
1

PRESENTATION GENERALE DE LA METHODOLOGIE

Cette première partie rappelle les éléments caractéristiques et la démarche de la méthodologie MCSE (Méthodologie pour la Conception des Systèmes Electroniques) décrits dans l'ouvrage "Spécification et conception des systèmes, une Méthodologie MCSE" de l'auteur [CALVEZ-90]. Ce schéma d'organisation des développements est illustré par toutes les études de cas qui suivent.

La méthodologie MCSE couvre les phases de Spécification, de Conception Fonctionnelle aussi appelée par ailleurs Conception Préliminaire, de définition de la Réalisation ou Conception Détaillée.

Développée spécialement pour les applications temps-réel en contrôle/commande, elle a été appliquée sur de multiples problèmes industriels. L'adéquation du modèle à la problématique traitée a ainsi été validée ainsi que l'intérêt de toutes les étapes.

Si cette méthodologie concerne plus particulièrement le domaine des systèmes électroniques temps-réel, il se trouve que la démarche suivie pour les premières phases du développement - spécification, conception architecturale - se trouve indépendante de la réalisation, puisqu'il s'agit d'une approche SYSTEME. Des expériences intéressantes ont été entreprises quant à son utilisation pour des applications variées : systèmes de contrôle/commande, réseaux et protocoles, systèmes répartis, outils interactifs, conception de composants intégrés.

Une large expérimentation de la méthodologie a montré l'importance du modèle, puis l'importance des règles et conseils qui engendrent chez les concepteurs des solutions de qualité. Nous avons aussi constaté que les solutions de qualité peuvent s'exprimer sous la forme de modèles génériques. Ainsi, la connaissance de tels modèles génériques de solutions pour diverses classes de problèmes facilite la tâche du concepteur et favorise la production de solutions de qualité au sens : lisibilité, maintenabilité, efficacité ... et donc coût.

En aval de la conception, la phase de définition de la réalisation propose une démarche systématique qui conduit, à partir d'une solution fonctionnelle, à choisir la technique de réalisation la plus appropriée pour le problème (type de technologie, répartition matériel/logiciel, respect des contraintes de temps...).

1.1. DESCRIPTION D'UN SYSTEME

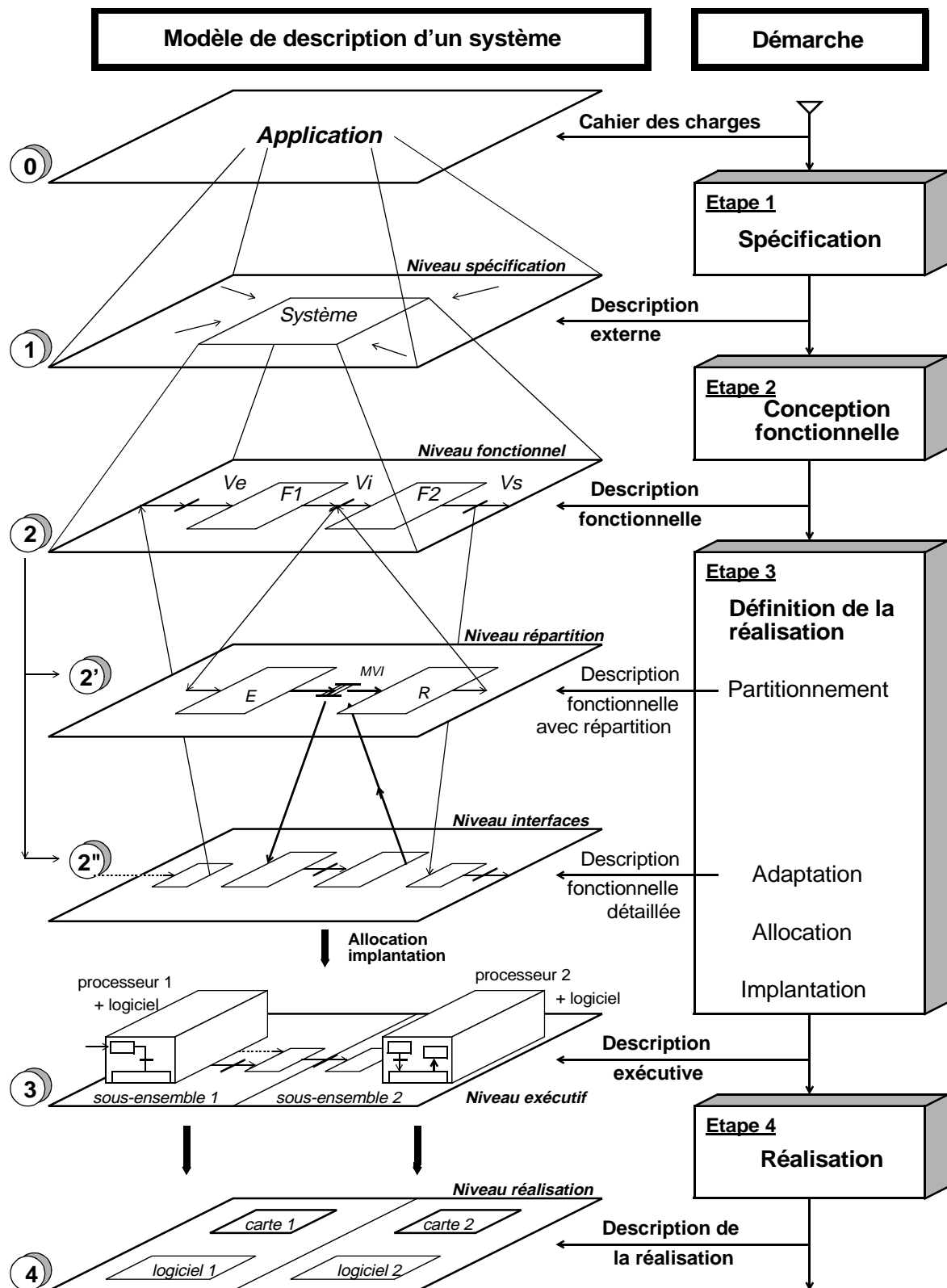
Une méthodologie est un guide pour le développement d'applications. MCSE préconise une démarche globalement descendante qui conduit à rechercher par approches successives, une réalisation appropriée.

La décomposition en étapes est basée sur la possibilité de décrire un système selon plusieurs niveaux d'abstraction. Ainsi une méthodologie se comprend bien lorsque le modèle de description des systèmes à concevoir est explicite.

Pour résumer la démarche préconisée, la structuration d'un système pour MCSE est décrite selon un ensemble de niveaux comme l'indique la figure ci-après.

Si on part du niveau le plus abstrait pour descendre jusqu'au niveau de la réalisation, on trouve:

- le niveau **Cahier des charges**, qui correspond à l'expression du besoin. Pour ce niveau, le système n'est pas encore défini. Les informations concernent l'application dans son ensemble avec les objectifs souhaités.
- le niveau **Spécification**. Il correspond à la description du système à concevoir selon une vue purement externe. Les spécifications incluent toutes les contraintes auxquelles doit satisfaire le système.
- le niveau **Description Fonctionnelle**, exprimant la structure du système sur le plan fonctionnel. Cette description interne a la particularité d'être indépendante de toutes les contraintes technologiques.
- le niveau **Description exécutive**, qui explicite les spécifications complètes de la réalisation en décrivant la partie matérielle par une structure d'exécution et la partie logicielle par des schémas d'implantation logicielle. Entre ce niveau et le précédent 2 sous-niveaux sont intéressants comme intermédiaires:
 - le niveau fonctionnel avec les **interfaces** qui découle du sous-niveau précédent avec ajout de toutes les interfaces physiques nécessaires pour le couplage avec l'environnement réel du système.
 - le niveau fonctionnel avec **répartition** qui résulte de la description fonctionnelle en y ajoutant la contrainte de répartition géographique, ce qui se traduit par l'utilisation de mécanismes pour le transport des informations entre les sous-ensembles répartis.
- le niveau **Réalisation** qui décrit la solution finale comme un ensemble de cartes et de logiciels.



-Figure 1.1- Niveaux de description d'un système.

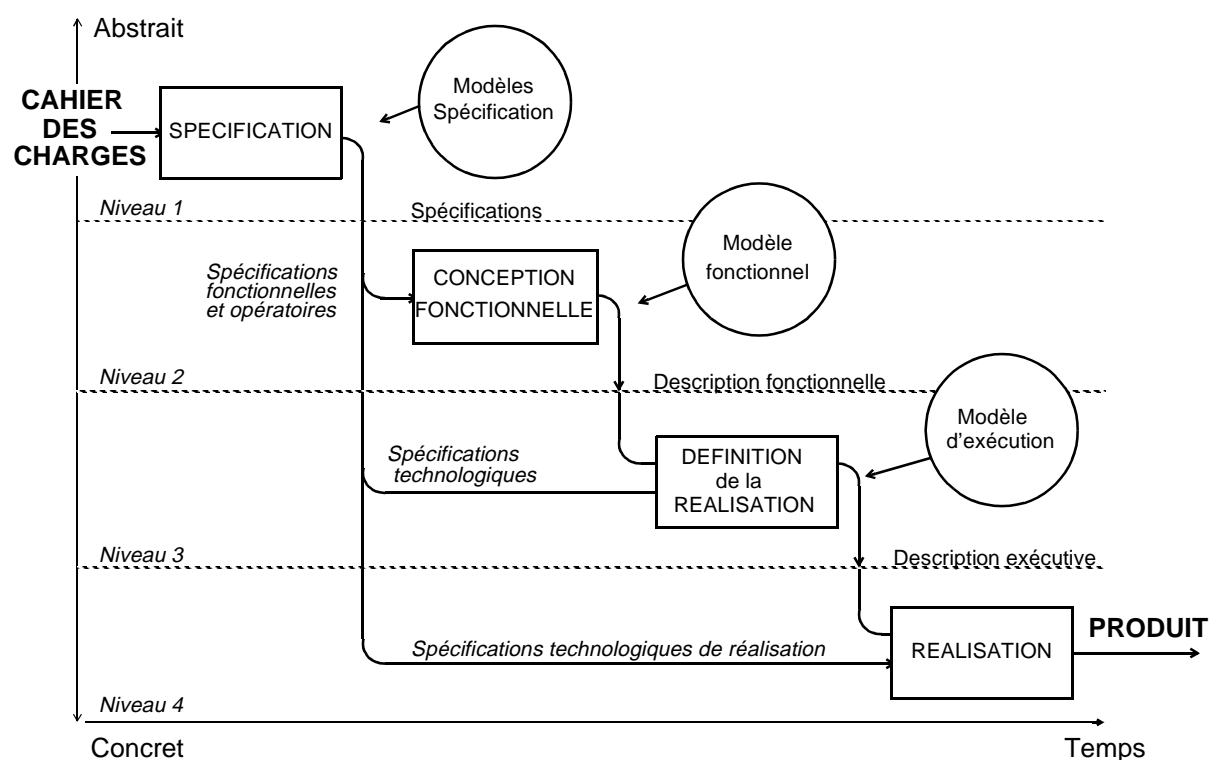
Cette décomposition en niveaux permet d'induire assez simplement la démarche à suivre. En partant du problème posé, chaque étape a pour objectif d'enrichir la description par ajout d'informations à partir des contraintes à satisfaire.

1.2. LA DEMARCHE

La démarche de conception basée sur l'emploi du modèle précédent exprime le processus de réflexion que doit suivre le concepteur pour aboutir à une description conforme au modèle, tout en répondant à des critères de qualité : robustesse, modularité, lisibilité, maintenabilité. Chaque niveau de description sert d'intermédiaire entre 2 étapes consécutives. Le développement s'effectue selon 4 étapes :

- L'**élaboration des spécifications**, de manière à exprimer à partir du besoin une vue purement externe du système (WHAT).
- La **conception fonctionnelle**, qui a pour objectif de trouver la description fonctionnelle, composée d'une structure fonctionnelle pour l'organisation interne de la solution et du comportement de toutes les fonctions nécessaires (HOW).
- La **définition de la réalisation** (aussi appelée conception détaillée), le but étant de trouver une structure d'exécution comme support matériel, ainsi qu'une implantation logicielle sur la structure matérielle retenue, en considérant toutes les contraintes technologiques: contraintes de répartition, contraintes de temps, contraintes électriques.
- La **réalisation** conduisant à un système opérationnel.

La figure suivante décrit l'enchaînement des 4 étapes.



-Figure 1.2- Enchaînement des étapes pour MCSE.

Pour chaque étape, le concepteur dispose en entrée : de la description d'un niveau intermédiaire comme résultat de l'étape précédente, de renseignements complémentaires que sont les contraintes imposées dans les spécifications. L'étape produit une description du niveau suivant conforme au modèle pour l'étape. Une vérification de conformité est possible à l'issue de chaque étape.

Dans les paragraphes suivants et sans entrer dans les détails, nous passons en revue les différentes étapes de la méthodologie en précisant les principes.

1.2.1. Elaboration des spécifications

Pour pouvoir concevoir, il faut tout d'abord disposer des spécifications. Par spécifications, il faut entendre une description complète mais purement externe du système à concevoir. Plus ces spécifications sont détaillées et conformes à des modèles formels, plus il est facile de déduire une solution. Mais cette spécification doit aussi être vérifiable, en particulier par le demandeur.

Le point de départ est le cahier des charges décrivant le besoin du demandeur. Pour décrire ce que doit faire un système, celui-ci est considéré comme observant et agissant sur les objets de son environnement. Il faut donc tout d'abord connaître cet environnement. Le connaître, c'est dans un premier temps modéliser les objets, et dans un second temps, expliciter les relations entre eux sous la forme d'une description fonctionnelle.

Ensuite, expliciter le rôle du système et donc exprimer ses spécifications consiste à énoncer et à caractériser les fonctions demandées. Ceci se fait en détaillant le comportement souhaité des objets de l'environnement sous le contrôle du système, ainsi que toutes les contraintes imposées.

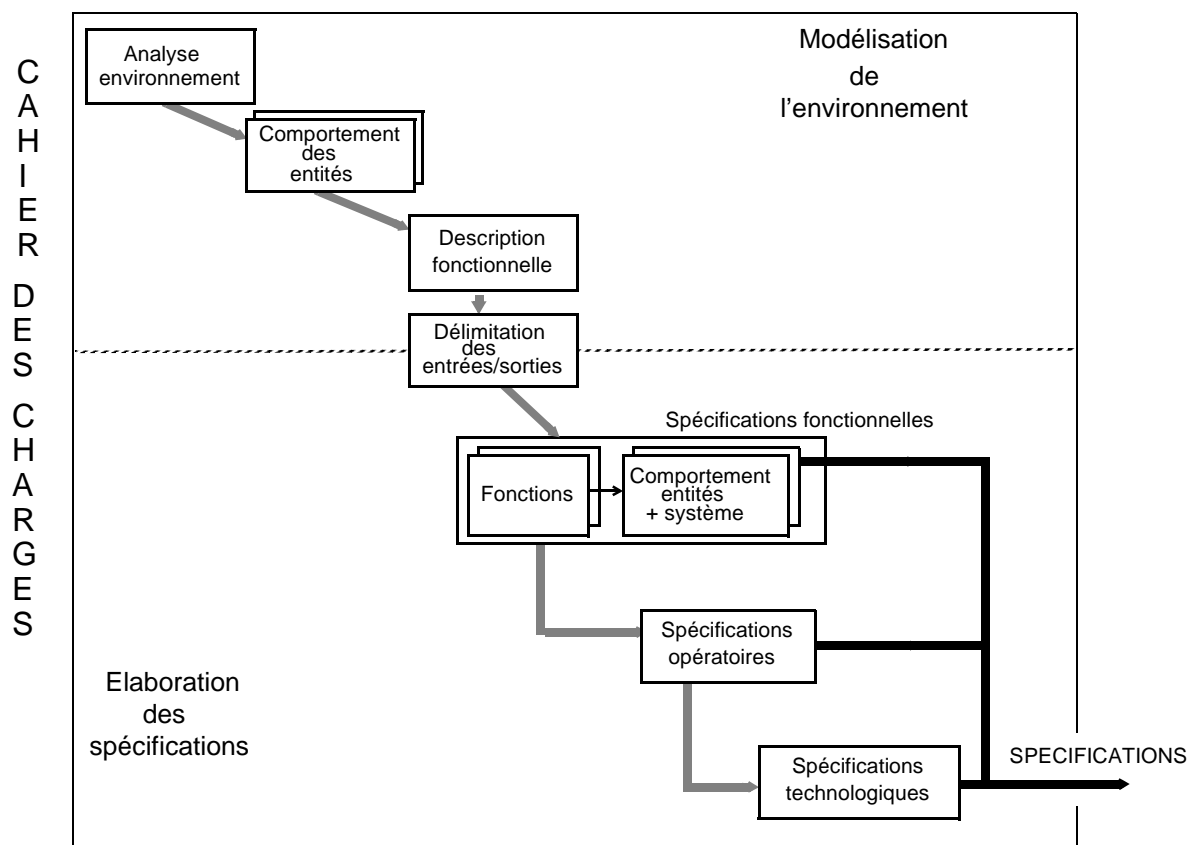
Par cette approche, la méthodologie fait apparaître une similitude de raisonnement entre la démarche pour obtenir les spécifications et celle pour concevoir. L'analyse de l'environnement conduit à une synthèse de la réalité sous la forme d'un modèle, et l'introduction des objectifs à atteindre conduit à un enrichissement de la modélisation précédente en considérant en supplément l'apport du système.

Cette étape permet d'obtenir 3 types de spécifications:

- les **spécifications fonctionnelles**: elles comprennent la liste des fonctions du système pour l'application (fonctions externes) et la description du comportement du système et de l'environnement pour ces fonctions.
- les **spécifications opératoires**, qui concernent le comportement, les performances, précisions, les méthodes à utiliser ...
- les **spécifications technologiques**, qui incluent : les contraintes de temps et de répartition, les caractéristiques des interfaces physiques, les contraintes de réalisation.

Les spécifications fonctionnelles et opératoires sont utilisées durant l'étape de conception fonctionnelle, tandis que les spécifications technologiques ne servent que pour les étapes de définition de la réalisation et de réalisation.

Les phases pour cette étape sont décrites par la figure ci-après.



-Figure 1.3- Déroulement pour l'étape de spécification.

1.2.2. Conception fonctionnelle

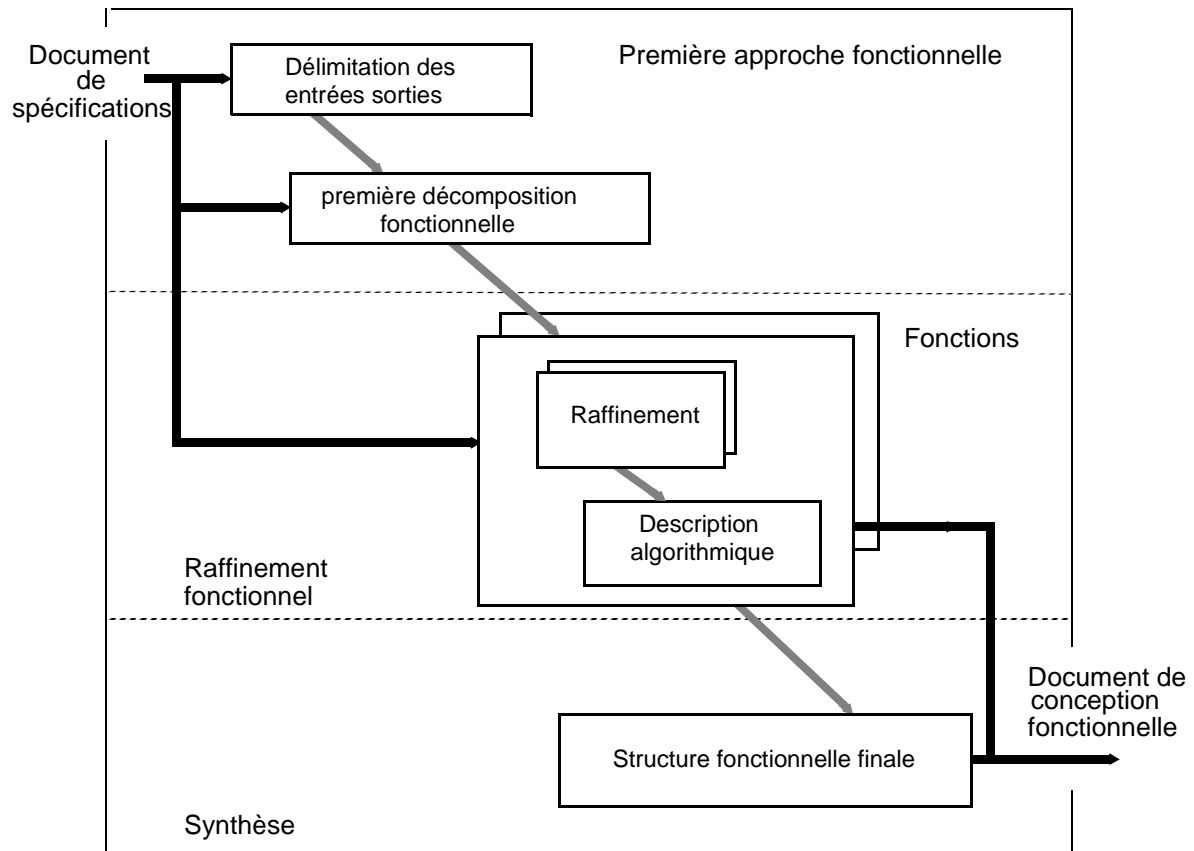
La solution pour cette étape se déduit d'une analyse des spécifications fonctionnelles. Elle doit être conforme au modèle fonctionnel.

La recherche de la structure fonctionnelle se fait tout d'abord à partir de la délimitation du système avec ses entrées et ses sorties. Il s'agit ensuite de trouver une première décomposition fonctionnelle. Cette première approche est importante car elle induit la qualité ou la non-qualité pour le reste du développement.

La démarche consiste ensuite à rechercher par raffinements successifs et pour chaque fonction à concevoir, les variables et événements internes caractéristiques nécessaires et si possible suffisants. Se déduisent alors les fonctions qui exploitent et assurent la mise à jour de ces variables ainsi que le comportement de chaque fonction. Le raffinement est poursuivi jusqu'à l'obtention de fonctions élémentaires qui peuvent s'exprimer par une description purement séquentielle.

L'expérience nous a montré que cette approche basée sur les données conduit à des structures fonctionnelles simples (réduction des couplages exprimant des relations d'ordre) et plus structurées que l'approche basée sur les fonctions, qui conduit à exprimer la structure comme décrivant un enchaînement de transformations.

La figure ci-dessous décrit la procédure à suivre.



-Figure 1.4- Les phases pour l'étape de Conception fonctionnelle.

1.2.3. Définition de la réalisation

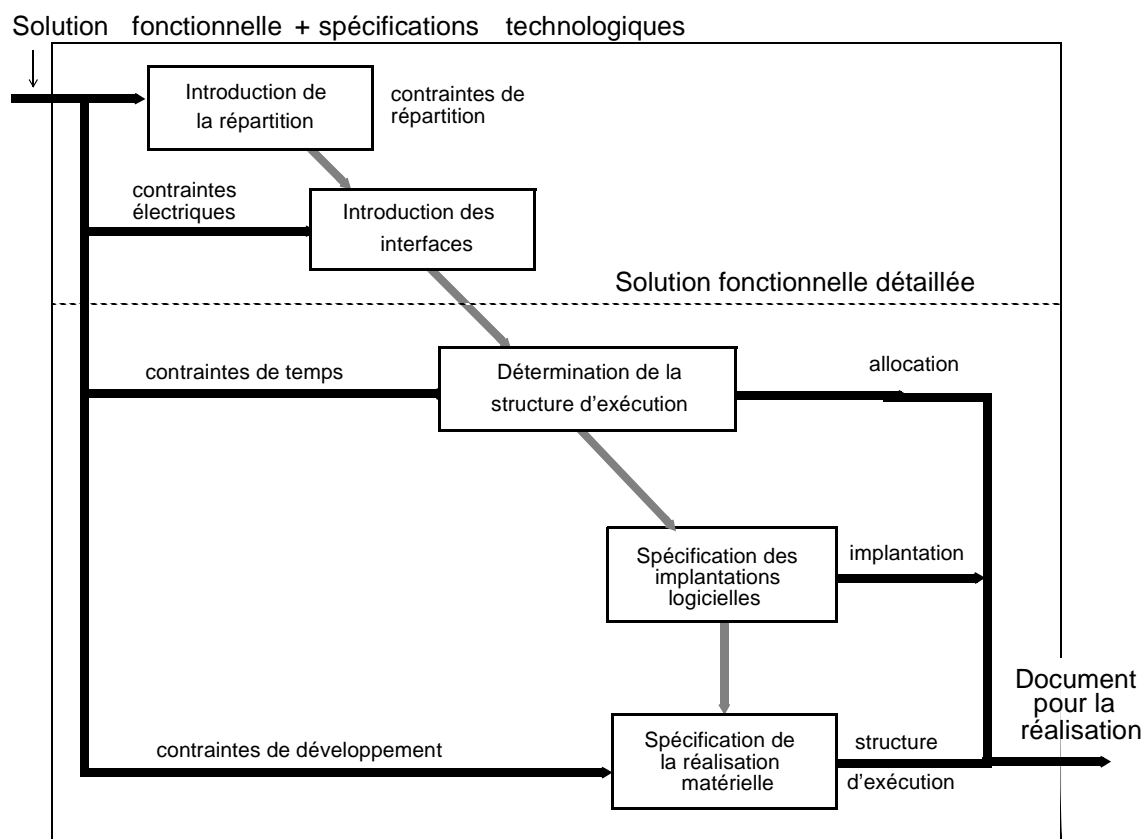
La troisième étape consiste à rechercher, d'une part le support exécutif, d'autre part la manière d'y implanter les fonctions réalisées par logiciel.

Tout d'abord, la description fonctionnelle doit être affinée, détaillée, enrichie pour tenir compte des contraintes technologiques que sont: la répartition géographique (si nécessaire), les interfaces physiques, les interfaces utilisateur.

Les contraintes de temps sont ensuite analysées pour déduire la répartition matériel/logiciel. La partie matérielle est spécifiée par une structure d'exécution. L'intégration ou allocation décrit complètement l'implantation de la description fonctionnelle sur la structure d'exécution.

Chaque sous-ensemble fonctionnel à réaliser par logiciel est décrit par un schéma d'implantation logicielle qui exprime la priorité de chaque tâche, et les relations de dépendance spatiale (par des données) ou temporelles. Cette implantation résulte de l'utilisation de règles qui permettent d'effectuer des transformations tenant compte du support matériel.

L'enchaînement des phases pour cette étape est représenté par la figure 1.5 ci-après.



-Figure 1.5- Les phases pour l'étape de définition de la réalisation.

1.2.4. Réalisation

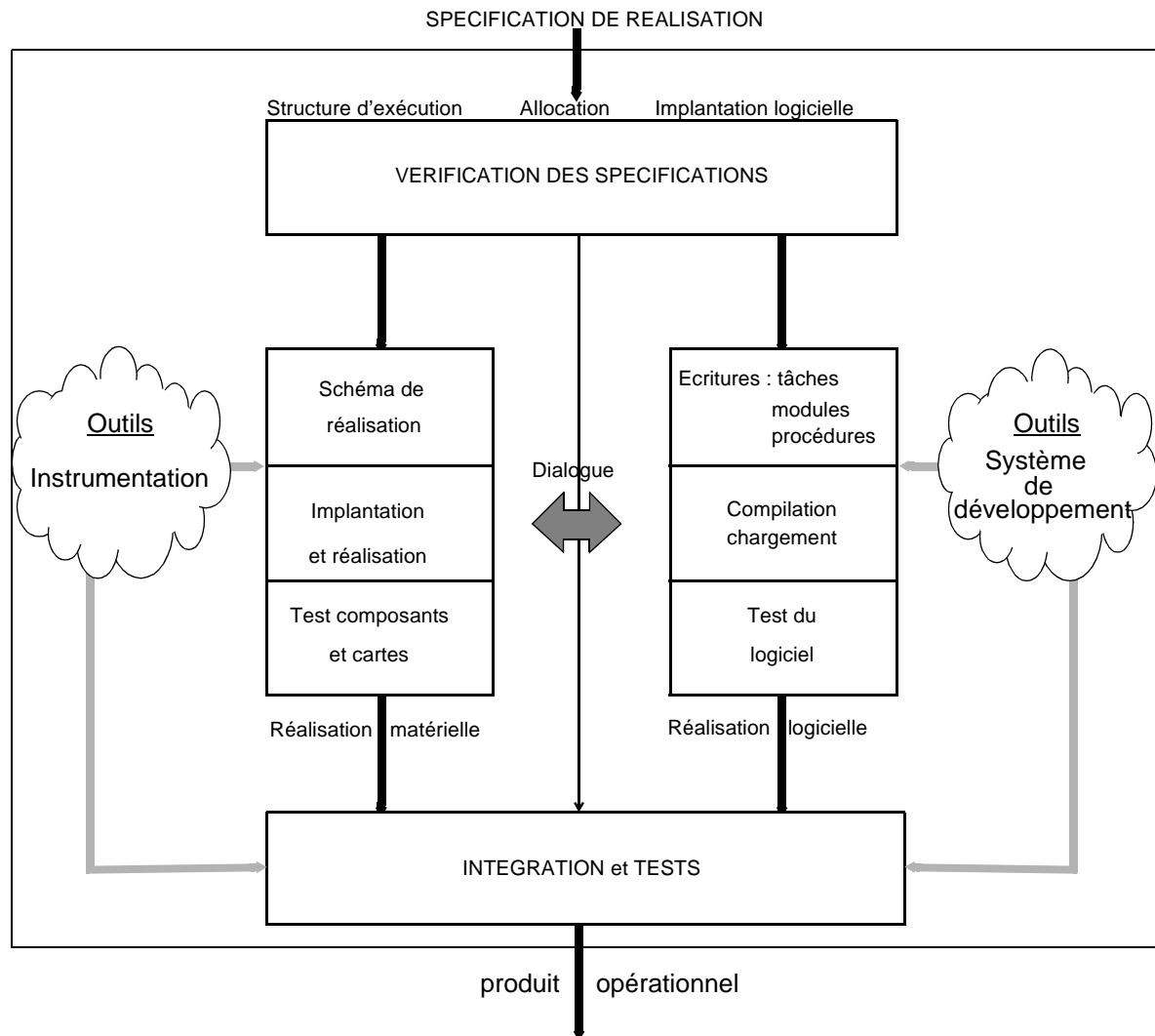
Les 2 parties - support matériel, implantation du logiciel - favorisent le travail de réalisation d'un prototype, l'intégration et le test.

Il faut être conscient à ce stade de la variété des stratégies de réalisation qui dépendent d'au moins 3 facteurs : les spécifications en entrée, les techniques à mettre en oeuvre, les outils et méthodes disponibles.

La réalisation est une démarche ascendante puisqu'elle consiste à assembler. Il s'agit de développer partie par partie la solution en faisant apparaître des fonctionnalités de plus en plus abstraites pour se rapprocher de l'objectif. Chaque niveau de la construction est validé par une vérification de la conformité aux spécifications du niveau correspondant de la démarche descendante. Réalisation matérielle et réalisation du logiciel peuvent se développer simultanément, ce qui permet de réduire le temps de la réalisation et de faire intervenir conjointement des spécialistes des 2 domaines.

Pour achever la réalisation, l'intégration et le test ont pour objectif de réunir toutes les parties des développements de manière à fournir un système opérationnel conforme aux souhaits du demandeur.

Cette démarche pour la réalisation est représentée ci-après.



-Figure 1.6- Démarche pour la réalisation.

1.3. CARACTERISTIQUES DE MCSE

MCSE est une démarche complète qui permet de passer du problème à une réalisation. Nous reprenons ici les aspects généraux essentiels de la méthodologie.

-A- UN MODELE DE DESCRIPTION COMME BASE

Toute méthodologie est basée sur un ou des modèles, ceci permet une décomposition en étapes. MCSE est basée sur un modèle de description interne en 3 composantes. Ce modèle incite à décrire tout système selon une hiérarchie de niveaux de description. Chaque niveau sert d'intermédiaire entre 2 étapes. La plupart des modèles sont graphiques, favorisant ainsi une compréhension globale et rapide.

-B- UNE DEMARCHE GLOBALEMENT DESCENDANTE POUR LA CONCEPTION

Chaque étape de la méthodologie permet de passer d'un niveau de description au niveau plus détaillé suivant en enrichissant la solution d'une composante supplémentaire.

La progression est donc globalement descendante puisqu'elle part du problème posé jusqu'à aboutir à une réalisation opérationnelle.

-C- UNE PROGRESSION ITERATIVE

Un développement ne peut pas se faire sans erreurs ou omissions. Des corrections sont toujours nécessaires. Basée sur la correction par retours-arrières, une phase de vérification en fin de chaque étape permet la détection des erreurs et induit un travail itératif avec des retours à l'intérieur de l'étape ou sur les étapes précédentes.

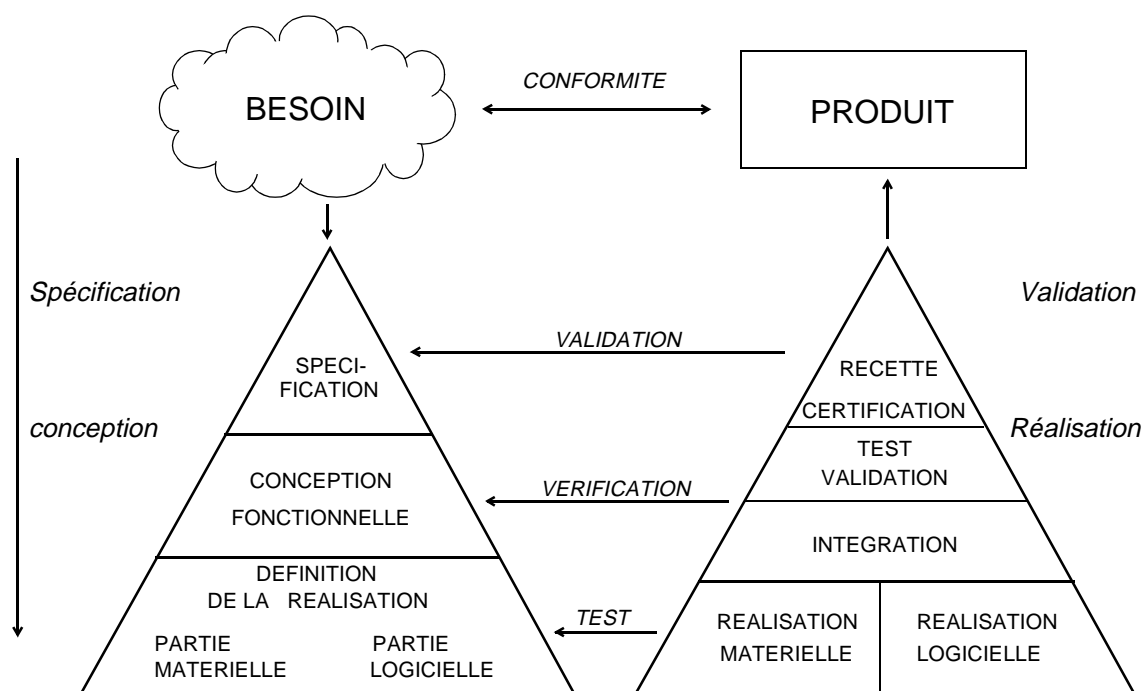
-D- UNE METHODE SPECIFIQUE POUR CHAQUE ETAPE

Le modèle de description de la solution à l'issue de chaque étape n'est pas suffisant (le QUOI). Le concepteur doit disposer pour chaque étape d'un guide précis lui expliquant COMMENT passer de la spécification en entrée à une solution possédant des qualités. Ce guide est la méthode à suivre : technique d'analyse, séquence des décisions, critères de choix. Par opposition à une recherche intuitive, l'emploi d'une méthode garantit l'obtention rapide d'une solution a priori de qualité.

Au-delà de l'aspect méthode, l'idée des modèles génériques de solutions a un intérêt certain. De tels modèles ayant la particularité pour la décomposition fonctionnelle d'être générateurs de multiples solutions, sont retenus car possédant des qualités intrinsèques: lisibilité, maintenabilité, simplicité, adéquation au modèle global MCSE. La connaissance de tels modèles, en complément des méthodes, améliore notablement le pourcentage des développements corrects.

-E- UNE DEMARCHE GLOBALEMENT ASCENDANTE POUR LA REALISATION

L'assemblage n'est possible qu'après disponibilité des constituants. Ainsi, la réalisation débute par la réalisation des plus petits sous-ensembles, puis remonte progressivement par assemblage et intégration de fonctions plus globales. Le travail de réalisation est représentable par un triangle juxtaposé à celui de la conception comme l'indique la figure suivante. La largeur du triangle pour chaque stade indique la quantité d'informations à maîtriser.



-Figure 1.7- Forme en double triangle pour le développement.

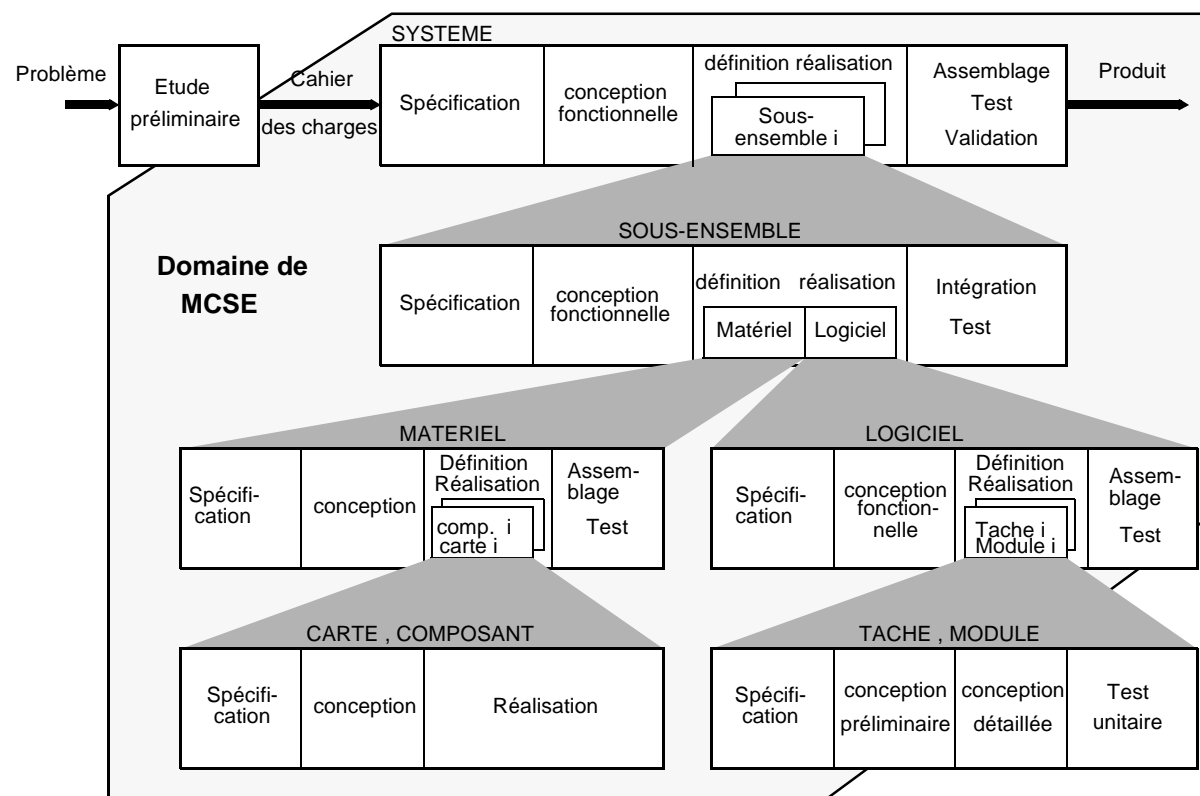
Chaque niveau de la réalisation est vérifiable, ce qui assure sa conformité au niveau correspondant de la conception. La figure 1.7 illustre la correspondance entre les 2 démarches complémentaires.

-F- UN MODELE DE CYCLE DE DEVELOPPEMENT HIERARCHIQUE

Pour un système relativement complexe, le modèle ci-dessus en double triangle n'est pas suffisamment précis.

L'étape de définition de la réalisation conduit à mettre en évidence les 2 parties : matériel, logiciel. Chaque partie est à nouveau à développer selon une démarche en 3 phases : spécification, conception, définition de la réalisation, et ceci jusqu'à la mise en évidence des constituants disponibles (composants matériels ou logiciels).

Ainsi comme le montre la figure ci-après, le modèle de cycle de développement est un emboîtement de développements. Au premier niveau, la conception est générale et concerne l'application dans son ensemble. Au fur et à mesure du raffinement de la solution, les développements concernent des problèmes plus spécifiques en rapport avec la réalisation : développement d'un composant, d'une fonction spécifique, d'un module logiciel ...



-Figure 1.8- Modèle hiérarchique pour le développement.

-G- GUIDE POUR LA DOCUMENTATION

Le modèle de description induit directement la structure des documents à produire durant le développement. Chaque document est le résultat d'une étape, et sert comme informations pour l'étape suivante.

Ainsi, en respectant la démarche par étapes successives, la documentation est générée au fur et à mesure du développement et non en final. Elle possède alors une réelle qualité quant à la forme et au fond puisqu'elle relate, en plus de la solution, la démarche suivie et l'argumentation qui justifie les décisions importantes.

Produite de cette manière, la documentation est utilisable durant le cycle de développement: pour les phases de vérification selon un cycle auteur-lecteurs, pour l'observation de l'état d'avancement, mais aussi pour les étapes ultérieures et en particulier pour la maintenance.

-H- GUIDE POUR LA CONDUITE DE PROJET

Le modèle de cycle de développement est utilisable pour la mise en place d'une procédure de conduite d'un ensemble de projets. Pour chaque projet, cette activité concerne :

- le management : planification, organisation, direction, contrôle et suivi du projet,
- l'obtention de la conformité : planification des tests, nature des tests techniques à utiliser, les résultats, conformité et certification,
- la gestion de la documentation : spécification des documents, planification des revues, méthodes de gestion, mises à jour ...
- la gestion de la maintenance : domaine et procédures de maintenance, solutions et outils, planification,
- la gestion de la qualité : assurance qualité, méthode pour l'obtention de la qualité, procédures de contrôle.

-I- UNE METHODOLOGIE OUVERTE ET COMPLEMENTAIRE

MCSE ne se trouve pas restreinte à une méthode particulière et à un domaine bien spécifique. Au contraire, pour chaque étape, plusieurs méthodes sont utilisables et c'est au concepteur de choisir selon des critères, celle qui lui permet de résoudre au mieux son problème.

Développée au départ pour les systèmes de contrôle/commande temps-réel à microprocesseurs, l'expérience nous a montré son adéquation pour une large classe d'application et de techniques, et tout particulièrement pour les applications qui utilisent l'électronique et l'informatique. MCSE n'est pas à opposer aux autres méthodologies, bien au contraire, elle se veut complémentaire. La plupart des modèles proposés par différents auteurs s'avèrent utilisables. Par exemple, SADT et les méthodes de spécification de WARD et MELLOR et de HATLEY favorisent la tâche d'analyse du problème. La méthodologie de JACKSON est dans l'esprit relativement proche de MCSE. Les méthodologies DARTS et SDWMC de BUHR ont aussi des points communs.

1.4. CONSEILS POUR UNE UTILISATION RATIONNELLE DE MCSE

Nous rappelons ici les principes essentiels de la Méthodologie qu'il est conseillé de suivre pour obtenir les meilleurs résultats. Ces principes répondent à des difficultés couramment rencontrées par les concepteurs.

Au préalable, il est bon de rappeler que le développement est une activité humaine mettant en jeu une méthodologie et un système social. Les 2 parties ont une importance pour la réussite. Aussi, l'attitude volontariste des concepteurs est une première condition strictement nécessaire pour le succès de la démarche.

Ensuite, un bon concepteur doit savoir se placer au dessus des aspects techniques et ceci en utilisant le plus possible des concepts. Il ne doit pas se laisser guider par une solution a priori; dans ce cas probablement, il aboutira à la conclusion qu'elle est correcte sans avoir évalué d'autres alternatives. Il est aussi essentiel de chercher à satisfaire la demande et non pas se satisfaire de son développement.

Nous citons ci-dessous pour chaque étape, quelques-unes des difficultés que tout utilisateur de la méthodologie risque de rencontrer et le point de vue à adopter.

-A- ETAPE DE SPECIFICATION

Concernant l'étape de spécification, il y a 2 difficultés. Tout d'abord pour la modélisation de l'environnement, il faut décider du modèle le plus approprié pour cet environnement, à la fois sa nature et son niveau de détail. Pour être bonne, l'analyse doit se faire sur un plan fonctionnel et non matériel. Il faut donc penser aux entités fonctionnelles. Ceci est une difficulté qui conduit à des erreurs fréquentes. De l'analyse se déduit la nature du modèle qu'il faut rechercher dans l'ordre de complexité: modèle statique données/événements, modèle dynamique global, modèle des activités. La modélisation est à faire selon un niveau de détails nécessaire mais juste suffisant pour résoudre le problème. Elle ne doit pas faire intervenir le système à concevoir.

Ensuite, pour l'élaboration des spécifications fonctionnelles, celles-ci doivent se limiter à une description externe du système. Les fonctions du système ne doivent être que des fonctions de l'application, qu'il ne faut surtout pas confondre avec les fonctions internes du système.

S'assurer de la complétude d'une spécification est aussi une réelle difficulté. Celle-ci doit apparaître cohérente, claire et doit être vérifiée par le demandeur.

-B- ETAPE DE CONCEPTION

Pour cette étape, le premier point important est de faire une conception indépendante de la technologie. La première difficulté est donc de faire abstraction des contraintes technologiques et donc de toute idée de réalisation. Pour cela, il faut éliminer tous les problèmes de répartition géographique et d'interfaces pour se consacrer au seul aspect fonctionnel. Il faut donc déjà partir des entrées et sorties fonctionnelles et non pas physiques.

Ensuite, la recherche d'une solution est à faire sur la base des variables internes nécessaires et non pas sur la base des fonctions internes, alors que la tendance naturelle consiste à appréhender le problème par les fonctions. L'approche préconisée conduit à une solution plus simple à comprendre et plus simple à implanter. Les idées de solutions peuvent aussi se déduire de modèles génériques de solution. L'intérêt est de disposer d'un guide pour la recherche de variables.

D'une manière générale, le concepteur doit avoir des idées pour être créatif mais doit faire abstraction de son idée a priori, sinon il conduira son développement sur la base de cette idée inhibant alors l'évaluation de toute autre solution peut-être meilleure.

C- ETAPE DE DEFINITION DE LA REALISATION

Une des tendances naturelles consiste à entreprendre la réalisation rapidement: développement ou choix du matériel très tôt et écriture du logiciel. L'étape de définition de la réalisation est une étape de réflexion importante qui conduit à réduire l'effort de développement pour en particulier la réalisation. Cette étape permet de s'adapter aux

caractéristiques de l'environnement, en introduisant les interfaces. Pour cela, il faut rechercher la solution la plus appropriée, c'est particulièrement le cas pour l'interface homme-machine. D'autre part, les contraintes de temps doivent être évaluées correctement pour le cas le plus défavorable de manière à pouvoir prouver a priori le bon fonctionnement de l'application.

De cette analyse découle ensuite la recherche de la répartition matériel/logiciel qui a pour objectif de réduire au maximum le développement. La démarche conduit à minimiser le matériel, puis le logiciel. Concernant le logiciel, il est composé de 2 parties: la partie opératoire strictement nécessaire pour le fonctionnement et donc relativement incompressible si la spécification comportementale a été bien élaborée, et la partie dite organisationnelle qui concerne les relations entre les fonctions. Cette deuxième partie est compressible, et l'objectif des transformations pour obtenir le schéma d'implantation logicielle est de réduire au maximum cette partie. La Méthodologie favorise cette réduction. Cette approche est tout à fait justifiée pour les applications temps-réel dédiées où le développement est assuré une fois pour toutes.

1.5. SUIVI D'UN PROJET

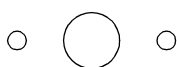
Développer un projet nécessite de mener en parallèle 3 activités:

- le développement à proprement dit en procédant étape par étape,
- la vérification du développement,
- la documentation du projet.

La fiche jointe sur la page suivante a été établie comme guide simplifiée utilisable dans le cas d'un projet qui peut se développer d'une manière linéaire. Pour des projets plus conséquents, une telle fiche peut aussi servir mais en limitant son rôle à chaque sous-ensemble à développer.

Les concepteurs et les responsables de projets sont invités à utiliser une telle fiche ou une variante de celle-ci, de manière à disposer d'une observation sur l'avancement du travail, ceci pour mieux répondre aux objectifs de coût, de délais et de qualité.

Cette présentation succincte, loin d'être suffisante, n'est donnée ici qu'à titre de rappel et de guide pour la méthodologie MCSE. Pour des informations plus précises sur la démarche et pour mieux comprendre les études de cas décrites dans les chapitres suivants, le lecteur est invité à lire l'ouvrage: "Spécification et Conception des systèmes, une méthodologie MCSE" de l'auteur.



1

PRESENTATION GENERALE DE LA METHODOLOGIE

Cette première partie rappelle les éléments caractéristiques et la démarche de la méthodologie MCSE (Méthodologie pour la Conception des Systèmes Electroniques) décrits dans l'ouvrage "Spécification et conception des systèmes, une Méthodologie MCSE" de l'auteur [CALVEZ-90]. Ce schéma d'organisation des développements est illustré par toutes les études de cas qui suivent.

La méthodologie MCSE couvre les phases de Spécification, de Conception Fonctionnelle aussi appelée par ailleurs Conception Préliminaire, de définition de la Réalisation ou Conception Détaillée.

Développée spécialement pour les applications temps-réel en contrôle/commande, elle a été appliquée sur de multiples problèmes industriels. L'adéquation du modèle à la problématique traitée a ainsi été validée ainsi que l'intérêt de toutes les étapes.

Si cette méthodologie concerne plus particulièrement le domaine des systèmes électroniques temps-réel, il se trouve que la démarche suivie pour les premières phases du développement - spécification, conception architecturale - se trouve indépendante de la réalisation, puisqu'il s'agit d'une approche SYSTEME. Des expériences intéressantes ont été entreprises quant à son utilisation pour des applications variées : systèmes de contrôle/commande, réseaux et protocoles, systèmes répartis, outils interactifs, conception de composants intégrés.

Une large expérimentation de la méthodologie a montré l'importance du modèle, puis l'importance des règles et conseils qui engendrent chez les concepteurs des solutions de qualité. Nous avons aussi constaté que les solutions de qualité peuvent s'exprimer sous la forme de modèles génériques. Ainsi, la connaissance de tels modèles génériques de solutions pour diverses classes de problèmes facilite la tâche du concepteur et favorise la production de solutions de qualité au sens : lisibilité, maintenabilité, efficacité ... et donc coût.

En aval de la conception, la phase de définition de la réalisation propose une démarche systématique qui conduit, à partir d'une solution fonctionnelle, à choisir la technique de réalisation la plus appropriée pour le problème (type de technologie, répartition matériel/logiciel, respect des contraintes de temps...).

1.1. DESCRIPTION D'UN SYSTEME

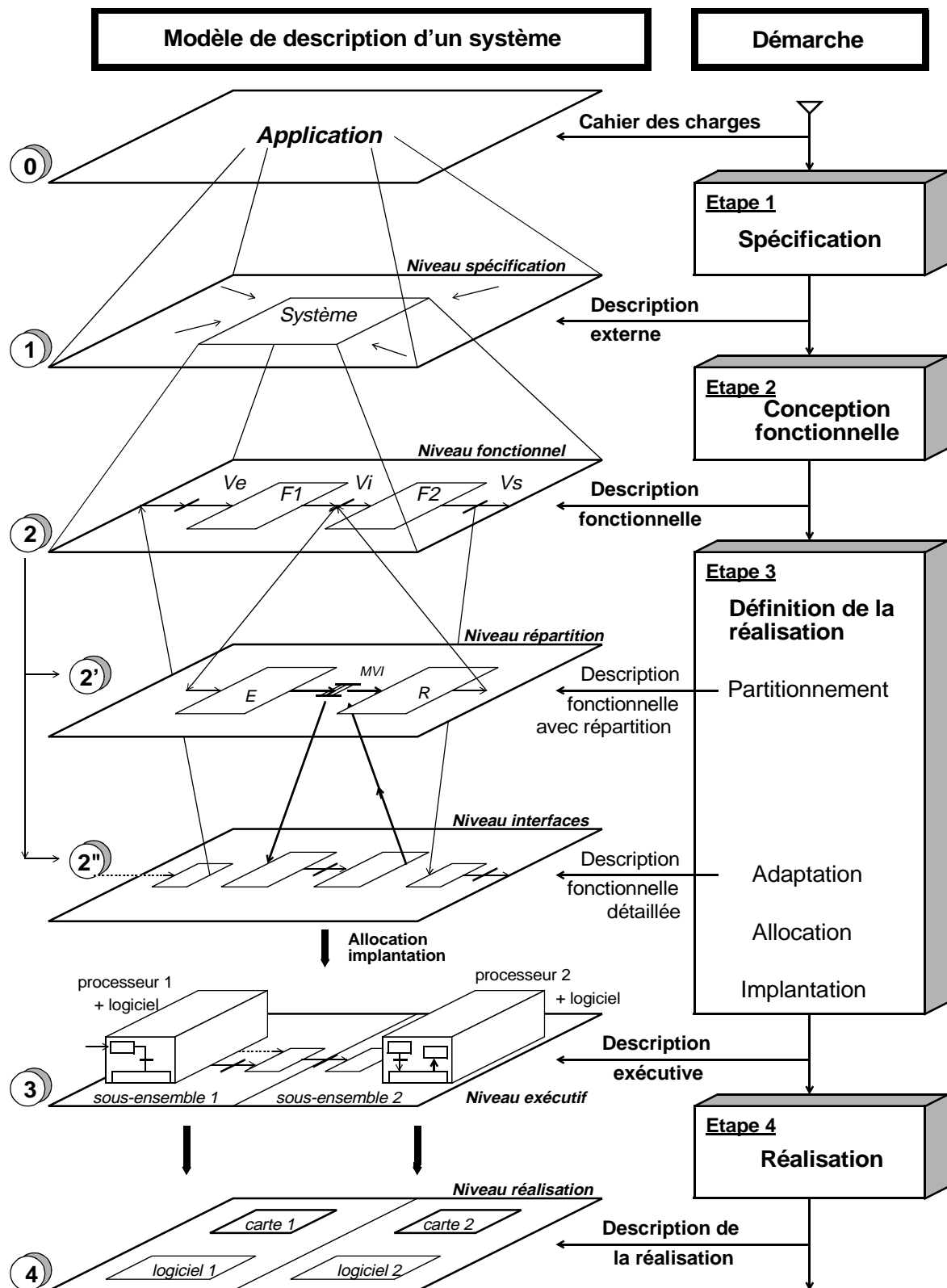
Une méthodologie est un guide pour le développement d'applications. MCSE préconise une démarche globalement descendante qui conduit à rechercher par approches successives, une réalisation appropriée.

La décomposition en étapes est basée sur la possibilité de décrire un système selon plusieurs niveaux d'abstraction. Ainsi une méthodologie se comprend bien lorsque le modèle de description des systèmes à concevoir est explicite.

Pour résumer la démarche préconisée, la structuration d'un système pour MCSE est décrite selon un ensemble de niveaux comme l'indique la figure ci-après.

Si on part du niveau le plus abstrait pour descendre jusqu'au niveau de la réalisation, on trouve:

- le niveau **Cahier des charges**, qui correspond à l'expression du besoin. Pour ce niveau, le système n'est pas encore défini. Les informations concernent l'application dans son ensemble avec les objectifs souhaités.
- le niveau **Spécification**. Il correspond à la description du système à concevoir selon une vue purement externe. Les spécifications incluent toutes les contraintes auxquelles doit satisfaire le système.
- le niveau **Description Fonctionnelle**, exprimant la structure du système sur le plan fonctionnel. Cette description interne a la particularité d'être indépendante de toutes les contraintes technologiques.
- le niveau **Description exécutive**, qui explicite les spécifications complètes de la réalisation en décrivant la partie matérielle par une structure d'exécution et la partie logicielle par des schémas d'implantation logicielle. Entre ce niveau et le précédent 2 sous-niveaux sont intéressants comme intermédiaires:
 - le niveau fonctionnel avec les **interfaces** qui découle du sous-niveau précédent avec ajout de toutes les interfaces physiques nécessaires pour le couplage avec l'environnement réel du système.
 - le niveau fonctionnel avec **répartition** qui résulte de la description fonctionnelle en y ajoutant la contrainte de répartition géographique, ce qui se traduit par l'utilisation de mécanismes pour le transport des informations entre les sous-ensembles répartis.
- le niveau **Réalisation** qui décrit la solution finale comme un ensemble de cartes et de logiciels.



-Figure 1.1- Niveaux de description d'un système.

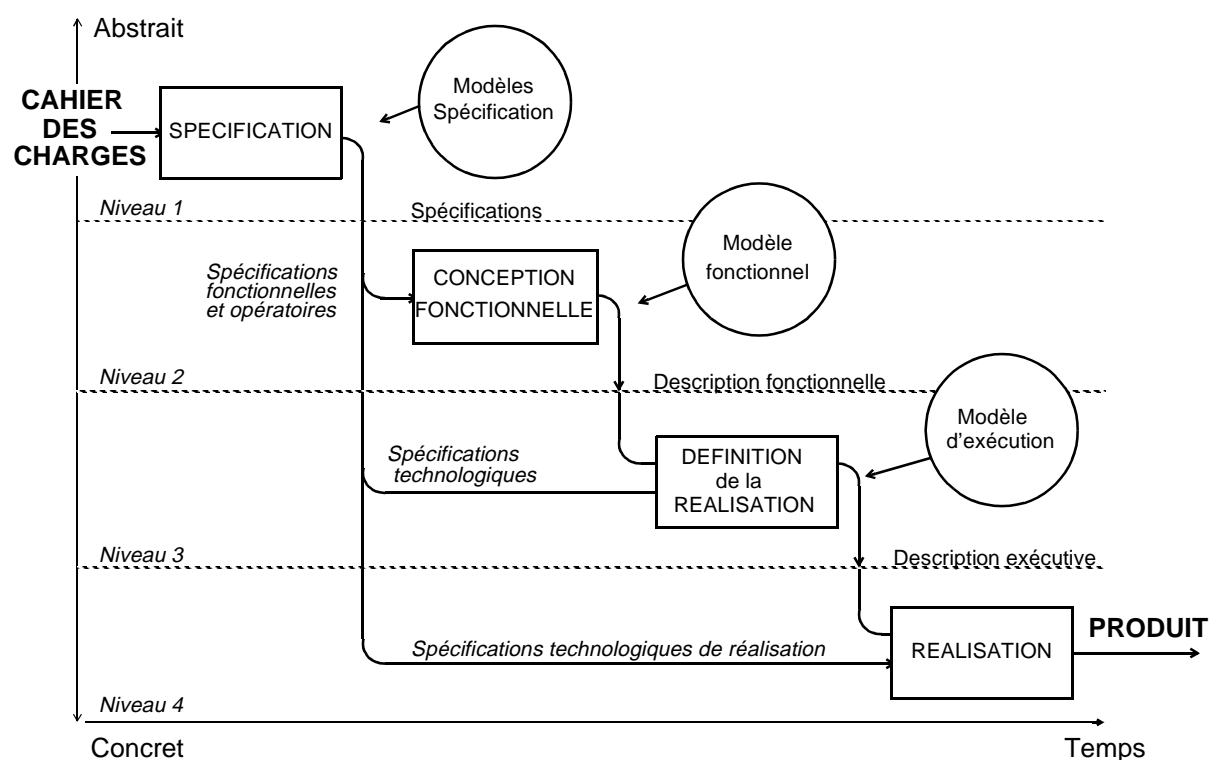
Cette décomposition en niveaux permet d'induire assez simplement la démarche à suivre. En partant du problème posé, chaque étape a pour objectif d'enrichir la description par ajout d'informations à partir des contraintes à satisfaire.

1.2. LA DEMARCHE

La démarche de conception basée sur l'emploi du modèle précédent exprime le processus de réflexion que doit suivre le concepteur pour aboutir à une description conforme au modèle, tout en répondant à des critères de qualité : robustesse, modularité, lisibilité, maintenabilité. Chaque niveau de description sert d'intermédiaire entre 2 étapes consécutives. Le développement s'effectue selon 4 étapes :

- L'**élaboration des spécifications**, de manière à exprimer à partir du besoin une vue purement externe du système (WHAT).
- La **conception fonctionnelle**, qui a pour objectif de trouver la description fonctionnelle, composée d'une structure fonctionnelle pour l'organisation interne de la solution et du comportement de toutes les fonctions nécessaires (HOW).
- La **définition de la réalisation** (aussi appelée conception détaillée), le but étant de trouver une structure d'exécution comme support matériel, ainsi qu'une implantation logicielle sur la structure matérielle retenue, en considérant toutes les contraintes technologiques: contraintes de répartition, contraintes de temps, contraintes électriques.
- La **réalisation** conduisant à un système opérationnel.

La figure suivante décrit l'enchaînement des 4 étapes.



-Figure 1.2- Enchaînement des étapes pour MCSE.

Pour chaque étape, le concepteur dispose en entrée : de la description d'un niveau intermédiaire comme résultat de l'étape précédente, de renseignements complémentaires que sont les contraintes imposées dans les spécifications. L'étape produit une description du niveau suivant conforme au modèle pour l'étape. Une vérification de conformité est possible à l'issue de chaque étape.

Dans les paragraphes suivants et sans entrer dans les détails, nous passons en revue les différentes étapes de la méthodologie en précisant les principes.

1.2.1. Elaboration des spécifications

Pour pouvoir concevoir, il faut tout d'abord disposer des spécifications. Par spécifications, il faut entendre une description complète mais purement externe du système à concevoir. Plus ces spécifications sont détaillées et conformes à des modèles formels, plus il est facile de déduire une solution. Mais cette spécification doit aussi être vérifiable, en particulier par le demandeur.

Le point de départ est le cahier des charges décrivant le besoin du demandeur. Pour décrire ce que doit faire un système, celui-ci est considéré comme observant et agissant sur les objets de son environnement. Il faut donc tout d'abord connaître cet environnement. Le connaître, c'est dans un premier temps modéliser les objets, et dans un second temps, expliciter les relations entre eux sous la forme d'une description fonctionnelle.

Ensuite, expliciter le rôle du système et donc exprimer ses spécifications consiste à énoncer et à caractériser les fonctions demandées. Ceci se fait en détaillant le comportement souhaité des objets de l'environnement sous le contrôle du système, ainsi que toutes les contraintes imposées.

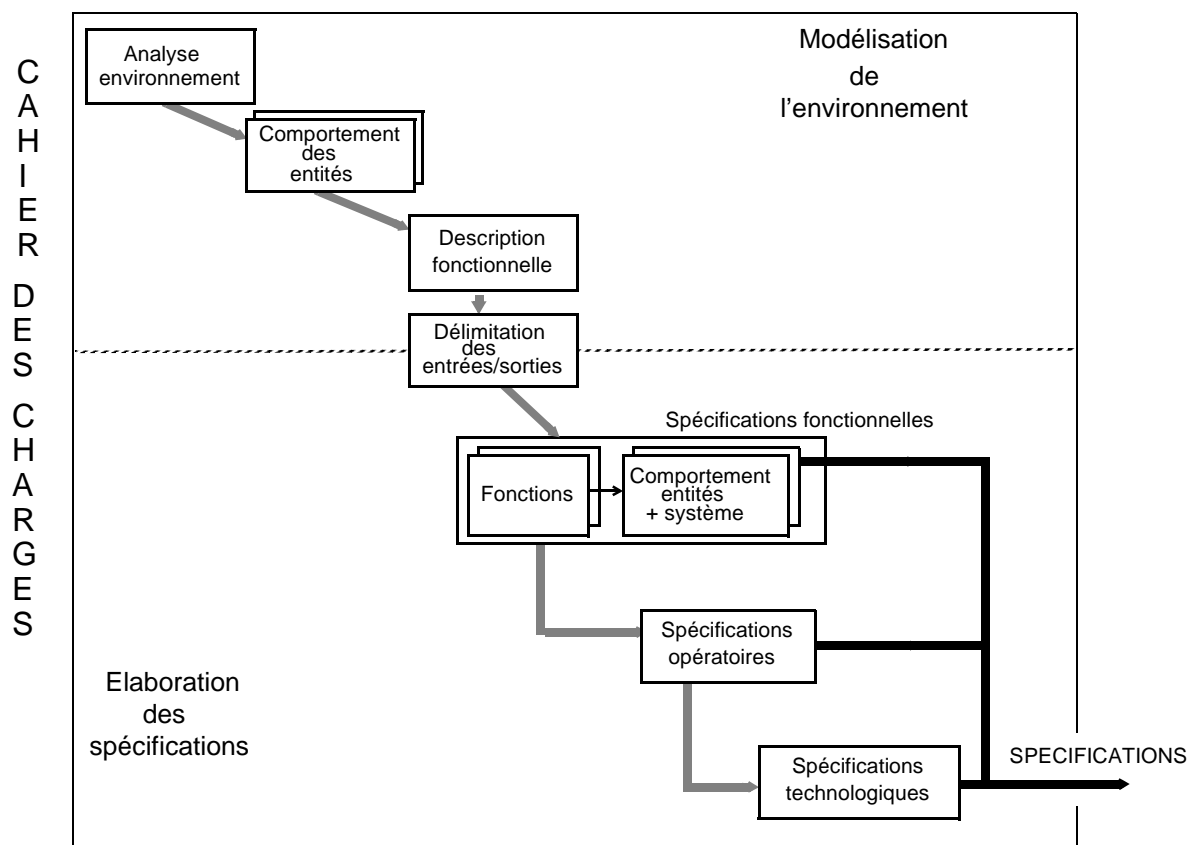
Par cette approche, la méthodologie fait apparaître une similitude de raisonnement entre la démarche pour obtenir les spécifications et celle pour concevoir. L'analyse de l'environnement conduit à une synthèse de la réalité sous la forme d'un modèle, et l'introduction des objectifs à atteindre conduit à un enrichissement de la modélisation précédente en considérant en supplément l'apport du système.

Cette étape permet d'obtenir 3 types de spécifications:

- les **spécifications fonctionnelles**: elles comprennent la liste des fonctions du système pour l'application (fonctions externes) et la description du comportement du système et de l'environnement pour ces fonctions.
- les **spécifications opératoires**, qui concernent le comportement, les performances, précisions, les méthodes à utiliser ...
- les **spécifications technologiques**, qui incluent : les contraintes de temps et de répartition, les caractéristiques des interfaces physiques, les contraintes de réalisation.

Les spécifications fonctionnelles et opératoires sont utilisées durant l'étape de conception fonctionnelle, tandis que les spécifications technologiques ne servent que pour les étapes de définition de la réalisation et de réalisation.

Les phases pour cette étape sont décrites par la figure ci-après.



-Figure 1.3- Déroulement pour l'étape de spécification.

1.2.2. Conception fonctionnelle

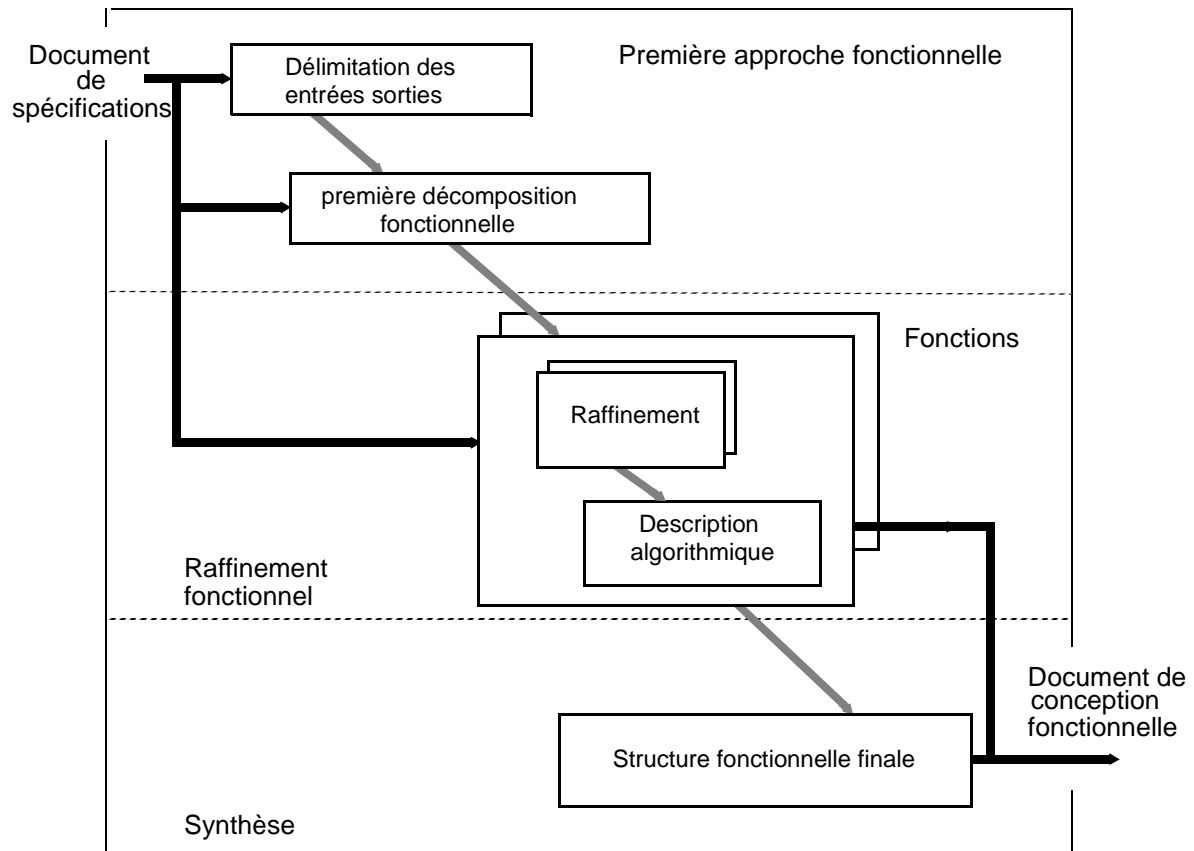
La solution pour cette étape se déduit d'une analyse des spécifications fonctionnelles. Elle doit être conforme au modèle fonctionnel.

La recherche de la structure fonctionnelle se fait tout d'abord à partir de la délimitation du système avec ses entrées et ses sorties. Il s'agit ensuite de trouver une première décomposition fonctionnelle. Cette première approche est importante car elle induit la qualité ou la non-qualité pour le reste du développement.

La démarche consiste ensuite à rechercher par raffinements successifs et pour chaque fonction à concevoir, les variables et événements internes caractéristiques nécessaires et si possible suffisants. Se déduisent alors les fonctions qui exploitent et assurent la mise à jour de ces variables ainsi que le comportement de chaque fonction. Le raffinement est poursuivi jusqu'à l'obtention de fonctions élémentaires qui peuvent s'exprimer par une description purement séquentielle.

L'expérience nous a montré que cette approche basée sur les données conduit à des structures fonctionnelles simples (réduction des couplages exprimant des relations d'ordre) et plus structurées que l'approche basée sur les fonctions, qui conduit à exprimer la structure comme décrivant un enchaînement de transformations.

La figure ci-dessous décrit la procédure à suivre.



-Figure 1.4- Les phases pour l'étape de Conception fonctionnelle.

1.2.3. Définition de la réalisation

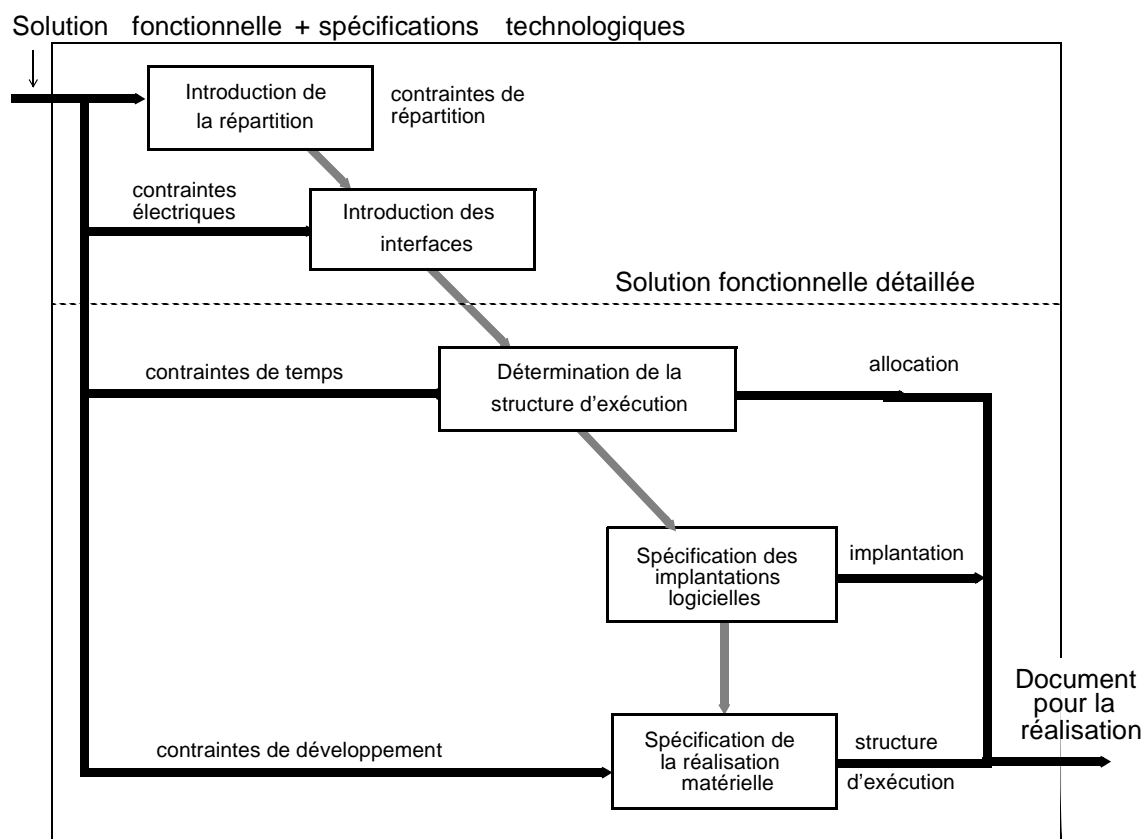
La troisième étape consiste à rechercher, d'une part le support exécutif, d'autre part la manière d'y implanter les fonctions réalisées par logiciel.

Tout d'abord, la description fonctionnelle doit être affinée, détaillée, enrichie pour tenir compte des contraintes technologiques que sont: la répartition géographique (si nécessaire), les interfaces physiques, les interfaces utilisateur.

Les contraintes de temps sont ensuite analysées pour déduire la répartition matériel/logiciel. La partie matérielle est spécifiée par une structure d'exécution. L'intégration ou allocation décrit complètement l'implantation de la description fonctionnelle sur la structure d'exécution.

Chaque sous-ensemble fonctionnel à réaliser par logiciel est décrit par un schéma d'implantation logicielle qui exprime la priorité de chaque tâche, et les relations de dépendance spatiale (par des données) ou temporelles. Cette implantation résulte de l'utilisation de règles qui permettent d'effectuer des transformations tenant compte du support matériel.

L'enchaînement des phases pour cette étape est représenté par la figure 1.5 ci-après.



-Figure 1.5- Les phases pour l'étape de définition de la réalisation.

1.2.4. Réalisation

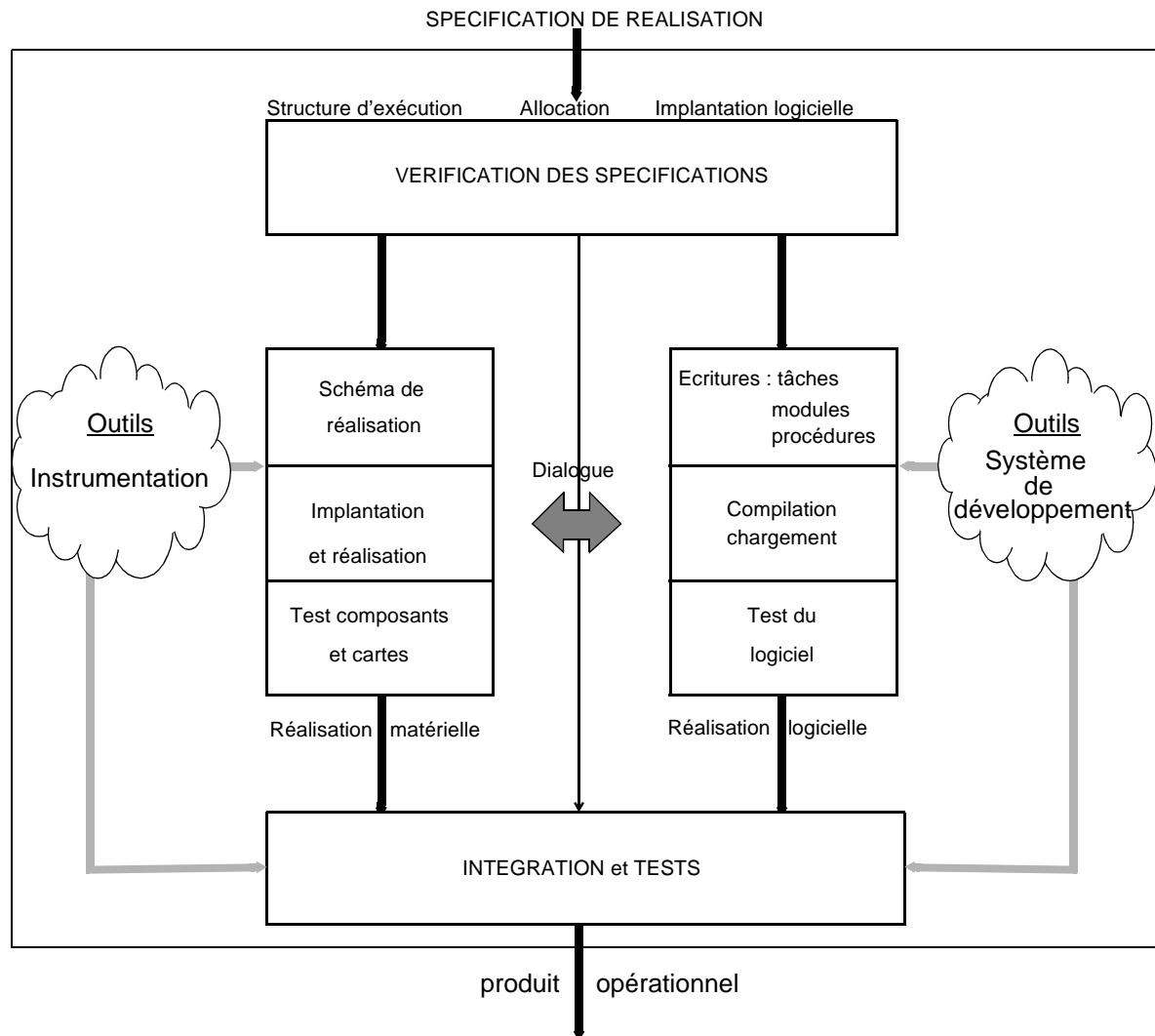
Les 2 parties - support matériel, implantation du logiciel - favorisent le travail de réalisation d'un prototype, l'intégration et le test.

Il faut être conscient à ce stade de la variété des stratégies de réalisation qui dépendent d'au moins 3 facteurs : les spécifications en entrée, les techniques à mettre en oeuvre, les outils et méthodes disponibles.

La réalisation est une démarche ascendante puisqu'elle consiste à assembler. Il s'agit de développer partie par partie la solution en faisant apparaître des fonctionnalités de plus en plus abstraites pour se rapprocher de l'objectif. Chaque niveau de la construction est validé par une vérification de la conformité aux spécifications du niveau correspondant de la démarche descendante. Réalisation matérielle et réalisation du logiciel peuvent se développer simultanément, ce qui permet de réduire le temps de la réalisation et de faire intervenir conjointement des spécialistes des 2 domaines.

Pour achever la réalisation, l'intégration et le test ont pour objectif de réunir toutes les parties des développements de manière à fournir un système opérationnel conforme aux souhaits du demandeur.

Cette démarche pour la réalisation est représentée ci-après.



-Figure 1.6- Démarche pour la réalisation.

1.3. CARACTERISTIQUES DE MCSE

MCSE est une démarche complète qui permet de passer du problème à une réalisation. Nous reprenons ici les aspects généraux essentiels de la méthodologie.

-A- UN MODELE DE DESCRIPTION COMME BASE

Toute méthodologie est basée sur un ou des modèles, ceci permet une décomposition en étapes. MCSE est basée sur un modèle de description interne en 3 composantes. Ce modèle incite à décrire tout système selon une hiérarchie de niveaux de description. Chaque niveau sert d'intermédiaire entre 2 étapes. La plupart des modèles sont graphiques, favorisant ainsi une compréhension globale et rapide.

-B- UNE DEMARCHE GLOBALEMENT DESCENDANTE POUR LA CONCEPTION

Chaque étape de la méthodologie permet de passer d'un niveau de description au niveau plus détaillé suivant en enrichissant la solution d'une composante supplémentaire.

La progression est donc globalement descendante puisqu'elle part du problème posé jusqu'à aboutir à une réalisation opérationnelle.

-C- UNE PROGRESSION ITERATIVE

Un développement ne peut pas se faire sans erreurs ou omissions. Des corrections sont toujours nécessaires. Basée sur la correction par retours-arrières, une phase de vérification en fin de chaque étape permet la détection des erreurs et induit un travail itératif avec des retours à l'intérieur de l'étape ou sur les étapes précédentes.

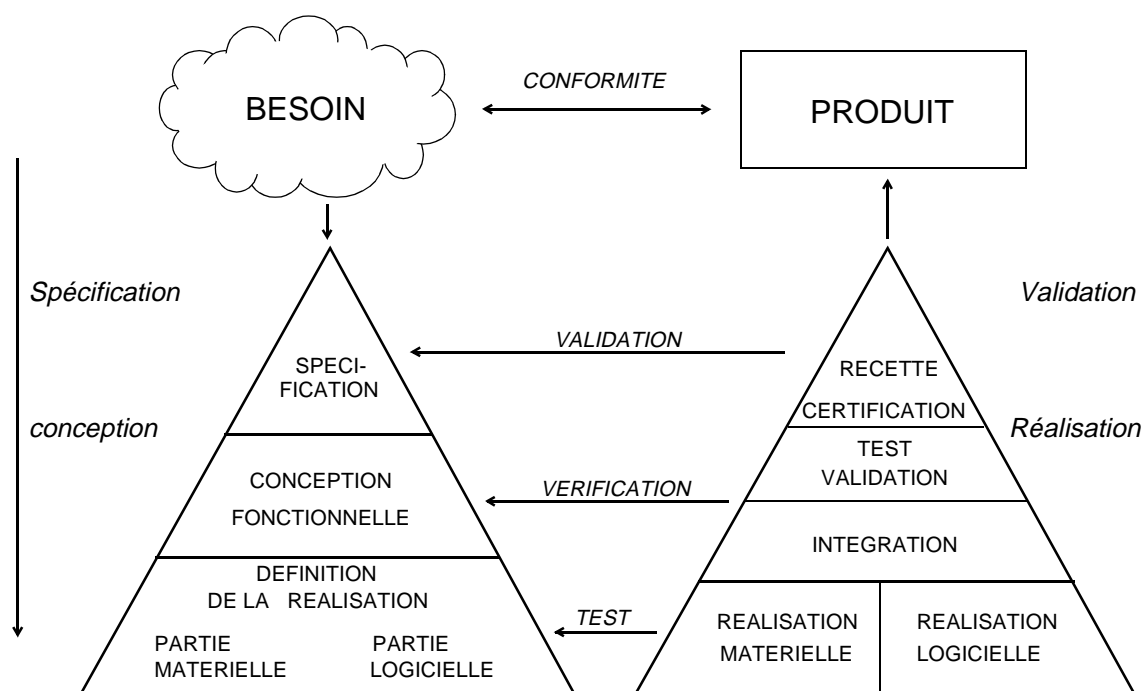
-D- UNE METHODE SPECIFIQUE POUR CHAQUE ETAPE

Le modèle de description de la solution à l'issue de chaque étape n'est pas suffisant (le QUOI). Le concepteur doit disposer pour chaque étape d'un guide précis lui expliquant COMMENT passer de la spécification en entrée à une solution possédant des qualités. Ce guide est la méthode à suivre : technique d'analyse, séquence des décisions, critères de choix. Par opposition à une recherche intuitive, l'emploi d'une méthode garantit l'obtention rapide d'une solution a priori de qualité.

Au-delà de l'aspect méthode, l'idée des modèles génériques de solutions a un intérêt certain. De tels modèles ayant la particularité pour la décomposition fonctionnelle d'être générateurs de multiples solutions, sont retenus car possédant des qualités intrinsèques: lisibilité, maintenabilité, simplicité, adéquation au modèle global MCSE. La connaissance de tels modèles, en complément des méthodes, améliore notablement le pourcentage des développements corrects.

-E- UNE DEMARCHE GLOBALEMENT ASCENDANTE POUR LA REALISATION

L'assemblage n'est possible qu'après disponibilité des constituants. Ainsi, la réalisation débute par la réalisation des plus petits sous-ensembles, puis remonte progressivement par assemblage et intégration de fonctions plus globales. Le travail de réalisation est représentable par un triangle juxtaposé à celui de la conception comme l'indique la figure suivante. La largeur du triangle pour chaque stade indique la quantité d'informations à maîtriser.



-Figure 1.7- Forme en double triangle pour le développement.

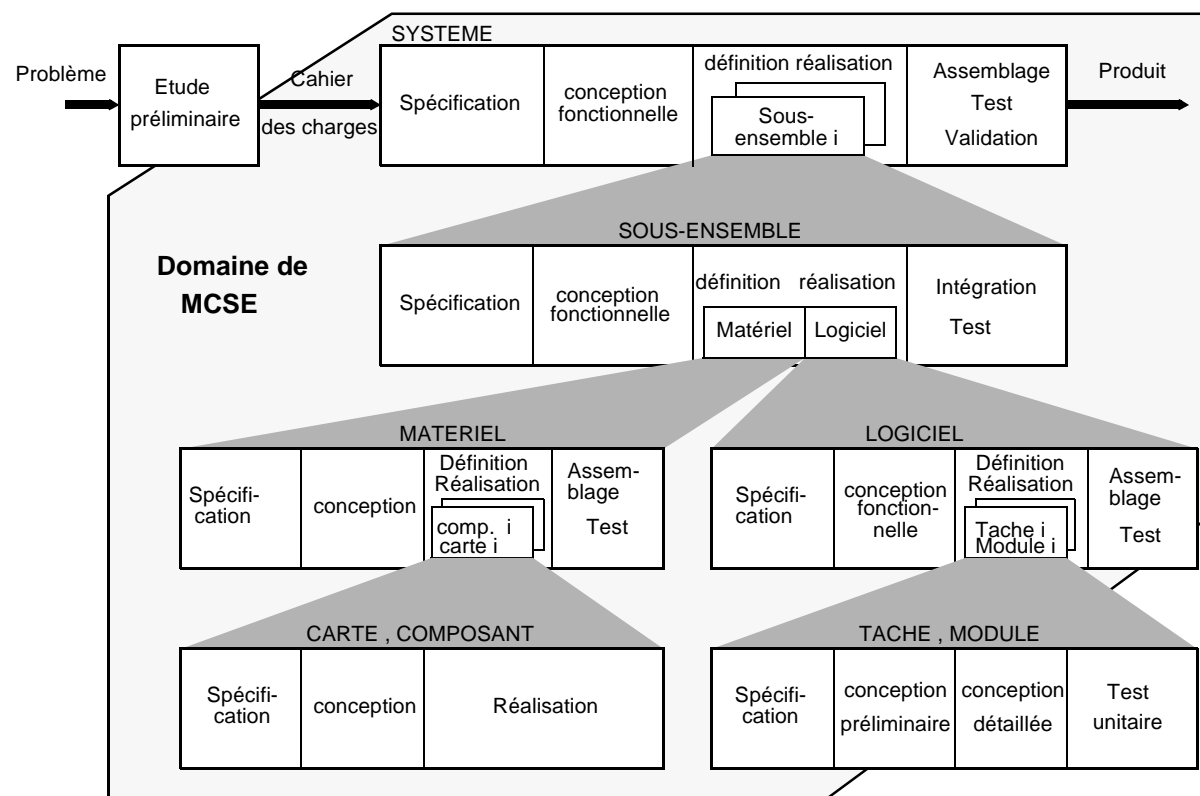
Chaque niveau de la réalisation est vérifiable, ce qui assure sa conformité au niveau correspondant de la conception. La figure 1.7 illustre la correspondance entre les 2 démarches complémentaires.

-F- UN MODELE DE CYCLE DE DEVELOPPEMENT HIERARCHIQUE

Pour un système relativement complexe, le modèle ci-dessus en double triangle n'est pas suffisamment précis.

L'étape de définition de la réalisation conduit à mettre en évidence les 2 parties : matériel, logiciel. Chaque partie est à nouveau à développer selon une démarche en 3 phases : spécification, conception, définition de la réalisation, et ceci jusqu'à la mise en évidence des constituants disponibles (composants matériels ou logiciels).

Ainsi comme le montre la figure ci-après, le modèle de cycle de développement est un emboîtement de développements. Au premier niveau, la conception est générale et concerne l'application dans son ensemble. Au fur et à mesure du raffinement de la solution, les développements concernent des problèmes plus spécifiques en rapport avec la réalisation : développement d'un composant, d'une fonction spécifique, d'un module logiciel ...



-Figure 1.8- Modèle hiérarchique pour le développement.

-G- GUIDE POUR LA DOCUMENTATION

Le modèle de description induit directement la structure des documents à produire durant le développement. Chaque document est le résultat d'une étape, et sert comme informations pour l'étape suivante.

Ainsi, en respectant la démarche par étapes successives, la documentation est générée au fur et à mesure du développement et non en final. Elle possède alors une réelle qualité quant à la forme et au fond puisqu'elle relate, en plus de la solution, la démarche suivie et l'argumentation qui justifie les décisions importantes.

Produite de cette manière, la documentation est utilisable durant le cycle de développement: pour les phases de vérification selon un cycle auteur-lecteurs, pour l'observation de l'état d'avancement, mais aussi pour les étapes ultérieures et en particulier pour la maintenance.

-H- GUIDE POUR LA CONDUITE DE PROJET

Le modèle de cycle de développement est utilisable pour la mise en place d'une procédure de conduite d'un ensemble de projets. Pour chaque projet, cette activité concerne :

- le management : planification, organisation, direction, contrôle et suivi du projet,
- l'obtention de la conformité : planification des tests, nature des tests techniques à utiliser, les résultats, conformité et certification,
- la gestion de la documentation : spécification des documents, planification des revues, méthodes de gestion, mises à jour ...
- la gestion de la maintenance : domaine et procédures de maintenance, solutions et outils, planification,
- la gestion de la qualité : assurance qualité, méthode pour l'obtention de la qualité, procédures de contrôle.

-I- UNE METHODOLOGIE OUVERTE ET COMPLEMENTAIRE

MCSE ne se trouve pas restreinte à une méthode particulière et à un domaine bien spécifique. Au contraire, pour chaque étape, plusieurs méthodes sont utilisables et c'est au concepteur de choisir selon des critères, celle qui lui permet de résoudre au mieux son problème.

Développée au départ pour les systèmes de contrôle/commande temps-réel à microprocesseurs, l'expérience nous a montré son adéquation pour une large classe d'application et de techniques, et tout particulièrement pour les applications qui utilisent l'électronique et l'informatique. MCSE n'est pas à opposer aux autres méthodologies, bien au contraire, elle se veut complémentaire. La plupart des modèles proposés par différents auteurs s'avèrent utilisables. Par exemple, SADT et les méthodes de spécification de WARD et MELLOR et de HATLEY favorisent la tâche d'analyse du problème. La méthodologie de JACKSON est dans l'esprit relativement proche de MCSE. Les méthodologies DARTS et SDWMC de BUHR ont aussi des points communs.

1.4. CONSEILS POUR UNE UTILISATION RATIONNELLE DE MCSE

Nous rappelons ici les principes essentiels de la Méthodologie qu'il est conseillé de suivre pour obtenir les meilleurs résultats. Ces principes répondent à des difficultés couramment rencontrées par les concepteurs.

Au préalable, il est bon de rappeler que le développement est une activité humaine mettant en jeu une méthodologie et un système social. Les 2 parties ont une importance pour la réussite. Aussi, l'attitude volontariste des concepteurs est une première condition strictement nécessaire pour le succès de la démarche.

Ensuite, un bon concepteur doit savoir se placer au dessus des aspects techniques et ceci en utilisant le plus possible des concepts. Il ne doit pas se laisser guider par une solution a priori; dans ce cas probablement, il aboutira à la conclusion qu'elle est correcte sans avoir évalué d'autres alternatives. Il est aussi essentiel de chercher à satisfaire la demande et non pas se satisfaire de son développement.

Nous citons ci-dessous pour chaque étape, quelques-unes des difficultés que tout utilisateur de la méthodologie risque de rencontrer et le point de vue à adopter.

-A- ETAPE DE SPECIFICATION

Concernant l'étape de spécification, il y a 2 difficultés. Tout d'abord pour la modélisation de l'environnement, il faut décider du modèle le plus approprié pour cet environnement, à la fois sa nature et son niveau de détail. Pour être bonne, l'analyse doit se faire sur un plan fonctionnel et non matériel. Il faut donc penser aux entités fonctionnelles. Ceci est une difficulté qui conduit à des erreurs fréquentes. De l'analyse se déduit la nature du modèle qu'il faut rechercher dans l'ordre de complexité: modèle statique données/événements, modèle dynamique global, modèle des activités. La modélisation est à faire selon un niveau de détails nécessaire mais juste suffisant pour résoudre le problème. Elle ne doit pas faire intervenir le système à concevoir.

Ensuite, pour l'élaboration des spécifications fonctionnelles, celles-ci doivent se limiter à une description externe du système. Les fonctions du système ne doivent être que des fonctions de l'application, qu'il ne faut surtout pas confondre avec les fonctions internes du système.

S'assurer de la complétude d'une spécification est aussi une réelle difficulté. Celle-ci doit apparaître cohérente, claire et doit être vérifiée par le demandeur.

-B- ETAPE DE CONCEPTION

Pour cette étape, le premier point important est de faire une conception indépendante de la technologie. La première difficulté est donc de faire abstraction des contraintes technologiques et donc de toute idée de réalisation. Pour cela, il faut éliminer tous les problèmes de répartition géographique et d'interfaces pour se consacrer au seul aspect fonctionnel. Il faut donc déjà partir des entrées et sorties fonctionnelles et non pas physiques.

Ensuite, la recherche d'une solution est à faire sur la base des variables internes nécessaires et non pas sur la base des fonctions internes, alors que la tendance naturelle consiste à appréhender le problème par les fonctions. L'approche préconisée conduit à une solution plus simple à comprendre et plus simple à implanter. Les idées de solutions peuvent aussi se déduire de modèles génériques de solution. L'intérêt est de disposer d'un guide pour la recherche de variables.

D'une manière générale, le concepteur doit avoir des idées pour être créatif mais doit faire abstraction de son idée a priori, sinon il conduira son développement sur la base de cette idée inhibant alors l'évaluation de toute autre solution peut-être meilleure.

C- ETAPE DE DEFINITION DE LA REALISATION

Une des tendances naturelles consiste à entreprendre la réalisation rapidement: développement ou choix du matériel très tôt et écriture du logiciel. L'étape de définition de la réalisation est une étape de réflexion importante qui conduit à réduire l'effort de développement pour en particulier la réalisation. Cette étape permet de s'adapter aux

caractéristiques de l'environnement, en introduisant les interfaces. Pour cela, il faut rechercher la solution la plus appropriée, c'est particulièrement le cas pour l'interface homme-machine. D'autre part, les contraintes de temps doivent être évaluées correctement pour le cas le plus défavorable de manière à pouvoir prouver a priori le bon fonctionnement de l'application.

De cette analyse découle ensuite la recherche de la répartition matériel/logiciel qui a pour objectif de réduire au maximum le développement. La démarche conduit à minimiser le matériel, puis le logiciel. Concernant le logiciel, il est composé de 2 parties: la partie opératoire strictement nécessaire pour le fonctionnement et donc relativement incompressible si la spécification comportementale a été bien élaborée, et la partie dite organisationnelle qui concerne les relations entre les fonctions. Cette deuxième partie est compressible, et l'objectif des transformations pour obtenir le schéma d'implantation logicielle est de réduire au maximum cette partie. La Méthodologie favorise cette réduction. Cette approche est tout à fait justifiée pour les applications temps-réel dédiées où le développement est assuré une fois pour toutes.

1.5. SUIVI D'UN PROJET

Développer un projet nécessite de mener en parallèle 3 activités:

- le développement à proprement dit en procédant étape par étape,
- la vérification du développement,
- la documentation du projet.

La fiche jointe sur la page suivante a été établie comme guide simplifiée utilisable dans le cas d'un projet qui peut se développer d'une manière linéaire. Pour des projets plus conséquents, une telle fiche peut aussi servir mais en limitant son rôle à chaque sous-ensemble à développer.

Les concepteurs et les responsables de projets sont invités à utiliser une telle fiche ou une variante de celle-ci, de manière à disposer d'une observation sur l'avancement du travail, ceci pour mieux répondre aux objectifs de coût, de délais et de qualité.

Cette présentation succincte, loin d'être suffisante, n'est donnée ici qu'à titre de rappel et de guide pour la méthodologie MCSE. Pour des informations plus précises sur la démarche et pour mieux comprendre les études de cas décrites dans les chapitres suivants, le lecteur est invité à lire l'ouvrage: "Spécification et Conception des systèmes, une méthodologie MCSE" de l'auteur.

