

# LA METHODE MERISE :: INTRODUCTION

La conception d'un système d'information n'est pas évidente car il faut réfléchir à l'ensemble de l'organisation que l'on doit mettre en place. La phase de conception nécessite des méthodes permettant de mettre en place un modèle sur lequel on va s'appuyer. La modélisation consiste à créer une représentation virtuelle d'une réalité de telle façon à faire ressortir les points auxquels on s'intéresse. Ce type de méthode est appelé *analyse*. Il existe plusieurs méthodes d'analyse, la méthode la plus utilisée en France étant la méthode MERISE (Méthode d'étude et de Réalisation Informatique de systèmes d'entreprises).

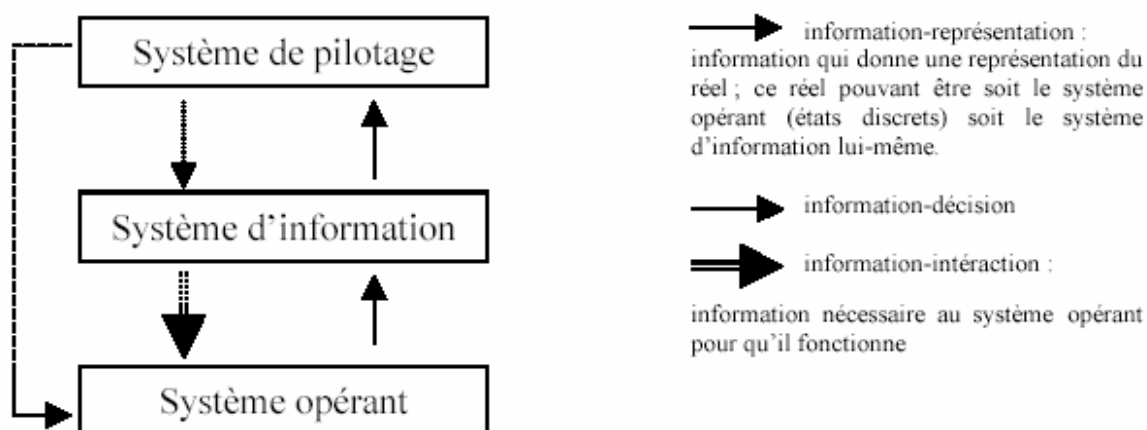
Le but de cette méthode est d'arriver à concevoir un système d'information. La méthode MERISE est basée sur la séparation des données et des traitements à effectuer en plusieurs modèles conceptuels et physiques. La séparation des données et des traitements assure une longévité au modèle. En effet, l'agencement des données n'a pas à être souvent remanié, tandis que les traitements le sont plus fréquemment.

La méthode MERISE date de 1978-1979, et fait suite à une consultation nationale lancée en 1977 par le ministère de l'Industrie dans le but de choisir des sociétés de conseil en informatique afin de définir une méthode de conception de systèmes d'information. Les deux principales sociétés ayant mis au point cette méthode sont le CTI (Centre Technique d'Informatique) chargé de gérer le projet, et le CETE (Centre d'Etudes Techniques de l'Équipement) implanté à Aix-en-provence.

Merise étant une méthode de conception et de développement de système d'information, l'objectif de ce chapitre est d'introduire la notion de système d'information et d'en proposer une description formelle.

## I. Le système d'information dans l'entreprise

L'entreprise est un système complexe dans lequel transitent de très nombreux flux d'informations. Sans un dispositif de maîtrise de ces flux, l'entreprise peut très vite être dépassée et ne plus fonctionner avec une qualité de service satisfaisante. L'enjeu de toute entreprise qu'elle soit de négoce, industrielle ou de services consiste donc à mettre en place un système destiné à collecter, mémoriser, traiter et distribuer l'information (avec un temps de réponse suffisamment bref). Ce système d'information assurera le lien entre deux autres systèmes de l'entreprise : le système opérant et le système de pilotage.



- Le système de pilotage décide des actions à conduire sur le système opérant en fonction des objectifs et des politiques de l'entreprise,
- Le système opérant englobe toutes les fonctions liées à l'activité propre de l'entreprise : facturer les clients, régler les salariés, gérer les stocks, ...

Une telle décomposition prend bien en compte :

- la différence de besoin en matière d'information des modules opérants et pilotes,

- la nécessité pour le système d'information de ne pas se contenter de transmettre les informations mais d'en changer le niveau de synthèse.

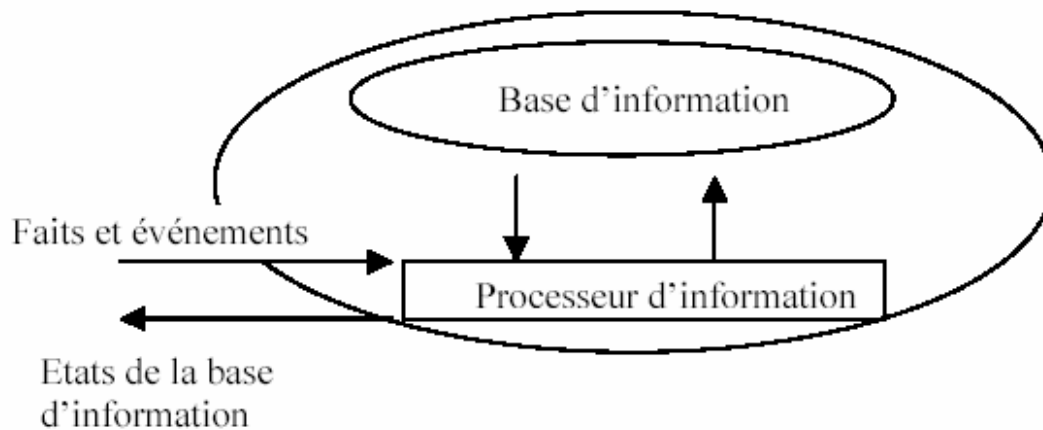
Dans certaines organisations, on peut trouver des formes plus intégrées du système d'information. Cette intégration peut se faire soit au niveau du système opérant, soit au niveau du système de pilotage.

- Un système d'information intégré au système opérant ne décrit plus le fonctionnement du système opérant mais il est intégré à ce fonctionnement. Par exemple dans un système de GPAO (Gestion de Production assistée par Ordinateur), les décisions de pilotage sont directement traduites en des décisions d'exécution de règles incluses dans une gamme opératoire.

- Un système d'information intégré au système de pilotage doit permettre d'engranger les décisions prises lors de diverses situations afin de rendre le pilotage plus intelligent. Ces Systèmes Interactifs d'Aide à la Décision (S.I.A.D) ont une architecture proche de celle des systèmes experts et font donc largement appel pour leur conception aux techniques de l'intelligence artificielle.

## **II. Architecture et conception d'un système d'information**

Le système d'information doit décrire (on dit encore représenter) le plus fidèlement possible le fonctionnement du système opérant. Pour ce faire, il doit intégrer une base d'information dans laquelle seront mémorisés la description des objets, des règles et des contraintes du système opérant. Cette base étant sujette à des évolutions, le système d'information doit être doté d'un mécanisme (appelé processeur d'information) destiné à piloter et à contrôler ces changements. Le schéma suivant synthétise l'architecture d'un système d'information.



Le processeur d'information produit des changements dans la base d'information à la réception d'un message. Un message contient des informations et exprime une commande décrivant l'action à entreprendre dans la base d'information. Le processeur d'information interprète la commande et effectue le changement en respectant les contraintes et les règles. Si le message exprime une recherche sur le contenu de la base d'information, le processeur interprète la commande et émet un message rendant compte du contenu actuel de la base d'information. Dans tous les cas, l'environnement a besoin de connaître si la commande a été acceptée ou refusée. Le processeur émet, à cet effet, un message vers l'environnement.

Relativement à la conception d'un système d'information, l'architecture présentée ci-dessus induit une double conception :

- celle de la base d'information (aspect statique)
- celle du processeur de traitement (aspect dynamique)

Pour aider le concepteur dans ces deux tâches, la méthode Merise propose un ensemble de formalismes et de règles destinées à modéliser de manière indépendante les données et les traitements du système d'information. Ces modèles ne sont qu'une base de réflexion pour le concepteur et un moyen de communication

entre les divers acteurs du système d'information dans l'entreprise. Seul la validation de l'ensemble se fera en commun.

### **III. Système d'information et système informatique**

Parmi les informations qui appartiennent au système d'information, certaines doivent ou peuvent faire l'objet d'un traitement automatisé grâce aux outils informatiques. Pour assurer la cohérence du système d'information, la méthode Merise propose une démarche d'informatisation comportant les étapes suivantes :

- le schéma directeur : dont le rôle est de définir, de manière globale, la politique d'organisation et d'automatisation du système d'information. Pour ce faire, il est nécessaire de répertorier l'ensemble des applications informatiques existantes à modifier et à développer.

Pour rendre contrôlable et modulable ce développement, il est nécessaire de découper le système d'information en sous-ensembles homogènes et relativement indépendant. Ces sous ensembles sont appelés domaines. Par exemple, on peut trouver le domaine « Approvisionnement », le domaine « Personnel ». Les résultats attendus à la fin de cette étape sont une définition précise des domaines, une planification du développement de chaque domaine et un plan détaillé, année par année, des applications qui doivent être réalisées.

- l'étude préalable par domaine: qui doit aboutir à une présentation générale du futur système de gestion (modèles des données et des traitements) en indiquant les principales novations par rapport au système actuel, les moyens matériels à mettre en œuvre, les bilans coût – avantage. Cette étude est réalisée en 4 phases :

- une phase de recueil qui a pour objectif d'analyser l'existant afin de cerner les dysfonctionnements et les obsolescences les plus frappantes du système actuel.
- une phase de conception qui a pour objectif de formaliser et hiérarchiser les orientations nouvelles en fonction des critiques formulées sur le système actuel

et d'autre part des politiques et des objectifs de la direction générale. Cela revient à modéliser le futur système avec une vue pertinente de l'ensemble.

- une phase d'organisation dont l'objectif est de définir le système futur au niveau organisationnel: qui fait quoi ?
  - une phase d'appréciation dont le rôle est d'établir les coûts et les délais des solutions définies ainsi que d'organiser la mise en œuvre de la réalisation. A cet effet un découpage en projets est effectué.
- l'étude détaillée par projet qui consiste d'une part à affiner les solutions conçues lors de l'étude préalable et d'autre part à rédiger, pour chaque procédure à mettre en œuvre, un dossier de spécifications détaillé décrivant les supports (maquettes d'états ou d'écran) ainsi que les algorithmes associés aux règles de gestion... A l'issue de cette étude, il est possible de définir le cahier des charges utilisateurs qui constitue la base de l'engagement que prend le concepteur vis à vis des utilisateurs. Le fonctionnement détaillé du futur système, du point de vue de l'utilisateur, y est entièrement spécifié.
- la réalisation dont l'objectif est l'obtention des programmes fonctionnant sur un jeu d'essais approuvés par les utilisateurs.
- la mise en œuvre qui se traduit par un changement de responsabilité : l'équipe de réalisation va en effet transférer la responsabilité du produit à l'utilisateur. Cette étape intègre en particulier la formation des utilisateurs. Après une période d'exploitation de quelques mois, la recette définitive de l'application est prononcée.
- la maintenance qui consiste à faire évoluer les applications en fonction des besoins des utilisateurs, de l'environnement et des progrès technologiques.

## IV. Conclusion

Même si la méthode MERISE étant, avant tout, une méthode de conception de systèmes d'information, et non de systèmes informatiques, il apparaît aujourd'hui que les systèmes d'information sont largement gérés par des applications informatiques. Les modèles MERISE doivent donc être utilisés pour faciliter le développement de ces applications en s'appuyant sur les technologies logicielles actuelles telles que les bases de données relationnelles et/ou l'architecture client-serveur.

De plus, il apparaît que les méthodes traditionnelles, composées d'étapes menées séquentiellement depuis l'analyse du besoin jusqu'à la recette, présentent l'inconvénient d'être rigides et peu réactives. Ainsi, le temps écoulé entre les spécifications et la phase de livraison est parfois tellement important que les besoins ont changé de nature. Pour pallier ces défauts, il faut envisager des démarches qui impliquent beaucoup plus l'utilisateur dans le processus global d'informatisation et qui procèdent par affinements successifs. Ainsi, une démarche basée sur des méthodes traditionnelles, comme MERISE pour l'aspect conceptuel, et plus modernes, comme le RAD pour produire des prototypes, pourrait s'avérer être un compromis avantageux pour la conception d'applications informatiques.

Ce cours s'inscrit dans cette logique : il ne détaillera donc pas les étapes de la méthode Merise dans le processus d'informatisation, mais sera axé sur les formalismes et concepts de Merise utiles aux descriptions statique et dynamique du système d'information à automatiser.

Relativement à ces descriptions (encore appelées modèles) la méthode Merise préconise 3 niveaux d'abstraction :

- le niveau conceptuel qui décrit la statique et la dynamique du système d'information en se préoccupant uniquement du point de vue du gestionnaire.

- le niveau organisationnel décrit la nature des ressources qui sont utilisées pour supporter la description statique et dynamique du système d'information. Ces ressources peuvent être humaines et/ou matérielles et logicielles.
- le niveau opérationnel dans lequel on choisit les techniques d'implantation du système d'information ( données et traitements)

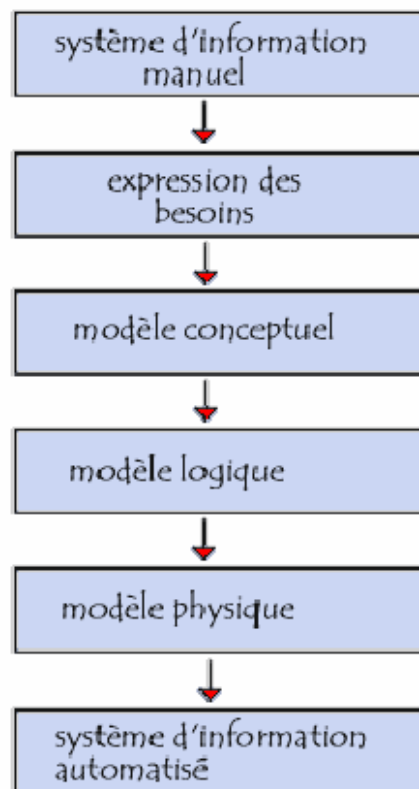
Du fait de ce découpage (qui a été introduit pour faciliter l'analyse d'un problème) seul le premier niveau est réellement indépendant de toute considération technologique : logicielle ou matérielle. Par exemple, si les données du futur système d'information doivent être gérées par un SGBD, c'est au niveau organisationnel que le choix du type du SGBD (relationnel, réseau ou objets) devra être effectué. La description statique du système d'information à ce niveau sera donc basée sur l'organisation des bases relationnelles, ou réseau, ou objets. Le troisième niveau est encore plus dépendant de l'aspect technologique puisqu'il cherchera à optimiser l'implantation. Il suppose donc une connaissance très pointue de l'architecture et des fonctions du SGBD qui gèrera le système d'information.

L'étude des technologies logicielles, telles que les types de SGBD ou encore l'architecture client-serveur, sortant du cadre de ce cours, celui-ci se focalisera sur le niveau conceptuel tant au niveau des données que des traitements. L'apprentissage des formalismes associés à ce niveau suffit à illustrer la richesse, la puissance et parfois même les faiblesses des formalismes Merise en général et, donne ainsi une bonne idée des principaux aspects de la méthode.

Enfin, l'utilisation de l'Atelier de Génie Logiciel AMC\*Designor permettra de découvrir comment Merise a été intégré à un outil de conception ainsi que son apport dans le développement d'une application client-serveur.



## Cycle d'abstraction de conception des S.I.



La conception du système d'information se fait par étapes, afin d'aboutir à un système d'information fonctionnel reflétant une réalité physique. Il s'agit donc de valider une à une chacune des étapes en prenant en compte les résultats de la phase précédente. D'autre part, les données étant séparées des traitements, il faut vérifier la concordance entre données et traitement afin de vérifier que toutes les données nécessaires aux traitements sont présentes et qu'il n'y a pas de données superflues.

Cette succession d'étapes est appelée cycle d'abstraction pour la conception des systèmes d'information:

L'expression des besoins aboutit au MCC (Modèle conceptuel de la communication) qui définit les flux d'informations à prendre compte.

L'étape suivante consiste à mettre au point le MCD (Modèle conceptuel des données) et le MCT (Modèle conceptuel des traitements) décrivant les règles et les contraintes à prendre en compte.

Le modèle organisationnel consiste à définir le MLD (Modèle logique des données) qui représente un choix logiciel pour le système d'information et le MOT (Modèle organisationnel des traitements) décrivant les contraintes dues à l'environnement (organisationnel, spatial et temporel).

Enfin, le modèle physique reflète un choix matériel pour le système d'information.

L'étape suivante consiste à mettre au point le MCD (Modèle conceptuel des données) et le MCT (Modèle conceptuel des traitements) décrivant les règles et les contraintes à prendre en compte.

Le modèle organisationnel consiste à définir le MLD (Modèle logique des données) qui représente un choix logiciel pour le système d'information et le MOT (Modèle organisationnel des traitements) décrivant les contraintes dues à l'environnement (organisationnel, spatial et temporel).

Enfin, le modèle physique reflète un choix matériel pour le système d'information.

<i>Niveau</i>	<i>Statique (données)</i>	<i>Dynamique (traitements)</i>	
Conceptuel	<b>MCD</b>	<b>MCT</b>	Indépendant du système: <i>QUOI ?</i>
Organisationnel ou logique	<b>MLD</b> <i>(OU ?)</i>	<b>MOT</b> <i>(QUI ? QUAND ?)</i>	Choix du SGBD: <i>QUI ? QUAND ? OU ?</i>
Opérationnel ou physique	<b>MPD</b>	<b>MOPT</b>	Haute connaissance du SGBD: <i>COMMENT ?</i>

# DESCRIPTION DYNAMIQUE DU S.I.

Comme il a été dit dans le chapitre d'introduction, tout système d'information est composé d'une base d'information et d'un processeur d'information qui représentent respectivement sa statique et sa dynamique. A l'instar du Modèle Conceptuel des Données (MCD) qui schématise les données du système d'information, le Modèle Conceptuel des Traitements (MCT) décrit les traitements et plus précisément toutes les activités découlant des échanges entre le domaine étudié et le monde extérieur. Il exprime donc ce que fait le domaine sans se poser le problème de savoir qui le fait, quand et comment.

## I. Les concepts de base

### 1.1) L'acteur

Un acteur est une personne morale ou physique capable d'émettre ou de recevoir des informations. *Par exemple, l'élève de terminale qui souhaite s'inscrire à un DEUG préparé par la faculté des sciences est un acteur du domaine « Gestion des inscriptions » de cette faculté.* On distingue deux types d'acteurs :

- les acteurs internes qui appartiennent au système d'information étudié. *Pour le domaine cité ci-dessus, le service des inscriptions*

*ou le service comptabilité de la faculté des sciences sont des acteurs internes.*

- les acteurs externes qui n'appartiennent pas au système d'information mais qui sont l'origine ou la destination de flux d'informations reçus ou émanant du système d'information.

L'élève de terminale qui effectue une demande de préinscription à la faculté des sciences est un exemple d'acteur externe.

Dans le Modèle Conceptuel de Traitements, seuls les acteurs externes sont modélisés; d'une part parce qu'on ne cherche qu'à formaliser les traitements découlant d'interactions avec l'environnement et que d'autre part, on ne cherche pas à connaître les acteurs internes qui réalisent les traitements en question.

**Remarque** : pour certains Ateliers de Génie Logiciel (AGL), la notion d'acteur est implicite : ils n'apparaissent donc pas graphiquement dans les différents modèles de traitements produits.

## **1.2) L'événement**

L'événement matérialise un fait, qui en se produisant, doit déclencher une réaction du système. Plus précisément cette notion recouvre deux aspects :

- le fait qui survient et sa perception. *La décision d'un élève de terminale de s'inscrire à la faculté des sciences illustre cet aspect.*

- le compte rendu de cette perception faite auprès du système d'information. *Ainsi, dans le cas de la pré-inscription, c'est le remplissage du dossier qui constitue le compte rendu du souhait de l'élève.*

Seul le second aspect est pris en compte dans la dynamique du système d'information et correspond à la définition d'événement. Du fait de cette restriction l'événement (au sens du modèle conceptuel des traitements) est porteur d'informations qui doivent être obligatoirement digérées par le système d'information sans quoi il ne répondrait pas à ses objectifs. Parmi les événements, on distingue les événements externes et les événements internes :

- les événements déclencheurs externes sont des événements émis par un acteur externe. *Par exemple le dépôt d'un dossier de pré-inscription est un événement externe déclenché par un futur bachelier souhaitant intégrer un DEUG à la faculté des sciences.*

- les événements internes sont des événements qui surviennent lorsqu'une opération se termine. *Ce peut être par exemple l'acceptation de la pré-inscription après vérification du contenu du dossier.* Un événement interne n'a lieu d'être que si le compte rendu de la fin d'une opération doit être soit suivi d'une nouvelle réaction du système d'information, soit de l'émission d'un message vers l'environnement.

Certains événements externes sont liés au temps. Par exemple, pour déclencher un traitement en début d'année civile, on introduira l'événement « Début d'année ». L'événement « Date actuelle est JJ/MM/AAAA » permettra d'exécuter un traitement à une date donnée. Dans le MCT, chaque événement est identifié au moyen d'un libellé générique tel que « Dépôt d'un dossier de pré-inscription ». Compte tenu de ce qui vient d'être dit, cet intitulé est très insuffisant pour décrire l'événement car il ne fait pas apparaître les données du compte rendu associé à l'événement. *Par exemple, le dépôt d'un dossier de pré-inscription apporte de nombreuses informations telles que l'état civil de l'élève qui effectue le dépôt, des données sur sa scolarité actuelle, le DEUG qu'il souhaite intégrer, etc.* Dans le cas où le nombre d'informations contenues dans le message associé à l'événement est peu important il est recommandé de les citer en annexe du MCT. Dans le cas contraire, il sera utile de dégager les principales entités figurant dans le compte rendu. Le terme « entité » ne fait pas référence au

modèle conceptuel des données ; il est pris dans son sens très général pour désigner tout objet

abstrait ou concret caractérisé par un ensemble de propriétés. *Par exemple, l'événement*

*« Dépôt de dossier de pré-inscription » sera détaillé ainsi : « Dépôt du dossier de préinscription de l'élève E de la terminale T à la formation F ». Ce texte, qui constitue le message associé à l'événement, fait intervenir trois « entités » : Elève, Terminale et FormationSup.*

Notons enfin que l'intitulé générique étant parfois long, on lui associe, sur le MCT un alias afin de le référencer plus facilement. Les alias seront codés ainsi : « ext » ou « int » pour indiquer le type de l'événement suivi d'un numéro séquentiel. *Par exemple « ext1 » désignera l'événement « Dépôt d'un dossier de pré-inscription ».*

#### a) Occurrences d'événements

L'occurrence d'un événement correspond à la réalisation effective d'un événement. *Par exemple, le dépôt du dossier de pré-inscription de M.*

*Girard de terminale ES pour le DEUG*

*MASS, constitue une occurrence de l'événement déclencheur « Dépôt du dossier de réinscription*

». Deux occurrences d'un même événement peuvent être distinguées soit par des valeurs de propriétés (ou d'entités) différentes, soit si les valeurs sont identiques, par le moment précis où l'événement s'est produit.

La notion d'occurrence d'événements n'est, en général, pas modélisée, par contre la capacité d'un événement, qui est le nombre maximum d'occurrences acceptées par le processeur d'information, et la fréquence d'apparition des occurrences le sont. *Ainsi, on peut fixer à 3000 le nombre maximum d'occurrences acceptées par le processeur pour l'événement « Dépôt du dossier de pré-inscription ».*

#### b) Participation et cardinalité d'un événement

La participation d'un événement définit le nombre d'occurrences différentes nécessaires au lancement de l'opération. *Dans le processus de gestion des inscriptions, le traitement du dossier déposé doit être déclenché à chaque apparition d'une occurrence de l'événement « Dépôt d'un dossier de pré-inscription ». La participation de l'événement au traitement est donc égal à 1.*

La cardinalité d'un événement est le nombre d'occurrences identiques d'un événement résultat. Si l'on associe à l'événement interne « Carte étudiant éditée » la cardinalité 2, cela impliquera une émission en deux exemplaires de la carte d'étudiant. Si ces deux caractéristiques ne sont pas précisées sur le MCT, elles prennent la valeur 1 par défaut.



### 1.3) L'opération

La réponse à l'arrivée d'un événement est le déclenchement d'un ensemble de traitements appelé opération. *Le traitement d'enregistrement d'une préinscription est une opération déclenchée lors du dépôt de dossier de préinscription*

Lors de son exécution une opération ne peut pas être interrompue par l'attente d'un événement externe. L'exécution d'une opération se ramène à l'exécution d'actions élémentaires effectuées sur la base d'informations à partir des données portées par le ou les événement(s) déclencheur(s). Ces actions élémentaires portent sur des occurrences d'entités ou d'associations du modèle conceptuel des données et peuvent appartenir à l'un des quatre types suivants :

- insertion
- la modification
- l'effacement
- recherche

La logique d'enchaînement des actions élémentaires n'est pas toujours séquentielle et peut faire intervenir des structures alternatives (Si .. Alors ... Sinon) ou itératives (Tant que ..., Répéter ..., Pour ...).

Sur le MCT, une opération est identifiée par un libellé et peut être décrite, de manière détaillée, en annexe du MCT, en présentant la logique algorithmique du déclenchement des actions élémentaires. *Par exemple l'opération d'enregistrement du dossier de préinscription pourrait être détaillée ainsi :*

- *création d'une occurrence de l'entité « Préinscrit »*
- *création d'une occurrence de l'association « Demande » (reliant l'entité « Préinscrit » à l'entité « Formation »)*
- ...

Il est possible d'associer à une opération une durée qui représente le temps maximal qui lui est alloué pour qu'elle s'exécute.

#### **1.4) La règle d'émission**

La production effective d'une ou de plusieurs occurrences d'un événement interne est soumise à une règle d'émission, c'est-à-dire à une proposition logique qui s'applique au contenu de la base d'information après exécution de l'opération. L'événement est produit si la proposition logique est vraie. A l'issue de l'enregistrement d'un dossier de pré-inscription deux cas peuvent se présenter :

- soit le dossier est complet et une occurrence de l'événement « Pré-inscription de l'élève E à la formation F réalisée le JJ/MM/AAAA » est émise

- soit le dossier est incomplet (certaines propriétés du MCD n'ont pas été renseignées) et une occurrence de l'événement « Dossier D Mis en attente le JJ/MM/AAAA » est produite.

Si la plupart des règles d'émission sont basées sur une structure alternative et donne donc

lieu à une seule occurrence d'événement interne, certaines peuvent intégrer une structure

itérative de type « Pour – Tout » afin de produire n occurrences d'un événement interne. *Par*

*exemple pour envoyer en début d'année des lettres de renouvellement d'adhésion, on introduira la règle d'émission suivante :*

*Pour tout adhérent A enregistré dans la base d'informations, créer une occurrence de int1 (Renouvellement d'adhésion envoyé à A le JJ/MM/AAAA)*

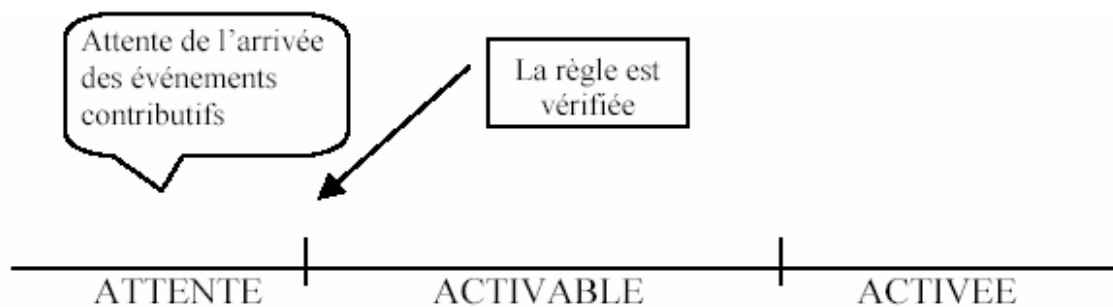
*Fin pour*

## **1.5) La synchronisation**

La synchronisation d'une opération est composée de deux éléments :

- d'une part la liste des événements (internes ou externes) qui doivent être arrivés avant de déclencher l'opération.
- et d'autre part la règle sous forme d'une proposition logique qui précise de quelle manière les événements participent au déclenchement de l'opération.

Le cycle de vie d'une synchronisation peut être représenté ainsi :



Pour des raisons de lisibilité ce sont les alias des événements participant à la synchronisation qui sont mentionnés, sur le MCT, dans l'expression logique de la synchronisation.

*Par exemple la condition ext1 et ext2 signifie que la synchronisation sera activable lorsque :*

- le nombre d'occurrences de l'événement ext1 sera égal à la participation de ext1
- et le nombre d'occurrences de l'événement ext2 sera égal à la participation de ext2

A cette proposition logique sont associées des conditions locales qui permettent de préciser, lorsque plusieurs occurrences d'un événement sont présentes comment choisir celles qui participera effectivement à la synchronisation. Les conditions locales portent obligatoirement sur les valeurs des propriétés ou des entités associées aux messages des événements à synchroniser. Une synchronisation ne peut pas consulter la base d'informations.

*Par exemple, pour modéliser le déclenchement de la mise à jour d'un dossier incomplet suite à la réception des pièces manquantes, on introduira une synchronisation admettant en entrée les deux événements suivants :*

*Événement int2 : « Dossier D Mis en attente le JJ/MM/AAAA »*

*Événement ext2: « Réception des pièces manquantes du dossier D »*

*Proposition logique : int2 et ext2*

*Conditions locales : int2.D = ext2.D*

Lorsque l'expression logique de la synchronisation est vérifiée, l'opération est déclenchée

et toutes les occurrences d'événements qui ont permis ce déclenchement sont alors consommées par l'opération. Par contre si elle n'est pas vérifiée, les occurrences d'événement restent en attente. Quand une opération est déclenchée par un seul événement, la synchronisation est facultative.

Comme pour les règles d'émission, les conditions locales d'une synchronisation peuvent comporter une structure itérative de type « Pour tout ». Cette possibilité permet de traiter (on dit encore consommer) n occurrences d'un événement contributif à la synchronisation. *La synchronisation définie ci-dessous permet de lancer la suppression de tous les dossiers mis en attente et pour lesquels les pièces manquantes n'ont pas été fournies dans un délai de 8 jours.*

*Événement int2 : « Dossier D mis en attente le JJ/MM/AAAA »*

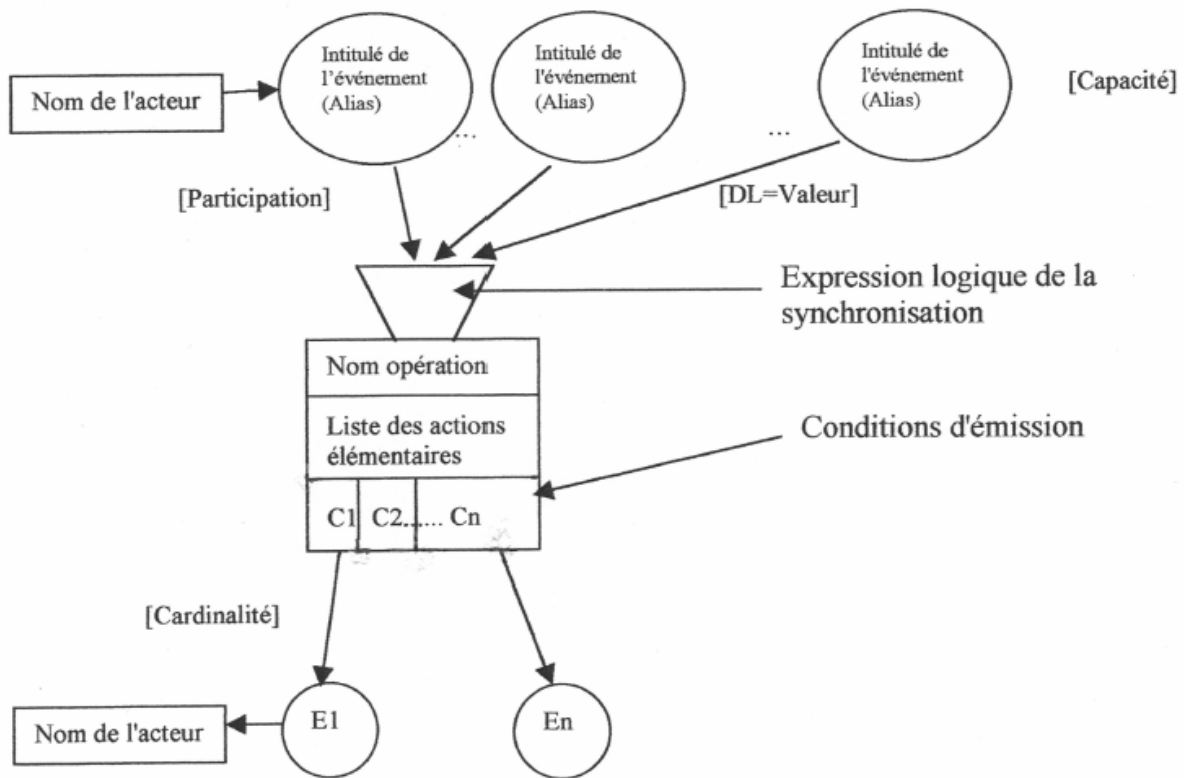
*Événement ext3 : « la date actuelle est JJ/MM/AAAA »*

*Proposition logique : int2 et ext3*

*Conditions locales : Pour tout int2 ayant*

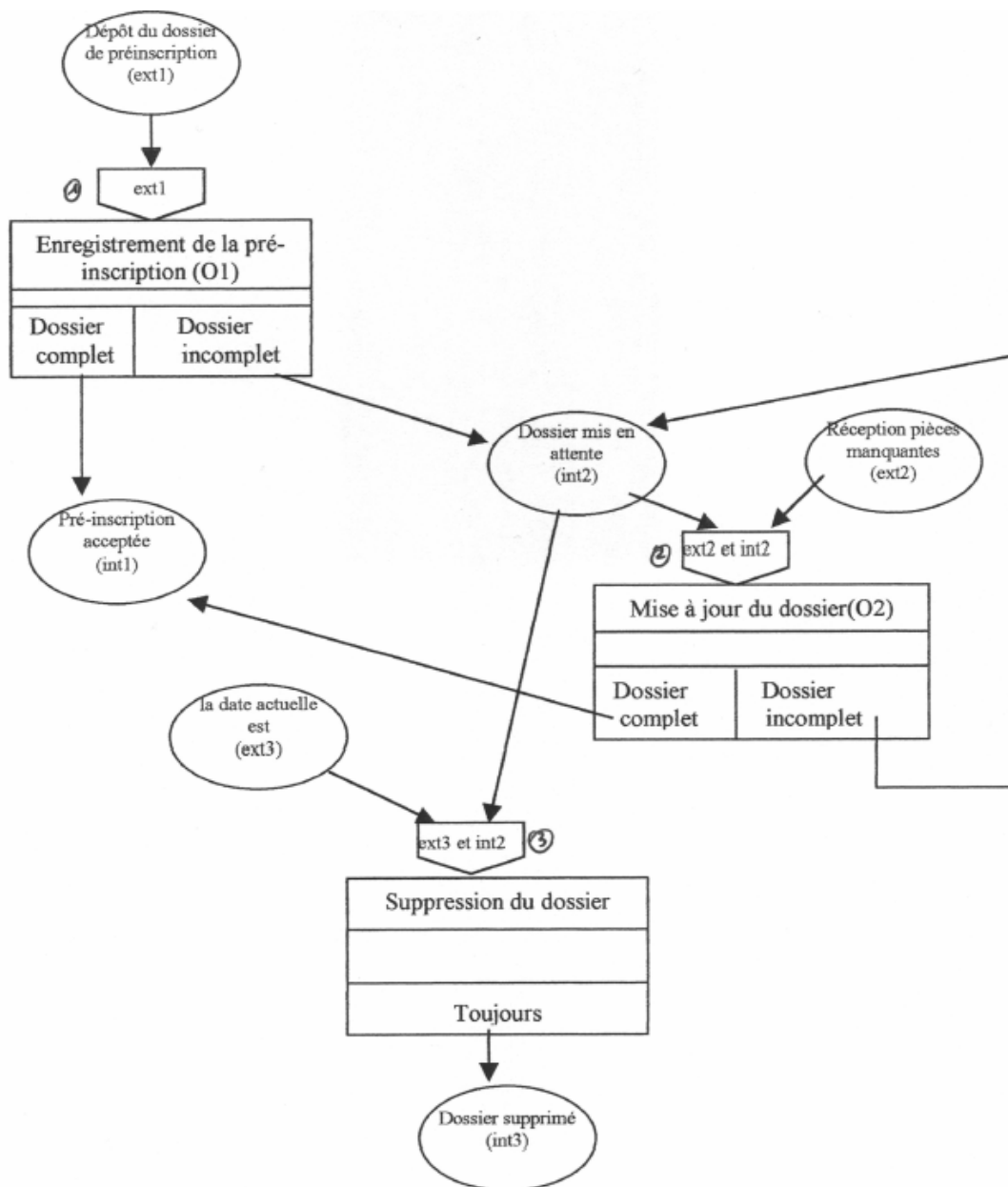
*int2.JJ/MM/AAAA = ext3.JJ/MM/AAAA*

## **1.6) Représentation graphique**



**Remarque :** Le MCT conditionne complètement l'interface graphique du S.I.

### 1.7) Extrait du MCT de la gestion des inscriptions



Evénement ext1 : Dépôt du dossier de pré-inscription de l'élève E pour la formation F

Evénement ext2 : Réception des pièces manquantes du dossier n° D

Evénement ext3 : La date actuelle est JJ/MM/AAAA



Evènement int1 : Dossier de pré inscription n° D accepté le  
JJ/MM/AAAA

Evènement int2 : Dossier de pré-inscription n° D mis en attente le  
JJ/MM/AAAA

Evènement int3 : Avis de suppression du dossier n° D émis le  
JJ/MM/AAAA

Synchronisation : Si ext2 et int2 Avec  $\text{ext2.D} = \text{int2.D}$

Synchronisation : Si int2 et ext3 Avec Pour tout int2 ayant  
 $\text{int2.JJ/MM/AAAA} + 8 = \text{ext3.JJ/MM/AAAA}$

**Remarque** : ce MCT fait apparaître un cycle caractérisé par une opération qui admet comme événement contributif un événement dont elle déclenche elle-même l'émission.

## **II. Fonctionnement d'un modèle dynamique**

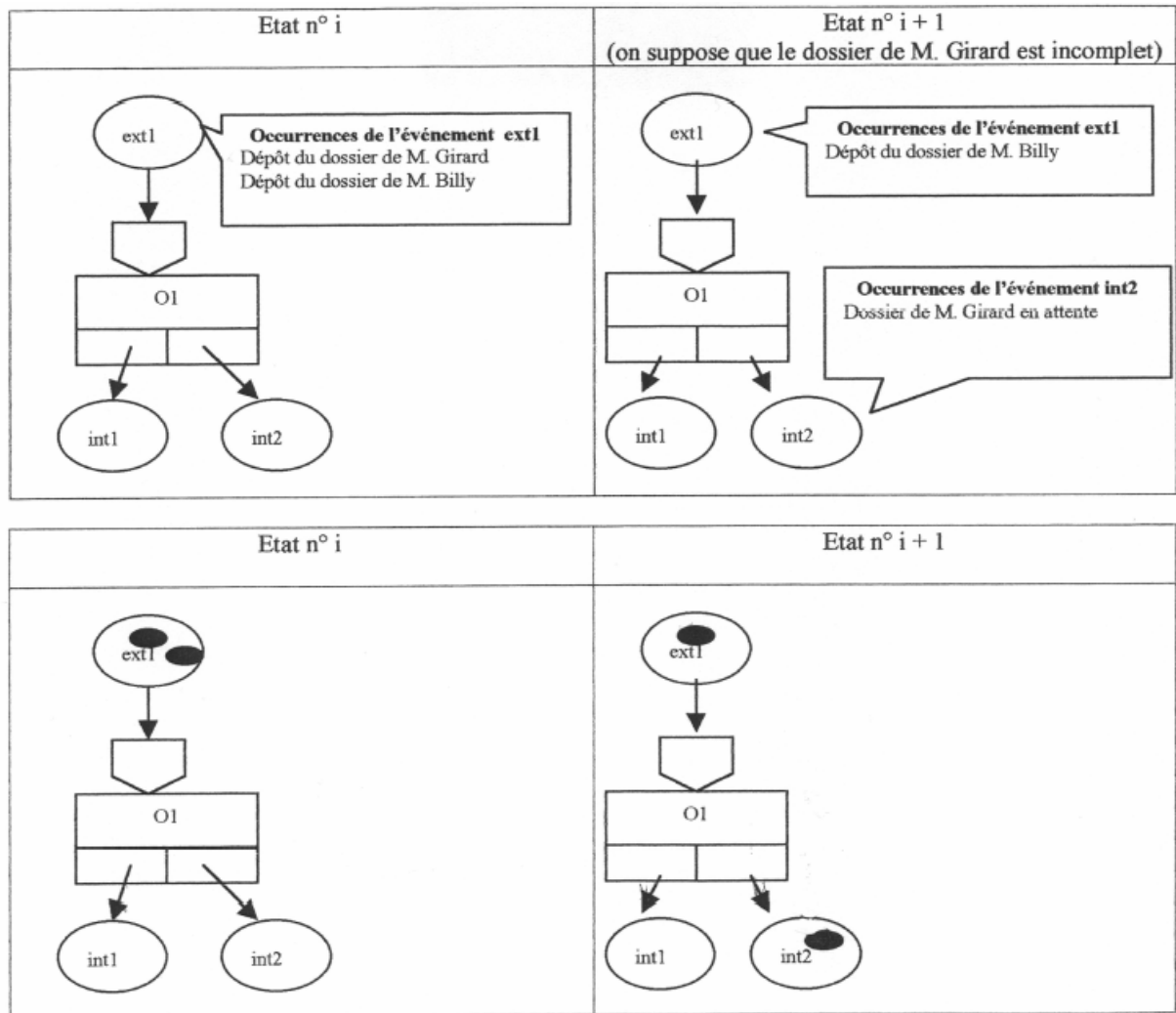
### **2.1) Fonctionnement d'un modèle dynamique**

L'arrivée d'un événement externe dans le système d'information provoque l'apparition d'une occurrence nouvelle pour cet événement. On appelle jeton cette occurrence d'événement. Une synchronisation, lorsqu'elle est en attente, devient activable, lorsque la proposition logique associée et les conditions locales deviennent vraies par l'arrivée d'un nouveau jeton. Lorsque la synchronisation est activée, il y a

consommation d'un ou de plusieurs jetons par événement qui a contribué à rendre vrai le prédicat et les conditions locales de synchronisation.

La synchronisation déclenche le démarrage de l'opération qui s'exécute et qui provoque l'apparition d'un ou de plusieurs jetons supplémentaires dans tous les événements en sortie de l'opération pour lesquels la règle d'émission est vérifiée.

Les tableaux qui suivent illustrent ce principe de consommation en premier lieu en terme d'occurrences d'événements et ensuite avec les jetons :



## 2.2) Règles de vérification du fonctionnement

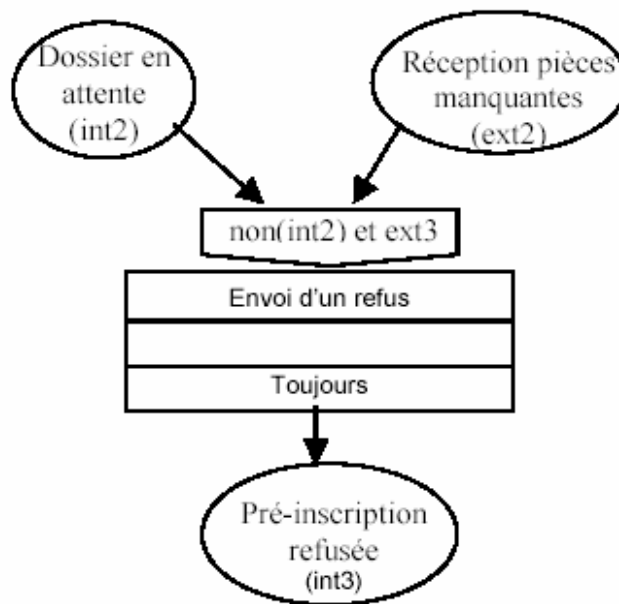
Compte tenu du principe de fonctionnement exposé précédemment, un modèle dynamique admet un ensemble d'états qui se matérialisent au moyen de jetons répartis dans ses différents événements. Pour que le modèle fonctionne correctement, différentes règles relatives à la consommation de ces jetons devront être vérifiées. On devra par exemple

s'assurer que des jetons ne s'accumulent pas dans un événement. Une telle situation signifierait que le système ne remplit pas sa fonction de consommation ou de traitement.

*Dans le processus de pré-inscription, il peut y avoir accumulation de jetons dans l'événement*

*« Réception de pièces manquantes » dès lors que le dossier mis en attente a été supprimé du fait du délai de 8 jours dépassé. C'est un choix du gestionnaire d'ignorer ou de traiter ces jetons. Si on décide, par exemple, d'envoyer un avis à l'élève afin de l'informer que ses pièces sont arrivées trop tardivement, on devra enrichir le MCT précédent par le diagramme suivant :*

*(ce traitement suppose que l'événement « pièces manquantes » soit porteur d'informations pour effectuer l'envoi ).*



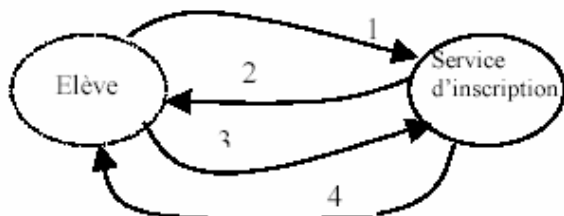
**Remarque** : il existe de nombreuses autres règles de vérification de fonctionnement détaillées dans l'ouvrage « La méthode Merise – principes et outils ».

### **III. Règles de construction d'un M.C.T.**

Comme pour le modèle conceptuel des données, il n'existe pas de méthode algorithmique permettant d'aboutir à un modèle conceptuel des traitements. Si la présentation de ses concepts peut en effet être entièrement formalisée et explicitée, leur assemblage pour résoudre un problème donné exige des qualités d'analyse et de réflexion que seule l'expérience peut accroître. Il existe cependant des outils ou des démarches d'aide à la conception d'un MCT.

Ainsi certains auteurs préconisent, pour faciliter la conception d'un MCT, l'élaboration d'un Modèle Conceptuel de Communication (MCC). Ce modèle consiste à recenser la liste de tous les acteurs intervenants dans le système d'information et à schématiser les flux d'information qu'ils échangent. Les acteurs internes - ceux qui appartiennent au système d'information - sont représentés dans des cercles en trait plein, les acteurs externes au système d'information sont représentés dans des cercles en pointillés. Les flux d'information sont schématisés par des arcs

entre acteurs. *Le schéma ci après représente les flux inhérents à la pré-inscription à la faculté des sciences.*



- |  |
|--|
| 1 Dépôt du dossier de préinscription<br>2. Pré-inscription acceptée ou dossier incomplet<br>3. Envoi des pièces manquantes<br>4. Pré-inscription refusée |
|--|

Ce diagramme met en évidence les événements externes du MCT.

Chaque flux échangé d'un acteur externe vers un acteur interne devient en effet un événement déclencheur externe.

Les flux à destination d'un acteur externe deviendront des événements internes. Les opérations ainsi que leurs événements résultats ne sont pas aisément déductibles de ce schéma.