

mips

conception assistée
des
systèmes d'information

MÉTHODE · MODÈLES · OUTILS

F. BODART Y. PIGNEUR

2^e édition



MASSON

BIBLIOTHEQUE DU CERIST

conception assistée des systèmes d'information

MÉTHODE · MODÈLES · OUTILS

BIBLIOTHEQUE DU CERIST

CHEZ LE MÊME ÉDITEUR

Dans la même collection :

LES SCHÉMAS DIRECTEURS : Démarche pratique, par G. BALANTZIAN. 1989, 3^e édition, 264 pages.

MAÎTRISER LA MODÉLISATION CONCEPTUELLE, par R. PLANCHE. 1988, 256 pages.

APPRENDRE ET PRATIQUER MERISE, par J. GABAY. Préface de A. Rochfeld. 1989, 240 pages.

CONCEPTION ET RÉALISATION ASSISTÉES PAR ORDINATEUR DE LOGICIELS DE GESTION, par PHAN HUY DUONG. 1983, 408 pages.

SQL ET DB2, par M. KOUTCHOUK. 1989, 224 pages.

BASES DE DONNÉES ET SGBD, de la conception à la mise en œuvre. Démarche pratique, par F. KRAMARZ et O. PERRAULT. 1986, 200 pages.

TRANSMISSIONS ET RÉSEAUX LOCAUX. Architecture IEEE 802, par P. MILLET. 1987, 232 pages.

INTÉGRATION VOIX ET DONNÉES. Principes et concepts, par P. MILLET. 1988, 368 pages.

Autres ouvrages :

CONCEPTION ASSISTÉE DES APPLICATIONS INFORMATIQUES.

Tome 2. Conception de la base de données. Par J.-L. HAINAUT. *Méthode + Programmes*. 1986, 224 pages.

MÉTHODE GÉNÉRALE D'ANALYSE D'UNE APPLICATION INFORMATIQUE, par X. CASTELLANI.

Tome 0. Macro-analyse. Étude préalable et analyse conceptuelle des systèmes d'information. 1987, 696 pages.

Tome 1. Étapes et points fondamentaux de l'analyse conceptuelle. 1986, 7^e édition révisée, 328 pages.

Tome 2. Étapes et points fondamentaux de l'analyse organique et de la programmation. 1986, 6^e édition révisée et augmentée, 268 pages.

LA CONCEPTION DES SYSTÈMES D'INFORMATION. Un modèle, un dossier standard, des méthodes, par J.-L. CAVARERO et D. HERIN-AIME. Préface de J.-C. Boussard. *Manuels Informatiques Masson*. 1982, 208 pages.

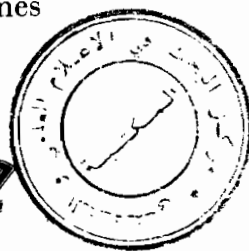
APPLICATION SYSTEM, un système IBM de 4^e génération, par J. RAMBAUD. Préface de N. Manson. *Manuels Informatiques Masson*. 1989, 2^e tirage, 272 pages.

L'AUDIT INFORMATIQUE, méthodes, règles, normes, par M. THORIN. *Manuels Informatiques Masson*. 1984, 2^e édition revue et augmentée, 152 pages.

MIPS

C 2101

Méthodes Informatiques et Pratique des Systèmes
Collection coordonnée par A. CHAMPENOIS



conception assistée des systèmes d'information

MÉTHODE · MODÈLES · OUTILS

François BODART
*Professeur aux Facultés
Universitaires Notre-Dame
de la Paix à Namur*

Yves PIGNEUR
*Professeur à l'Université
de Lausanne*

2^e édition entièrement refondue

MASSON

Paris Milan Barcelone Mexico
1989

Cet ouvrage est la seconde édition de :
Conception assistée des applications informatiques
Tome 1 : Étude d'opportunité et analyse conceptuelle
 paru en 1983 dans la collection Méthode + Programmes

Tous droits de traduction, d'adaptation et de reproduction par tous procédés, réservés pour tous pays.

Toute reproduction ou représentation intégrale ou partielle, par quelque procédé que ce soit, des pages publiées dans le présent ouvrage, faite sans l'autorisation de l'éditeur est illicite et constitue une contrefaçon. Seules sont autorisées, d'une part, les reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective, et d'autre part, les courtes citations justifiées par le caractère scientifique ou d'information de l'œuvre dans laquelle elles sont incorporées (loi du 11 mars 1957 art. 40 et 41 et Code Pénal art. 425).

Des photocopies payantes peuvent être réalisées avec l'accord de l'éditeur. S'adresser au : Centre Français du Copyright, 6 bis, rue Gabriel-Laumain, 75010 Paris. Tél. : 48.24.98.30.

© Masson, Paris, 1983, 1989

ISBN : 2-225-81807-X

ISSN : 0757-7206

5644

MASSON
 MASSON ITALIA EDITORI
 MASSON S.A.
 MASSON EDITORES

120, Bd Saint-Germain, 75280 Paris Cedex 06
 Via Statuto 2, 20121 Milano
 Balmes 151, 08008 Barcelona
 Dakota 383, Colonia Napoles, 03810 Mexico D.F.

Préface

François Bodart et Yves Pigneur nous livrent ici un exposé synthétique des bases de la **méthode IDA** (Interactive Design Approach)*, dont ils sont les auteurs, et qui est aujourd'hui utilisée avec succès par des centaines d'informaticiens et de non-informaticiens pour concevoir des systèmes d'information de gestion.

Une oeuvre rigoureuse et pratique

Les auteurs sont tous deux des universitaires et leur recherche s'est effectuée dans le cadre de l'Institut d'Informatique des Facultés Universitaires Notre-Dame de la Paix à Namur où François Bodart est Professeur. Yves Pigneur enseigne aujourd'hui à l'Ecole des Hautes Etudes Commerciales de Lausanne.

Cette origine universitaire d'une méthode d'informatisation pourrait paraître suspecte aux pragmatistes : elle est un gage de rigueur et de profondeur. Le livre qui suit n'a pas été rédigé à la sauvette à l'issue de quelques réunions, c'est le fruit de douze années de travail de toute une équipe de chercheurs ouverte à ce qui se passe dans les centres de recherche et sur le terrain, et qui se soucie à la fois de participer activement à l'évolution de la recherche sur les systèmes d'information et de rendre des services réels à des utilisateurs de leur méthode, experts ou non, avec lesquels ils n'ont cessé de dialoguer.

Dans un domaine où règnent l'argument d'autorité et la persuasion commerciale, l'exposé solide que nous livrent François Bodart et Yves Pigneur est indispensable pour échapper au dogmatisme et à l'illusion de la facilité. C'est un exposé qui porte parce qu'il est loyal, complet, bien construit et qu'il est mis à la portée de tous ceux qui sont susceptibles de concevoir avec méthode des systèmes d'information.

Mais la visée de nos auteurs n'est pas seulement pédagogique : ils ont voulu élaborer et fournir une véritable méthode qui soit effectivement utilisée parce qu'elle condense en des outils informatiques le savoir-faire aujourd'hui formulable et facilite ainsi la tâche de ceux qui s'y soumettent sans leur enlever ce qui constitue la noblesse de leur métier.

En tant que dirigeant de la société commerciale fondée pour diffuser la méthode et l'atelier logiciel IDA, je puis témoigner de la fécondité de la recherche de François Bodart et Yves Pigneur et de leurs collègues. Cette fécondité a été amplement démontrée entre les deux versions de leur ouvrage, la

* IDA est une marque commerciale de la Société METSI.

présente édition bénéficiant du champ d'expérience que nous avons labouré avec eux depuis six ans et où ils n'ont cessé de puiser pour préciser, améliorer et étendre les formulations méthodologiques et les outils qu'ils présentent ici. Si cela était encore nécessaire, l'aventure d'IDA prouverait à l'évidence qu'"il n'y a rien de plus pratique qu'une bonne théorie".

Une oeuvre originale dès le départ

Lorsqu'en 1977, François Bodart a commencé à travailler à l'élaboration d'une méthode de conception assistée des systèmes d'information de gestion, on parlait abondamment de méthodes d'analyse et de programmation, mais celles-ci contenaient d'autant moins de savoir-faire réel et formulé qu'elles exigeaient d'avantage de discipline dans le parcours à suivre en imposant l'élaboration manuelle d'une documentation plus lourde. On commençait à parler de modélisation mais celle-ci n'intéressait que les concepteurs de bases de données. Les outils se limitaient à la programmation et à la génération de code à partir de langages d'un peu plus haut niveau que le COBOL.

C'est dans ce contexte, et après avoir participé aux projets ISDOS de l'Université de Michigan où est née la conception des systèmes d'information assistés par ordinateur, que François Bodart s'est fixé pour objectif de réaliser une méthode entièrement outillée, fondée sur des principes analogues à ceux de la CAO industrielle. L'approche était originale et ambitieuse.

- Au lieu de considérer comme inévitable le développement des systèmes d'information à travers une suite cahotique de traductions approximatives et de remises en cause, IDA pensait le développement comme un travail progressif de spécification par affinements successifs et prise en compte séquentielle des contraintes sans retour en arrière.
- Au lieu de concevoir d'emblée les systèmes d'information à travers les contraintes spécifiques de moyens informatiques contingents, IDA s'attachait à une spécification conceptuelle, strictement fonctionnelle, et indépendante des solutions informatiques éventuelles. Cette étape de spécification était tenue pour essentielle, puisqu'elle avait pour objet de constituer une fondation solide au développement des applications informatiques.
- L'expression de cette spécification fonctionnelle fut envisagée dès l'origine dans un langage formel de haut niveau. Ce langage constitue un univers logique strict et homogène, et correspond à un ensemble de modèles riches dont la sémantique, analysable par l'ordinateur, est bien définie.
- Ce choix du langage formel était nécessaire pour garantir la pertinence des systèmes d'information à l'aide de tests puissants au niveau de la spécification conceptuelle elle-même.

La rigueur de l'univers des modèles et du langage associé permettait de vérifier la complétude et la cohérence des spécifications dans cet univers par l'application informatique de règles selon une stratégie de validation qui n'est plus tributaire d'une documentation à produire par l'analyste.

- Le caractère formel des spécifications devait également autoriser une approche expérimentale de la spécification. Il a été mis à profit très tôt pour

rendre les spécifications fonctionnelles directement exécutables. Sous la forme de simulateurs, on vérifie les attentes des traitements et la charge des ressources de manière à s'assurer de la faisabilité du système et de ses performances de gestion. Sous la forme de prototypes offrant des vues internes du système, on vérifie la pertinence des règles de gestion, l'articulation fonctionnelle des traitements et des données et la conformité aux besoins des utilisateurs.

- Le langage formel était également nécessaire pour générer le plus automatiquement possible les applications informatiques à partir des spécifications conceptuelles. Ces spécifications élaborées avec la méthode et les outils IDA sont aujourd'hui transformées en schémas de fichiers et de bases de données, en architectures de programmes, en spécifications informatiques et en codes exploitables, à travers divers outils de réalisation.
- Les propriétés principales de la méthode IDA que nous venons d'évoquer sont conciliées avec la plus grande souplesse.

Grâce au caractère non procédural du langage, la démarche est totalement adaptable. L'extension du langage est autorisée à l'utilisateur qui crée ses propres attributs. Enfin, le langage est modifiable à l'aide d'un méta-système sans avoir à modifier les outils associés, ce qui permet aux utilisateurs d'envisager sereinement les progrès à venir en matière de modélisation des systèmes d'information.

Lorsque la première édition du présent ouvrage est parue en 1983, IDA était la première véritable méthode complète de conception des systèmes d'information de gestion. Elle apportait un savoir-faire condensé dans des règles opératoires et dans l'interaction avec des outils d'aide évolués. Elle offrait des moyens puissants de test et d'expérimentation. Elle couvrait tous les aspects des systèmes d'information : données, traitements, ressources et autorisait la conception de très grands systèmes.

Une seconde édition entièrement refondue et complétée

Depuis la parution de cette première édition, les outils d'IDA ont été stabilisés, améliorés et complétés. L'utilisation concrète de la méthode et des outils s'est généralisée à grande échelle sur de multiples sites. Un savoir-faire approfondi a été acquis, formulé et introduit dans les outils. La présente édition est enrichie de cette expérience comme des recherches que les auteurs et leurs équipes ont poursuivies depuis lors.

Des précisions utiles ont été apportées à la sémantique des modèles (chapitre II). Une présentation plus générale et plus aisément accessible du langage formel de spécification nous est maintenant offerte (chapitre IV). Le caractère expérimental de la méthode est mis en valeur et les versions nouvelles de la simulation et du prototypage nous sont rapidement présentées (chapitre V).

Les innovations majeures de l'ouvrage portent sur les règles de validation des spécifications, sur la démarche de l'analyse fonctionnelle, sur l'ouverture vers la réalisation qui tient lieu de conclusion, et sur l'ergonomie de l'atelier logiciel IDA évoquée tout au long de l'ouvrage.

- L'ensemble de règles de validation des modèles qui nous est proposé au chapitre III donne un contenu opératoire précis aux notions de complétude et de cohérence auxquelles beaucoup de concepteurs de systèmes d'information se réfèrent sans leur donner un sens explicite. L'énoncé systématique de règles claires se situe loin des querelles byzantines que se livrent les spécialistes à ce sujet. Il affirme sans prétention la très grande maîtrise de la modélisation à laquelle les auteurs sont parvenus et qu'ils mettent aujourd'hui à la portée de chacun. Il illustre aussi la méthode de recherche des auteurs qui ont déjà traduit en outils de test la plupart des règles de validation ici formulées.
- La précédente édition ne faisait qu'évoquer la démarche de l'analyse fonctionnelle. Celle-ci nous fournit, au chapitre VI, l'énoncé détaillé et bien formulé d'une démarche type qui définit une architecture de tâches et pour chacune d'elles des objectifs, un contenu, des règles à respecter et des outils à utiliser. L'utilisateur dispose désormais d'un guidage à l'écran et la démarche peut être complètement industrialisée en cataloguant les commandes de test et de documentation associées, les outils devenant alors totalement transparents derrière le poste de travail.

Une coopération Université-Industrie exemplaire

Entre les deux éditions de l'ouvrage de François Bodart et Yves Pigneur un très important travail de recherche a été accompli par les auteurs et leurs équipes. Je me fais un devoir d'ajouter que cette recherche, loin de s'enfermer sur elle-même, s'est engagée dans une coopération Université-Industrie que je tiens pour exemplaire entre les Facultés Universitaires Notre-Dame de la Paix à Namur et la Société METSI.

METSI fut créée pour distribuer la méthode et les logiciels IDA au moment où le texte de la première édition de cet ouvrage était remis à l'éditeur. Pendant cinq ans, cette société a soutenu la recherche réalisée aux Facultés de Namur et les Facultés ont assuré le développement et la maintenance IDA. Peu à peu, METSI a pris une part croissante dans le développement en s'attachant à l'intégration de la démarche dans l'atelier logiciel et à l'interfaçage de cet atelier avec divers outils de réalisation.

METSI, après avoir racheté aux Facultés les droits des logiciels IDA, vient de créer son propre centre de développement industriel qui est hébergé pour trois ans dans la pépinière d'entreprises des Facultés. La coopération continue sur une nouvelle base, l'équipe des Facultés se dégageant rapidement du développement pour se consacrer de plus en plus exclusivement à la recherche dans le même domaine.

Une innovation de grande portée

A travers cette coopération durable entre les Facultés de Namur et METSI, et dans l'amitié qui unit ses protagonistes, une contribution importante a été apportée à l'émergence d'une innovation majeure dont nous commençons seulement à saisir la portée.

Je voudrais évoquer brièvement les implications des choix faits dès 1977 dans l'orientation des recherches de François Bodart et Yves Pigneur :

- La spécification des systèmes d'information, telle qu'elle est envisagée avec IDA, est une condition pour que les systèmes d'information aient une existence propre et puissent être **gérés comme une ressource**. Cette condition satisfaite, nous pourrions voir un jour l'organisation fonctionnelle à l'actif du bilan des entreprises.
- L'énoncé formel des spécifications, l'élucidation progressive des règles de travail de spécification et des règles de transformation de ces spécifications en applications informatiques sont à la base même de **l'industrialisation du logiciel**. Une standardisation très poussée est déjà possible, elle entraîne l'entrée massive des automatismes dans une activité informatique qui pouvait jusqu'ici se vanter d'avoir pour raison d'être d'automatiser artisanalement le travail des autres.
- Les systèmes d'information, en autorisant de s'éloigner sans risque de l'existant sacro-saint, permettent de supprimer la plus grosse partie de la maintenance ultérieure des logiciels de gestion, maintenance imputable, comme on le sait, aux seuls glissements des spécifications qui accompagnent l'apprentissage progressif des utilisateurs.
- Ces quelques évocations suffisent pour entrevoir les changements profonds que vont connaître les structures et les métiers de l'informatique du fait de méthodes et d'outils comme ceux d'IDA.

Les années passant, les choix initiaux se sont avérés pertinents. Ils se sont précisés, approfondis et enrichis. Pris ensemble, ils constituent aujourd'hui une stratégie de génie logiciel cohérente, propre à orienter les progrès à venir.

Il s'agit bien d'une stratégie, ouverte aux apports de la recherche et de l'expérience, qui a déjà creusé une trace profonde et continue d'innover sans se laisser distraire par les gadgets.

Elle est cohérente en ce sens qu'elle poursuit d'une même élan trois objectifs qui sont indissociables dans la phase initiale du génie logiciel : gestion des systèmes d'information, industrialisation du développement de logiciel, pérennité des investissements effectués sur les ateliers logiciels et à travers leur emploi.

Marcel Bourgeois,
Président-Directeur général
de METSI

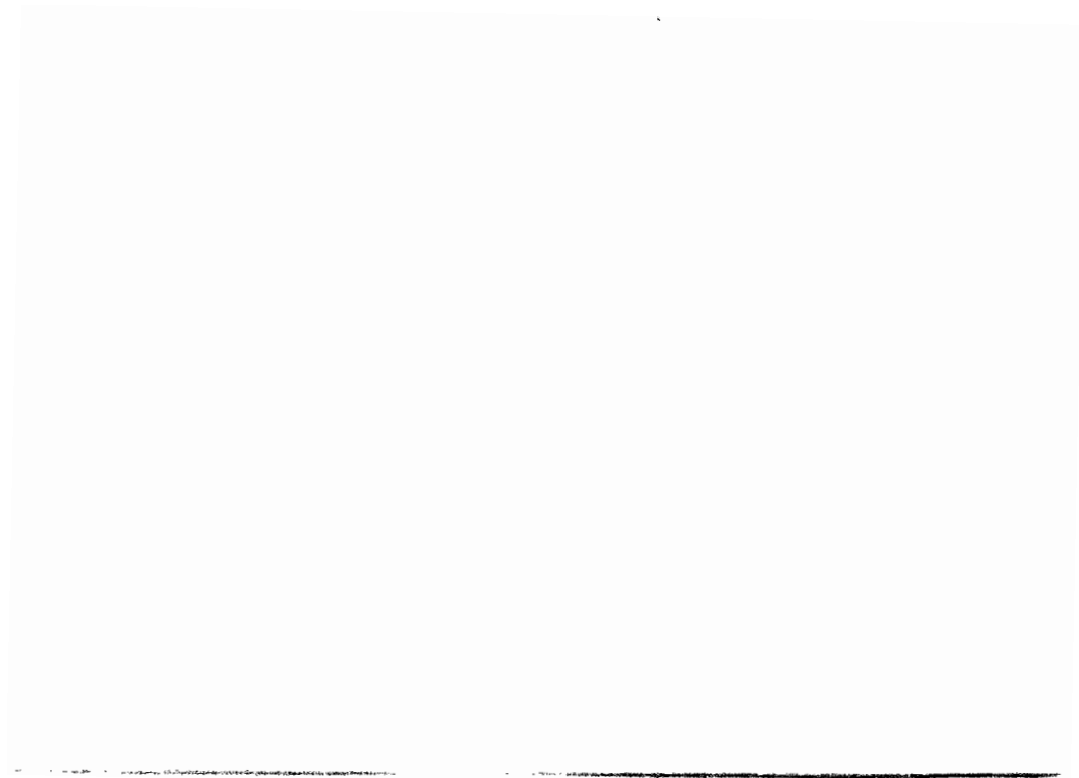


Table des matières

PREFACE	i
Chapitre I - LA CONCEPTION DES SYSTEMES D'INFORMATION	1
I.1 Quels systèmes d'information	1
I.2 Le développement des systèmes d'information	4
1.2.1 <i>Le cycle de vie d'un S.I.</i>	4
1.2.2 <i>L'analyse fonctionnelle</i>	6
I.3 Une méthodologie de conception des S.I.	7
Chapitre II - MODELES D'AIDE À L'ANALYSE FONCTIONNELLE	11
II.1 Introduction	11
II.2 Le modèle Entité-Association (E.A.)	12
II.2.1 <i>Définition et objectifs</i>	13
II.2.2 <i>Concepts de base</i>	15
II.2.2.1 Concept d'entité	15
II.2.2.2 Concept d'association	17
II.2.2.3 Concepts d'attribut et de valeur	20
II.2.2.4 Représentation graphique des concepts de base du modèle E.A.	24
II.2.3 <i>Mécanismes d'abstraction sous-jacents à la structuration des informations</i>	25
II.2.4 <i>Les contraintes d'intégrité</i>	29
II.2.4.1 Notions	29
II.2.4.2 Connectivité	32
II.2.4.3 Identification	33
II.2.4.4 Existence	37
II.2.4.5 Inclusion, Exclusion, Egalité de rôles	38
II.2.4.6 Inclusion, Exclusion, Egalité d'association	40
II.2.4.7 Sous-typage	40
II.2.4.8 Domaine de valeurs	42
II.2.4.9 Dépendances fonctionnelles	43
II.2.5 <i>Concepts dérivés</i>	49
II.2.5.1 Concept de mémoire	49
II.2.5.2 Concept de message	49

II.3 Modèle de structuration des traitements	50
II.3.1 <i>Définition et objectifs</i>	50
II.3.2 <i>Structuration arborescente des traitements</i>	51
II.3.3 <i>Nomenclature standard de structuration des traitements</i>	52
II.3.3.1 Le concept de "projet"	54
II.3.3.2 Le concept d' "application"	54
II.3.3.3 Le concept de "phase"	56
II.3.3.4 Le concept de "fonction"	60
II.4 Modèle de la statique des traitements	63
II.4.1 <i>Définition et Objectifs</i>	63
II.4.2 <i>Modèle de la boîte noire</i>	63
II.4.3 <i>Règles de traitement</i>	65
II.5 Modèle de la dynamique des traitements	70
II.5.1 <i>Définition et objectifs</i>	70
II.5.2 <i>Le concept de Processus</i>	71
II.5.3 <i>Le concept d'événement et le mécanisme de déclenchement</i>	72
II.5.4 <i>Le mécanisme de synchronisation</i>	77
II.5.5 <i>Le mécanisme de sélection</i>	84
II.5.6 <i>Le mécanisme de duplication</i>	86
II.5.7 <i>Schéma de la dynamique</i>	88
II.5.8 <i>Les éléments de quantification</i>	89
II.6 Modèle des ressources	91
II.6.1 <i>Définition et objectifs</i>	91
II.6.2 <i>Le concept de Ressource</i>	92
II.6.3 <i>Le mécanisme de réquisition</i>	93
II.6.4 <i>Les notions de priorité et de préemption</i>	95
II.6.5 <i>Evolution temporelle des processus</i>	96
II.7 Modèle des flux de messages : le diagramme des flux	98
II.7.1 <i>Définition et Objectif</i>	98
II.7.2 <i>Les types d'objets du diagramme des flux</i>	98
II.7.3 <i>Les types de relations du diagramme des flux</i>	100
II.7.4 <i>Quantification du diagramme des flux</i>	101
II.7.5 <i>Représentation graphique du diagramme des flux</i>	101
II.7.6 <i>Rapprochement avec le modèle de la dynamique</i>	103

<i>Table des matières</i>	XIII
Chapitre III - REGLES DE VALIDATION DES SPÉCIFICATIONS	105
III.1 Introduction	105
III.2 Validation d'un schéma Entité-Association	106
III.2.1 <i>Règles de complétude</i>	106
III.2.2 <i>Règles de cohérence</i>	110
III.2.3 <i>Forme canonique et normalisation</i>	132
III.2.3.1 Définition et caractéristiques générales du modèle relationnel	132
III.2.3.2 Normalisation	133
III.2.3.3 Expression relationnelle du modèle E.A.	135
III.2.3.4 Forme canonique du modèle E.A. et normalisation du modèle relationnel	137
III.3 Validation d'un schéma de structuration des traitements	138
III.3.1 <i>Règles de complétude</i>	138
III.3.2 <i>Règles de cohérence</i>	138
III.4 Validation d'un schéma de la statique	139
III.4.1 <i>Règles de complétude</i>	139
III.4.2 <i>Règles de cohérence</i>	139
III.5 Validation d'un schéma de la dynamique	142
III.5.1 <i>Règles de complétude</i>	142
III.5.2 <i>Règles de cohérence</i>	147
III.6 Validation d'un schéma des ressources	154
III.6.1 <i>Règles de complétude</i>	154
III.6.2 <i>Règles de cohérence</i>	154
III.7 Validation d'un diagramme des flux	156
III.7.1 <i>Règles de complétude</i>	156
III.7.2 <i>Règles de cohérence</i>	157
Chapitre IV - CONCEPTION ASSISTEE DES SYSTEMES D'INFORMATION	159
IV.1 Introduction	159
IV.2 DSL, un langage de spécification	161
IV.2.1 <i>Présentation générale</i>	161
IV.2.2 <i>Structuration des données en DSL</i>	168
IV.2.3 <i>Décomposition des traitements en DSL</i>	171

IV.2.4 Statique des traitements en DSL	173
IV.2.5 Dynamique des traitements en DSL	175
IV.2.6 Description des ressources en DSL	177
IV.2.7 Diagramme des flux en DSL	179
IV.3 DSL/SPEC, un environnement de spécification	180
IV.3.1 Présentation générale	180
IV.3.2 Le dictionnaire des spécifications	182
IV.3.3 Le méta-dictionnaire du modèle	184
IV.3.4 Les utilitaires d'acquisition et mémorisation	185
IV.3.5 Les utilitaires de sélection et validation	186
IV.3.6 Les utilitaires d'extraction et restitution	190
IV.3.7 Le gestionnaire du dialogue	195
IV.4 En synthèse	195
Chapitre V - VERS UNE APPROCHE EXPERIMENTALE DANS LA CONCEPTION DES SYSTEMES D'INFORMATION	199
V.1 Introduction	199
V.2 Evaluation de la faisabilité par simulation	200
V.2.1 Objectifs	200
V.2.2 Outil d'évaluation - DSL/SIM	204
V.2.3 Simulation du comportement	206
V.2.4 Mesures du comportement	208
V.2.4.1 Mesures concernant les événements	209
V.2.4.2 Mesures concernant les synchronisations	210
V.2.4.3 Mesures concernant les processus	213
V.2.4.4 Mesures concernant les ressources	215
V.2.4.5 Animation	217
V.3 Evaluation de la conformité par prototypage	219
V.3.1 Objectifs	219
V.3.2 L'outil d'évaluation de la conformité : DSL/PROTO	222
V.3.3 Vérification de la conformité	227
Chapitre VI - DEMARCHE DE L'ANALYSE FONCTIONNELLE	231
VI.1 Introduction	231
VI.2 L'analyse d'opportunité	234
VI.2.1 Les étapes élémentaires de l'analyse d'opportunité	235
VI.2.2 Un cadre d'interprétation : le modèle de Leavitt	236
VI.2.3 L'identification du projet	238
VI.2.4 Définition du projet cadre	240

VI.2.5 Etude critique du S.I. existant	246
VI.2.6 Point de contrôle du Projet Cadre	250
VI.2.7 Elaboration et évaluation des solutions	251
VI.2.8 Choix d'une solution	254
VI.2.9 Point de contrôle de la solution recommandée.	255
VI.2.10 Mise à jour du projet cadre	255
VI.2.11 Planning des tâches	256
VI.2.12 Point de contrôle de l'étude d'opportunité	256
VI.3 L'analyse conceptuelle	257
VI.3.1 Introduction	257
VI.3.2 L'élaboration du schéma conceptuel des informations	259
VI.3.2.1 Tri des phases de l'application	260
VI.3.2.2 Analyse d'une phase	261
VI.3.2.3 Elaboration ou extension du schéma brut	264
VI.3.2.4 Mise sous forme canonique	273
VI.3.2.5 Extraction des sous-schémas externes	275
VI.3.2.6 Point de contrôle du schéma conceptuel des informations	276
VI.3.3 Elaboration du schéma conceptuel des traitements	277
VI.3.3.1 Introduction	277
VI.3.3.2 Description des fonctions de la phase	278
VI.3.3.3 Schéma de la dynamique des fonctions	286
VI.3.3.4 Prototypage de la phase	289
VI.3.3.5 Point de contrôle de la spécification de la phase	289
VI.3.4 Consolidation des spécifications	290
VI.3.4.1 Introduction	290
VI.3.4.2 Consolidation proprement dite	290
VI.3.4.3. Quantifications	292
VI.3.4.4 Point de contrôle final	293
Chapitre VII - CONCLUSION	295
VII.1 Une continuité dans le développement	295
VII.2 Conception assistée des spécifications fonctionnelles	296
VII.3 Réalisation assistée des applications informatiques	298
VII.4 En synthèse	301
BIBLIOGRAPHIE	303
INDEX	315