter ni retirer une ligne de commande à une commande existante (ce comportement est assez classique avec les lignes d'écritures comptables !)

Quand à la prise en compte au niveau logique, elle reste assez délicate. En effet, vu de la base de données, il est difficile de distinguer une modification due à une évolution "naturelle" du système et une modification due à une "correction

d'erreur". On devra cependant tenir compte de cette notion de stabilité dans la conception des interfaces utilisateurs en ne permettant pas la modification des valeurs des propriétés concernées dans des transactions "normales" et en réservant des transactions particulières aux corrections des erreurs.

Comment exprimer une relation réflexive ?

Auteurs : Nanci ,

La notion de relation dite « réflexive » intervient lorsqu'une même entité est impliquée plusieurs fois dans une relation (plusieurs « pattes » sur la même entité).

Dans une relation binaire, la relation semble boucler sur elle-même, d'où le nom de réflexive ; mais on peut rencontrer cette situation dans les relations ternaires et plus.

Une relation binaire « réflexive » exprime par exemple :

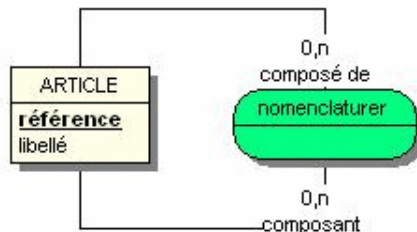
- un Contrat est l'avenant d'un Contrat précédent
- un Projet est la suite d'un autre Projet
- un Article se décompose en Articles
- à partir d'un Diplôme, on peut obtenir d'autres Diplômes


Les cardinalités maxi peuvent être 1 ou n, selon le cas.

Les cardinalités mini sont obligatoirement 0 (sinon on crée une « boucle » sans fin#).

Pour distinguer chaque « patte », on lui affecte un rôle.

Exemple:



lien :  [Comment transformer une relation réflexive ?](#)

Comment déterminer un bon identifiant (unique) pour une entité ?

Auteurs : Nanci , Cian ,

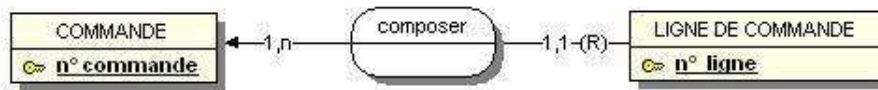
L'identification peut être obtenue de plusieurs façons :

a) de manière absolue, où toutes les propriétés contribuant à l'identification appartiennent à l'entité, soit :

- une propriété spécifiquement inventée à cette fin (code, référence, n°) qui n'est pas obligatoirement numérique
- une propriété (ou une composition de propriétés) naturelle(s) intrinsèques à l'entité; par exemple nom + prénom + date et lieu de naissance d'une personne

b) de manière relative, où l'identification nécessite la présence d'identifiant(s) appartenant à une (ou plusieurs autres) entité(s) reliée(s) à l'entité à identifier

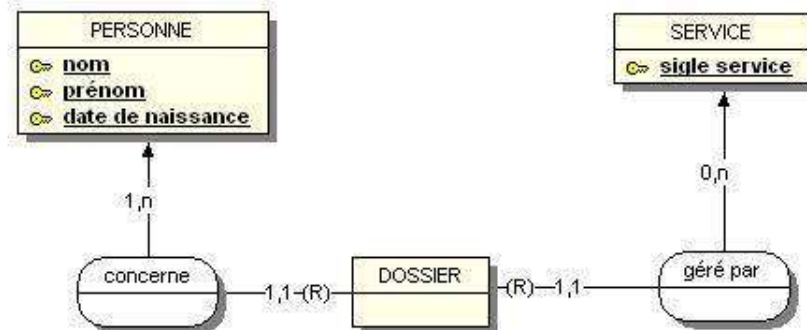
- l'entité comporte une propriété intrinsèque, participant à l'identification relative (cas classique du n° de ligne de commande par rapport à la commande)



Ici, la Ligne de commande est identifiée par n° commande + n° ligne

- l'entité est identifiée relativement à plusieurs autres entités et ne comporte pas de propriété intrinsèque ; on qualifie alors cette entité d'entité faible.

(cas d'un Dossier identifié par la Personne concernée et le Service gestionnaire, sans aucun n° spécifique de dossier ; dans chaque Service, le Dossier est identifié par le nom/prénom/date de naissance de la Personne, chaque Service ayant son Dossier (évidemment différent) sur chaque Personne)



Ici, le Dossier est identifié par sigle service + nom + prénom + date de naissance.

Dossier n'a pas d'identifiant « propre » et pourtant ce n'est pas une relation !

L'entité Dossier participera naturellement à des relations, comme toute autre entité.

Bien sûr, on trouvera toujours un informaticien pour imposer un n° de dossier ignoré de tous, confondant ainsi le niveau conceptuel et le niveau logique.

Attention : Il s'agit de la modélisation conceptuelle qui sera ensuite traduite en logique relationnel avec la redondance due aux clés étrangères.

Comment exprimer que des caractéristiques ne puissent être pertinentes que pour certaines occurrences d'une entité ?

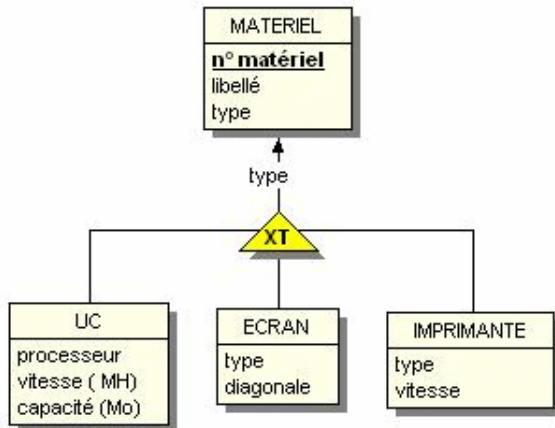
Auteurs : Nanci , Cian ,

Dans un modèle où l'on souhaiterait désigner sous une même entité un ensemble d'occurrences, on peut se retrouver bloqué face au fait que certaines caractéristiques (nécessaires à certaines occurrences) ne puissent pas être renseignées pour toutes les occurrences. Par exemple, si l'on considère la gestion d'un parc informatique : les caractéristiques d'un écran (diagonale, nbHz) ne correspondent pas aux caractéristiques d'un lecteur de CD. Pourtant ce sont tout deux des matériels informatiques.

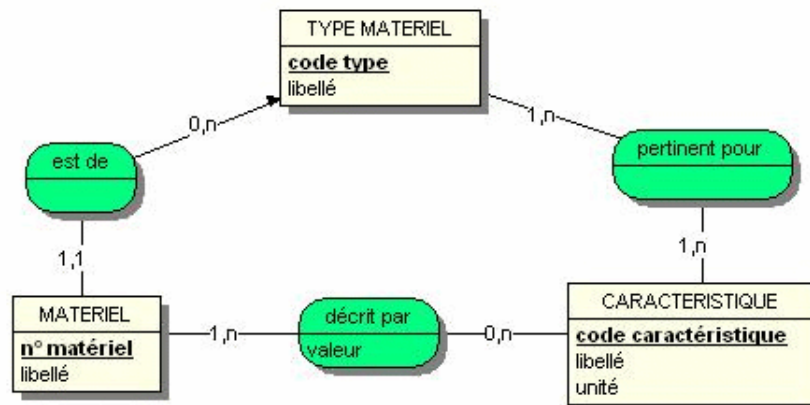
Ce problème est connu sous le nom de "liste variable de propriétés". Il exprime que certaines propriétés n'ont de pertinence que pour certaines occurrences de l'entité.

Dans notre exemple, voici trois types de solutions (résumées ici), chacune ayant des conditions de mise en oeuvre ainsi que des avantages et inconvénients.

- 1/ Accepter que ces propriétés n'aient pas de pertinence pour certaines occurrences de l'entité
- 2/ Expliciter les sous-populations correspondantes par une modélisation en termes de sous-types



- 3/ Recourir à la métamodélisation, c'est à dire modéliser les propriétés en tant qu'occurrences d'entités-types



Comment traduire un MCD en un diagramme de classe UML ?

Auteurs : Nanci , Cian ,

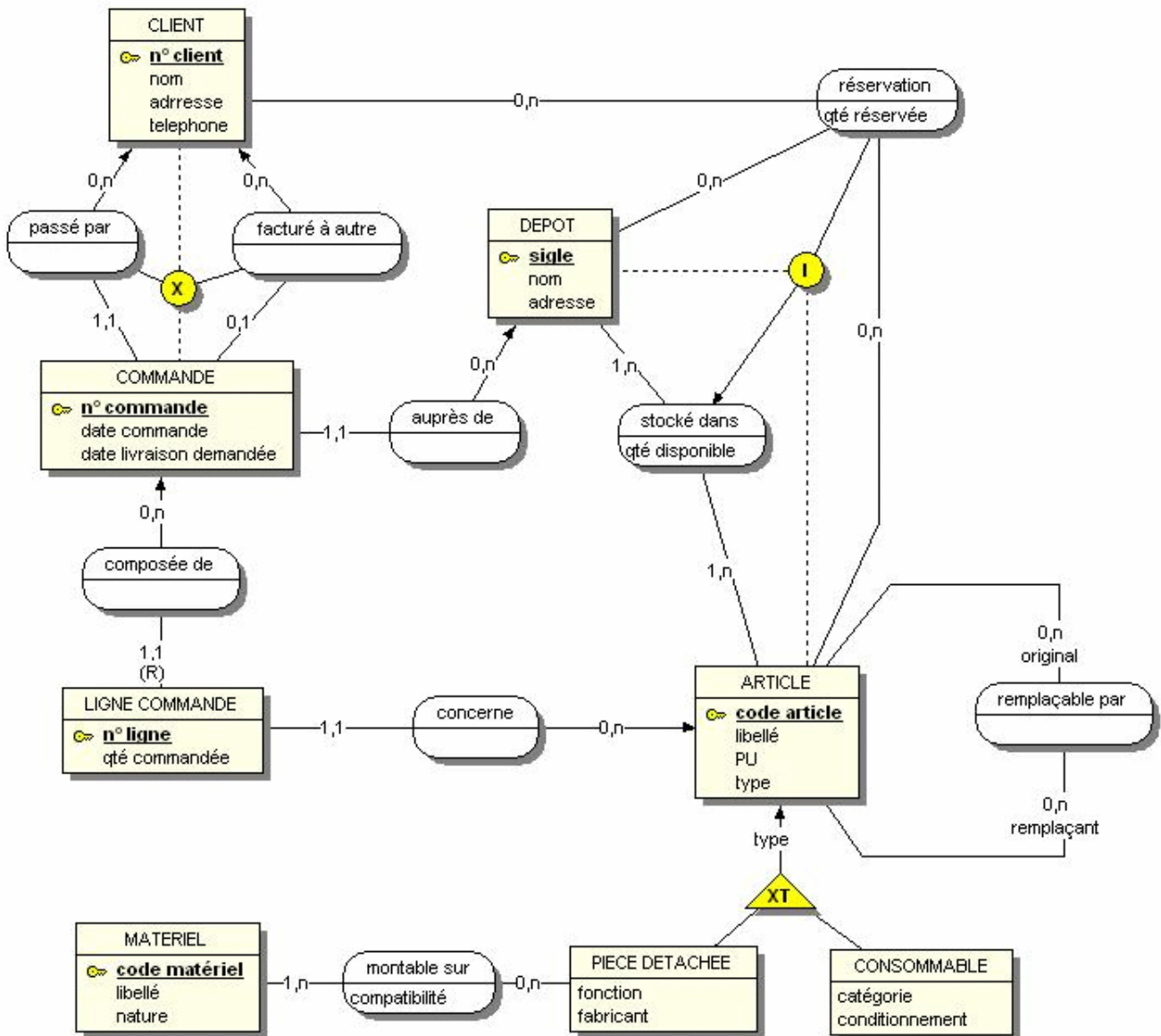
Il existe des règles de transformation d'un MCD Merise en un modèle de classe UML.

Ces règles ont été élaborées dès l'apparition d'OMT (il y a environ un dizaine d'années) et publiées dans de nombreux ouvrages (Exemple : De UML à SQL de C.Soutou, Eurolles).

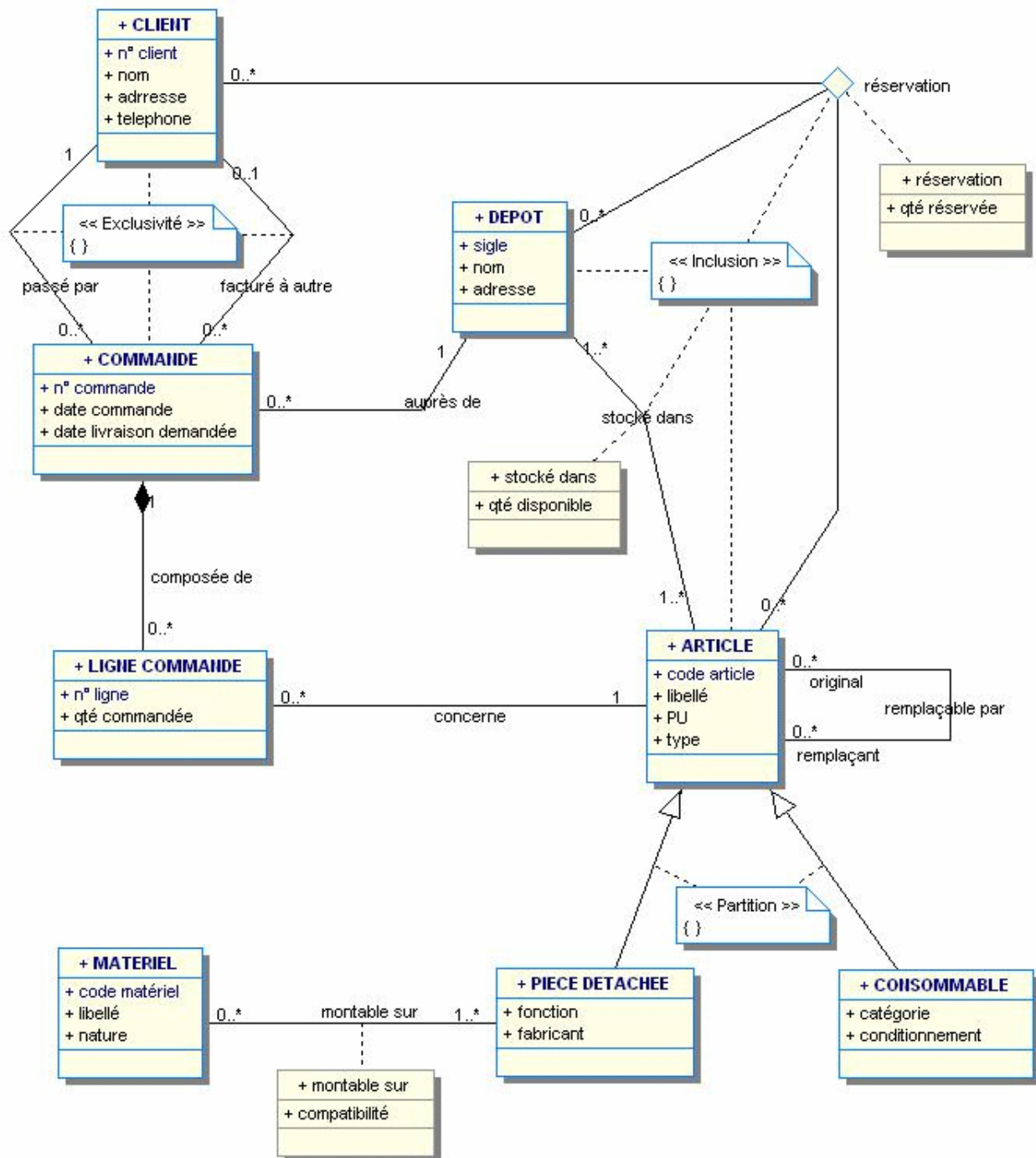
Elles sont d'ailleurs mises en oeuvre dans les outils qui disposent de ces deux modélisations (Win'Design et Power AMC).

Illustration et explication sur un cas :

Le MCD :



Le diagramme de classe correspondant :



> toute entité est transformée en classe

Ses propriétés deviennent des attributs. Son identifiant devient un attribut identifiant ou clé.

> toute relation est transformée en association Attention bien qu'UML propose des relations n-aires, nombre d'outils se limitent aux relations binaires. Les cardinalités des "pattes" de relation deviennent des multiplicités des terminaisons des associations. Attention, en UML, les multiplicités sont notées sur la terminaison opposée (ça marche bien sur les associations binaires, plus problématique avec les n-aires...) :

=> 0,1 -> 0..1

=> 1,1 -> 1

=> 0,n -> 0..*

=> 1,n -> 1..*

- > une relation porteuse de propriétés est en plus transformée en une classe-association, reliée à l'association représentant la relation et accueillant les attributs correspondant aux propriétés
 - > les relations "réflexives" suivent la règle ci-dessus
 - > l'héritage est transformé en généralisation
- Attention: UML perd le fait qu'il y avait éventuellement plusieurs héritages sémantiquement distincts dans le MCD
- > une relation "identifiante" (participant à une identification relative) se transforme en une association avec une agrégation
 - > les contraintes inter-relation se transforment en contraintes (éventuellement stéréotypées)
- Ces règles couvrent la majorité des cas (ce qui veut dire que l'on pourrait sans doute en identifier d'autres).

CIF (ou dépendance fonctionnelle) de A à Z

Auteurs : Nanci ,

1/ Dépendance fonctionnelle et CIF

Dans Merise, ces deux termes recouvrent globalement la même notion. Le terme dépendance fonctionnelle fait référence à une notion mathématique entre ensembles.

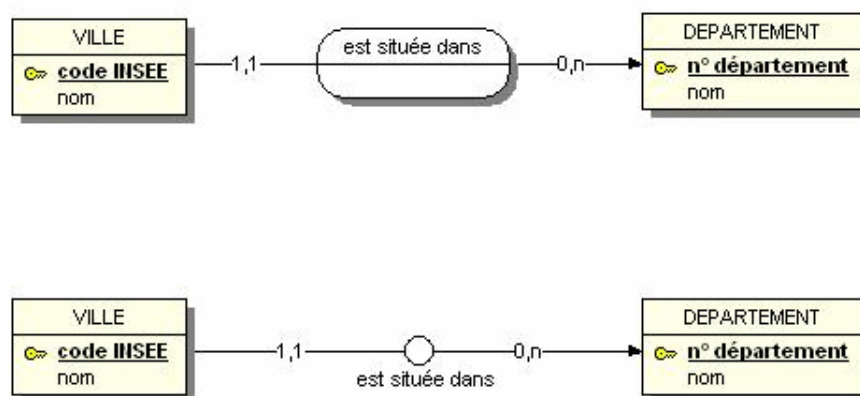
On dit que, entre deux ensembles A et B, il existe une dépendance fonctionnelle si à un élément (a) de A ne correspond qu'un élément (b) de B. On note cette dépendance $A \rightarrow B$. On appelle A l'origine, et B la cible de la dépendance fonctionnelle. Il existe également des dépendances plus complexes lorsque l'ensemble d'origine est composé de "n-uples" d'éléments $A \times B \rightarrow C$

Dans Merise cette notion s'applique explicitement aux relations; elle ne se modélise pas indépendamment des relations; elle ne prend son sens que dans le cadre d'une relation citée. On dit alors que cette relation est porteuse d'une dépendance fonctionnelle ou d'une CIF (contrainte d'intégrité fonctionnelle).

La mise en oeuvre de la notion dépend de la dimension de la relation.

a/ Dans une relation binaire, une cardinalité maxi à 1 (0,1 ou 1,1) sur l'une des pattes induit obligatoirement une dépendance fonctionnelle.

Dans l'exemple suivant, on dit usuellement que la relation "est située dans" est (porteuse d') une dépendance fonctionnelle. Il est inutile d'utiliser un symbole explicite.



b/ Dans une relation de dimension > 2 (ternaire ou plus), la notion de dépendance fonctionnelle peut impliquer tout ou partie des entités de la relation et n'est pas systématiquement liée aux cardinalités. Elle a donc une modélisation explicite et prend alors le nom de contrainte d'intégrité fonctionnelle, représentée par un rond noté CIF, relié à la relation porteuse (trait pointillé) et aux entités concernées (trait plein) dont l'un est porteur d'une flèche (la cible de la CIF).

Si la dimension de la CIF (nombre d'entités impliquées) est inférieure à celle de la relation porteuse, alors on doit envisager une décomposition de la relation en plusieurs relations, avec généralement intégration de la CIF comme simple dépendance fonctionnelle d'une relation issue de la décomposition (cf. tout bon ouvrage sur Merise...).

Sinon, la CIF reste explicitement modélisée car elle a des conséquences sur la transformation en relationnel. Dans la pratique, seules des relations ternaires ou plus peuvent conserver une CIF (qui implique alors la totalité des entités de la relation concernée). Pour les cardinalités, si l'une des pattes de la relation est dotée d'une cardinalité maxi à 1, alors il y a alors autant de CIF vers les autres entités, et décomposition de la relation initiale.

En conséquence, les seules CIF "persistantes" explicitement et obligatoirement modélisées portent sur des relations ternaires (ou plus) avec toutes les cardinalités maxi à (n).

Dans l'exemple ci-dessous, qui évoque une organisation d'un examen multi sites, : - un Candidat est affecté à un Site ,
 - un Site comporte des Salles ,

- l'examen comporte des Matières,

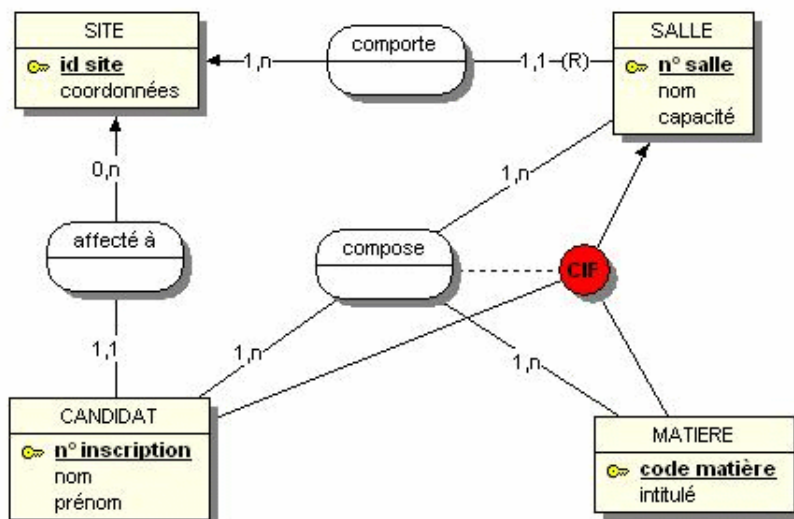
- un Candidat compose une Matière dans une Salle ,

- pour une Matière, un Candidat ne compose que dans une Salle (CIF)

Pourtant une Matière peut être composée par plusieurs Candidats et dans plusieurs Salles (1,n)

une Salle peut être le lieu de composition de plusieurs Candidats et plusieurs Matières (1,n)

un Candidat peut composer dans plusieurs Matières et plusieurs Salles (1,n)



Attention, ce sont les pattes de la relation qui portent les cardinalités et non celles de CIF. De plus la CIF doit être relié à la relation à laquelle la CIF s'applique.

Comment gérer la notion de code postal ?

Auteurs : Nanci ,

Le code postal en France identifie le bureau distributeur qui achemine le courrier dans une commune. En conséquence et d'après cette définition, il n'EXISTE pas de relation entre le code postal et le code du département de la commune (relisez la définition si vous n'êtes pas convaincu !). Prenons l'exemple d'une commune limitrophe d'un département, « La FEUILLADE », dont le code postal est 19600, est située dans le département « 24 (Dordogne) ».

Le code postal n'identifie que le bureau distributeur; et il assez rare qu'une application gère explicitement les bureaux distributeurs (sauf les spécialistes du routage...). Il n'y a pas de correspondance biunivoque entre le code postal et une ville. Une commune peut avoir plusieurs codes postaux, un code postal peut recouvrir plusieurs communes !

D'ailleurs, la norme Poste de codage des adresses distingue bien Ville, Code postal, Bureau distributeur .

Dans cette non correspondance entre code postal et département, il y a toute la Corse ! Les codes postaux (restés numériques) sont tous 20xxx 20000 Ajaccio (Corde du Sud) , 20200 Bastia (Haute-Corse).

Attention donc dans vos modèles!

Quels sont les différents héritages possibles ?

Auteurs : Cian ,

Soit l'exemple suivant : Une personne est soit un contact soit un utilisateur. L'entité "mère" est PERSONNE. Les entités CONTACT et UTILISATEUR héritent de PERSONNE.

Ils existent trois types d'héritages possibles à choisir selon les contraintes de son projet.

1/ XT (exclusion et totalité sous Win Design) = héritage avec partition :

Une personne est soit un contact, soit un utilisateur (mais pas les deux). L'union des contacts et des utilisateurs constitue l'ensemble des personnes.

2/ X = héritage avec exclusion :

Pour les cas où l'union des contacts et des utilisateurs ne constitue pas l'ensemble des personnes : une personne n'est ni un contact ni un utilisateur. C'est pour utiliser l'héritage comme pourvoyeur de propriétés identiques plus que dans un contexte sémantique semble-t-il.

3/ T= héritage avec totalité :

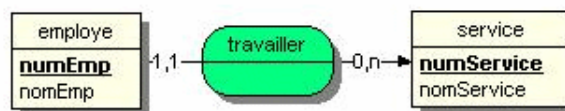
Pour représenter qu'il existe des personnes à la fois contact et client.

Comment créer une clé étrangère dans mon MCD ?

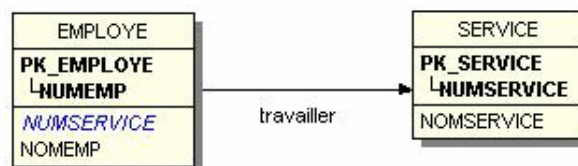
Auteurs : Cian ,

On ne crée pas de clé étrangère dans un MCD!! La clé étrangère sera générée automatiquement lors de la génération du MLD avec l'outil.

Soit le MCD suivant :



Le MLD généré est :



On remarque la création d'une clé étrangère (NUMSERVICE) dans l'entité employé sachant qu'un employé travaille dans un et un seul service.

Sommaire > MLD

Comment transformer un MCD en MLD ?

Auteurs : Nanci ,

Les règles basique sont les suivantes:

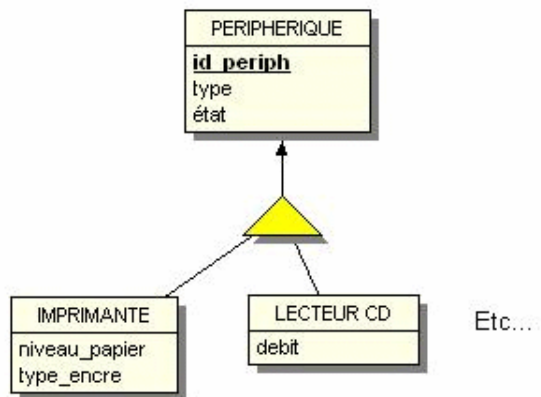
- 1 Pour les entités**
 - Toute entité devient une table.
 - L'identifiant de l'entité devient une clé primaire de cette table.
 - Les propriétés de l'entité deviennent des attributs
- 2 Pour les relations, cela dépend de la dimension et des cardinalités**
 - Relation binaire avec une cardinalité *,1
 - La relation devient un lien référentiel avec une clé étrangère dans la table correspondant à l'entité coté cardinalité *,1
 - Relation binaire avec des cardinalités *,n -*,n
 - La relation devient une table et des liens référentiels vers les tables correspondant aux entités composant la relation. La clé primaire de cette table est composée des clés étrangères référant aux clés primaires. Les éventuelles propriétés de la relation deviennent des attributs de la table.
 - Relation ternaire ou plus : Idem 2.2

lien :  [A voir également](#)

Comment transformer l'héritage, les sous-types du MCD dans le MLD ?

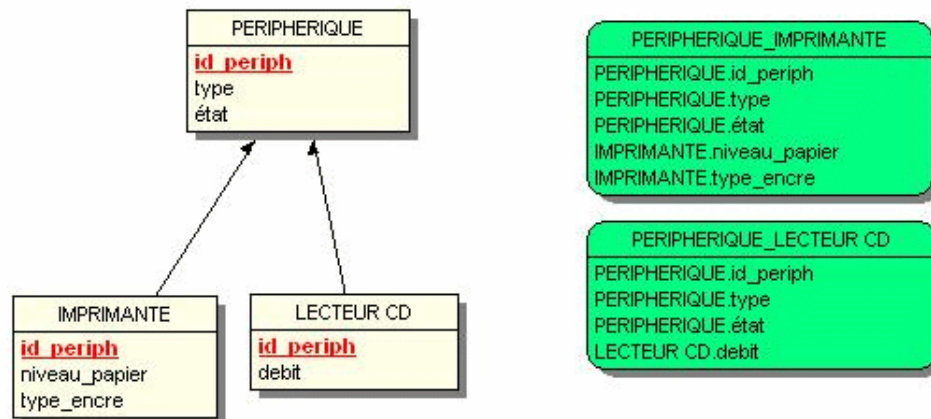
Auteurs : Nanci ,

Soit le MCD suivant :



La théorie nous indique qu'il existe 4 possibilités de transformation

- **Ne dupliquer dans les tables sous-type que l'identifiant**
 - Cette solution ne pose aucun problème de redondance. Elle est celle qui minimise la place. On peut "reconstituer" les héritages par des vues SQL.
- Le MLD suivant est UNE solution possible :



- **Dupliquer la totalité du surtype dans les sous types**
 - On a certes une redondance à gérer mais l'accès est plus facile car on a l'intégralité des informations d'un périphérique spécifique dans une table.
- **Ne conserver que les sous-types après avoir dupliqué le contenu du surtype**
 - Conserve les avantages de la solution 2.2 en supprimant la redondance; mais cela exige que l'héritage est un partition.
- **Tout remonter dans le surtype**
 - Solution de facilité, une seule table; mais sûrement un grand nombre de valeurs "à vide"; ça dépend comment la base de données gère ces vides.

Comment transformer une relation réflexive ?

Auteurs : Nanci ,

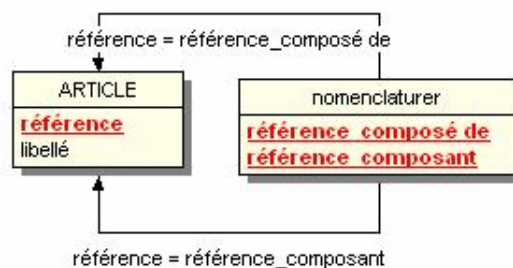
Les relations réflexives suivent les mêmes règles de base que les autres relations. Pour une relation réflexive 0n # 0,1 Elle devient un attribut clé étrangère dans la table transformée, attribut éventuellement suffixé par le rôle de la patte 0,1. Cette clé étrangère réfère à la clé primaire de cette même table.

Attention certains SGBD (Access par exemple) ne supportent pas cette modélisation, il faut alors dédoubler ces tables.

Pour une relation réflexives 01 # 0,1

Il y a alors deux clés étrangères (donc deux liens référentiels) nommées sur le principe ci-dessus. L'optimisation conduira certainement à la suppression d'une d'entre elles. Pour une relation réflexives 0n # 0,n

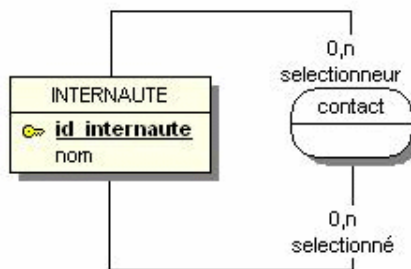
Elle devient une table, avec les éventuels attributs appartenant à la relation, dont la clé primaire est composée des deux attributs clé étrangères, référant respectivement à la clé primaire de la table (ex-entité). Ces attributs sont suffixés (ou renommés) par le rôle de chacune des pattes correspondantes (d'où l'importance de définir ces rôles dans les outils#).



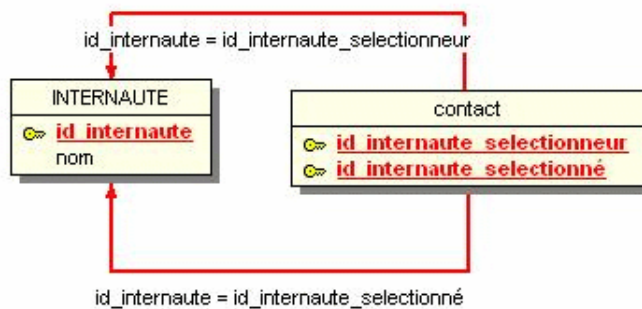
Comment faire pour générer un MLD ou un MPD avec des noms d'attributs "parlants" avec une relation réflexive dans son MCD ?

Auteurs : Nanci ,

Pour éviter d'obtenir des noms d'attributs quasi identiques dans chaque table (MPD) ou entité (MLD) lors d'une génération automatique (PowerAMC par exemple), il faut préciser les rôles sur chaque patte de la relation. Ainsi, le MCD suivant :



La transformation en MLD donne automatiquement le résultat suivant



Sommaire > Diagramme des flux

Que sont "les flux d'information" ?

Auteurs : Nanci ,

Les flux d'informations sont un échange d'informations (message) entre des acteurs (externes ou internes au système étudié) et le domaine étudié. On appelle Diagramme des flux, une modélisation qui représente uniquement ces flux échangés, sans chronologie et sans description des activités associées (en entrée ou sortie) à ces flux.

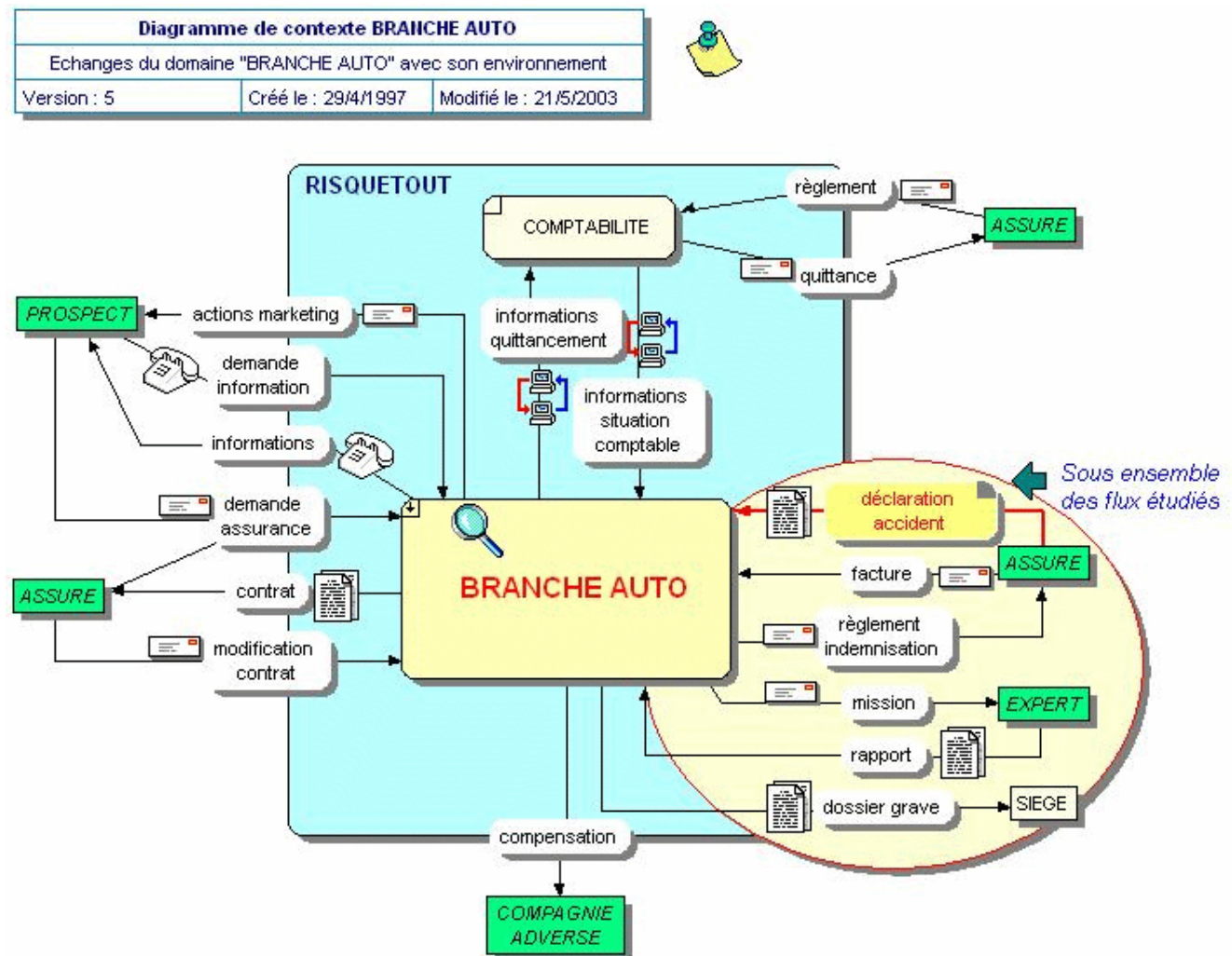
Qu'est-ce que le diagramme des flux ?

Auteurs : Nanci , Cian ,

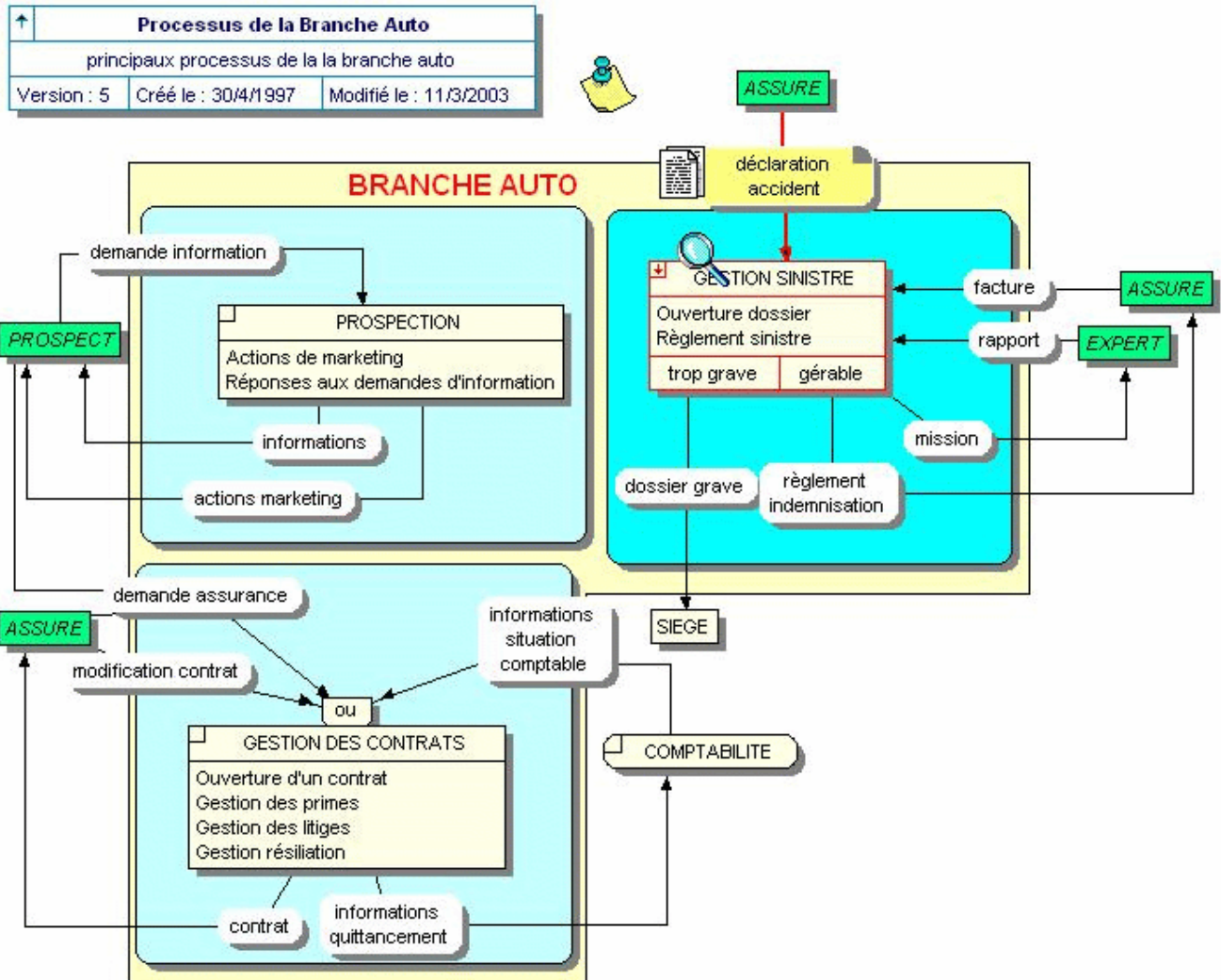
On appelle Diagramme des flux, une modélisation qui représente uniquement les flux échangés, sans chronologie ni description des activités associées (en entrée ou sortie) à ces flux.

On retrouvera évidemment ces mêmes flux dans un MCT (puis plus tard dans un MOT) mais cette fois-ci associés aux opérations qui composent le domaine étudié. Un flux (ou message) entrant dans le système déclenchera une opération; un flux sortant sera produit par une opération.

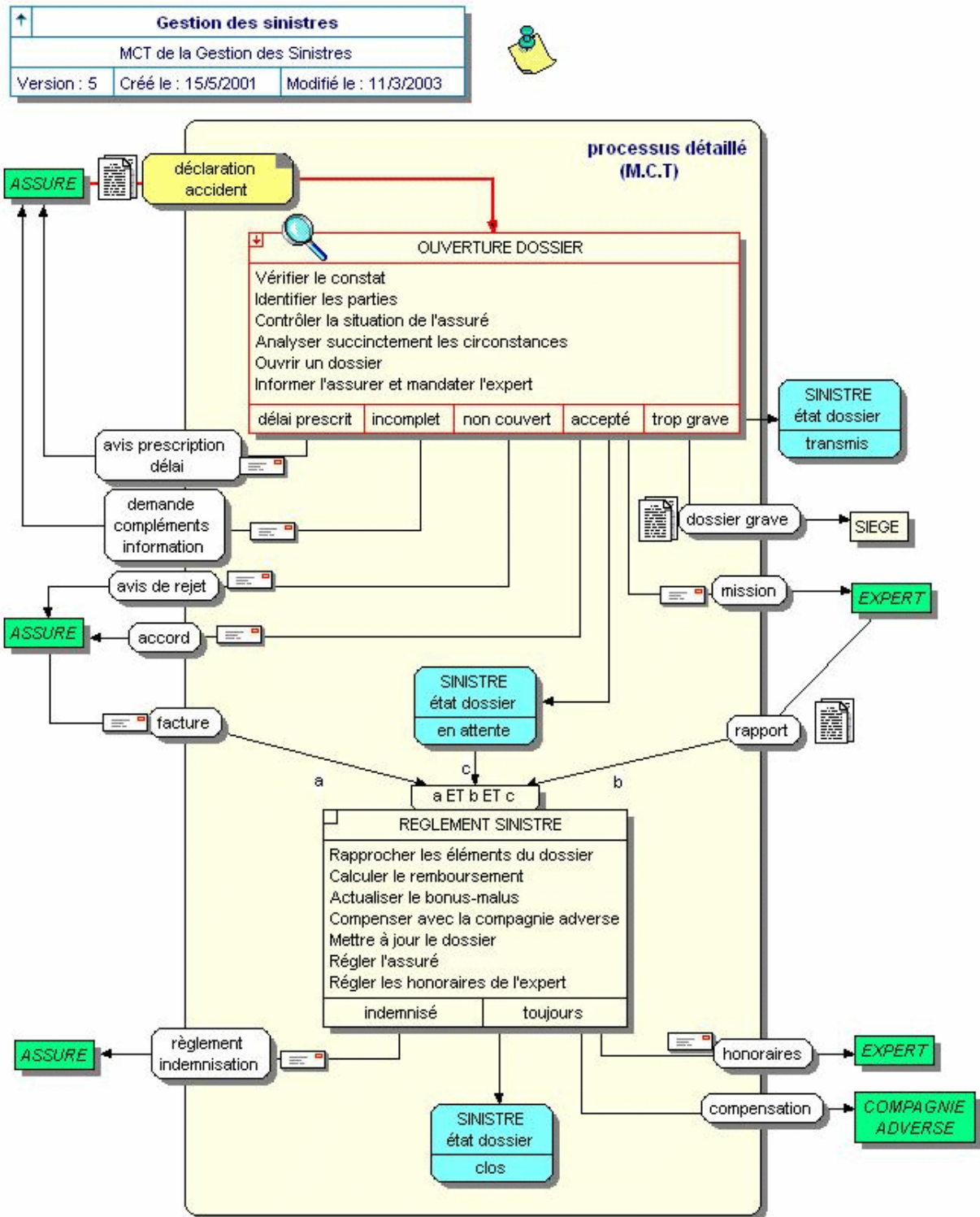
Le diagramme des flux ci-dessous est dit "de contexte" car il montre globalement les principaux flux entre le domaine étudié (la branche Auto) et les acteurs de son environnement.



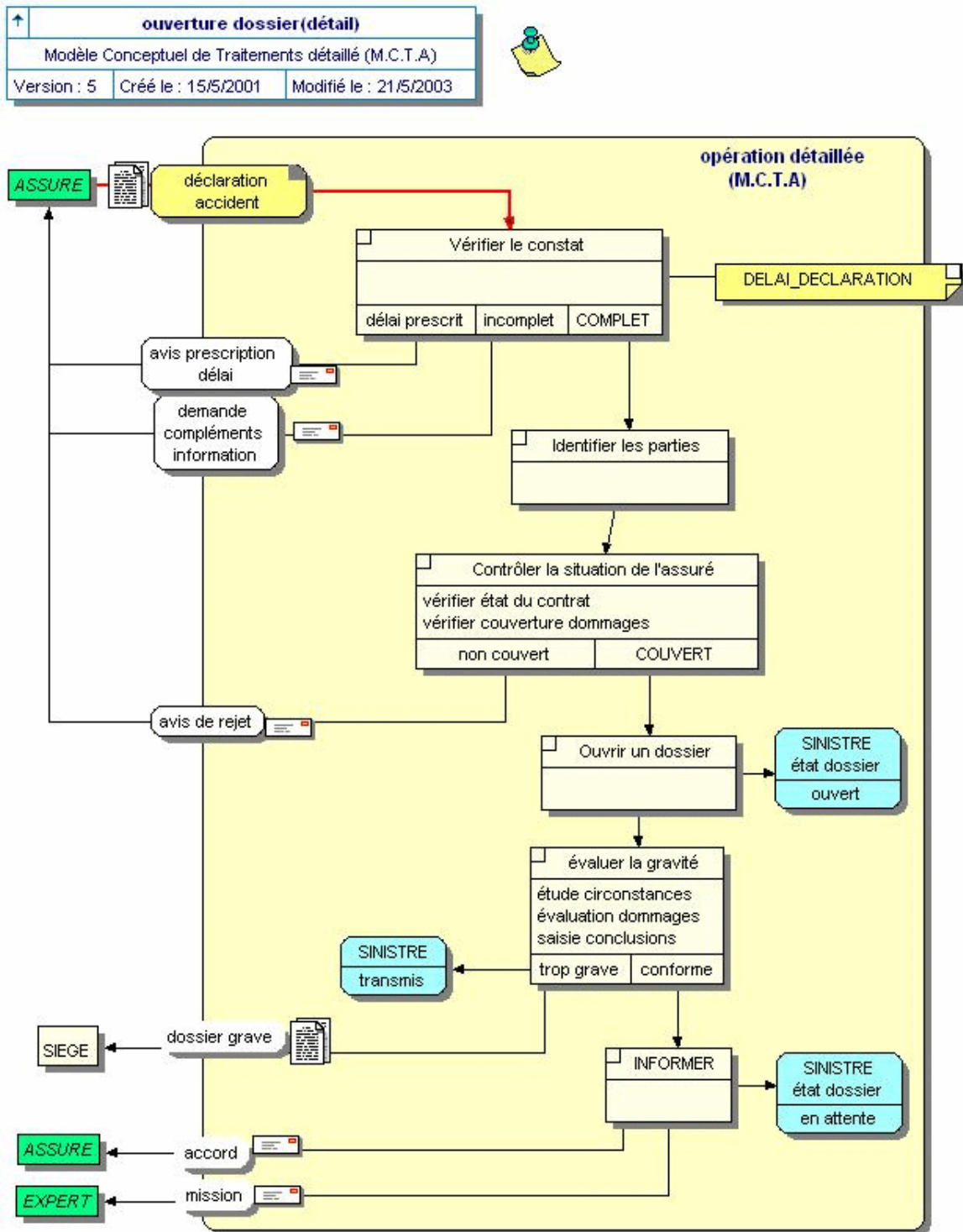
Le diagramme des flux suivant expose les principaux processus du domaine et les flux associés à ces processus. Un tel diagramme est une transition entre un diagramme des flux de contexte et un "macro-MCT".




Le diagramme ci-après est un MCT "classique" qui présente les deux opérations (les deux "temps" métiers) qui constituent le processus "Gestion des sinistres". Il existe des "règles" pour construire les opérations d'un MCT.



Ce dernier diagramme est un MCT analytique qui décompose en opérations élémentaires, l'opération initiale Ouverture du dossier.



A travers toutes ces modélisations, on voit bien que le flux d'informations "Déclaration d'accident" est partout présent avec des implications différentes. En terme de méthode, on perçoit d'abord les flux puis on dit à quoi il sert (quelles activités ils déclenchent).



lien :  [Que sont "les flux d'information" ?](#)

[Sommaire > MCT](#)

Qu'est-ce qu'un MCT ?

Auteurs : Nanci ,

Le MCT modélise les activités du domaine, activités conditionnées par les échanges avec l'environnement, sans prise en compte de l'organisation. Ainsi, chaque activité (nommée opération) regroupe un ensemble d'activités élémentaires réalisables au sein du domaine, sans autres informations extérieures (on a pas besoin de s'arrêter pour attendre des informations extérieures). De ce fait, une opération du MCT présente une vision macroscopique qui en fait l'intérêt dans l'analyse des processus métier, en particulier dans le Business Process Reengineering.

lien :  [Qu'est-ce que le diagramme des flux ?](#)lien :  [Qu'est-ce qu'une phase ?](#)

Peut-on réaliser le MCT avant le MCD ?

Auteurs : Nanci ,

On peut commencer l'analyse des processus par un MCT "classique" sans se préoccuper des données (et c'est comme ça que l'on fait sur le terrain !). Cette analyse est liée évidemment à l'étude des flux. Plus tard, beaucoup plus tard, quand on aura abordé les données, on pourra décomposer les opérations du MCT en opérations analytiques élémentaires (MCTA) autour des actions sur les ensembles homogènes de données.


Sommaire > MOT


Qu'est-ce qu'un MOT ?

Auteurs : Nanci ,

Le MOT modélise les activités du domaine, compte tenu de l'organisation et des ressources (qui, où, quand, comment). Ainsi, chaque activité (nommée tâche ou phase) est homogène en terme de finalité, degré d'informatisation, poste, instant de réalisation, ressources. Cette exigence conduit naturellement à la décomposition de l'opération (conceptuelle) en de nombreuses tâches (organisationnelles). Il y a, de fait, corrélation entre le changement de niveau (du conceptuel à l'organisationnel) et l'accroissement du degré de détail.

Le MOT décrit ce que font les gens du métier (leurs tâches ou phases) dont certaines sont informatisées. Il ne décrit pas comment marche l'application (partie informatisée) qui sert les tâches.

lien :  [Qu'est-ce qu'une phase ?](#)

lien :  [Qu'est-ce qu'une tâche ?](#)

Quels sont les types de phase d'un MOT ?

Auteurs : Nanci ,

Le type de phase recouvre ce que l'on appelle les caractéristiques organisationnelles d'une phase ou tâche (une activité de niveau organisationnel).

On distingue les caractéristiques suivantes :

- l'unité organisationnelle (ou poste)dans laquelle s'effectue l'activité.
 - Cela se représente en positionnant le symbole de l'activité dans une colonne (couloir de piscine....)
- le degré d'automatisation
 - il traduit l'utilisation des ressources humaines et/ou informatiques pour l'exécution de l'activité.

On distingue trois degrés

 - *manuels* :
 - seule la ressource humaine est mise en oeuvre (même si c'est une activité intellectuelle...)
 - *interactive ou conversationnelle* :
 - les ressources humaines et informatiques sont mobilisées dans le déroulement de l'activité (même de manière alternative dans le courant de l'activité, globalement, les 2 sont mobilisées)
 - *automatique* :
 - durant le déroulement de la phase ou tâche, seule la ressource informatique est utilisée
- le délai de réponse
 - il traduit la rapidité de prise en compte de l'événement (message) déclencheur de l'activité.

On distingue deux valeurs:


 - *réponse immédiate* :
 - dès la survenance de l'événement, et si les ressources sont disponibles, l'activité traite l'événement.
 - *réponse différée* :
 - le déclenchement de l'activité attend une condition complémentaire (délai, ordre, ...) .
- le mode de fonctionnement
 - Il traduit comment l'activité prend en charge les occurrences d'événements.

Il peut prendre deux valeurs:

 - *unitaire* :
 - l'activité traite les événements un par un et libère ses ressources à chaque occurrence qui deviennent alors disponibles soit pour prendre une autre occurrence avec la même activité, soit pour exécuter une autre activité.
 - *par lots* :

- **l'activité prend en charge un lot (dont la taille est à préciser) de messages et mobilise les ressources jusqu'à la fin du traitement.**

Ces caractéristiques organisationnelles sont citées par ordre d'importance. Pour chacune, il existe bien sur la valeur "indéterminée". C'est au fur et à mesure de l'avancement de l'analyse que l'on va préciser les valeurs de ces caractéristiques. Toutes les combinaisons sont possibles; cependant, certaines sont classiques et fréquentes (par exemple automatique, différé, par lot)


lien :  **Qu'est-ce qu'une phase ?**

Sommaire > Bibliographie

Quels sont les meilleurs ouvrages sur Merise selon developpez.com ?

Auteurs : Cian , SQLpro , Nanci ,

Les livres que nous vous conseillons sont indiqués ici :  Livres proposés par SqlPro

Egalement disponible, une panoplie d'articles sur www.developpez.com :  [articles developpez.com](http://www.developpez.com/articles)

lien :  [L'ingénierie des systèmes d'information avec Merise](#)

Sommaire > Outils

Sommaire > Outils > PowerDesigner/PowerAMC

Peut-on enlever le type des attributs dans le MCD ?

Auteurs : Cian , fadace ,

Dans les versions PowerDesigner post-v.7, le défaut se paramètre par Tools/Model Options.../Model/Default Data Type. Pour ne pas les afficher, c'est sous Tools/Display Preferences/Object View->Entity en décochant à droite Entity Attributes/Data types. Bien sûr, si c'est dans l'association, et pas dans l'entité que le problème se pose, agir de la même manière sous Object View->Association .

Comment donner le même nom à deux propriétés appartenant à deux entités différentes ?

Auteurs : fadace , Cian ,

On peut aisément le paramétrer sous Outils/Options du modèle, en décochant Code Unique.

Comment créer un MPD à partir de scripts SQL existants ?

Auteurs : fadace ,

Créer un nouvel MPD via Reverse engineering(Fichier -> Reverse Ingeneering -> base de données -> etc.^) en prenant comme source le fichier de script

Est-il possible de générer des bases de données MYSQL avec powerAMC ?

Auteurs : Cian , sophnouille ,

Oui, c'est possible dans la version 9.5.1 en tout cas : option proposée lors de la création du MPD (MYSQL 3.22 et MYSQL 3.23)

Comment donner les valeurs d'un champ de type SET ?

Auteurs : Cian , Bouboubou ,

Cliquer sur les propriétés de l'attribut et utiliser l'onglet "contrôles standard". Dans "Liste des valeurs possibles", ajouter les différentes valeurs (Exemple : 'Connecté', 'Déconnecté', 'Inactif'). Cela sera généré dans le MPD sous la forme de contrainte sur l'attribut.

Comment faire pour que l'outil remplace automatiquement les domaines définis par les types réels ?

Auteurs : Bouboubou ,

Dans les "options de génération de base de données", sous l'onglet "Tables et Vues" => "Colonnes" et décochez "Types utilisateur".

Sommaire > Outils > Win'design

Comment représenter un sous-type avec win'design 5.6 ?

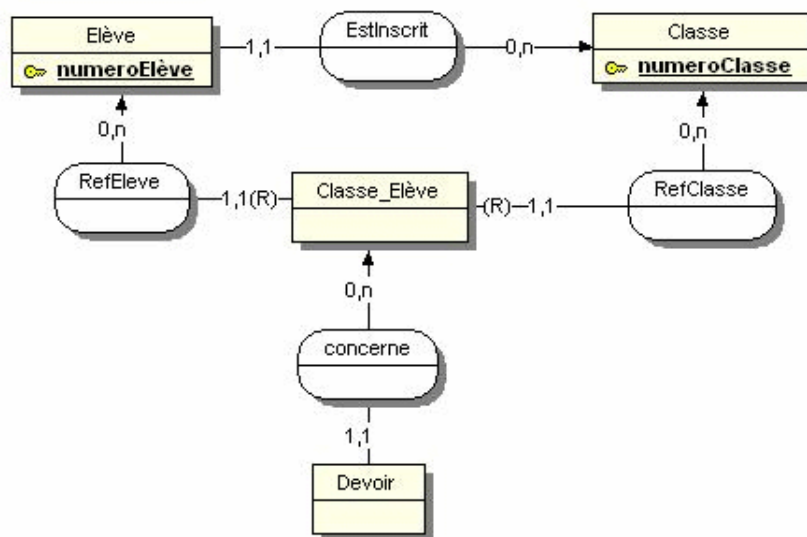
Auteurs : Nanci , Cian ,

Il suffit de dessiner les entités concernées , le sur-type et le(s) sous-type(s); puis de sélectionner l'ensemble et de choisir l'outil Héritage. Enfin dessiner ce symbole à l'emplacement souhaité puis spécifier enfin la patte relative au sur-type. Explication disponible dans l'aide de l'outil.

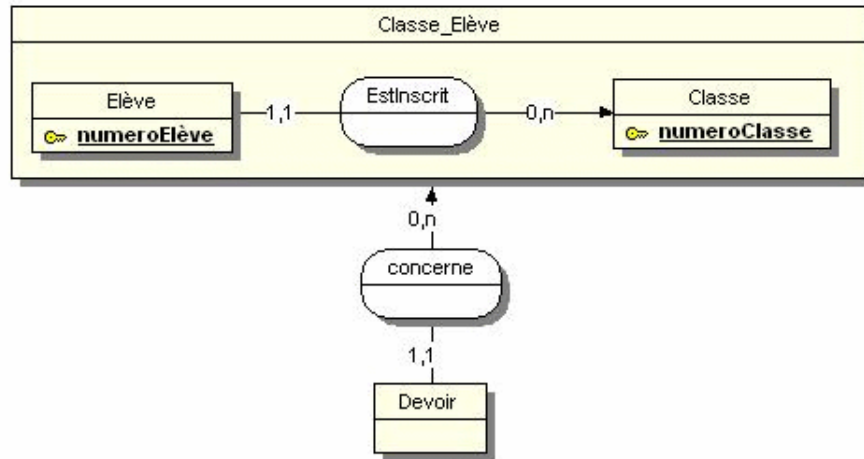
Comment représenter une agrégation avec win'design ?

Auteurs : Nanci ,

L'agrégation est définie comme "une abstraction qui transforme une association entre objets en un objet agrégé. Il est modélisable par la notion d'entité faible (weak entity), c'est à dire une entité doublement identifiée en relatif. (ce n'est plus une relation, mais c'est identifié comme une relation...). C'est la solution que met en oeuvre Win'Design. Le premier schéma présente la modélisation effective. L'entité `Class_Elève` est vide (pas d'identifiant propre, pas de propriété).



La suivante est la présentation graphique obtenue en masquant les relations Ref, en étendant l'entité `Classe_Elève` et en la supersosant avec les entités agrégées.



Vous remarquez l'utilisation d'un lien identifiant. Le lien identifiant, ou identification relative, indique que pour son identification, une entité a besoin d'un indentifiant attaché à une autre entité à laquelle il est relié (par une relation fonctionnelle). Dans le cas d'une double (ou plus) identification relative, on est équivalent à l'identification d'une relation.

Ainsi, l'agrégation, qui est une association qui devient un objet, va se modéliser par une entité faible qui ne comporte pas d'identifiant propre basé sur des propriétés intrinsèques. On reconstitue alors l'identification d'une relation, c'est à dire basée sur les identifiants des entités reliées, en ayant recours aux identifications relatives.







Il existe d'ailleurs dans Win'Design, une fonction qui transforme automatiquement une relation en une entité faible, c'est à dire avec autant de liens identifiants que de "pattes" de la relation initiale.

Sommaire > Exercices

Où trouver des exemples et exercices corrigés ?

Auteurs : Cian ,

Il existe peu de sites proposant des exercices de modélisation corrigés que ce soit pour Merise ou une autre méthode de modélisation. Cependant, www.developpez.com vous propose deux exemples : veuillez consulter l'url ci-dessous. N'hésitez pas non plus à parcourir le forum modélisation! Enfin, cette FAQ présente un certain nombre de petits exemples glanés dans le forum.

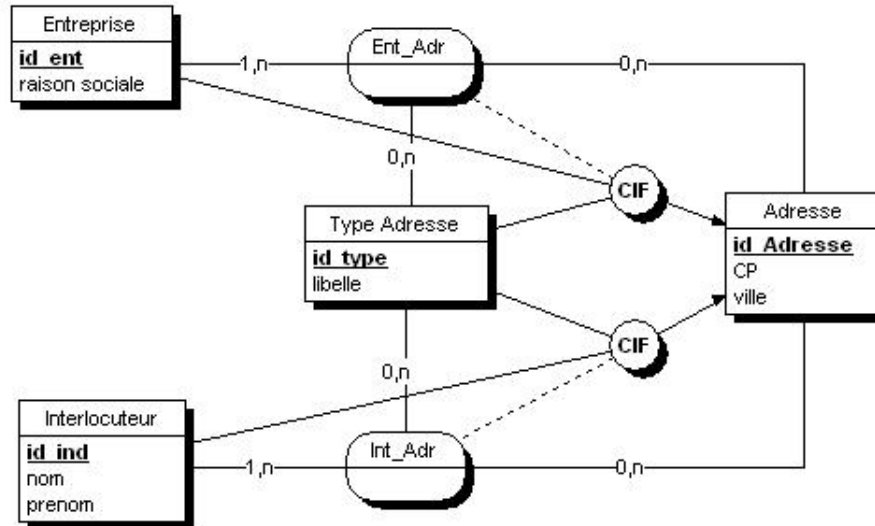
lien :  [Exemples de MCD](#)lien :  [Exemple de modélisation de la gestion d'un hôtel](#)lien :  [Forum Modélisation de Developpez.com](#)lien :  [Exemple de modélisation lié au thème de la bibliothèque](#)lien :  [Exemple de modélisation de page internet](#)lien :  [Exemple lié à la gestion d'un SAV](#)lien :  [Exemple lié à la gestion d'une vidéothèque](#)lien :  [Exemple d'une gestion d'appels téléphoniques](#)lien :  [Exemple lié à la gestion de construction de meubles](#)

Exemple d'utilisation d'une CIF

Auteurs : Nanci ,

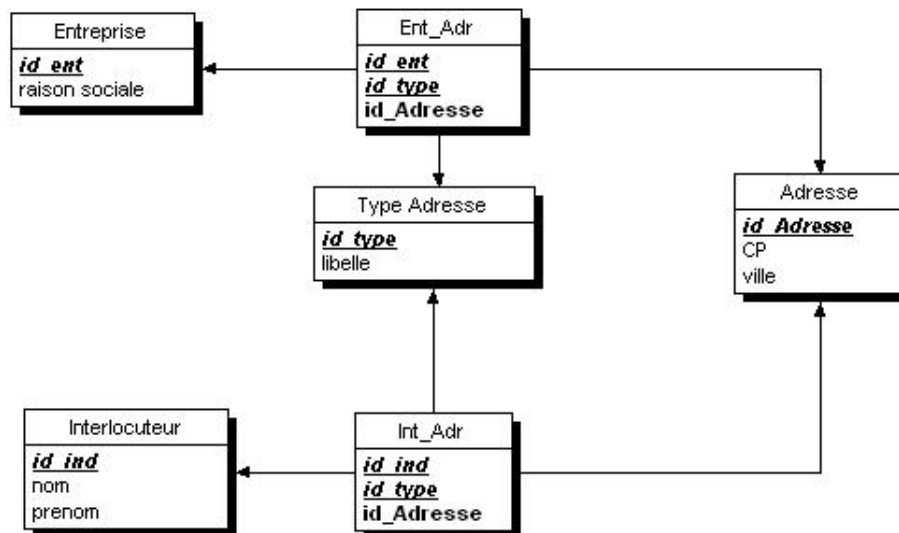
Énoncé : Les Interlocuteurs et les Entreprises ont plusieurs Adresses mais d'1 seul Type_Adresse à chaque fois (une entreprise ne peut pas avoir X adresse type "Siège Social") . Comment gérer cela ?

Modèle et explication: La question posée relève de la notion de contrainte d'intégrité fonctionnelle (CIF pour les branchés). En effet, une Entreprise (ou un Interlocuteur) ne peut avoir qu'une seule Adresse correspondant à un Type d'adresse donné; mais des adresses différentes pour des types différents. Soit Entreprise x Type_Adresse ---> Adresse. C'est exactement la définition d'une CIF. La modélisation conceptuelle est la suivante



Réalisé avec Win'Design

Elle se transforme en MLD relationnel suivant, conformément aux règles :



Exemple de modélisation lié au thème de la bibliothèque

Auteurs : Nanci , skapull , Cian ,

Enoncé :

L'énoncé est assez classique. Nous sommes dans un contexte de bibliothèque où un client peut emprunter des livres. Ces livres existent en plusieurs exemplaires. Comment gérer la disponibilité du livre compte-tenu de son nombre d'exemplaire ?

Modèle et explication :

1/ Il existe tout d'abord deux entités : LIVRE et EXEMPLAIRE. La relation Posséder entre Livre et Exemple est indiscutable :

[LIVRE]--1,n--(posséder)--1,1--[EXEMPLAIRE]

et donc à un exemplaire ne correspond qu'un livre !

2/ Maintenant, la relation Emprunter est plus subtile. En effet, un Client emprunte un Exemple (donc un Livre compte tenu de la remarque précédente); donc

[CLIENT]--1,n--(emprunter)--0,?--[EXEMPLAIRE]

La cardinalité 0,? mérite discussion.

Si 0,n , alors on aura l'historique des emprunts (vu d'un exemplaire) on pourrait alors avoir les dates d'emprunt/retour dans la relation.

Si 0,1, alors ça sera l'emprunt "ponctuel"; l'occurrence de la relation ne serait mise en place que pour la durée de l'emprunt,; la date d'emprunt serait une propriété de la relation .

Personnellement, je préfère le 0,n .

Dans la pratique, je ferais plutôt la modélisation suivante :

[CLIENT]--0,n--(effectuer)--1,1--[EMPRUNT]--1,n--(comporte)--0,n--[EXEMPLAIRE]

3/ La disponibilité mérite également discussion.

S'il s'agit de la disponibilité d'un exemplaire, alors Disponibilité est une propriété d'Exemple (le fait que la valeur de la disponibilité puisse être calculée n'est pas un motif à ce qu'elle ne soit pas identifiée comme étant une propriété). Toutefois, si l'on utilise la cardinalité 0,1, alors un Exemple disponible est un exemplaire qui n'est pas emprunté (donc sans Client!)

S'il s'agit de la disponibilité du Livre, alors, il se détermine par une règle : un Livre est disponible s'il existe au moins un exemplaire disponible. Ce qui incidemment renvoie à la disponibilité de l'exemplaire ! Cette règle sera probablement implémentée par une sélection.

En conclusion, il semble qu'une propriété Disponibilité soit vraiment nécessaire pour l'entité Exemple.

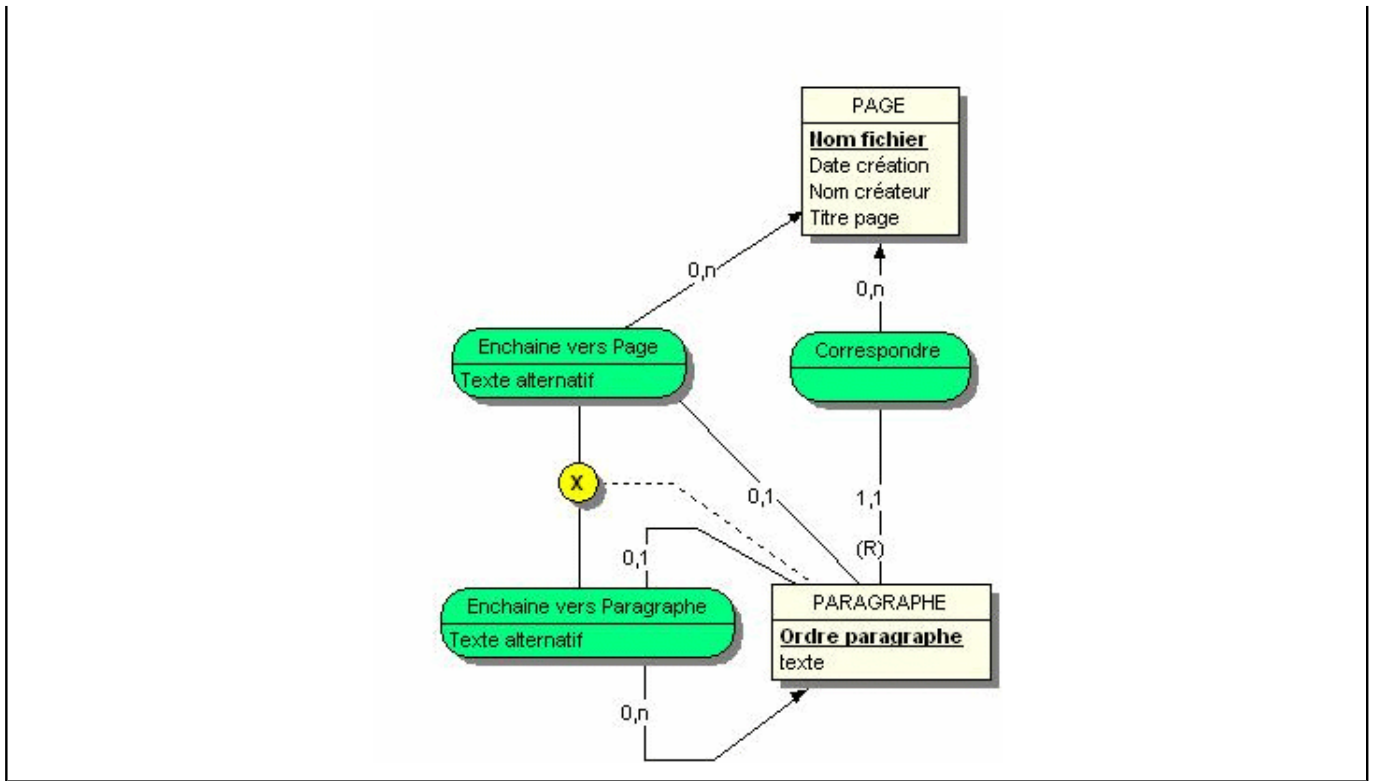
Exemple de modélisation de page internet

Auteurs : Nanci , Cian , Petrus ,

Enoncé :

Il s'agit de représenter le fait qu'"un paragraphe(de page web) peut être lié à un autre paragraphe ou une autre page web" . U

Modèle et explications :



Exemple lié à la gestion d'un SAV

Auteurs : Nanci ,

Énoncé :

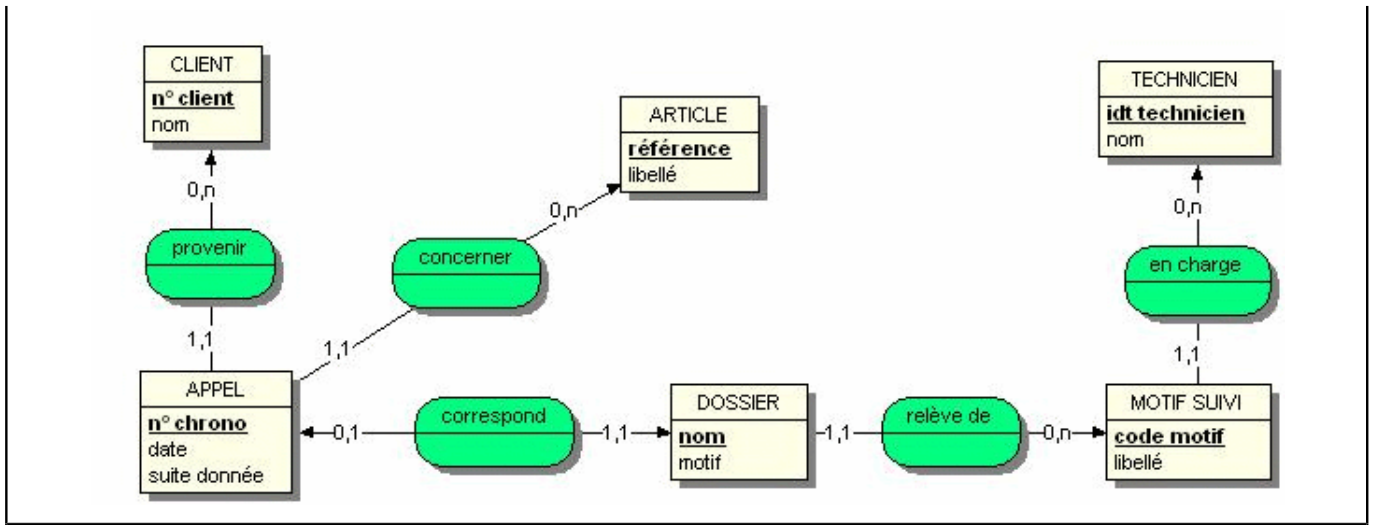
Gérer des appels téléphoniques arrivant à un service après-vente.

On veut savoir quels clients, quels articles et quelles périodes représente le plus de charges pour le service après-vente:

- 1- on note tous les appels entrant
- 2- L'appel doit provenir d'un client repertorié dans une liste de clients ayant droit au service après-vente.
- 3- L'appel doit concerner un article repertorié dans une liste d'article suivi en service après-vente.
- 4- On peut soit classer l'appel sans suite, soit ouvrir un dossier suivi, on donne alors un nom au dossier de suivi.
- 5- A chaque motif de suivi correspond un technicien de maintenance qui aura la charge du dossier.

Modèle et explications :

Une proposition de MCD :



Exemple lié à la gestion d'une vidéothèque

Auteurs : Nanci , Cian ,

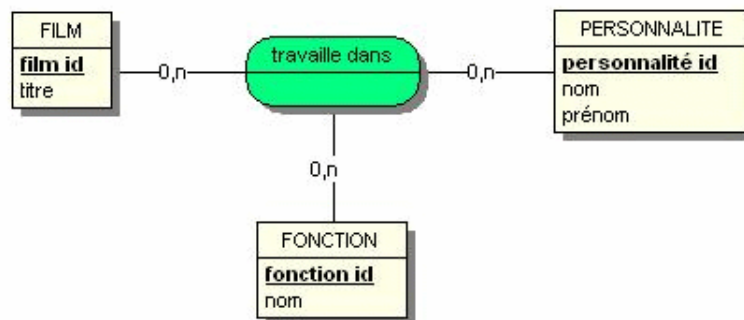
Enoncé :

On souhaite pouvoir gérer les films, les acteurs ainsi que leur rôle dans chacun des films.

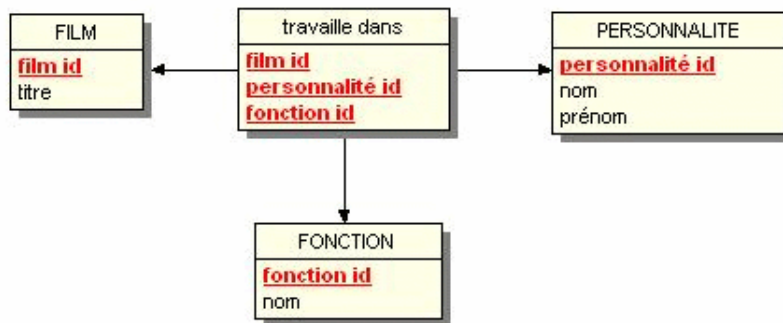
Modèle et explications :

Nous considérons donc 3 entités : **FILM**, **PERSONNALITE**, **FONCTION**. Naturellement, une **PERSONNALITE** est liée à un **FILM** par une relation **TRAVAILLE_DANS**. Dans ce cas, **TRAVAILLE_DANS** devra être une relation ternaire puisqu'il faut lier la **FONCTION** à la **PERSONNALITE** et au **FILM**.

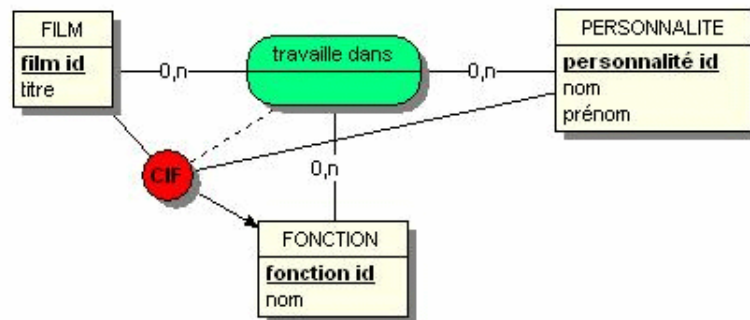
Le MCD correspondant :



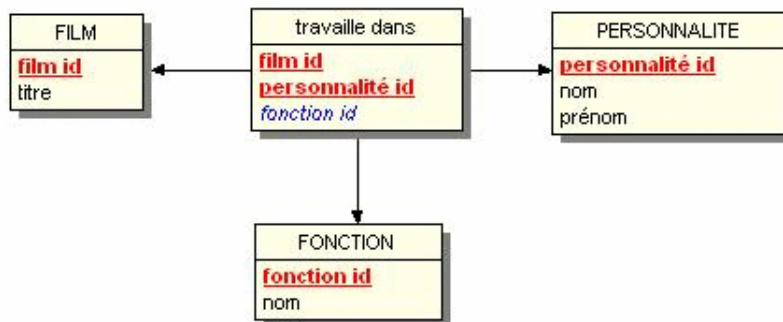
Le MLD correspondant :



Toutefois, si l'on optait pour des Fonctions composites (une personnalité pourrait être à la fois acteur/ réalisateur pour un film, sans saisie de deux enregistrements), on pourrait modéliser une contrainte d'intégrité fonctionnelle (CIF) au sein de la ternaire: une Personnalité ne travaille dans un Film qu'avec une Fonction (Personnalité x Film --> Fonction).



On obtiendrait alors la modélisation logique où Fonction_id n'est que clé étrangère dans la table Travaille .



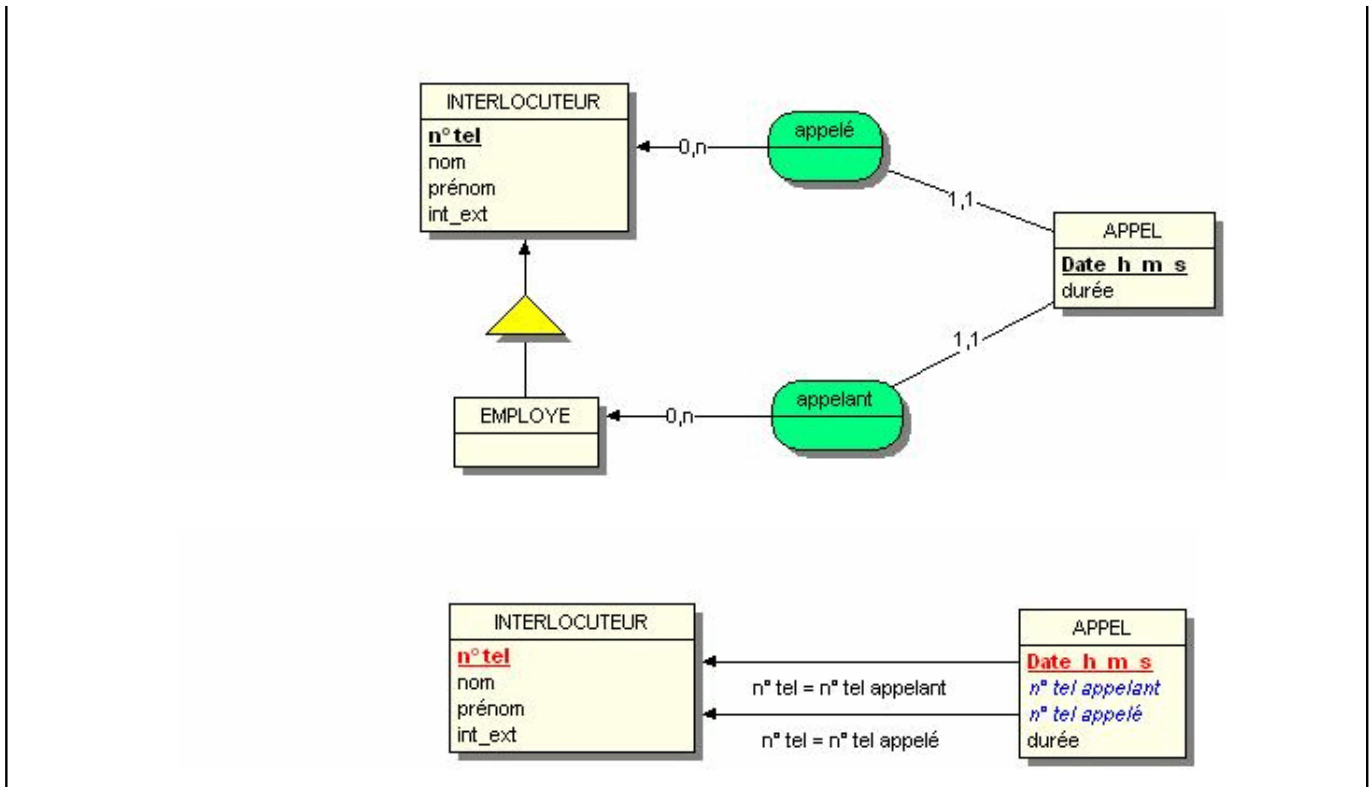
Exemple d'une gestion d'appels téléphoniques

Auteurs : Nanci ,

Enoncé :

un employé effectue des appels téléphoniques. Le correspondant peut être externe à l'entreprise ou être lui-même un employé.

Modèle simple proposé :



Exemple lié à la gestion de construction de meubles

Auteurs : Nanci ,

Enoncé : Un fabricant de meubles souhaite informatiser les modes de conception des différents meubles.

La fabrication d'un meuble nécessite divers matériaux. On aura par exemple besoin de 20 vis cruciformes 20x45 et de deux panneaux de 1x0,5 m ni de contre-plaqué 3 mm.

Chaque matériau est identifié par son nom (eau bois, vis, clou ...), sa matière (chêne, contre-plaqué, métal, laiton, plastique...) et ses caractéristiques éventuelles (épaisseur pour du bois, dimensions et type pour une vis...).

Le temps d'assemblage d'un meuble doit être stocké. Chaque matériau peut être acquis auprès de différents fournisseurs (à des prix différents évidemment). Les fournisseurs sont décrits par les propriétés habituelles. Bien entendu, il faut connaître l'unité de tarification de chaque matériau (le bois est tarifé au M2 ou au mètre selon sa nature -panneau ou baguette-, la quincaillerie au kilo ...). L'unité dépend du matériau.

Il faut également gérer les différentes étapes de conception du meuble dans l'ordre (1ère étape, 2ème étape ...). On devra connaître pour chaque étape

- sa durée ;
- les éléments du meuble qui sont concernés (par exemple : les deux panneaux de bois 1x0,5 m ni et 5 vis cruciformes 20x45)
- les outils nécessaires
- ce qu'il faut faire !

Chaque étape est d'un type précis (peinture, découpe, assemblage, vernissage ... À chaque type est associé un tarif horaire. Cela permettra de connaître le coût d'assemblage du meuble.

Proposition de modèle :

La contrainte d'inclusion exprime que tout matériau concerné par une étape de conception d'un meuble, fait partie des matériaux nécessaires à la réalisation de ce meuble

