

GUIDE
PRATIQUE

MODÈLE CONCEPTUEL
DES TRAITEMENTS

MODÈLE ORGANISATIONNEL
DES TRAITEMENTS

Présentation théorique de Merise

Objectifs

- Définir, analyser, concevoir et spécifier tout projet d'organisation d'un système d'information
- Ni méthode de conduite de projet, ni méthode de programmation ou d'algorithmique
- En aval du schéma directeur, en amont de la réalisation

Principes

- Approche globale, intégrant tous les sous-systèmes
- Conception descendante, partant des finalités de chaque activité
- Etude indépendante des données et des traitements, puis rapprochement pour valider l'étude des données avec les résultats de l'étude des traitements, et réciproquement.
- Approche par étapes (Conceptuelle, puis logique, enfin opérationnelle)
- Recherche des invariants du système d'informations
- Utilisation d'un formalisme facilitant la lecture et la communication

A partir des deux principes de séparation de l'analyse des données et de l'analyse des traitements d'une part, et d'une démarche en trois étapes, on obtient les questions à se poser dans le tableau suivant :

	Analyse des données	Analyse des traitements
Niveau conceptuel	Quelles informations manipule-t-on ?	Que veut-on faire ?
Niveau logique	Comment structurer ces données ?	Qui fait quoi, où, quand ?
Niveau physique	Où les stocker ?	Comment ?

A chacune de ces six questions, il s'agira d'amener des réponses. Le tableau suivant présente les documents qu'e la méthode Merise produit pour y répondre.

	Analyse des données	Analyse des traitements
Niveau conceptuel	Modèle conceptuel des données (M. C. D.)	Modèle conceptuel des traitements (M. C. T.)
Niveau logique	Modèle logique des données (M. L. D.)	Modèle organisationnel des traitements (M.O.T.)
Niveau physique	Tables et index	Procédures

Dans le cadre de l'utilisation d'un S.G.B.D., le concepteur est déchargé de l'implantation physique des tables. D'autre part, Merise ne guide pas le concepteur dans la production des procédures, car elles sont dépendantes du choix du système, des outils et des machines. Les seuls niveaux analysés sont donc les niveaux conceptuel et logique.

L'expérience m'a amené à douter de l'efficacité de l'analyse des traitements (M.C.T et M.O.T). De plus cette conception est en partie remise en cause par les technologies objet développées dans les outils modernes. Ce cours se contentera donc d'indiquer la théorie de l'élaboration d'un M.C.T, puis d'un M.O.T., sans approfondir les aspects pratiques.

Guide pratique de Merise

I - La réalisation d'un M.C.T.

I.1 - Ce qu'on attend d'un M.C.T.

But :

- Il s'agit de représenter, par un formalisme précis et en grande partie standardisé, l'ensemble des traitements que l'on doit réaliser pour répondre aux attentes du projet défini en amont de l'analyse (dans le schéma directeur).

principes :

- IL FAUT OUBLIER LES MOYENS QUI SERONT MIS EN ŒUVRE POUR LA RÉALISATION. (il s'agit uniquement de décrire le problème à traiter, et pas du tout de préciser, simplifier ou guider les choix qu'on sera plus tard amenés à faire)
- Par "moyens mis en œuvre", il faut entendre machines et systèmes d'exploitation, mais aussi S.G.B.D, langages, outils et aussi culture informatique et maîtrise des produits par les développeurs. Tous ces points doivent impérativement être oubliés dans cette phase.
- Chacune des huit étapes décrites répond à une question élémentaire. Il ne faut surtout pas essayer de préparer le terrain pour les étapes suivantes. Il faut modestement se concentrer sur la seule question traitée par cette étape.

Remarques :

I.2 - Les huit étapes de la réalisation d'un M.C.T.

I.2.A - Le collectage des acteurs et des événements-messages

- But :**
- Collecter l'ensemble des procédés amenant une modification des valeurs des attributs manipulés par le système, et conceptualiser ces procédés en événement-messages (actions amenant une modification des données) et acteurs (ressources à l'origine ou à la réception de l'événement-message)
- Moyens :**
- Collecter au moins deux occurrences de chacun des documents, écrits ou non, manipulés par le système.
 - Repérer chacun des acteurs du système. Par acteur, on entend l'émetteur ou le récepteur de tout ou partie d'un document.
 - Pour chaque type de document, analyser l'ensemble de ses occurrences, en repérant chaque événement-message porté par le document, chaque événement-message à l'origine de ce document, et chaque événement-message ayant pour conséquence une modification de ce document au cours du temps.
- précautions :**
- Il faut s'intéresser à ce qui va modifier les valeurs d'occurrences d'attributs ("modifier des données") sans s'intéresser aux données elles-mêmes.
 - Il n'est pas facile de déterminer si deux réponses différentes à une question représentent deux occurrences d'un message ou deux messages différents (ex : si réponse-devis est "accepté" ou "refusé", s'agit-il du même message ?). Considérer que si deux réponses différentes ont pour conséquence des traitements différents, il s'agit de deux messages distincts.
- Limites :**
- Les informations les moins abstraites des événements-messages sont les données que le M.C.D. structure. Par conséquent, une "concrétisation" du M.C.T. ne peut se faire qu'en s'appuyant sur le M.C.D.
 - On ne connaît pas de moyens de définir d'une façon non ambiguë "l'univers du discours", c'est-à-dire les traitements qui font ou ne font pas partie du problème à traiter.
- Remarques :**

EXEMPLE : Réparateur horloger

Acteurs :

Client
Réparateur
Service comptable

Événements-messages :

Dépôt de la montre
Devis
Acctptation de réparation
Refus de réparation
Montre réparée
Facture à régler
Paiement
Carte de garantie
Facture acquittée
PRésentatnion facture acquittée
Montre rendue

I.2.B - Le diagramme de flux des données (abr : DFD)

NB : afin de simplifier l'écriture, ce document remplacera indifféremment le terme "événement-message" par "événement" ou "message".

But :

- Représenter sous forme compacte, et par conséquent plus lisible, l'ensemble des acteurs et des messages les reliant.

Moyens :

- Présenter les acteurs dans des ovals, et les messages sous forme de flèches entre acteurs à l'origine et à la réception du message

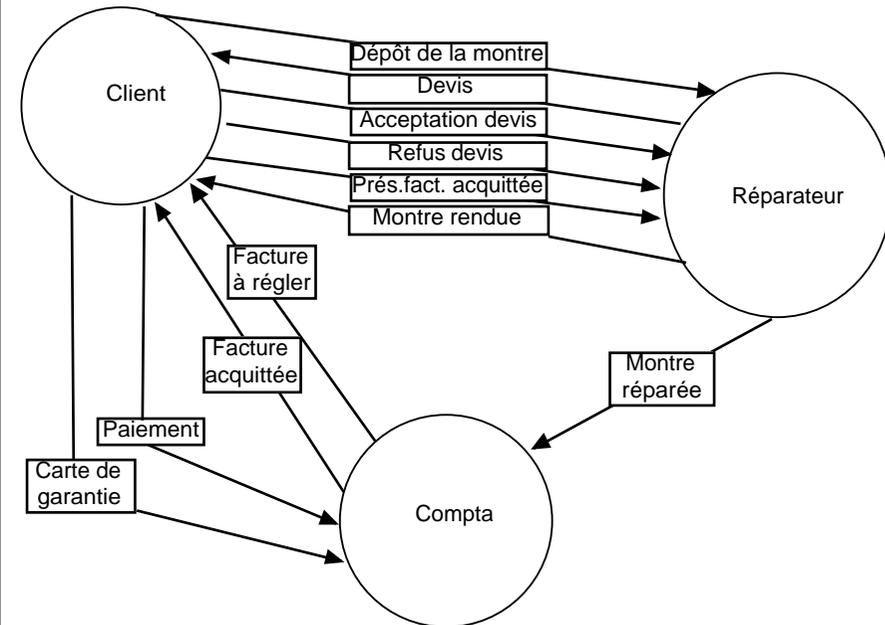
précautions :

- Ne pas tenter de séquencer les messages : considérer dans cette étape que chaque message est indépendant.

Remarques :

EXEMPLE : Réparateur horloger

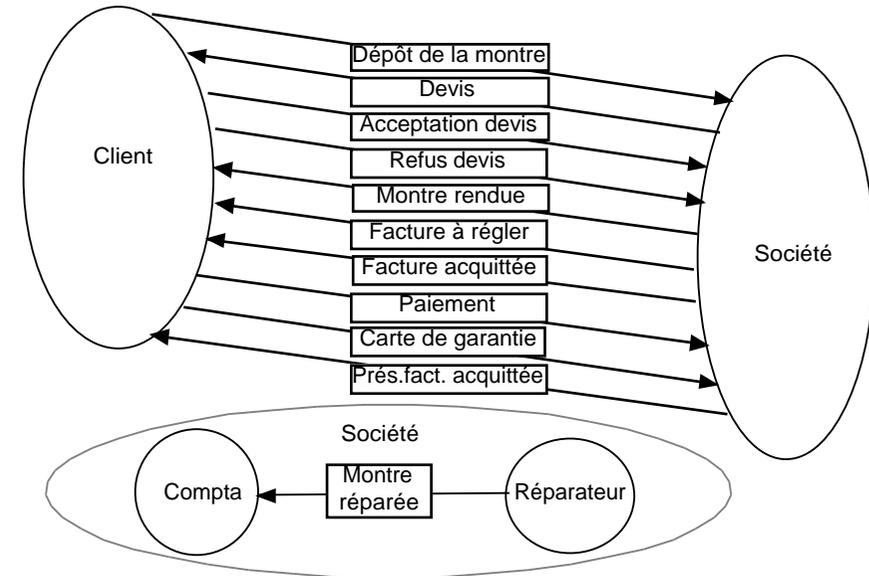
Diagramme de Flux des données :



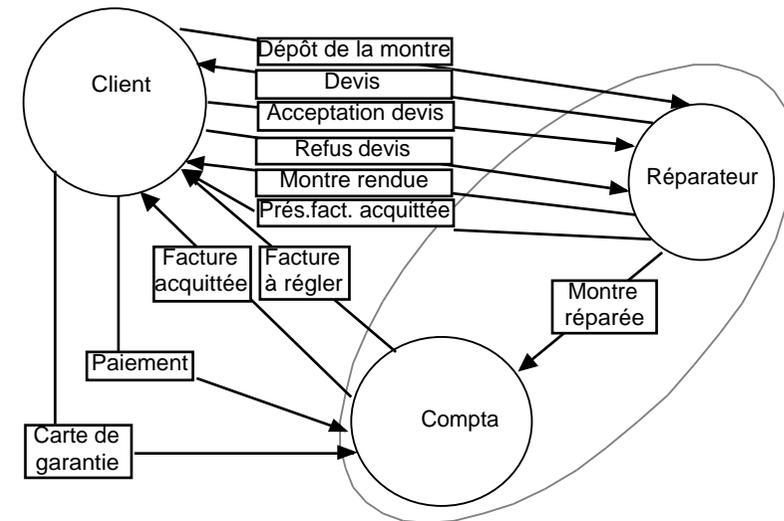
I.2.C - Reconnaissance de domaines

- But :**
- Hiérarchiser le diagramme obtenu, et donc le simplifier au niveau le plus haut de la hiérarchie.
- Moyens :**
- Dans le cas de projets complexes, regrouper les événement-messages selon qu'ils sont internes (source et cible internes à l'organisation) ou externes (source OU cible externe à l'organisation). *NB : en principe, les messages dont les deux acteurs sont externes ne nous concernent pas.* Puis reconnaître des domaines disjoints dans l'organisation, et déterminer, parmi les messages internes à l'organisation, ceux qui sont internes à un domaine et ceux qui en sont externes. Dans l'étude détaillée de ce domaine, on décrira les messages internes à ce domaine, mais dans un diagramme plus général, on négligera tout ce qui est interne à ce domaine. On peut ainsi, à l'aide d'une analyse descendante, ramener l'étude à des niveaux plus élémentaires. Ex : confondre dans un premier temps les acteurs "Réparateur" et service comptable en un domaine "Société", et négliger le message "Montre réparée". Le message "Montre réparée" n'apparaîtra que dans le diagramme de flux de l'organisation
- Remarques :**
- Dans cet exemple, la reconnaissance d'un domaine, quoique juste, est absolument dépourvue d'intérêt.

1° représentation : DFD simplifié + DFD complémentaire

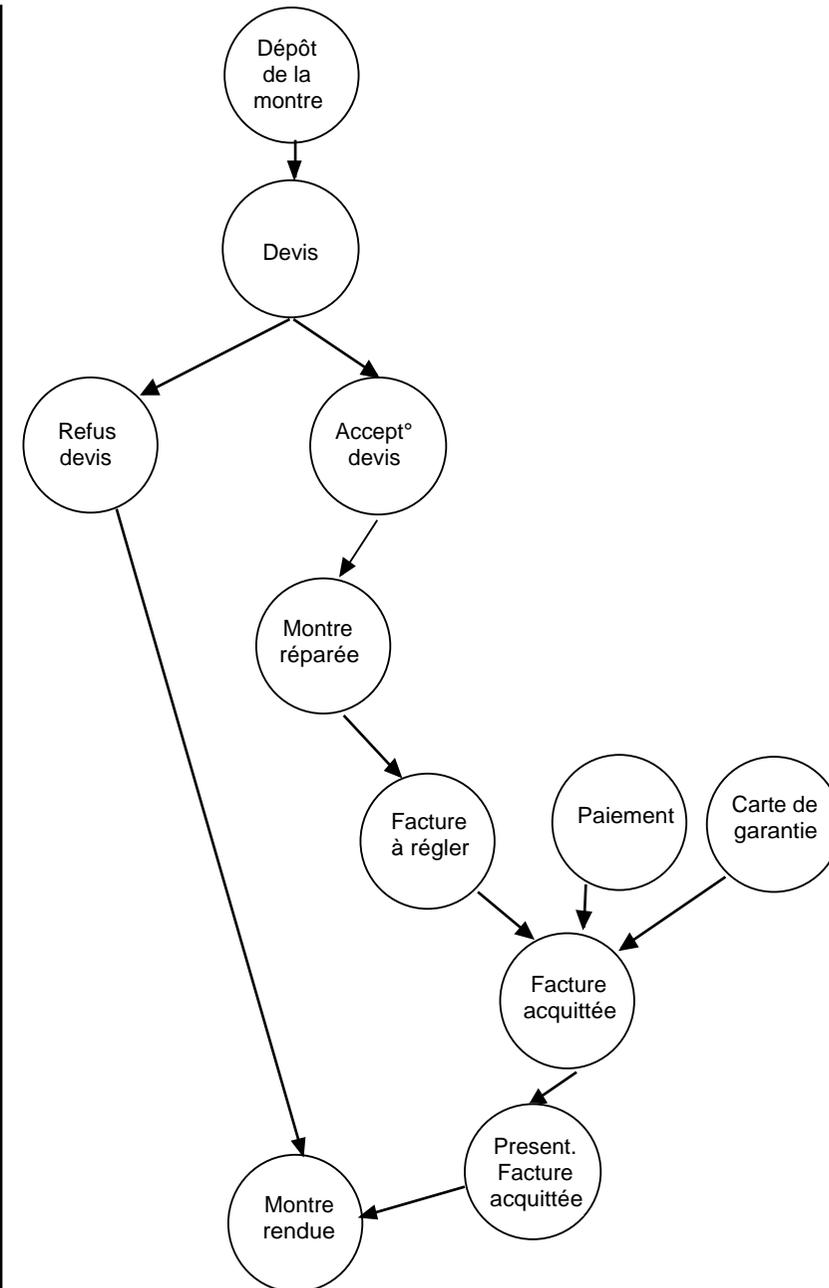


2° représentation : DFD hiérarchisé



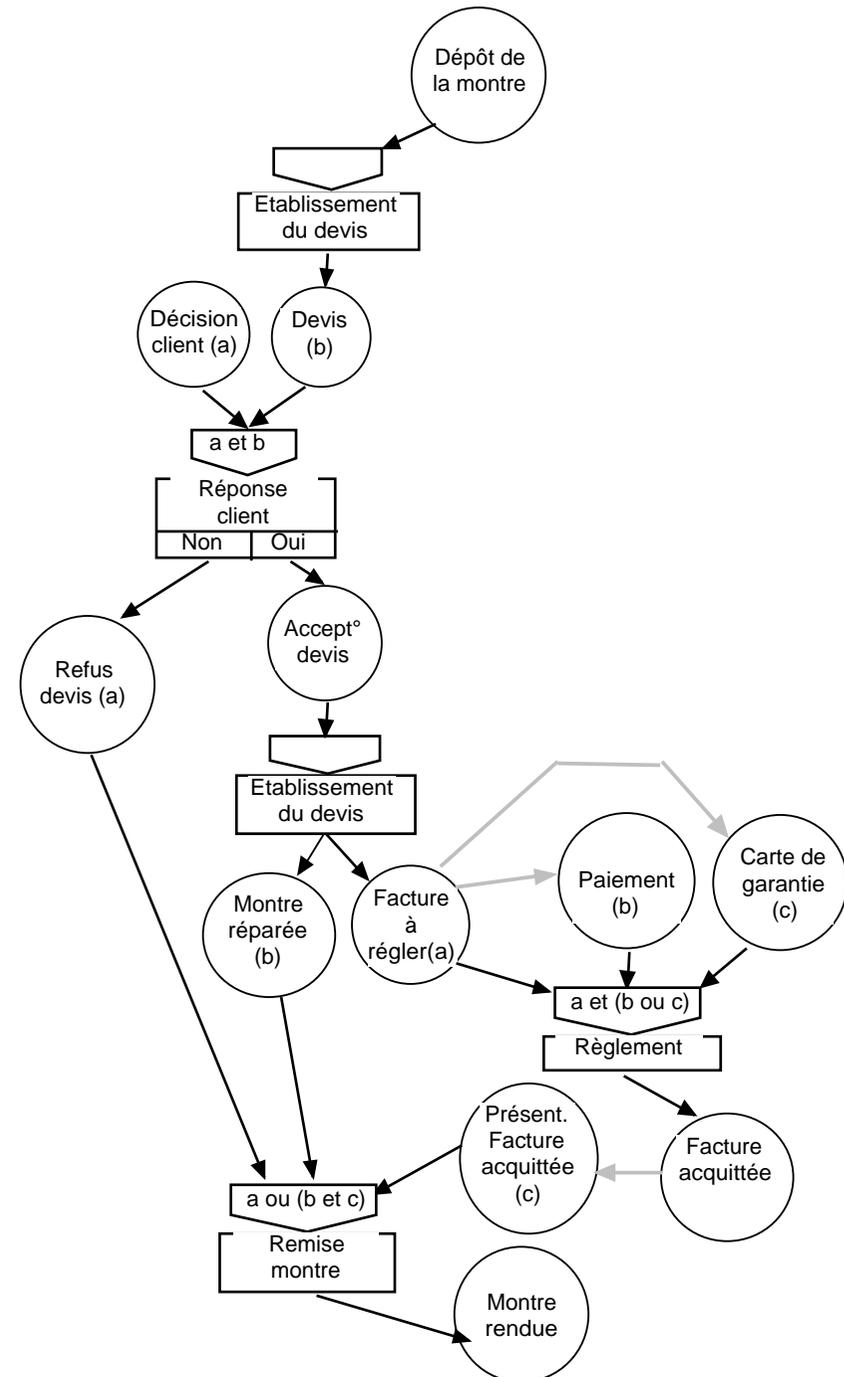
I.2.D - Diagramme ordonné des messages

- But :
- Faire apparaître la chronologie des messages, et par conséquent commencer à faire apparaître leurs interactions et dépendances.
- Moyens :
- Représenter dans un cercle chacun des événement-messages repérés dans l'étape 1.2.B.
 - Représenter par des flèches orientées les précédences chronologiques des événement-messages.
- précautions :
- Ici aussi, SE FIER À LA TECHNIQUE PLUTÔT QU'AU BON SENS : il arrive qu'un choix d'identifiant paraisse évident et soit erroné.
- Remarques :



I.2.E - Ebauche du M.C.T.

- But :**
- Décrire l'ensemble des dépendances entre événement-messages, en précisant, à partir du diagramme orienté, les actions permettant la génération de ces événement-messages, et les conditions de déclenchement de ces actions.
- Moyens :**
- Reprendre le diagramme précédent, et préciser, pour chaque flèche définie dans ce diagramme, un nom d'action (dans un rectangle), précédé des conditions de déclenchement (dans un "pentagone rectangle") et suivi des conditions de sortie. Les conditions de déclenchement sont appelées "synchronisations"
- précautions :**
- Donner des alias aux messages (a, b c...) en fonction de leur participations aux synchronisations suivantes et non pas en fonction de l'action placée en amont.
 - Ajouter si nécessaire les événements dont la source et la cible sont externes, mais qui ont une répercussion sur les synchronisations (ex : décision client).
 - A chaque ajout d'identifiant, il est indispensable de refaire l'épuration du dictionnaire des données, car l'identifiant créé peut rendre caducs certains attributs initialement retenus.
- Remarques :**

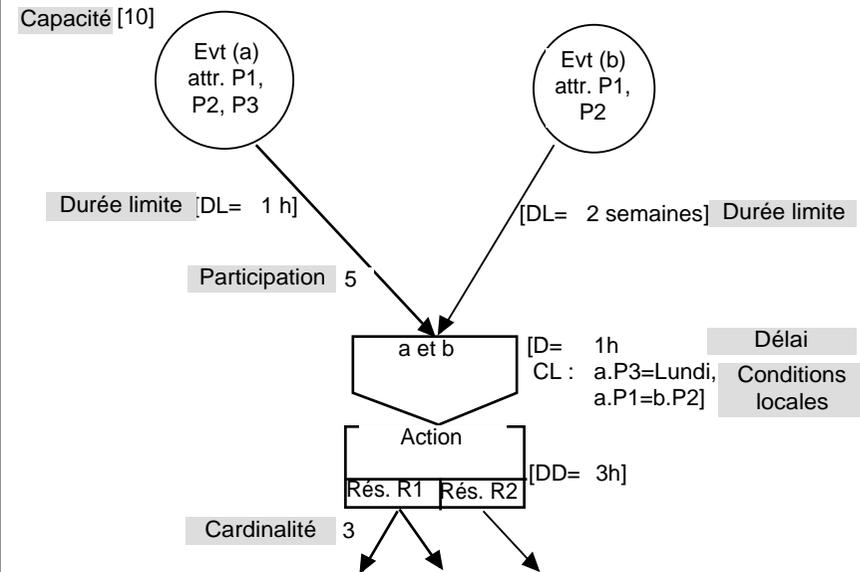


I.2.F - Enrichissement du M.C.T.

- But :**
- Préciser les conditions de déclenchement et les charges des différents événements, pour pouvoir plus tard vérifier la vie du système.
- Moyens :**
- Ajouter en amont de chaque événement la capacité du système (çàd le nombre maximum, s'il existe, d'occurrences de l'événement pouvant être en attente dans le système. Ce nombre sera supposé indéterminé s'il n'est pas précisé). Ce nombre est représenté entre crochets.
 - Ajouter, sur chaque flèche partant de de chaque événement, sa participation à l'action suivante (çàd le nombre d'occurrences de l'événement qui doit être fourni à l'action suivante. Ce nombre sera supposé de 1 s'il n'est pas précisé)
 - Ajouter si besoin, sur les synchronisations, les conditions locales et délais. Ces informations sont placées entre crochets sous la forme :
 [D=..... (nul si non renseigné)
 DL=..... (infinie si non renseignée)
 CL :
 CL :]
 On entend par condition locale toute règle de traitement ne faisant pas appel à des données autres que les attributs des événements entrants.
 - Ajouter si besoin, sur les pattes entrantes des synchronisations, les durées limites. Ces informations sont placées entre crochets sous la forme :
 [DL=.....](nul si non renseigné)
 - Ajouter si besoin, sur les actions, la durée de l'action. Cette information est placée entre crochets sous la forme :
 [DD=.....]
 - Ajouter si besoin, sur les pattes sortantes des actions, la cardinalité de l'événement-résultat, c'est à dire le nombre d'occurrences de l'événement-sortant produites par l'action. Si la cardinalité n'est pas renseignée, elle est supposée égale à 1
- précautions :**
- Lorsqu'on a mis en lumière l'existence d'une relation entre deux objets, vérifier s'il en existe une autre entre ces deux mêmes objets.
 - S'il existe à la fois une relation entre un objet A et un objet B, entre B et C et entre A et C, vérifier :
 - si l'une des relations peut être une conséquence immédiate des deux autres. Dans ce cas, la supprimer.
 - si l'on est en présence d'une seule relation entre trois entités.

Remarques :

EXEMPLE :



I.2.G - Vérification du M.C.T.

But : • S'assurer de la cohérence de chacune des actions décrites, en vérifiant, pour chacune d'entre elles, les 11 règles suivantes.

- Règles :
- 1 Si une synchronisation est associée à plus d'un événement contributif (e.c), elle ne doit pas être déclenchable par un seul e.c
 - 2 Si une action est précédée de plus d'un e.c, le prédicat de synchronisation ne doit pas être toujours fausse
 - 3 La participation d'un e.c doit être au plus égal à sa capacité.
 - 4 Chaque e.c doit contribuer à au moins une synchronisation sans durée limite
 - 5 Une synchronisation doit avoir au plus un e.c de durée limite égale à 0
 - 6 Les conditions locales portent uniquement sur les attributs des messages associés aux e.c
 - 7 La cardinalité d'un événement-résultat doit être au plus égale à sa capacité.
 - 8 La disjonction des règles de sortie doit être systématiquement vraie
 - 9 Toute propriété d'un événement-message doit figurer dans le M.C.D.
 - 10 Tout événement en entrée d'une action doit constituer un modèle externe valide (doit pouvoir être représenté dans le M.L.D. sous forme d'une vue ou d'un select)
 - 11 Tout événement en sortie d'une action rendant activable cette action doit constituer un modèle externe valide en mise à jour (doit pouvoir être représenté dans le M.L.D. sous forme d'une requête de mise à jour)

Remarques :

I.2.H - Cohérence globale du M.C.T.

But : • Vérifier que le M.C.T. ne nous amène pas à des dysfonctionnements répertoriés.

Moyens : • Vérifier que le modèle est “atteignable”, “borné”, “vivant”, “propre”, “bien formé”, et éventuellement “Déterministe”. Ces notions ne seront pas abordées ici

Remarques :