

# INSIA – SIGL 2

## La méthode MERISE

### 1 : Introduction

Bertrand LIAUDET

## BIBLIOGRAPHIE

La méthode MERISE. Tome 1 : Principes et outils, Les éditions d'organisation, 1986.  
Tardieu, Rochfeld, Colletti.

La méthode MERISE. Tome 2 : Démarche et pratiques, Les éditions d'organisation, 1985.  
Tardieu, Rochfeld, Colletti, Panet, Vahée.

Ingénierie des systèmes d'information : Merise - Deuxième génération, Eyrolles, 2001,  
4<sup>ème</sup> édition (ISIM). Nanci, Espinasse.

Ingénierie des systèmes d'information, sous la direction de Corine Cauvet et Camille Rosenthal-Sabroux, Hermes, 2001.

## SOMMAIRE

<b>INTRODUCTION</b>	<b>3</b>
<b>1. Génie Logiciel vs Ingénierie des systèmes d'information</b>	<b>3</b>
Génie logiciel - Software	3
Ingénierie des systèmes d'information - Brainware	4
Relations entre software engineering et brainware engineering	5
<b>2. Généralités sur la méthode</b>	<b>6</b>
Définitions	6
Méthode et méthodologie	6
<b>3. Méthode analytique vs méthode systémique</b>	<b>7</b>
Méthode analytique	7
Brève présentation de la méthode systémique	8

<b>4. Développement d'un logiciel : Les 4 distinctions capitales</b>	<b>10</b>
Première distinction : Développement = Conception + Réalisation	10
Deuxième distinction : Conception = Analyse fonctionnelle + Analyse organique	10
Troisième distinction : Analyse organique = Architecture système + Analyse détaillée	11
Quatrième distinction : données versus traitements : l'analyse des données.	11
<b>5. Le cycle en V</b>	<b>12</b>
Présentation	12
La production des documents	13
Réalisation et langage de programmation	15
Cycle en V et analyse des données	15
<b>6. Système d'information</b>	<b>16</b>
Présentation théorique	16
Distinction entre SIO et SII	17
Distinction entre système entreprise et système logiciel	17
Relations entre SIO, SII, système entreprise et système logiciel	18
<b>7. La méthode MERISE</b>	<b>19</b>
Définition	19
Historique	19
Les 3 dimensions de la démarche MERISE	20
La distinction entre données et traitement	20
Le cycle d'abstraction	20
Le cycle de vie	23
Le cycle de décision	24
Les plans types	24

Première édition : Novembre 2008

Deuxième édition : Janvier 2010

# INTRODUCTION

*Il est facile de décrire la méthode MERISE, encore que son application exige à coup sûr savoir et pratique.*

**MERISE** : c'est une méthode systémique de conception des systèmes d'information. Elle est en relation avec le développement des bases de données relationnelles (SQL).

**Conception** : c'est une partie du développement du logiciel.

**Système d'information** : c'est un ou plusieurs logiciels manipulant un ensemble d'informations structurées cohérentes. Par exemple : l'intra de l'EPITECH : des cours, des élèves, des profs, des horaires, des projets, des groupes, des notes, etc. On peut les consulter, les créer, les modifier, les détruire.

Avant de présenter la méthode MERISE, on va présenter quelques notions générales sur la méthode, la conception et le système d'information.

## 1. Génie Logiciel vs Ingénierie des systèmes d'information

### Génie logiciel - Software

Le terme de génie logiciel (*software engineering*) est né en Europe à la fin des années 60.

Le G.L. vise à transformer les besoins et attentes des utilisateurs en une application informatique.

**Besoins et attentes** —————> **Application informatique**

**Quoi** : software

**Qui** : les informaticiens.

Le G.L. regroupe :

- Des **METHODES** (organisation du travail)
- Des **TECHNIQUES** (langages de programmation, documentation des programmes)
- Des **OUTILS** (compilateurs, systèmes de gestion de la documentation)

de développement du logiciel.

### **Ingénierie des systèmes d'information**

Le terme d'ingénierie des systèmes d'information (*requirement engineering*) est né au début des années 90.

L' I.S.I vise à transformer les besoins et attentes des utilisateurs en spécifications formalisées d'une future application informatique.

**Besoins et attentes** —————> **Spécifications formalisées**

**Quoi** : brainware

**Qui** : les informaticiens, les gestionnaires et les autres utilisateurs du système d'information

L' I.S.I. regroupe :

- Des **METHODES** d'organisation du travail de spécification : MERISE
- Des **TECHNIQUES** de modélisations : MCD, MLD, etc.
- Des **OUTILS** de modélisation et de spécifications : logiciel Win'Design, logiciel AMC designer...

utilisés pour le développement des spécifications.

### **Le brainware**

Le concept de brainware, très peu usité, a été introduit par Tosio Kitagawa en septembre 1974 dans le n°39 des Research Report of Research Institute of Fundamental Information Science.

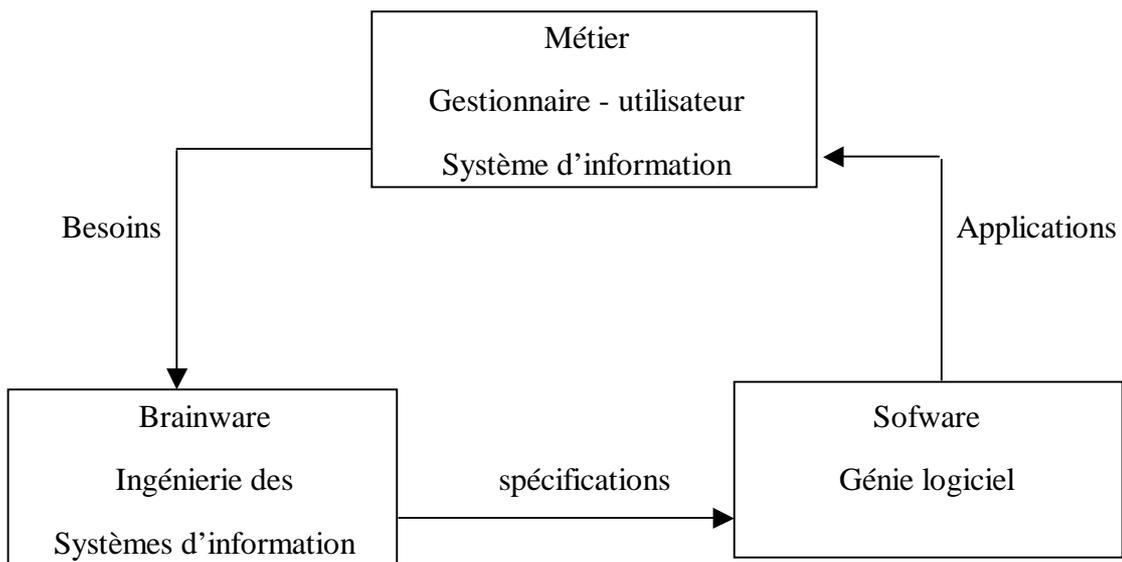
Le brainware est la fondation intellectuelle qui fonde le software.

Le brainware est un matériau (ware) en ce sens que c'est un stock objectif de connaissances et d'informations.

On peut donc distinguer entre :

**software ingenering - brainware ingenering**

**Relations entre software engineering et brainware engineering**



ISIM, p. 2

## 2. Généralités sur la méthode

### Définitions

1. Marche, ensemble de démarches que suit l'esprit pour découvrir et démontrer la vérité (dans les sciences).
2. Ordre réglant une activité; arrangement qui en résulte.
3. Ensemble de moyens raisonnés suivis pour arriver à un but

### Méthode et méthodologie

La méthodologie est la science des méthodes.

**Par abus de langage, la méthodologie est devenu la méthode** en tant que programme formel qui règle à l'avance une suite d'opérations à réaliser pour arriver à un résultat en signalant les difficultés à contourner.

C'est en général le résultat de principes théoriques et de retour d'expérience.

La méthodologie est une **connaissance très concrète et qui semble abstraite** car elle met en œuvre concrètement des opérations complexes et abstraites.

Si la maîtrise concrète de ces opérations complexes et abstraites n'est pas acquise, la méthodologie la méthodologie paraîtra très abstraite.

*C'est [ la méthode ] que l'on place le plus souvent en tête dans les écoles, comme propédeutique des sciences, alors que, selon le parcours de la raison humaine, elle est l'ultime étape, à laquelle la raison parvient uniquement quand la science est déjà terminée depuis longtemps et n'a plus besoin que de la dernière main pour être mise en ordre et atteindre la perfection. Car il faut que l'on connaisse les objets déjà à un assez haut degré, si l'on veut indiquer les règles selon lesquelles une science s'en peut mettre en œuvre.*

**Critique de la raison pure, 1781, Emmanuel Kant (1724-1804)  
Introduction de la logique transcendantale**

### 3. Méthode analytique vs méthode systémique

#### Méthode analytique

##### Discours de la méthode

La méthode analytique est la méthode de décomposition classique dont on retrouve les fondements chez Descartes :

*Certains chemins m'ont conduit à des considérations et des maximes dont j'ai formé une méthode par laquelle il me semble que j'ai moyen d'augmenter par degrés ma connaissance, et de l'élever peu à peu au plus haut point...*

*Au lieu de ce grand nombre de préceptes dont la logique est composée, je crus que j'aurais assez des quatre suivants, pourvu que je prisse une ferme et constante résolution de ne manquer pas une seule fois à les observer.*

***Le premier** était de ne recevoir jamais aucune chose pour vraie que je ne la connusse évidemment être telle; c'est-à-dire, d'éviter soigneusement la précipitation et la prévention, et de ne comprendre rien de plus en mes jugements que ce qui se présenterait si clairement et si distinctement à mon esprit, que je n'eusse aucune occasion de le mettre en doute.*

***Le second**, de diviser chacune des difficultés que j'examinerais, en autant de parcelles qu'il se pourrait, et qu'il serait requis pour les mieux résoudre.*

***Le troisième**, de conduire par ordre mes pensées, en commençant par les objets les plus simples et les plus aisés à connaître, pour monter peu à peu comme par degrés jusques à la connaissance des plus composés, et supposant même de l'ordre entre ceux qui ne se précèdent point naturellement les uns les autres.*

***Et le dernier**, de faire partout des dénombrements si entiers et des revues si générales, que je fusse assuré de ne rien omettre.*

Discours de la méthode, 1637, Descartes (1596-1650)

Ce discours met en avant quatre principes :

1. L'évidence contre les préjugés pour décrire la réalité.
2. L'analyse : La division du tout en partie.
3. La synthèse : la reconstruction du tout à partir des parties.
4. La totalité : ne rien omettre.

##### L'analyse descendante et ses étapes

La bonne méthode consiste à diviser le tout en parties, puis à réaliser les parties, avant de les réintégrer toutes ensembles. C'est l'analyse descendante.

Les étapes sont les suivantes :

1. Partir de la réalité : le problème à traiter.
2. S'en faire une idée claire et complète.
3. Diviser cette idée en parties.
4. Construire les parties une par une.
5. Intégrer les parties toutes ensemble.

### Les erreurs de méthode

#### ➤ *ne pas partir du tout*

Une erreur classique consiste à ne pas partir du tout, mais à partir directement des parties. C'est **l'analyse ascendante : on part de l'étape 4.**

Le défaut de cette méthode est que l'absence d'analyse initiale de l'idée de la totalité va conduire à des grandes difficultés au moment de l'intégration des parties.

#### ➤ *se tromper sur le tout*

En appliquant l'analyse descendante, on peut aussi se tromper en appliquant mal l'étape 2.

L'idée qu'on se fait du problème à traiter est incomplète ou fausse. En conséquence de quoi l'intégration des parties ne pourra pas donner un bon résultat.

### Breve présentation de la méthode systémique

La science des systèmes, ou systémique, est à l'origine du développement de la notion de système d'information.

### Caractéristiques des systèmes étudiés par la systémique

- Ils ont un projet identifiable : c'est **l'hypothèse téléologique** (télos = finalité)
- Ils sont ouverts sur leur environnement : c'est **l'hypothèse d'ouverture.**
- Ils sont décrits totalement, dans l'espace et le temps : c'est **l'hypothèse structuraliste.**

### Caractéristiques des systèmes ouverts

- La **rétroaction** : la rétroaction consiste à ce que les informations en sortie du système reviennent en entrée dans le système.
- **Equifinalité** : les mêmes conséquences peuvent avoir des origines différentes.
- La **simulation** : du fait de la rétroaction et de l'équifinalité, les systèmes ouverts s'étudient avec des simulations.

### Méthode d'analyse systémique

La méthode systémique consiste à analyser le système non pas par fonction, mais par structure.

On recherche les invariants au niveau fonctionnel. Ces invariants sont la structure du système : ce sont **les données brutes** et **les règles d'organisation**. Ils permettent la modélisation : la représentation du réel par un modèle.



#### 4. Développement d'un logiciel : Les 4 distinctions capitales

Il y a quatre distinctions capitales dans le développement d'un logiciel.

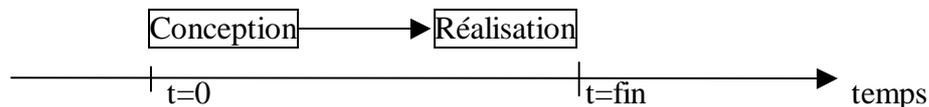
##### Première distinction : Développement = Conception + Réalisation

Le développement se compose de deux activités qu'on peut distinguer : la conception et la réalisation.

- **La conception** consiste à comprendre et prévoir ce qu'il a à faire.
- **La réalisation** consiste à faire concrètement ce qu'il y a à faire.

La distinction entre la conception et la réalisation est une façon d'organiser la division du travail.

Le premier principe de la méthode consiste à considérer ces deux activités comme deux étapes successives :



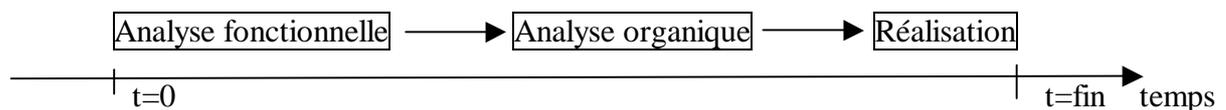
Le projet se déroule dans le temps : il commence avec la conception, il se termine avec la réalisation.

La division du travail consiste à mettre en évidence les étapes de la réalisation d'un logiciel.

##### Deuxième distinction : Conception = Analyse fonctionnelle + Analyse organique

La conception se divise en deux parties :

- **L'analyse fonctionnelle** (ou analyse générale, ou spécifications fonctionnelles) d'abord. L'analyse fonctionnelle s'occupe des fonctions (ou des services) que le système offre à ses utilisateurs.
- **L'analyse organique** (ou architectonique<sup>1</sup>) ensuite. L'analyse organique s'occupe de la façon dont sera construit le système pour répondre aux attentes de l'analyse fonctionnelle.



<sup>1</sup> L'architectonique c'est la technique de la construction, mais aussi la structure ou l'organisation de la construction. Est architectonique ce qui est conforme à la technique de l'architecture.

<b>ANALYSE FONCTIONNELLE</b>	<b>ANALYSE ORGANIQUE</b>
EXTERNE	INTERNE
Le QUOI	Le COMMENT
<b>Point de vue</b> de l'utilisateur et du client, le maître d'ouvrage, <b>MOA</b> : celui qui commande le logiciel	<b>Point de vue</b> de l'informaticien et du maître d'œuvre, <b>MOE</b> : celui qui réalise le logiciel
<i>Build the right system</i>	<i>Build the system right</i>

Avec cette distinction, on fait apparaître :

- le point de vue de l'utilisateur : le maître d'ouvrage (l'utilisateur, le client)
- le point de vue de l'informaticien : le maître d'œuvre.

**Pour l'utilisateur**, ce qui compte, c'est l'usage du système : les cas d'utilisation (vocabulaire UML). L'analyse fonctionnelle permettra de modéliser l'ensemble des cas d'utilisation.

**Pour l'informaticien**, ce qui compte c'est l'architecture interne du système.

L'analyse fonctionnelle garantit qu'on va bien faire ce qui est demandé : répondre aux exigences du client.

L'analyse organique garantit que ce qu'on va faire, on va bien le faire.

<b>Troisième distinction : Analyse organique = Architecture système + Analyse détaillée</b>
---

L'analyse organique se divise en deux parties :

- **L'architecture système** (ou analyse organique générale): elle s'occupe de l'organisation des sous-systèmes logiciels et matériels du système complet.
- **L'analyse détaillée** (ou analyse organique détaillée, ou spécifications détaillées) : elle s'occupe du découpage en procédure et en fonctions informatiques de chacun des sous-systèmes logiciels. A ce niveau vont apparaître les en-têtes des fonctions, voir leurs pseudo-codes.

<b>Quatrième distinction : données versus traitements : l'analyse des données.</b>
--

Les trois distinctions précédentes sont centrées sur la questions des traitements.

La dernière distinction est celle qui est faites entre les données et les traitements.

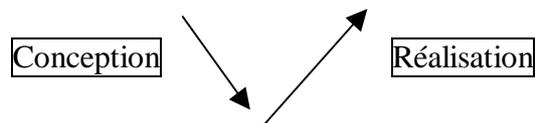
Les données seront analysées pour elle-même, indépendamment des traitements qu'on leur appliquera.

## 5. Le cycle en V

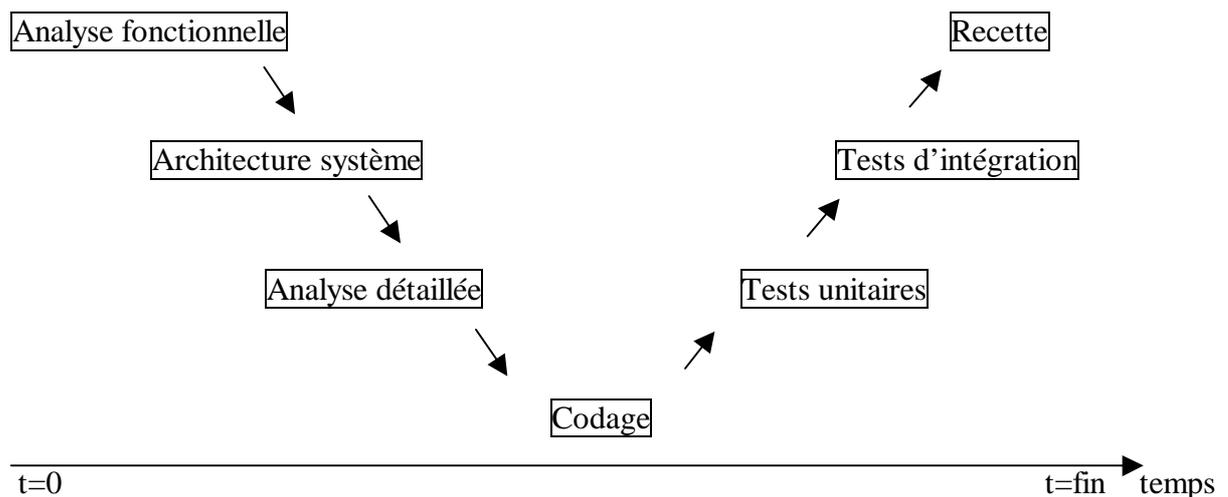
### Présentation

Le cycle en V c'est une méthode classique de développement du logiciel.

Dans cette méthode, la conception et la réalisation forment les deux branches du cycle en V :



Ces deux étapes sont détaillées en reprenant les 3 premières distinctions abordées précédemment et en ajoutant des distinctions dans la réalisation :



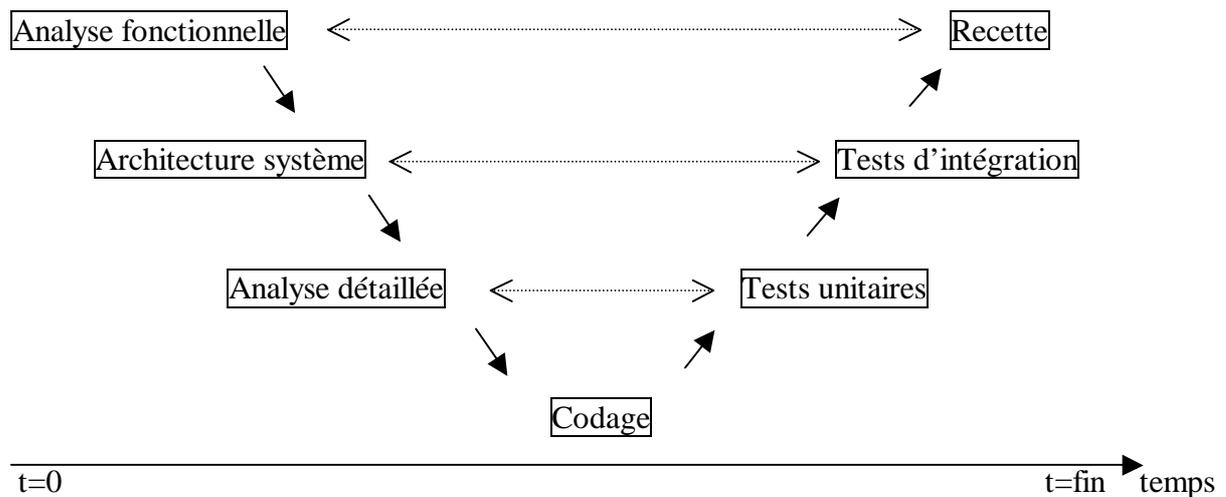
**Conception** = Analyse fonctionnelle + Architecture système + Analyse détaillée.

**Réalisation** = Codage + Tests unitaires + Tests d'intégration + Recette.

C'est le lien entre les étapes de chaque branche qui justifie le cycle en V :

- Quand on fait l'analyse fonctionnelle, on peut préparer la procédure de recette.
- Quand on fait l'architecture système, on peut préparer les tests d'intégration des sous-systèmes.
- Quand on fait l'analyse détaillée, on peut préparer les tests unitaires

Ainsi, cela permettra, en cas de problème de test (unitaire, d'intégration ou de recette), de revenir facilement à la partie de la conception à laquelle le problème correspond.



### La production des documents

Le cycle en V correspond aussi à un cycle de consommation et de production des documents.

Les activités du cycle en V utilisent les documents des activités précédentes et produisent des documents qui seront utilisés aux étapes suivantes : l'étape immédiatement suivante dans le cycle en V et l'étape de même niveau dans le cycle en V.

**Le client** (maître d'ouvrage) produit le cahier des charges.

**L'analyse fonctionnelle** se base sur le cahier des charges.

Elle aboutit à un document d'analyse fonctionnelle. Ce document pourra être validé par le client de façon à vérifier la bonne compréhension du cahier des charges par l'informaticien. Ce document servira d'entrée pour l'architecture et l'analyse détaillée.

Elle aboutit aussi à un document de recette et au manuel utilisateur. Le document de recette servira d'entrée pour la recette. Le manuel utilisateur servira au client une fois le produit livré.

**L'architecture** se base sur le document d'analyse fonctionnelle et éventuellement sur le cahier des charges.

Elle aboutit à une document d'architecture et à un document d'intégration.

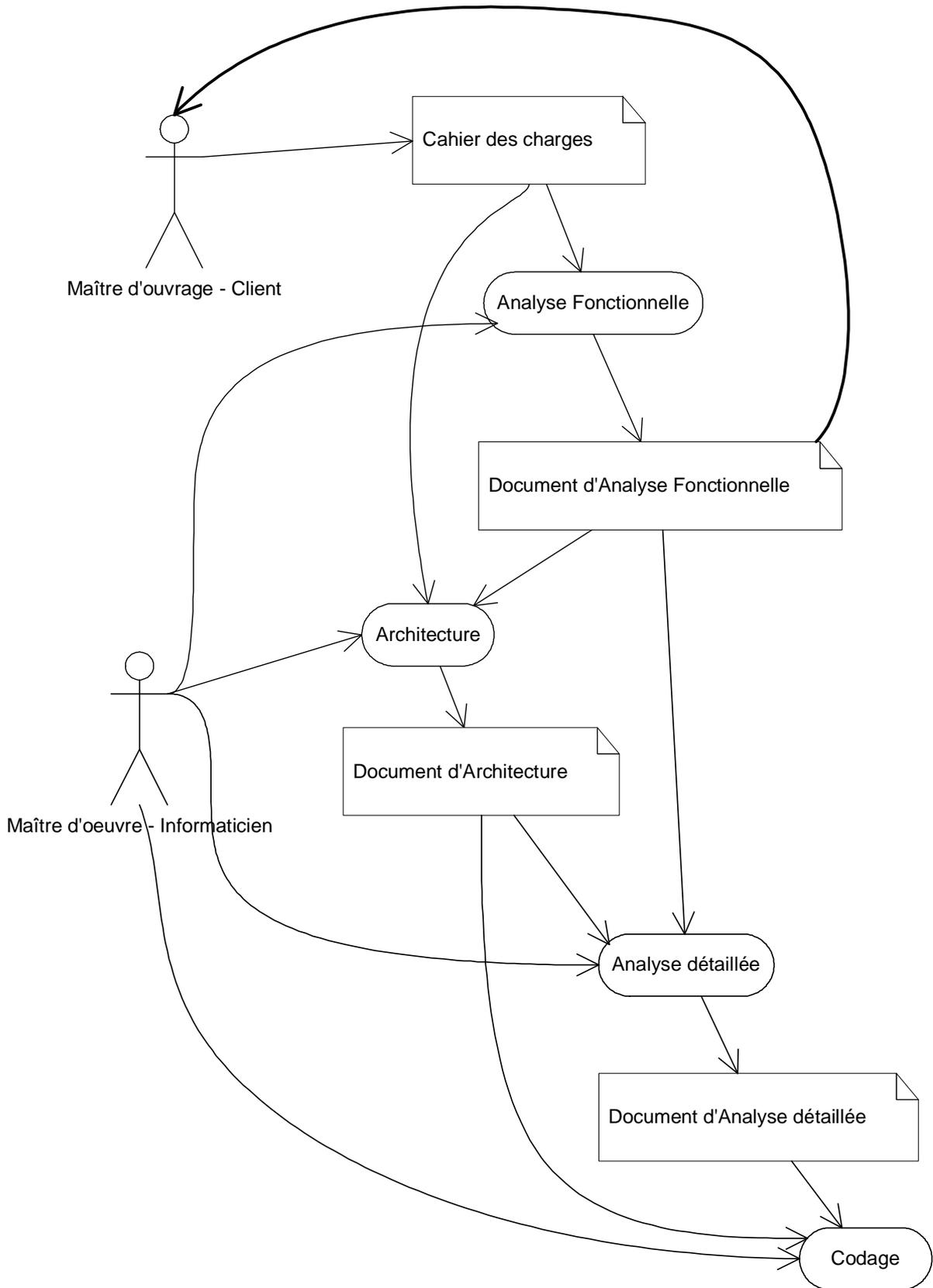
**L'analyse détaillée** se base sur le document d'analyse fonctionnelle et sur le document d'architecture. A ce niveau, on n'utilise plus le cahier des charges. Elle aboutit à un document d'analyse détaillée et à un document de test unitaire.

### Remarques

L'analyse fonctionnelle produit aussi un document de recettes qui sera utilisé à la fin par l'activité de recette.

L'architecture produit aussi un document d'intégration qui sera utilisé par l'activité d'intégration.

L'analyse détaillée produit aussi un document de tests unitaires qui sera utilisé par l'activité de tests unitaires.



**Cycle de la documentation**

## Réalisation et langage de programmation

Une fois la conception terminée, on passe à la réalisation.

La réalisation peut se faire avec n'importe quel langage.

Toutefois, dans le cas d'un SI centré sur une base de données, on utilisera probablement le SQL pour la partie directement liée à la base de données.

Pour l'interface utilisateur, on utilisera indifféremment : C,C++, Java, php, mais aussi des environnements de développement rapide du type de 4D (quatrième dimension) ou de Oracle Database XE (freeware depuis mars 2006).

## Cycle en V et analyse des données

Le cycle en V ne prend pas explicitement en compte l'analyse des données des données.

L'analyse des données peut être considérée comme une partie de chaque étape de la conception. On commence au niveau de l'analyse fonctionnelle : c'est le MCD MERISE et le diagramme des classes métier.

On continue au niveau de l'architecture : c'est le MOD MERISE.

On finit au niveau de l'analyse détaillée : c'est le MLD et MPD MERISE et le diagramme de classes détaillé.

## 6. Système d'information

### Présentation théorique

La notion de système d'information est une notion issue de la science des systèmes (ou systémique).

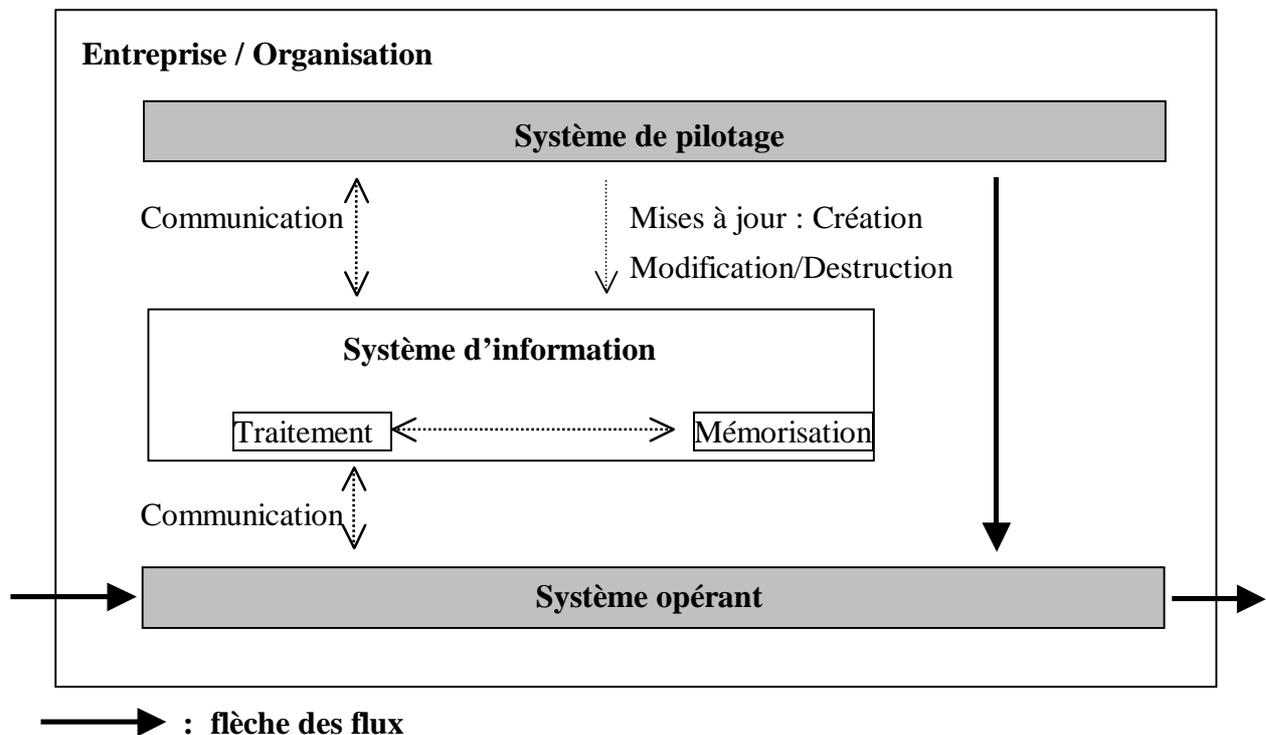
Un système est un ensemble d'éléments reliés entre eux et compris dans un ensemble plus grand.

Le système d'information est une représentation possible de n'importe quel système, notamment de tout système humain organisé.

Les systèmes d'information préexistent donc à l'informatique.

L'analyse systémique permet d'arriver à la modélisation suivante de l'entreprise :

### Environnement



**Le système opérant – SOP** - est le siège de l'activité productive de l'entreprise. Cette activité consiste en une transformation de données en entrée (les flux primaires). Ces flux primaires peuvent être des flux de matière, de finance, de personnel ou d'information. Par exemple : le système opérant reçoit une commande et la traite.

**Le système de pilotage** est le siège de l'activité décisionnelle de l'entreprise. Il permet le pilotage, la régulation et l'adaptation, par la communication avec le SI, la mise à jour du SI et l'envoi de décisions au SOP. Cette activité décisionnelle est très large et elle est assurée par de

nombreux acteurs de l'entreprise à des niveaux divers. Par exemple : le système de pilotage décide d'une campagne publicitaire ou de l'installation d'une nouvelle application informatique dans le système d'information.

**Le système d'information** est un système de mémorisation et de traitement de l'information au sens large, interfacé avec le système opérant et le système de pilotage. Ce système est en partie informatisé

**Distinction entre SIO et SII**

On distingue dans le SI entre :

- **Le système d'information organisationnel**, SIO. C'est une représentation possible de n'importe quel système, notamment de tout système humain organisé (donc de toute entreprise). Une entreprise peut donc être considérée comme un SIO.

Et

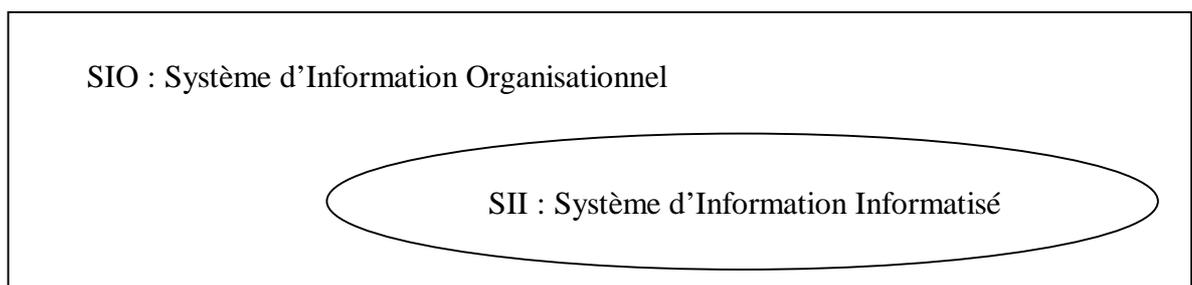
- **Le système d'information informatisé**, SII, c'est la partie informatisée du système d'information organisationnel à laquelle les utilisateurs ont accès. Il est constitué par le ou les logiciels (les applications) qu'on utilise dans l'entreprise, par les fichiers, la ou les bases de données, le ou les SGBD.

Un SIO peut contenir un SII, mais ce n'est pas obligatoire. Tout SII est inclus dans un SIO.

Le SIO est tourné vers les utilisateurs et fera appel à certaines disciplines des sciences de la gestion.

Le SII est sous la responsabilité des informaticiens et fera appel aux disciplines du génie logiciel.

Cependant, un SII est au service du SIO mis en place par les dirigeants de l'entreprise, et non l'inverse ! La conception du SII doit s'appuyer sur celle du SIO et non l'inverse !

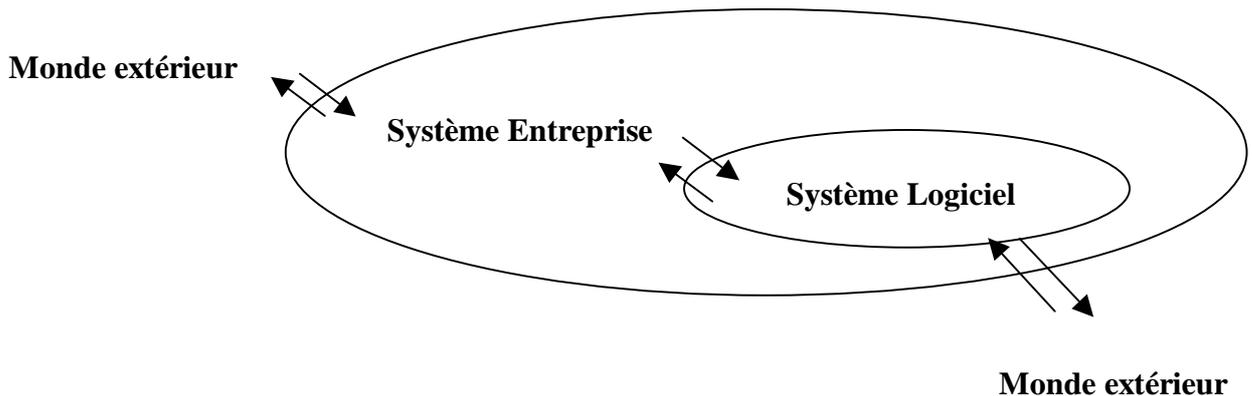


**Distinction entre système entreprise et système logiciel**

Dans une première analyse, on ne va pas s'intéresser directement à l'activité réalisée via un logiciel, mais plus généralement à l'activité réalisée par l'entreprise.

On peut distinguer 3 lieux : l'entreprise (le système entreprise), le monde extérieur et le logiciel (système logiciel).

Ces trois lieux sont des abstractions concentriques : l'entreprise inclut le logiciel et le monde extérieur inclut l'entreprise.



On va ensuite décrire les échanges entre ces trois lieux :

- Le monde extérieur communique avec l'entreprise.
- L'entreprise communique avec son système logiciel.
- Le monde extérieur peut aussi communiquer directement avec le système logiciel (borne automatique, site internet...).

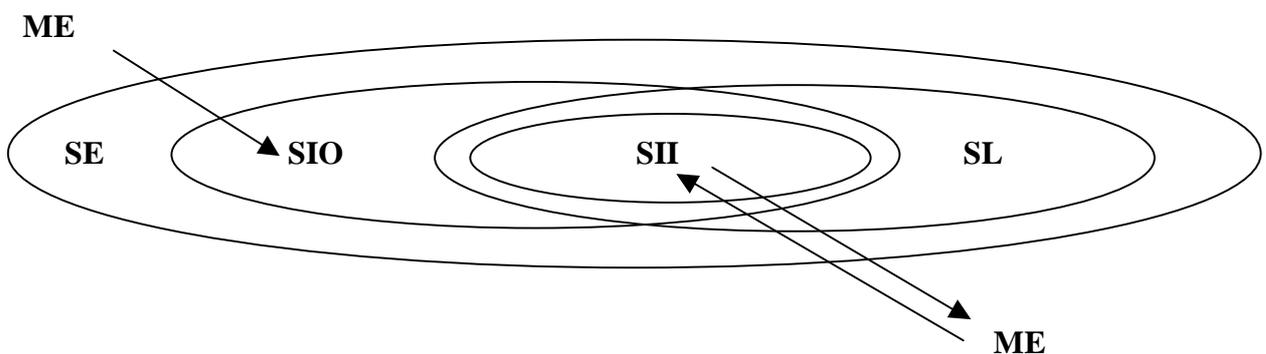
**Relations entre SIO, SII, système entreprise et système logiciel**

Le SIO est un ensemble inclus dans le système entreprise. On peut considérer en première approximation qu'ils sont identiques.

Le SIO contient une partie du système logiciel. On peut considérer en première approximation que le SIO contient la totalité du système logiciel.

Le SII, c'est la partie du système logiciel contenue par le SIO.

Le SII est donc inclus dans le système logiciel. On peut considérer en première approximation qu'ils sont identiques.



## 7. La méthode MERISE

### Définition

MERISE est une méthode systémique de conception des systèmes d'information.

Elle est en relation avec le développement des bases de données relationnelles.

### Historique

1970	Modèle Relationnel de Codd.
Années 70	Premiers prototypes de SQL
1976	Modèle Entité Association de Chen
1974-78	Le noyau de MERISE est établi par une équipe d'ingénieurs et de chercheurs aixois.
1978	Développement de MERISE : méthode française de conception de systèmes d'information, sous l'égide du ministère de l'industrie.
1979	Conception du système d'information, construction de la base de données, H. Tardieu, D. Nanci, D. Pascot (préfacé par J.-L. Le Moigne), Editions d'Organisation.
1979	Première version de SQL, proposé par ORACLE.
1983	La méthode MERISE - Tome 1 : principes et outils. H. Tardieu, A. Rochfeld, R. Colletti. Éditions d'Organisation.
1985	La méthode MERISE - Tome 2 : démarche et pratique. H. Tardieu, A. Rochfeld, R. Colletti, G. Panet, G. Vahée. Éditions d'Organisation.
1986	SQL ANSI (American National Standard Institute)
1989	SQL-1, ISO et ANSI (International Standard Organisation)
1989	La méthode MERISE - Tome 3 : gamme opératoire. A. Rochfeld, J. Moréjon. Édition d'Organisation.
1992	Ingénierie des systèmes d'information : MERISE. 1 <sup>ère</sup> édition. D. Nanci, B. Espinasse. Sybex.
1992	SQL-2, ISO et ANSI
fin années 90	PHP-MySQL
1999	SQL-3, ISO et ANSI
2001	Ingénierie des systèmes d'information : MERISE. 4 <sup>ème</sup> édition. D. Nanci, B. Espinasse. Vuibert.
2006	Oracle Database XE

En 2001, la méthode MERISE était encore la méthode de conception de systèmes d'information la plus largement pratiquée en France.

MERISE a pris en compte les évolutions de l'informatique et continue de s'adapter aux nouvelles technologies : architectures clients/serveur, interfaces graphiques, démarche de développement rapide, approche objet, applications intra/internet.

Aujourd'hui, la méthode MERISE correspond encore globalement aux savoir-faire actuels en ingénierie des systèmes d'information de gestion.

MERISE constitue un standard de fait en conception des systèmes d'information.

### Les 3 dimensions de la démarche MERISE

La démarche de développement proposée par MERISE s'inscrit dans trois dimensions :

- **Le cycle de vie** : c'est le découpage du projet en trois périodes : conception, réalisation et maintenance. Le cycle de vie rejoint le cycle en V.
- **Le cycle de décision** : c'est la liste de tous les moments où une décision est prise sur le projet (décision de faire le projet après une étude préalable, décision de valider l'analyse fonctionnelle et de passer à l'architecture, validation de la recette, etc.)
- **Le cycle d'abstraction** : c'est l'organisation structurelle des données et des traitements.

On va surtout s'intéresser au cycle d'abstraction.

### La distinction entre données et traitement

Le cycle d'abstraction est basé sur une distinction entre les données et les traitements.

C'est la dichotomie fondamentale de MERISE.

Elle est directement issue de l'approche base de données.

### Le cycle d'abstraction

Le cycle d'abstraction est découpé en quatre niveaux : conceptuel, organisationnel, logique et physique.

- **Le niveau conceptuel** : il exprime des choix fondamentaux de gestion (recherche d'éléments stables indépendamment des moyens à mettre en œuvre, de leurs contraintes et de leur organisation). Répond à la question : **QUOI**.
- **Le niveau organisationnel** : il exprime les choix d'organisation de ressources humaines et matérielles, au travers notamment de la définition d'acteurs et de postes de travail. Répond aux questions : **QUI, OU, QUAND**.
- **Le niveau logique** : il exprime les choix de moyens et de ressources informatiques, en faisant abstraction de leurs caractéristiques techniques précises. C'est le niveau du modèle relationnel (moyen informatique : base de données relationnelle), du diagramme des classes et des diagrammes de séquence objets (moyen informatique : langage orienté objet). Répond à la question : **COMMENT**.
- **Le niveau physique** : il traduit les choix techniques et la prise en compte de leurs spécificités. C'est le niveau du code dans un langage particulier.

<b>LE CYCLE D'ABSTRACTION</b>		
<b>Niveaux</b>	<b>DONNEES</b>	<b>TRAITEMENTS</b>
<b>CONCEPTUEL</b>  <b>QUOI</b>	<b>M C D</b>  <i>Modèle conceptuel des données</i>  Signification des informations sans contraintes techniques, organisationnelle ou économique.  <b>Modèle entité – association</b>	<b>M C T</b>  <i>Modèle conceptuel des traitements</i>  Activité du domaine sans préciser les ressources et leur organisation
<b>ORGA-NISATIONNEL</b>  <b>QUI, OU, QUAND</b>	<b>M O D</b>  <i>Modèle organisationnel des données</i>  Signification des informations avec contraintes organisationnelles et économiques. (Répartition et quantification des données ; droit des utilisateurs)	<b>M O T</b>  <i>Modèle organisationnel des traitements</i>  Fonctionnement du domaine avec les ressources utilisées et leur organisation (répartition des traitements sur les postes de travail)
<b>LOGIQUE</b>  <b>COMMENT</b>	<b>M L D</b>  <i>Modèle logique des données</i>  Description des données tenant compte de leurs conditions d'utilisation (contraintes d'intégrité, historique, techniques de mémorisation).  <b>Modèle relationnel</b>	<b>M L T</b>  <i>Modèle logique des traitements</i>  Fonctionnement du domaine avec les ressources et leur organisation informatique.
<b>PHYSIQUE</b>  <b>COMMENT</b>	<b>M P D</b>  <i>Modèle physique des données</i>  Description de la (ou des) base(s) de données dans la syntaxe du Système de Gestion des données (SG.Fichiers ou SG Base de Données)  Optimisation des traitements (indexation, dénormalisation, triggers).	<b>M P T</b>  <i>Modèle physique des traitements</i>  Architecture technique des programmes

D'après ISIM, p. 37

**SIO et SII dans le cycle d'abstraction**

<b>Niveau</b>	<b>DONNÉES</b>	<b>TRAITEMENTS</b>	<b>SI</b>
<b>Conceptuel</b>	<b>M C D</b>	<b>M C T</b>	<b>SIO</b> Système d'information organisationnel
<b>Organisationnel</b>	<b>M O D</b>	<b>M O T</b>	
<b>Niveau logique</b>	<b>M L D</b>	<b>M L T</b>	<b>SII</b> Système d'information informatisé
<b>Niveau physique</b>	<b>M P D</b>	<b>M P T</b>	

ISIM, p. 218

## Le cycle de vie

Le cycle de vie MERISE est une méthode de développement au même titre que le cycle en V.

Le cycle de vie est découpé en trois périodes: la conception, la réalisation et la maintenance.

LE CYCLE DE VIE		
Etapes de la démarche		Explications
<b>Conception</b>	Schéma directeur	Définition des orientations générales du développement à moyen terme des systèmes d'information
	Étude préalable	Proposition et évaluation de solutions d'organisation et de solutions techniques pour le SI d'un domaine.  Cette étape porte sur un sous-ensemble représentatif du domaine étudié.
	Étude détaillée	Spécifications complètes du futur SIO du point de <b>vue de l'utilisateur</b> (point de vue externe).  Elle comporte deux phases : <ul style="list-style-type: none"><li>• la conception générale (extension de l'étude préalable à tout le domaine)</li><li>• la conception détaillée ( description complète de chacune des tâches à automatiser).</li></ul>
<b>Réalisation</b>	Étude technique	Spécifications complètes du futur SII du <b>point de vue du réalisateur</b> (point de vue interne).
	Production logicielle	Écriture des programmes, générations des fichiers ou des bases de données, tests.
	Mise en service	Installation de l'application informatique, vérification du bon fonctionnement, mise en place de la nouvelle organisation, formation des utilisateurs.
<b>Maintenance</b>	Maintenance	Rectification des anomalies, améliorations, évolutions.

ISIM, p. 32

## Le cycle de décision

Le cycle de décision représente l'ensemble des choix qui doivent être faits durant le déroulement du cycle de vie.

Étapes de la démarche		Résultats	Décisions
Schéma directeur	MOA	Plan de développement des SI	Approbation et mise en application
Étude préalable	MOA	Dossier des choix, n solutions	Choix d'une solution ou arrêt
Étude détaillée	MOE	Spécifications fonctionnelles	Accord des utilisateurs sur les spécifications fonctionnelles
Étude technique	MOE	Spécifications techniques pour la réalisation	Accord des réalisateurs sur les spécifications techniques
Réalisation logicielle	MOE	Système réalisé en ordre de marche	Recette provisoire : conformité du système
Mise en service	MOE	Système installé dans l'organisation	Recette définitive : système en service
Maintenance	MOE	Système maintenu	Recette simplifiée : fin de maintenance

ISIM, p. 41

## Les plans types

Les étapes de la démarche du cycle de vie donnent lieu à la production de documents.

Comme toute autre méthode, la méthode MERISE propose des plans types pour tous les documents prévus par la méthode.

L'étude préalable et de l'étude détaillée sont les deux études sont les plus spécifiques à MERISE car elles font intervenir l'essentiel du cycle d'abstraction.

On présente ci-dessous les plans type de ces deux étapes.

### **Plan type de l'étude préalable : production du cahier des charges**

- **1. Recueil**
  - Préparation et réalisation des interviews
  - Recherche de la documentation
  - Description et bilan de l'existant
- **2. Conception**
  - Élaboration des divers scénarios

- Élaboration des MCD et MCT
- Maquette et prototype
- Élaboration du cahier des charges fonctionnel
- **3. Qualité**
  - Définition des exigences qualité globale
  - Définition des exigences qualité par fonction
- **4. Chiffrage**
  - Estimation prévisionnelle des charges, coût, délais
  - Planning prévisionnel
- **Résultats obtenus :**
  - **Cahier des charges fonctionnel**
  - Dossier de choix

### **Plan type de l'étude détaillée : production de spécifications**

- **1. Recueil complémentaire**
  - Préparation et réalisation des interviews des utilisateurs
  - Recherche de la documentation
  - Actualisation de l'étude préalable
- **2. Conception**
  - Mise à jour des MCD et MCT
  - Élaboration du MOT
  - Description des états et des écrans
  - Validation croisée MCD / MOT
  - Élaboration du MLD
- **3. Qualité**
  - Définition des facteurs qualité
- **4. Chiffrage**
  - Estimations globale et détaillée
  - Plannings global et détaillé
- **Résultats obtenus :**
  - **Dossier des spécifications fonctionnelles**
  - Plan de développement logiciel