



HAL
open science

De la gestion des données techniques pour l'ingénierie de production - Référentiel du domaine et cadre méthodologique pour l'ingénierie des systèmes d'information techniques en entreprise

Rebiha Bacha

► To cite this version:

Rebiha Bacha. De la gestion des données techniques pour l'ingénierie de production - Référentiel du domaine et cadre méthodologique pour l'ingénierie des systèmes d'information techniques en entreprise. Sciences de l'ingénieur [physics]. Ecole Centrale Paris, 2002. Français. NNT : . tel-00011949

HAL Id: tel-00011949

<https://theses.hal.science/tel-00011949>

Submitted on 14 Mar 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



ECOLE CENTRALE PARIS
THESE DE DOCTORAT

Spécialité
Productique – Génie industriel

Présentée par
Rebiha BACHA

Le 08 janvier 2002
Pour l'obtention du
GRADE DE DOCTEUR

**DE LA GESTION DES DONNEES TECHNIQUES POUR
L'INGENIERIE DE PRODUCTION**
*Référentiel du domaine et cadre méthodologique pour l'ingénierie des
systèmes d'information techniques en entreprise*

Composition du jury

Président	F. Vernadat – Professeur, Université de Metz
Rapporteurs	J-P Bourey – Professeur, Ecole Centrale Lille M. Tollenaere – Professeur, ENSGI - INPG
Examineurs	A-F Cutting Decelle – Professeur, Université d'Evry J-M Célerier – Renault SA D. Lemaire – Renault SA
Directeur de thèse	B. Yannou – MdC HDR, Ecole Centrale Paris

LABORATOIRE PRODUCTIQUE LOGISTIQUE – ECOLE CENTRALE PARIS
GRANDE VOIE DES VIGNES 92925 CHATENAY-MALABRY CEDEX

"You see, I keep thinking that we need a new language, a language of the heart...some kind of language between people that is a new kind of poetry, that is the poetry of the dancing bee, that tells us where the honey is. And I think that in order to create that language we're going to have to learn how you can go through a looking glass into another kind of perception, in which you have that sense of being united to all things, and suddenly you understand everything"

Wallace Shawn & André Gregory. My Dinner with André (1981)

Remerciements

Ma gratitude va tout d'abord à :

B. Yannou pour m'avoir accueillie au laboratoire et m'avoir encadrée avec rigueur et sans discontinuité.

J-P Bourey et M. Tollenaere pour avoir accepté d'étudier et de juger mon rapport de thèse. C'est un honneur de vous avoir comme rapporteurs.

A-F Cutting-Decelle et F. Vernadat ; vous avez bien voulu m'honorer par votre participation au jury en tant qu'examineurs.

J-M Celerier et D. Lemaire, pour votre foi en mes travaux et pour avoir gentiment accepté de participer à ce jury.

La thèse s'est déroulée dans le groupe GSP de la direction de la recherche du Technocentre Renault. Je dois dire que j'ai été heureuse d'avoir vécu parmi vous...

Mes remerciements vont à ceux pour qui j'ai travaillé dans le cadre des projets d'entreprise : les maîtres d'œuvres de la DTSI et la maîtrise d'ouvrage de la DIV/DM ; vous êtes trop nombreux pour être tous nommés ! Vous m'avez donné l'opportunité de pratiquer ma recherche et vous avez jugé mes modestes contributions utiles. De mon côté, les expérimentations que nous avons menées ensemble m'ont beaucoup apporté aussi bien du point de vue de la thèse que vis-à-vis de mon accomplissement personnel.

Sans la communauté de la recherche, mon travail de thèse serait immature et manquerait d'objectivité. Ma reconnaissance s'adresse à vous chercheurs, malgré vos diverses occupations, vous avez toujours répondu à mes requêtes avec un dévouement naturel et humilité. Ces échanges ont été plus que jamais éclairants et utiles pour ma thèse.

Merci particulièrement à B. Morand, D. Rieu, C. Rolland, K. Rouibah, J. A. Harding, A. Al-Ashaab...

Le même sentiment de gratitude va aux consultants et partenaires :

Y. Connetable pour tous le temps que tu as consacré à débroussailler mes premiers modèles. Tu as su me passionner pour l'Objet et plus particulièrement pour les Design Pattern.

C. Bertrand pour ta qualité exceptionnelle de vendeur qui a assuré merveilleusement bien le SAV ! Aussi, pour la passion que nous partageons pour Zachman Framework, SA de Popkin Software...

MT Consulting, pour vos diverses prestations gratuites...

DELMIA/Dassaut Systèmes, pour l'intérêt sincère, témoigné à l'égard de mes travaux de thèse.

Je ne pourrais oublier le Centre des Ressources Technologiques de Renault et la bibliothèque de l'Ecole Centrale de Paris pour mes innombrables commandes réalisées toujours avec promptitude.

La période de thèse a été radieuse en votre présence, mes amis : Faïza, Pascal et Ron.

Enfin, ce travail n'aurait pu se faire sans ton amour, ma grande famille et à toi aussi que je dédie ce mémoire...

J'adresse par avance mes excuses à ceux qui ont contribué dans mon entreprise et dont les noms et sans doute par négligence ou simple oubli de ma part, n'apparaîtront pas ici. A tous, je suis infiniment reconnaissante.

Rapport de Michel Tollenaere en qualité de rapporteur

Mademoiselle Rebiha BACHA présente en vue de l'obtention d'un doctorat à l'École Centrale de Paris un mémoire intitulé "De la gestion des données techniques pour l'ingénierie de production : référentiel du domaine et cadre méthodologique pour l'ingénierie des systèmes d'information techniques en entreprise". Ce thème d'études s'inscrit dans la problématique générale des outils de l'ingénierie de production manufacturière, et en particulier dans les méthodes de conception, réalisation et mise en place d'outils informatiques de productivité collective au sein de ces organisations. Les données techniques de production n'ont pas fait l'objet dans un passé récent du même intérêt que celui qui a pu être apporté aux données techniques des produits. Or il est notoire que 80 % des volumes d'études en vue de produire de nouveaux véhicules, portent sur les procédés et processus de production, les 20% restant portant sur la définition et le comportement du produit. De plus, les principaux retards sur les délais de développement sont dus à des problèmes liés aux process. Ce sujet d'études, conduit en partenariat étroit avec un grand constructeur automobile, nous apparaît donc comme extrêmement pertinent et porteur de performances des systèmes productifs dans un futur proche.

Le mémoire, divisé en cinq chapitres totalisant 163 pages, est présenté avec tous les standards de qualité attendus d'un mémoire de thèse : qualité de la typographie, qualité des figures, table des matières et des figures, liste des acronymes, glossaire... Cette présentation, assortie d'un style d'écrit précis et concis, facilite grandement l'étude du contenu.

Le premier chapitre s'appuie sur une étude de terrain pour en dégager la problématique qui sera abordée dans cette thèse. Ce chapitre situe l'importance de l'ingénierie du process et met l'accent, en les particularisant à son domaine, sur certaines difficultés classiques liées aux évolutions des systèmes d'information techniques d'entreprises (SIT). Ces évolutions des SIT sont, dès que leur taille le justifie, conduites en mode projet avec définition d'une maîtrise d'ouvrage (MOA), d'une maîtrise d'oeuvre (MOE), de jalons qui évaluent et planifient les engagements des différents acteurs.

Les besoins coté maîtrise d'oeuvre et maîtrise d'ouvrage des systèmes d'information nous semblent très pertinemment décrits, brochant un tableau somme toute complet des problèmes auxquels cette thèse s'attaque. L'objectif de ce travail d'apporter, au travers d'une formalisation de la création du cahier des charges commun MOA/MOE, des méthodes performantes au domaine de la conduite de projets SIT est très ambitieux, très pertinent et parfaitement inscrit dans l'actualité scientifique.

Le second chapitre est un état de l'art des outils et méthodes pour l'ingénierie des systèmes d'information techniques. Cette problématique conduit l'auteur à s'intéresser aux travaux portant sur les modèles métier d'entreprises manufacturières (tels STEP, EXPRESS, CIMOSA, GRAI,...). L'auteur puise ses informations dans la littérature scientifique, mais également des pratiques industrielles ou du conseil (MERISE par exemple), et des outils logiciels du marché. Elle s'appuie sur les résultats de quelques projets européens récents, tel que Rapid PDM, dont les objectifs sont en effet connexes à sa problématique dans le domaine des systèmes d'information dédiés aux produits. L'analyse du domaine de l'ingénierie de process se base très judicieusement sur MANDATE, travail de normalisation conduit dans le cadre ISO STEP. Enfin, ce chapitre se termine par une analyse des méthodes d'ingénierie de systèmes d'information et conclut par la nécessité de faire supporter ces méthodes par les AGL utilisés. Cet état de l'art nous semble parfaitement adapté à la problématique abordée. De plus, la rédaction de ce chapitre montre que l'auteur s'est approprié les travaux analysés et les a resitués dans le contexte décrit dans le premier chapitre.

Le chapitre 3 met en relation les besoins exprimés sur le terrain avec l'état de l'art pour proposer un plan d'action et un cahier des charges pour ce travail de thèse. Il débute par un parallèle entre

les méthodes d'ingénierie des produits et celles d'ingénierie des systèmes d'information. Ce parallèle a d'ailleurs conduit dans de multiples activités d'ingénierie (véhicule, satellite...) à adopter l'ingénierie système dont l'auteur aurait pu s'inspirer.

Face à l'étendue des données, processus et acteurs de l'ingénierie du process, l'auteur propose un premier outil d'analyse des périmètres d'étude basé sur les évolutions des projets "site de production" (de l'avant projet à la vie série), sur les métiers des process automobile (mécanique, emboutissage, tôlerie, montage), enfin sur des activités transverses (logistique, implantation, tolérancement, qualité...) présenté de façon synthétique sur la figure 3.2. Une limitation du périmètre de l'étude de cas qui validera la thèse, est proposée. L'auteur peut alors présenter les axes qui articuleront ses propositions :

- la prise en compte d'une démarche mixte itérative ascendante (bottom up) et descendante (top down) permettant de prendre en compte à la fois des volontés stratégiques, mais aussi des acquis de terrain ou des systèmes existants.
- la présentation du "cahier des charges commun MOA/MOE" sous forme de deux modèles, un business model représentant les processus et un data model définissant les objets métier.
- une méthode d'acquisition des connaissances permettant la réutilisation inter-projets pour la construction de ces deux modèles.

Ce chapitre permet de structurer très clairement les propositions de l'auteur qui forment un ensemble très cohérent ; nous identifions seulement un risque sur lequel nous reviendrons, celui de la maintenance de cohérence entre les "data model" et "business model" qui semble à ce stade reposer sur le savoir faire des modélisateurs.

Le quatrième chapitre constitue avec le suivant le coeur de l'apport scientifique de ce travail de thèse : ce chapitre propose en effet un référentiel du domaine de l'ingénierie de production. Cette contribution a été structurée en trois parties distinctes : la première présente la méthodologie de construction du référentiel, la seconde un modèle des processus d'entreprises (business model), enfin la dernière détaille le modèle des objets métiers (data model). La méthodologie s'appuie sur le cadre de Zachman, dont le lecteur perçoit ici tout l'intérêt, pour proposer un ensemble de modèles et des démarches de construction afin de viser au plus tôt une mise en accord maîtrise d'ouvrage, maîtrise d'oeuvre. Le business model identifie les processus de l'organisation et les rôles d'acteurs à même d'y intervenir : il peut prendre plusieurs formes (IDEF 0, IDEF 3... ARM de STEP), selon les pratiques métier et les AGL utilisés, mais nous retiendrons essentiellement les Use Cases, rôles et diagrammes de séquences inter-acteurs d'UML. Le "data model" est organisé autour de quatre méta entités : les produits, les process, les ressources et les systèmes industriels. Les liens établis entre ces entités répondent aux "questions" du cadre de Zachman. L'auteur utilise principalement les modèles de classes UML pour décrire les entités du data model.

Ce chapitre offre donc une définition très claire de la structure du cahier des charges proposé avec les "business" et "data" modèles. Un des grands intérêts de ces propositions est la prise en compte des démarches descendante et ascendante : en effet, cette dernière permet la réconciliation de processus élémentaires, mais surtout l'identification des données redondantes et la construction de modèles de données prenant en compte les existants dans de nombreuses BD locales. Les propositions de méta entités, en cohérence à la fois avec le terrain et avec la normalisation MANDATE, semblent pertinentes. Toutefois, je regrette que les liens entre business model et data model n'aient pas été plus approfondis, la maintenance des deux modèles ayant beaucoup à y gagner. L'approfondissement de ces liens passe par la définition au niveau de chaque classe d'objets des aspects dynamiques : les méthodes (ou services) offertes et les états-transitions possibles des objets ; ces services répondent alors aux requêtes exprimées au niveau des diagrammes de séquence du "business model". A titre d'exemple, le paragraphe V.2 (des cas d'utilisation métier aux objets potentiels) porte en germe des réponses à la question ouverte de conclusion "quel niveau de détail dans la modélisation ?" ; ce paragraphe aurait gagné à être mis plus en lumière, un peu au coeur de la démarche.

Le chapitre 5 présente une démarche d'usage du référentiel présenté précédemment. Cette démarche est heureusement et judicieusement illustrée par le cas test industriel qui a servi à valider ces propositions ; le lecteur se trouve ainsi confronté avec la complexité des données manipulées dans les métiers de l'ingénierie.

Ce chapitre permet d'éclairer plusieurs points remarquables de la thèse :

- le cas présenté s'appuie sur le logiciel de GED "Documentum" et permet donc de mettre en évidence la mise en oeuvre de la démarche avec une implantation dans un progiciel, aboutissement auquel peu de travaux de thèse parviennent, les propositions restant trop souvent "théoriques".
- il est symptomatique que le diagramme de la figure 5.5 présente un diagramme d'états-transitions d'objets répondant ainsi à l'une de nos remarques sur le chapitre 4 du mémoire. Les aspects dynamiques des objets sont bien pris en compte, mais restent à mon sens trop en filigrane.

Le chapitre de conclusion de la thèse met l'accent sur les contributions scientifiques de ce travail, référentiel du domaine et cadre méthodologique, souligne les facteurs clés de succès attendus de l'usage de la démarche : accélération et rationalisation du développement, indépendance par rapport à l'offre de progiciels, intégration des systèmes existants, approche combinée ascendante et descendante, capitalisation itérative au sein de l'AGL utilisé, appui sur le cadre de Zachman.... La conclusion dégage quelques perspectives, entre autres normatives, que nous encourageons l'auteur à poursuivre. Comme souvent dans cette nature de travaux de recherche, le problème de la validation des propositions qui se veulent génériques est quelque peu occulté ; seule la mise en oeuvre sur un cas valide les propositions, il y a là pour un chercheur matière à réflexion.

Outre les aspects dynamiques des objets et le lien entre business et data models, le problème de la tenue à jour des référentiels de connaissance à des fins de réutilisation mériterait à lui seul un nouveau travail de thèse.

En conclusion, Mademoiselle Rebiha Bacha présente un mémoire remarquablement riche et structuré ; le souci de clarifier des connaissances confuses et diffuses, de se nourrir d'un ensemble large de références bibliographiques, d'organiser ces éléments dans une démarche pragmatique permettant de conduire des projets SIT, contribue à un résultat de thèse dont nous pensons qu'il fera date dans la littérature scientifique du domaine. Ce travail, qui devrait faire référence, constitue un apport tout à fait majeur au domaine des systèmes d'information techniques et justifie la tenue d'une soutenance en vue de l'obtention du doctorat de l'École Centrale de Paris, spécialité Productique - Génie industriel.

Fait à Grenoble, le 28 décembre 2001

Michel Tollenaere

Professeur INPG

Rapport de Jean-Pierre Bourey en qualité de rapporteur

Le travail présenté par Mademoiselle Rebiha BACHA a été effectué au sein du Laboratoire Productique-Logistique de l'École Centrale de Paris et de la Direction de la Recherche du Technocentre Renault. Le mémoire est intitulé "De la gestion des données techniques en ingénierie de production. Référentiel du domaine et cadre méthodologique pour l'ingénierie des systèmes d'information techniques en entreprise".

Le thème principal du travail concerne les problèmes méthodologiques rencontrés dans les projets de mise en oeuvre des systèmes d'information techniques en entreprise. La contribution principale se situe dans la proposition d'un référentiel et d'une démarche d'élaboration de ce référentiel.

Le mémoire est constitué de cinq chapitres principaux, d'une bibliographie et de trois annexes.

Dans le premier chapitre, Mademoiselle BACHA présente la problématique industrielle à l'origine de son travail de recherche à partir d'une étude de terrain menée chez Renault. Après avoir dressé un bilan des difficultés liées à la gestion des données techniques et de leurs conséquences dans cette entreprise, les besoins des principaux acteurs (maîtrise d'oeuvre et maîtrise d'ouvrage) sont mis en évidence ainsi que les principaux goulets d'étranglement de la mise en oeuvre des systèmes d'information techniques.

Dans cette partie, Mademoiselle BACHA montre très clairement sa capacité à dégager une problématique de recherche à partir d'une étude de terrain.

Le second chapitre est consacré à l'état de l'art des méthodes, formalismes et outils susceptibles d'entrer dans le cadre de la problématique présentée dans le chapitre précédent. Mademoiselle BACHA s'intéresse d'abord aux technologies de l'information et supports d'intégration puis aux techniques et outils de modélisation. Elle souligne, à juste titre, que s'il existe de nombreux travaux dans le domaine de la structuration de données techniques, la littérature est beaucoup moins abondante sur les aspects méthodologiques liés à l'ingénierie des systèmes d'information techniques en entreprise.

Ce chapitre montre sans équivoque la culture pluridisciplinaire de Mademoiselle BACHA et son souci de recherche de la solution la plus adaptée à ses besoins. Chaque élément est abordé du point de vue de ses apports et de ses limites par rapport à l'objectif visé. L'étude bibliographique est représentative du domaine et un argumentaire pertinent est développé de manière rigoureuse.

A partir du troisième chapitre, on entre véritablement dans le coeur de la contribution de Mademoiselle BACHA. Cette contribution s'organise autour, d'une part, du cahier des charges de la maîtrise d'oeuvre-d'ouvrage appelé "référentiel du domaine" et, d'autre part, d'une démarche pour la conduite des projets de mise en oeuvre des systèmes d'information techniques. Les différents choix effectués (techniques de modélisation de données, de processus et orientées objet) sont présentés et correctement justifiés. Ils sont placés dans le cadre de Zachman permettant ainsi de les positionner par rapport aux modèles "classiques" de la littérature.

Sur le plan de la démarche, Mademoiselle BACHA décrit la méthodologie préconisée chez Renault en précisant dans quelle mesure elle ne peut être retenue pour l'étude. La démarche qui est alors préconisée, de manière très pragmatique, s'appuie sur une double approche : une approche descendante pour l'intégration de nouveaux besoins et une approche ascendante pour l'intégration des systèmes existants.

Le quatrième chapitre est consacré à la construction du référentiel. Les éléments nécessaires à la construction, la démarche d'élaboration du référentiel ainsi que ses deux composants (le "Business Model" et le "Data Model") y sont décrits. Le cadre de Zachman est utilisé à la fois pour positionner les deux modèles constituant le référentiel mais également pour préciser les trois cadres (produit, systèmes d'information techniques et entreprise) qui concernent le travail présenté dans le

mémoire. Si les deux composants du référentiel sont décrits de manière très claire et très détaillée, il est cependant regrettable que la partie consacrée au scénario de construction du référentiel, qui constitue, de mon point de vue, l'une des parties les plus originales du mémoire, ne soit pas plus largement développée.

Le dernier chapitre est consacré à une mise en application des propositions de Mademoiselle BACHA. Le contexte d'application est bien décrit ainsi que les différentes phases de mise en oeuvre. Il est ainsi possible de voir concrètement la portée et l'impact de la contribution de ce travail de recherche.

De mon point de vue, la contribution de Mademoiselle BACHA est tout à fait significative et représente un très bon exemple de ce qu'est un travail de recherche en Génie Industriel. En effet, à partir d'une problématique industrielle concrète, elle a réussi à prendre du recul, à dégager un axe de recherche clair et précis et à apporter une contribution méthodologique importante débouchant sur une mise en oeuvre ayant, de toute évidence, un impact direct sur la performance de l'entreprise.

Sur le plan de la forme, l'ensemble du mémoire, dont la structure est bien équilibrée, est rédigé de façon claire, pédagogique et rigoureuse. Le mémoire est abondamment illustré et les différents encarts permettant d'avoir une lecture à différents niveaux de détails.

La conclusion, je pense que le travail présenté par Mademoiselle BACHA est de très bonne qualité et je donne un avis favorable à la soutenance en vue de l'obtention du grade de Docteur de l'Ecole Centrale de Paris.

Villeneuve d'Ascq, le 08 décembre 2001

J.P. Bourey
Professeur des universités à l'Ecole Centrale de Lille

Rapport de François Vernadat en qualité de président du jury.

Mlle BACHA Rebiha a su clairement exposer son travail de thèse sur un rythme soutenu mais néanmoins dans un style agréable alliant maîtrise, aisance et rigueur. L'exposé, sur un sujet novateur, difficile, vaste, aux facettes multiples, dans un domaine pertinent mais jugé risqué au départ par les parties prenantes, a montré qu'elle a su défricher, structurer et synthétiser ce thème du génie industriel avec succès et avec l'enthousiasme qui lui est propre.

Ses réponses détaillées, empruntées d'une grande honnêteté intellectuelle, aux questions nombreuses et variées du jury ont démontré le recul pris par Mlle BACHA par rapport à son sujet, l'applicabilité avérée de ses résultats sur le plan industriel et leur apport potentiel pour les normes en productique. En conséquence, le jury a été unanime pour lui décerner le titre de docteur avec la mention très honorable avec les félicitations du jury.

Symboles et abréviations

Note : liste des symboles et abréviations fréquemment utilisés dans le manuscrit.

AGL	Atelier de Génie Logiciel – traduction de CASE (Computer-Aided Software Engineering)
BE	Bureau d'Etudes
BdC*	Bord de Chaîne
BdD	Base de Données
BPR	Business Process Reengineering
CdC	Cahier des Charges
CIM	Computer Integrated Manufacturing
CRUD	Create Read Update Delete
E-A/E-R	Entité-Association / Entité-Relation
GDT	Gestion des Données Techniques
GED	Gestion Electronique de Documents
I PAO	Ingénierie de Process ou de Production Assistée par Ordinateur - traduction littérale de CAPE (Computer-Aided Process Engineering)-
MANDATE	MANufacturing DATa Exchange
MOA*	Maîtrise d'OuvrAge
MOE*	Maîtrise d'Œuvre
MTC*	Mise en Titre de Critères
OO	Orienté-Objet
PSL	Process Specification Language
SDLC	System Development Life Cycle
SGBD	Système de Gestion des Bases de Données
SGDT	Système de Gestion des Données Techniques - traduction de PDM (Product Data Management) ou plus récemment de cPDM (collaborative Product Definition management)
SIGNE*	Système d'Information pour la Gestion des Nomenclatures d'Entreprise
SI/SIT	Système d'Information / Système d'Information Technique
SPR*	Système de Production Renault
TIC	Technologie d'Information et de Communication - traduction de IT (Information Technology)
UML	Unified Modeling Language

* propre au jargon Renault

TABLE DES MATIERES

La table est le seul endroit où l'on ne s'ennuie jamais pendant la première heure.

Anthelme BRILLAT-SAVARIN (1755-1826). Physiologie du goût

Table des matières

REMERCIEMENTS	
LISTE DES SYMBOLES ET ABREVIATIONS	
SOMMAIRE	
FIGURES, TABLEAUX ET ENCARTS	

INTRODUCTION GENERALE **1**

Chapitre premier - Regard sur les systèmes d'informations techniques en entreprise. Problématique industrielle

I RAISONS D'APPARITION ET D'EVOLUTION DE LA GDT EN ENTREPRISE ... 5

II D'ABORD QUELQUES CONSTATS..... 7

II.1	DE L'INGENIERIE DU PRODUIT A L'INGENIERIE DU PROCESS.....	7
II.2	CONCEPTION VERTICALE DES OUTILS D'IPAO.....	9
II.3	MORCELLEMENT DU LONG PROCESSUS D'INDUSTRIALISATION DU VEHICULE.....	11
II.4	LES CONSEQUENCES SUR LA GDT.....	12

III BESOINS DE GDT EN ENTREPRISE : SPECIFICATIONS INTERNES ET EXTERNES..... 17

III.1	SPECIFICATIONS EXTERNES : BESOINS DE LA MAITRISE D'OUVRAGE.....	18
III.2	SPECIFICATIONS INTERNES : BESOINS DE LA MAITRISE D'ŒUVRE.....	20

IV GOULOTS D'ETRANGLEMENT A LA MISE EN ŒUVRE DES SIT EN ENTREPRISE..... 22

IV.1	CdC DES BESOINS EN GDT : MODELES D'EXPRESSION INCOMPLETS ET INFORMELS.....	22
IV.2	MODE D'EMPLOI : ABSENT OU GENERATEUR DE RUPTURES.....	23
IV.3	"BUSINESS REQUIREMENT" : DIFFICULTE D'APPREHENDER SA DYNAMIQUE.....	23
IV.4	JEUNESSE INHERENTE DU DOMAINE DE LA GDT PROCESS.....	24

Chapitre deuxième - Quels outils et quelles méthodes pour l'ingénierie des systèmes d'information techniques en entreprise. Etat de l'art

I NOUVELLE PHASE DE RUPTURE POUR LES SI EN ENTREPRISE 25

I.1	INTRODUCTION.....	25
I.2	CE QUI CARACTERISE LES SYSTEMES D'INFORMATION TECHNIQUES DES AUTRES SI.....	26

II TECHNOLOGIES D'INFORMATION ET SUPPORTS D'INTEGRATION 28

II.1	LES SGDT ET LES FONCTIONS INTEGREES AUX SGDT.....	28
II.2	TRAVAUX DE RECHERCHE AUTOUR DES FONCTIONS CLASSIQUES DE SGDT.....	30
II.3	ACTIONS NORMATIVES ET AUTRES TECHNOLOGIES D'INTEGRATION.....	31
II.4	SYNTHESE : APPORT DES TECHNOLOGIES D'INTEGRATION.....	33

III	TECHNIQUES ET OUTILS DE MODELISATION.....	34
III.1	MODELES ET LANGAGES DE SPECIFICATION.....	35
III.2	ARCHITECTURES DES SI ET CADRES D'ENTREPRISE.....	43
III.3	OUTILLAGE DES TECHNIQUES : LES ATELIERS DE GENIE LOGICIEL	48
III.4	BILAN ET ANALYSE	51
IV	METHODES D'INGENIERIE DES SI ET DEVELOPPEMENT DES SIT EN ENTREPRISE.....	53
IV.1	METHODOLOGIES EN INGENIERIE DES SI	54
IV.2	APPORTS ET LIMITES DES METHODES EN INGENIERIE DES SI	58
IV.3	DEVELOPPEMENT DES SYSTEMES D'INFORMATION TECHNIQUES EN ENTREPRISE	60
IV.4	CE QUE NOUS DEVONS RETENIR DES TRAVAUX LIES AUX DEVELOPPEMENTS DES SIT.....	65
V	CONCLUSION	67

Chapitre troisième - Orientations de la thèse et plan d'actions

I	INTRODUCTION.....	69
I.1	GRANDES ORIENTATIONS DE NOTRE THESE	69
I.2	INGENIERIE VEHICULE – INGENIERIE SI : ELEMENTS DE COMPARAISON	69
II	PERIMETRE DE L'INGENIERIE DE PRODUCTION.....	70
II.1	JALONNEMENT DU PROJET	70
II.2	METIER PROCESS.....	71
II.3	ACTIVITES TRANSVERSALES	71
III	CHOIX DE LA TECHNOLOGIE D'INTEGRATION.....	72
IV	QUELS MODELES DE SPECIFICATION ?.....	73
IV.1	TECHNIQUES DE MODELISATION DE DONNEES	73
IV.2	TECHNIQUES DE MODELISATION DES PROCESSUS (BPM).....	73
IV.3	UML ET PATRONS D'ANALYSE ET DE CONCEPTION	74
V	CADRE DE ZACHMAN : DIFFERENTS USAGES	76
VI	L'ATELIER LOGICIEL : SYSTEM ARCHITECT DE POPKIN SOFTWARE	77
VII	QUELLES APPROCHES METHODOLOGIQUES ?	79
VII.1	METHODOLOGIE PRECONISEE EN ENTREPRISE : POINTS FORTS ET POINTS DURS.....	80
VII.2	DEMARCHE INTEGREE : STRATEGIE ASCENDANTE ET STRATEGIE DESCENDANTE	82
VII.3	BUSINESS ET DATA MODEL : DEUX MODELES EN INTERACTIONS PERMANENTES	82
VII.4	INGENIERIE DE DOMAINE : SOURCES D'ACQUISITION DES CONNAISSANCES	84
VII.5	APPROCHE CONTEXTUELLE : TRIPTYQUE CLE (OBJECTIF, PRODUIT, CONTRAINTE)	85
VIII	SYNTHESE.....	87

Chapitre quatrième - Mise en oeuvre du référentiel du domaine.

SECTION 1 – CONSTRUCTION DU REFERENTIEL : INGREDIENTS ET MODE D'EMPLOI.....	89
I CADRE DE ZACHMAN POUR GERER LES CONNAISSANCES D'ENTREPRISE	89
I.1 CADRES DU SYSTEME : INTERDEPENDANCES ET RECURSIVITE LOGIQUE	89
I.2 PERIMETRE DU REFERENTIEL : DIMENSIONS ET PERSPECTIVES	93
II METHODOLOGIE DE CONSTRUCTION DU REFERENTIEL	94
II.1 BUSINESS ET DATA MODEL : QUELQUES CONCEPTS HOMONYMES.....	94
II.2 PRINCIPAUX JALONS D'ELABORATION DU REFERENTIEL : UN SCENARIO DE CONCEPTION	95
SECTION 2 – LE BUSINESS MODEL	98
III DECOUVERTE DU DOMAINE : MODELES CONTEXTUELS	98
III.1 VISION STRATEGIQUE	99
III.2 DIMENSION TEMPORELLE	100
III.3 VUE ORGANISATIONNELLE	100
III.4 DECOMPOSITION DU PROCESSUS METIER (BUSINESS PROCESS)	101
IV MODELES OPERATIONNELS	103
IV.1 CONSTRUCTION DES MODELES DE PROCESSUS	103
IV.2 DOCUMENTATION DES PROCESSUS ELEMENTAIRES.....	106
V MAPPING VERS LES DONNEES : MODELES DE TRANSFORMATION	108
V.1 DES PROCESSUS AUX CAS D'UTILISATION METIER.....	109
V.2 DES CAS D'UTILISATION METIER AUX OBJETS POTENTIELS	112
SECTION 3 – LE DATA MODEL	114
VI DIAGNOSTIC DES DONNEES	114
VI.1 RETROINGENIERIE : ANALYSE DES SIMILARITES ET RECHERCHE D'ANALOGIES	114
VI.2 META ENTITES ET CLASSIFICATION DES OBJETS METIER.....	123
VII STRUCTURATION ET RAFFINEMENT DES OBJETS METIER	127
VII.1 ORGANISATION EN PAQUETAGES.....	128
VII.2 DES OBJETS POTENTIELS AUX CLASSES D'OBJETS PERSISTANTES	129
VII.3 LES NOMENCLATURES	131
VII.4 LIENS ENTRE NOMENCLATURES.....	133
VII.5 VUES DES OBJETS METIERS.....	135
VIII CONCLUSION ET OUVERTURES	138

Chapitre cinquième - Usages du référentiel du domaine

I	CONTEXTE DU CAS INDUSTRIEL.....	141
I.1	INTENTION ET MOTIVATION : NOUVEAU SYSTEME DOCUMENTAIRE.....	141
I.2	PRODUIT FINAL ATTENDU : APPLICATION INFORMATIQUE DE GRANDE TAILLE.....	142
I.3	CONTRAINTES ENVIRONNEMENTALES : INTEGRATION DES SIT EXISTANTS.....	143
II	ADAPTATION DU REFERENTIEL ET CDC DU PROJET	144
II.1	DES PAQUETAGES AUX COMPOSANTS : ARCHITECTURE GENERALE DES DONNEES.....	144
II.2	TRAITEMENT DES PROCEDURES ET DES CAS D'UTILISATION METIER.....	145
II.3	STRUCTURE DES DONNEES : PRODUIT, PROCESS, RESSOURCE ET SYSTEME INDUSTRIEL	147
II.4	MISE EN ŒUVRE DE L'APPLICATION.....	147
III	FONCTIONNALITES ACQUISES.....	150
III.1	STRUCTURE D'ACCUEIL CONVIVIALE : VISION MULTI-VUE.....	150
III.2	GESTION CONJOINTE DES DOCUMENTS ET DES DONNEES	152
III.3	TRAÇABILITE ET GESTION DES CONFIGURATIONS DU DOCUMENT	154
III.4	LE WORKFLOW ET AUTOMATISATION DES PROCEDURES METIER.....	155
III.5	REQUETES DE RECHERCHE ET EDITION DE RAPPORTS	156
IV	CONCLUSION ET OUVERTURES DE L'APPLICATION.....	158
	CONCLUSION GENERALE	161
	REFERENCES BIBLIOGRAPHIQUES	165
	ANNEXE A – GLOSSAIRE	
	ANNEXE B – PROBLEMATIQUES EN INGENIERIE DE PRODUCTION	
	ANNEXE C – ELEMENTS SUR L'EVALUATION DES AGL	

FIGURES, TABLEAUX ET ENCARTS

Il ne faut point s'étonner si les figures, quand elles sont employées à propos, donnent de la vivacité, de la force, ou de la grâce au discours ; car outre la propriété d'exprimer les pensées, comme tous les autres assemblages de mots, elles ont encore, si j'ose parler ainsi, l'avantage de leur habit, je veux dire de leur modification particulière, qui sert à réveiller l'attention, à plaire ou à toucher.

César CHESNEAU, SIEUR DU MARSAIS (1676-1756). Des tropes ou des différents sens dans lesquels on peut prendre un même mot dans un même langage. Première partie

Encarts, Figures et Tableaux

Chapitre premier

ENCART 1.1 APPELLATIONS DES SGDT : DU PDM AU cPDM	5
ENCART 1.2 PARADIGME DES SYSTEMES D'INFORMATION TECHNIQUES (SIT)	6
ENCART 1.3 STRATEGIE DE DEVELOPPEMENT DES OUTILS D'IPAO	11
ENCART 1.4 MISSIONS DE LA MAITRISE D'OUVRAGE ET DE LA MAITRISE D'ŒUVRE (SOURCE RENAULT).....	17
FIGURE 1.1 POSITION ET FONCTIONNALITES DES SGDT EN ENTREPRISE.....	6
FIGURE 1.2 CROISSANCE DE L'OFFRE ET DE L'OFFREUR EN PROGICIELS DE SGDT (SOURCE CIMDATA [CIM01]).....	7
FIGURE 1.3 POSITION CHRONOLOGIQUE DES OUTILS D'INGENIERIE ET VOLUME DES DONNEES	8
FIGURE 1.4 CONCEPTION VERTICALE DE L'IPAO : NOTION DE "ISLANDS OF AUTOMATION"	10
FIGURE 1.5 ARCHITECTURE DES OUTILS EN MONTAGE : CONCEPTION SEQUENTIELLE DU PROCESS.....	11
FIGURE 1.6 CHAINE D'INDUSTRIALISATION DU VEHICULE : METIERS PRINCIPAUX (SOURCE RENAULT).....	13
FIGURE 1.7 SIGNE : APPLICATION DE GDT D'ENTREPRISE	15
FIGURE 1.8 ETAT ACTUEL DES SIT EN INGENIERIE DE PRODUCTION	16
FIGURE 1.9 CLASSIFICATION ET REPRESENTATION DES DONNEES	21
FIGURE 1.10 CONCEPTION BOTTOM-UP ET TOP-DOWN DU PROCESSUS D'INDUSTRIALISATION DU VEHICULE	22
TABLEAU 1.1 CONCEPTION DU PRODUIT ET CONCEPTION DU PROCESS : ELEMENTS DE COMPARAISON.....	9
TABLEAU 1.2 SYNTHESE DES BESOINS EN GDT ET LES COMPOSANTS SGDT CORRESPONDANTS.....	22

Chapitre deuxième

ENCART 2.1 ORIENTATION ET PLAN DU CHAPITRE : INTERDEPENDANCE DES THEMATIQUES.....	27
ENCART 2.2 QUELQUES TRAVAUX DANS LE DOMAINE DE LA GESTION DE CONFIGURATIONS DU PRODUIT	30
ENCART 2.3 EXEMPLES DE TRAVAUX DANS LE DOMAINE DES WORKFLOW	31
ENCART 2.4 PRINCIPES ET TERMES CLES DANS LA MODELISATION DES PROCESSUS	39
ENCART 2.5 DEFINITIONS DE BASE DU DOMAINE DE L'OBJET.....	42
ENCART 2.6 DAVANTAGE DE LECTURES SUR LE CADRE DE ZACHMAN	46
ENCART 2.7 BESOIN DE METHODOLOGIES POUR LES METIERS D'INGENIERIE.....	53
ENCART 2.8 INTERROGATIONS EPISTEMOLOGIQUES AUTOUR DES METHODOLOGIES	53
ENCART 2.9 ON PARLE DE JUNGLE DE METHODES.....	55
ENCART 2.10 PANORAMA DE QUELQUES METHODES D'INGENIERIE DES SI	56
ENCART 2.11 INSTRUMENTATION DES METHODOLOGIES DANS LES AGL.....	67
FIGURE 2.1 LE TRIANGLE STRATEGIQUE : POSITION DU SI EN ORGANISATION (SOURCE [TAR91]).....	25
FIGURE 2.2 ITRM : MODELE DE REFERENCE DE L'AIT-IP (SOURCE [AIT98] OU WWW.AIT.ORG.UK)	32
FIGURE 2.3 LES CLEFS DE LA "DESIGN CHAIN" (ADAPTEE DE : SIMULOG/ESPRIT CONCEPT, WWW.SIMULOG.FR)	34
FIGURE 2.4 CONTEXTE DE DEVELOPPEMENT D'UN SI EN ENTREPRISE. DIAGRAMME "RICH PICTURES" (A GAUCHE)	36
FIGURE 2.5 CONTEXTE DE DEVELOPPEMENT D'UN SI EN ENTREPRISE. TECHNIQUE "ROOT DEFINITION" (A DROITE) ...	36
FIGURE 2.6 PROCESSUS DE VALIDATION D'UNE SPECIFICATION D'UN SI. DIAGRAMME "CONCEPTUAL MODEL".....	36
FIGURE 2.7 TECHNIQUES DE MODELISATION DES DONNEES.....	38
FIGURE 2.8 PATRON DE DOCUMENTATION D'UNE ENTITE (ADAPTEE DE [AVI00]).....	39
FIGURE 2.9 MODELE DE PROCESSUS CLASSIQUE	39
FIGURE 2.10 CONCEPTION PRODUIT/PROCESS EN ENTREPRISE MANUFACTURIERE. DIAGRAMME DFD.....	40
FIGURE 2.11 ALTERNATIVES DE DECISION POUR L'AFFECTATION D'UNE OPERATION AU POSTE.	40
FIGURE 2.12 DONNEE-APPLICATION : MATRICE CRUD (CREATE READ UPDATE DELETE).....	41
FIGURE 2.13 CYCLE DE VIE DE L'ENTITE "OPERATION"	41
FIGURE 2.14 LES DIAGRAMMES UML	42

FIGURE 2.15 CUBE CIMOSA (SOURCE : HTTP://CIMOSA.CNT.PL A GAUCHE)	45
FIGURE 2.16 MAISON ARIS (SOURCE : WWW.IDS-SCHEER.DE A DROITE)	45
FIGURE 2.17 CADRE DE ZACHMAN (SOURCE : WWW.ZIFA.COM)	47
FIGURE 2.18 ARCHITECTURE D'UN CASE –AGL- (ADAPTEE DE [WHI01])	49
FIGURE 2.19 ORGANISATION DE L'ENCYCLOPEDIE, NOYAU DES AGL : MODELE OBJET SIMPLIFIE.....	49
FIGURE 2.20 EVOLUTION DES METHODES EN INGENIERIE DES SYSTEMES D'INFORMATION (SOURCE [GZA00]).....	55
FIGURE 2.21 LE MODELE MDM A L'AIDE DU DIAGRAMME DE CLASSES UML	62
FIGURE 2.22 STRUCTURE GENERALE DU STANDARD MANDATE	63
FIGURE 2.23 PROCESSUS POUR ET PAR LA REUTILISATION (ADAPTEE DE [GZA00])	65
TABLEAU 2.1 QUELQUES AUTRES TECHNOLOGIES D'INTEGRATION (DIVERSES SOURCES).....	33
TABLEAU 2.2 CLASSIFICATION DES TECHNIQUES (SOURCE [AVI00])	42
TABLEAU 2.3 COUVERTURE DES METHODOLOGIES D'INGENIERIE DES SI (SOURCE [AVI00])	58

Chapitre troisième

ENCART 3.1 EXEMPLES DE PATRONS QUE NOUS AVONS ADAPTES	75
ENCART 3.2 COMMENT REDUIRE LA COMPLEXITE DES DIAGRAMMES (MODELES DE SPECIFICATION).....	86
ENCART 3.3 DIFFERENTS ASPECTS DE REUTILISATION EN INGENIERIE DES SIT.....	87
FIGURE 3.1 INGENIERIE VEHICULE – INGENIERIE SI : ELEMENTS DE COMPARAISON	70
FIGURE 3.2 PERIMETRE D'ETUDE : INGENIERIE DE PRODUCTION	71
FIGURE 3.3 TECHNOLOGIE D'IMPLEMENTATION DU SIT.....	73
FIGURE 3.4 CADRE DE ZACHMAN : COUVERTURE DES MODELES RETENUS	77
FIGURE 3.5 ENVIRONNEMENT UTILISATEUR DE SYSTEM ARCHITECT DE POPKIN SOFTWARE	78
FIGURE 3.6 NOTATIONS ET TECHNIQUES SUPPORTEES PAR L'AGL SYSTEM ARCHITECT DE POPKIN SOFTWARE	79
FIGURE 3.7 CYCLE DE VIE DES PROJETS INFORMATIQUES CHEZ RENAULT (SOURCE RENAULT)	81
FIGURE 3.8 NOTRE DEMARCHE INTEGREE : ASCENDANTE ET DESCENDANTE (A GAUCHE)	83
FIGURE 3.9 BUSINESS MODEL ET DATA MODEL : REFERENTIEL DU DOMAINE (A DROITE)	83
FIGURE 3.10 INGENIERIE DE DOMAINE : SOURCES D'ACQUISITION DES CONNAISSANCES	85

Chapitre quatrième

ENCART 4.1 ELEMENTS DE MODELISATION DES DONNEES - QUELQUES COMPARAISONS AVEC UML	115
ENCART 4.2 ARCHITECTURE DES SIT ET DUPLICATION DES OBJETS COMMUNS	122
FIGURE 4.1 INTERACTIONS : CADRE DE PRODUIT – CADRE D'ENTREPRISE	92
FIGURE 4.2 LIENS POTENTIELS : CADRE DE SI – CADRE D'ENTREPRISE	92
FIGURE 4.3 SCENARIO D'ELABORATION DU REFERENTIEL ET D'UN NOUVEAU SIT EN ENTREPRISE	96
FIGURE 4.4 OBJECTIFS STRATEGIQUES ET NOUVEAUX REQUIREMENTS DE L'ENTREPRISE	99
FIGURE 4.5 MATRICE "GROUPE DE TRAVAIL/ROLE" – ASPECT DE LA DIMENSION ORGANISATIONNELLE	101
FIGURE 4.6 DECOMPOSITION DES PROCESSUS METIER :	102
FIGURE 4.7 ACTIVITE DE PREPARATION EN MONTAGE : CONCEPTION EN BOTTOM-UP.....	104
FIGURE 4.8 PROCESSUS DE DEFINITION DES LIAISONS EN TOLERIE : MULTIPLICITE DES BDD	104
FIGURE 4.9 CONNECTIONS "RESULTAT - EVENEMENT" : RECOLLER LES PROCESSUS MORCELES.....	105
FIGURE 4.10 L'INGENIERIE PROCESS ET SES PRINCIPAUX LIVRABLES	105
FIGURE 4.11 DOCUMENTATION UNIFORME DES PROCESSUS ELEMENTAIRES (1/2)	106
FIGURE 4.12 DOCUMENTATION UNIFORME DES PROCESSUS ELEMENTAIRES (2/2)	107
FIGURE 4.13 REDUCTION DE LA COMPLEXITE DES DIAGRAMMES.....	107
FIGURE 4.14 QUELQUES APPLICATIONS ET FLUX EN INGENIERIE DE MONTAGE	108
FIGURE 4.15 DES ROLES AUX ACTEURS : HIERARCHISATION DES ACTEURS DU METIER	110
FIGURE 4.16 LES TYPES DE LIENS ENTRE CAS D'UTILISATION.....	110
FIGURE 4.17 CONCEPTION DU BORD DE CHAINE. DIAGRAMME DE CAS D'UTILISATION.....	111

FIGURE 4.18 SCENARIO DU CAS D'UTILISATION "DEFINIR UN PICKING" .	112
FIGURE 4.19 TYPOLOGIE DES OBJETS (SOURCE [KET98])	113
FIGURE 4.20 CHEMINS POUR ACCEDER AU DIAGRAMME DE CLASSES	114
FIGURE 4.21 FRAGMENT DE MODELE LOGIQUE DE DONNEES	117
FIGURE 4.22 TABLES ET LIENS DANS UN MODELE PHYSIQUE DE DONNEE	117
FIGURE 4.23 IMPORT DES BASES DE DONNEES EXISTANTES	118
FIGURE 4.24 ANALYSE DES SIMILARITES : GENERALISATION DE CANDIDATS REUTILISABLES.	121
FIGURE 4.25 CORRESPONDANCES ANALOGIQUES : TRAITEMENT DE LA DIVERSITE PRODUIT/PROCESS	121
FIGURE 4.26 ARCHITECTURE ACTUELLE DES SIT EN ENTREPRISE	123
FIGURE 4.27 LES QUATRE META ENTITES DU DATA MODEL : LIENS CENTRES PROCESS	124
FIGURE 4.28 DIAGRAMME INITIAL DES PAQUETAGES	129
FIGURE 4.29 ASSEMBLE DE PIECES VU PAR LE PREPARATEUR TOLERIE. DIAGRAMME INITIAL DE CLASSES.	130
FIGURE 4.30 CYCLE DE VIE DE LA CLASSE "ASSEMBLE" EN INGENIERIE. DIAGRAMME DES ETATS	131
FIGURE 4.31 COMPOSITION FONCTIONNELLE ET STRUCTURELLE DU PRODUIT	132
FIGURE 4.32 GRAPHE DU PROCESS	132
FIGURE 4.33 NOMENCLATURE DE RESSOURCES	133
FIGURE 4.34 DECOUPAGE STRUCTUREL DU SYSTEME INDUSTRIEL	134
FIGURE 4.35 LIENS ENTRE ELEMENTS DE NOMENCLATURES : VUE MICROSCOPIQUE	134
FIGURE 4.36 ROLES DE LA RESSOURCE EN INGENIERIE DE PRODUCTION.	135
FIGURE 4.37 REPARTITION DES SURFACES : UNE VUE DU SYSTEME INDUSTRIEL.	136
FIGURE 4.38 STRUCTURE GENERIQUE DU DOCUMENT	137
TABLEAU 4.1 CADRE DU PRODUIT : DIFFERENTES REPRESENTATIONS DU VEHICULE INDUSTRIEL	90
TABLEAU 4.2 REFERENTIEL DE DOMAINE : VUES ET NIVEAUX D'ABSTRACTION	94
TABLEAU 4.3 BUSINESS MODEL ET DATA MODEL - CONCEPTS HOMONYMES	95
TABLEAU 4.4 PRINCIPAUX JALONS SIGNIFICATIFS EN ENTREPRISE (SOURCE RENAULT)	100
TABLEAU 4.5 DESCRIPTION TEXTUELLE DU CAS D'UTILISATION "DEFINIR DES PICKINGS"	111
TABLEAU 4.6 ANALYSE DES OBJETS CANDIDATS : EXEMPLES.	113
TABLEAU 4.7 QUELQUES EQUIVALENCES : E-R ET UML	118

Chapitre cinquième

FIGURE 5.1 NOUVEAU SYSTEME DOCUMENTAIRE A CONCEVOIR	142
FIGURE 5.2 CONTENU INFORMATIONNEL DU DOCUMENT FOP A	143
FIGURE 5.3 ARCHITECTURE GENERALE DES DONNEES. DIAGRAMME DE COMPOSANTS	145
FIGURE 5.4 CONTROLE D'ACCES ET DE MODIFICATIONS DES UTILISATEURS. MATRICE CRUD	146
FIGURE 5.5 CYCLE DE VIE DU DOCUMENT FOP A. DIAGRAMME DES ETATS	146
FIGURE 5.6 FRAGMENT DU MODELE FEDERATEUR DE DONNEES.	147
FIGURE 5.7 LES ROLES DANS LE PROJET DE DEVELOPPEMENT DE L'APPLICATION DE GED.	148
FIGURE 5.8 PHASES DE MISE EN ŒUVRE DE L'APPLICATION DE GED.	149
FIGURE 5.9 INFRASTRUCTURE DE L'APPLICATION DE GED. DIAGRAMME DE DEPLOIEMENT	150
FIGURE 5.10 ENVIRONNEMENT UTILISATEUR DU PROGICIEL DOCUMENTUM.	151
FIGURE 5.11 STRUCTURE D'ACCUEIL DE LA GED : MULTIPLES VUES	151
FIGURE 5.12 FORMULAIRE DE SAISIE DES CHAMPS DU DOCUMENT	152
FIGURE 5.13 STOCKAGE UNIQUE DU DOCUMENT ET MULTIPLICITE D'ACCES	153
FIGURE 5.14 DOCUMENT CREE DEPUIS LE FORMULAIRE DE SAISIE	154
FIGURE 5.15 GESTION DES CONFIGURATIONS DU DOCUMENT : VERSIONNEMENT	155
FIGURE 5.16 DIFFUSION DU DOCUMENT SUIVANT LES PROCEDURES METIER	156
FIGURE 5.17 PROCEDURE D'APPROBATION D'UN DOCUMENT	156
FIGURE 5.18 REQUETE DE RECHERCHE STANDARD DU PROGICIEL	157
FIGURE 5.19 EXEMPLE DE REQUETE DE RECHERCHE PERSONNALISEE	157

INTRODUCTION GENERALE

Contexte, problématique et plan du mémoire

Une longue marche commence toujours par un petit pas.

Mao TSE-TOUNG (1893-1976)

Introduction générale

La Gestion des Données Techniques (GDT) ouvre un champ considérable d'activités de recherche et de développement, impliquant des domaines pluridisciplinaires tels :

- L'ingénierie des Systèmes d'Information (SI) et des Systèmes d'Information Techniques (SIT) au travers des outils et des méthodes de développement de ces systèmes.
- Le génie industriel et la productique par la capitalisation des expériences industrielles et des connaissances liées à l'ingénierie des produits manufacturiers.
- L'ingénierie du besoin, l'ingénierie des connaissances, les bases de données etc.

A ces domaines d'ingénierie, correspondent divers points de vue avec lesquels ils sont appréhendés : organisationnel et humain, méthodologique, technique, cognitif...

En effet, la GDT est ambitieuse car elle entend couvrir le cycle de vie tant des données que des documents, de la création à l'archivage en passant par la diffusion. Elle couvre la gestion des configurations, la gestion des articles et des nomenclatures, la gestion du travail collaboratif...

Faut-il souligner que la GDT repose sur deux systèmes distincts pour aborder ses principaux champs d'application en entreprise manufacturière :

- L'ERP (Enterprise Resource Planning) supporte le processus connu sous les noms de "Supply Chain" ou de "Order to Cash", correspondant au monde de l'exploitation (production). Celui-ci est caractérisé par des sous-processus courts mais rigides, nécessitant un suivi rigoureux des procédures et un usage intensif de workflows. En revanche, les modifications fréquentes des nomenclatures –faisant le lien avec le monde de l'ingénierie- sont freinées une fois acheminées vers l'exploitation.
- Le PDM (Product Data Management) supporte la définition du produit et du process, à travers son cycle de vie. Il est de ce fait appelé aussi PLM (Product Lifecycle Management) ou encore P²DM (P² pour Product/Process). Il s'agit du monde de l'ingénierie dans lequel les sous-processus sont fondamentalement longs à tel point que l'on cherche à les paralléliser –on parle de l'ingénierie simultanée. Ils fonctionnent par itérations successives, avec des allers-retours et des modifications continues sur l'objet de conception. Les données générées sont de grande taille et évoluent en fonction de la maturité du projet. L'usage des workflows est limité qu'aux jalons principaux du projet.

C'est à ce monde de l'ingénierie que nous nous intéressons dans ce travail de thèse et plus particulièrement à celui de **l'ingénierie de production**. Faut-il également rappeler la confusion longtemps présente entre la production stricte et celle de son ingénierie –appelée "Process Engineering" par les anglo-saxons. Celle-ci a été en effet longtemps négligée, relativement à l'ingénierie de produit, tant au niveau des pratiques industrielles que de la recherche.

Cependant, l'émergence des logiciels d'Ingénierie de Process (IPAO) et les mutations organisationnelles créant des plates-formes appelées désormais bureaux d'ingénierie produit/process –par opposition aux bureaux d'études et des méthodes- motivent ces communautés pour s'intéresser au domaine.

Ainsi, les projets de **mise en œuvre des SIT** pour l'ingénierie de production font sans doute rappeler les débuts de la CAO et ceux de la mise en place des PDM au service des bureaux d'études. Ayant les mêmes intentions, ces projets souffrent également des mêmes difficultés ; certaines sont plus délicates et constituent des goulots d'étranglement dans le processus de développement ; pour n'en citer que quelques-uns :

- La lenteur des projets et la présence des ruptures dans les jalons : les produits une fois livrés risquent d'être obsolètes face aux évolutions permanentes des besoins et des stratégies de l'entreprise. Ce goulot est lié en grande partie à :
- La difficulté de communication entre les principaux acteurs : la maîtrise d'ouvrage qui est en charge d'exprimer les besoins en matière de GDT, et la maîtrise d'œuvre devant traduire ces besoins en spécifications fonctionnelles. Notons la même difficulté dans les échanges avec les partenaires du projet (éditeur du PDM, sociétés de prestations...). La cause principale à ce problème est que les modèles d'expression utilisés sont informels, voire inexistantes en phases amont d'ingénierie de besoin.
- L'absence d'un guide méthodologique suffisamment flexible et contextuel pour mieux appréhender les singularités des projets. Les méthodes formelles proposées dans la littérature sont difficiles à adapter et ont échoué dans les projets antérieurs.
- Ajoutons à cela, le manque de rationalisation des processus d'ingénierie de production, qui pose des difficultés de formalisation de ses données. L'interdépendance des activités d'ingénierie produit/process ne facilite pas non plus cette tâche.

C'est dans ce contexte que s'inscrit notre problématique de thèse. Elle traite principalement deux aspects du domaine de l'ingénierie des SIT :

- Elaboration du **référentiel du domaine de l'ingénierie de production**. Celui-ci doit traduire, à l'aide de modèles de spécification suffisamment compréhensibles et standards, les besoins de la maîtrise d'ouvrage et de la maîtrise d'œuvre en GDT. Il pourra alors servir comme cadre de référence pour le développement des futurs SIT en entreprise.
- Rationalisation de la **démarche d'élaboration du référentiel** et plus globalement de la **conduite des projets de nouveaux SIT** en entreprise. La démarche se veut participative afin d'impliquer au mieux tous les acteurs et contextuelle pour répondre à la dynamique de tels projets.

Le travail de thèse s'est déroulé dans la Direction de la Recherche du Technocentre Renault et a été financé dans le cadre du projet "Usine Numérique". Il a été mené en collaboration avec la direction de l'ingénierie véhicule et la direction informatique de l'entreprise.

Le plan du mémoire est structuré en cinq chapitres. Le **premier chapitre** est consacré à la description de la problématique industrielle, telle que nous l'avons perçue chez le constructeur Renault. Nous introduisons d'abord quelques constats ayant des conséquences directes sur la GDT. Nous poursuivons notre analyse du terrain par la présentation des besoins des principaux acteurs impliqués dans ces projets : la maîtrise d'ouvrage et la maîtrise d'œuvre. Nous terminons ce chapitre par expliquer les goulots d'étranglements à la mise en œuvre des SIT en entreprise. Cela nous conduit à définir notre problématique de thèse.

Afin de résoudre cette problématique, nous explorons dans le **second chapitre** les solutions potentielles se trouvant dans la littérature. Cette investigation nous amène à distinguer diverses pistes que nous citons sommairement en :

- Technologies d'information et supports d'intégration.
- Modèles de spécifications en ingénierie des SI et langages dédiés aux données techniques.
- Architectures de références et techniques de modélisation en entreprise.
- Ateliers de génie logiciel, permettant d'automatiser de nombreuses tâches du projet.

- Méthodologies d'ingénierie des SI aussi bien académiques que pratiques.
- Méthodes de développement des SIT, s'inspirant essentiellement des méthodes de SI.

A l'issue de cette étude, nous sommes en mesure de présenter, dans le **troisième chapitre**, notre plan d'actions et les pistes de solutions retenues. Nous délimitons en premier le périmètre de l'ingénierie de production en un domaine représentatif et stratégique tant pour l'entreprise que pour le travail de recherche. Nous argumentons par la suite, les choix retenus quant aux modèles de spécifications. Pour élaborer le référentiel, nous avons opté pour une combinaison des techniques de modélisation des données, des processus et la notation UML. Notre référentiel est affiné en empruntant les patrons abstraits et d'analyse du domaine de l'objet. L'emploi de ces différentes techniques est aisément appréhendé grâce au cadre de Zachman et à l'atelier logiciel que nous avons retenus. Tous les aspects méthodologiques préconisés sont présentés dans la dernière partie de ce chapitre. Le choix a été fait d'intégrer, pour leur complémentarité, les approches ascendante et descendante. Nous intégrons par conséquent les différentes sources d'acquisition des connaissances en ingénierie du domaine. Nous expliquons par la suite notre démarche contextuelle guidée par le triptyque : intentions du projet, produit final attendu et contraintes environnementales.

Nous nous justifions par ailleurs sur les raisons pour lesquelles nous n'avons pas fait usage des outils et de la méthode homologués en entreprise.

Le **quatrième chapitre** aborde notre contribution basée sur les solutions retenues. Il est constitué de trois sections. La première section identifie les ingrédients du référentiel et un scénario d'élaboration de celui-ci. Nous mettons en avant quelques aspects originaux, comme l'usage du cadre de Zachman pour gérer les connaissances en phase d'analyse du domaine. La deuxième section porte sur la construction du business model, l'un des deux composants du référentiel. Il est composé d'un ensemble de modèles –contextuels, opérationnels et modèles de transformation logiques-, conçus chronologiquement et d'une façon contiguë et ce, suivant les vues stratégique, organisationnelle, temporelle et fonctionnelle. La dernière section du chapitre présente le data model, second composant du référentiel. L'élaboration du data model se présente en deux étapes : le diagnostic de données dans lequel les techniques de rétroingénierie et d'analyse de problèmes se confrontent pour produire des objets pertinents. Ces derniers sont structurés et raffinés dans la seconde étape. Celle-ci est basée sur les techniques objets et sur l'adaptation des patrons de la littérature.

Le **cinquième** chapitre décrit l'expérimentation du référentiel suivant le cadre méthodologique préconisé pour le développement de nouveaux SIT en entreprise. Nous introduisons le contexte de ce projet en identifiant les éléments du fameux triptyque. Nous décrivons par la suite la rédaction du cahier des charges de la maîtrise d'ouvrage, basée sur l'adaptation du référentiel. Enfin, nous illustrons, à l'aide des copies d'écran du produit final obtenu, les fonctionnalités acquises. Nous terminons ce chapitre par une conclusion portant sur les apports de ce cas industriel et les ouvertures possibles de l'application.

La **conclusion générale** du manuscrit fait le bilan de notre contribution, des difficultés rencontrées et propose des perspectives à ce travail.

CHAPITRE PREMIER

Regard sur les systèmes d'information techniques en entreprise. Problématique industrielle

[...] Donc toutes choses étant causées et causantes, aidées et aidantes, médiates et immédiates, et toutes s'entretenant par un lien naturel et insensible qui lie les plus éloignées et les plus différentes, je tiens impossible de connaître les parties sans connaître le tout, non plus que de connaître le tout sans connaître particulièrement les parties.

Blaise PASCAL (1623-1662). Les pensées – XV Disproportion de l'homme

Ce chapitre aborde la problématique de la GDT telle qu'elle est vécue chez l'industriel. Nous nous sommes inspirés essentiellement des données du terrain. De ce fait, les pointeurs bibliographiques sont très peu présents ici. Nous examinons en premier lieu, le contexte d'apparition du domaine de la GDT (I). Nous présentons par la suite les constats ayant des conséquences directes sur l'état actuel des SIT en entreprise (II). Nous formalisons les besoins en GDT suivant le point de vue des deux principaux acteurs : la maîtrise d'ouvrage et la maîtrise d'œuvre (III). Enfin, nous terminons ce chapitre par un bilan des problèmes cruciaux liés à l'ingénierie et au développement des SIT en entreprise (IV).

I Raisons d'apparition et d'évolution de la GDT en entreprise

Les mutations organisationnelles en entreprise manufacturière sont liées à l'évolution du périmètre de l'entreprise, aux nouvelles méthodes de travail et aux nouvelles exigences de flexibilité et de réactivité. Elles ont été sans doute le facteur clé d'apparition du domaine de la Gestion des Données Techniques (GDT) et d'une offre progicielle appelée SGDT (Encart 1.1).

La GDT se positionne parmi les plus grandes applications transversales en entreprise. Apparus au départ pour répondre aux premiers besoins d'automatisation des fonctions d'entreprise (comme la gestion documentaire), les SGDT deviennent des outils d'intégration entre diverses applications d'ingénierie ainsi qu'avec les ERP (Enterprise Resource Planning). Les SGDT sont considérés comme supports accompagnant le projet (gestion du travail collaboratif et gestion de projet). Ils sont aussi des composantes essentielles à la réalisation des maquettes numériques (Digital Mock-up pour les anglo-saxons). La gestion de configuration est une autre fonction de la GDT qui traite la diversité du produit et la complexité de son processus d'élaboration. Nous récapitulons l'ensemble de ces fonctions sur la Figure 1.1 que nous détaillons au chapitre 2 de la thèse.

Les technologies d'information et de communication (TIC) sont le deuxième facteur qui a contribué à l'évolution des fonctionnalités de la GDT. Les applications basées sur la technologie Web témoignent de l'implication des TIC dans la croissance des SGDT.

Encart 1.1 L'appellation des SGDT : du PDM au cPDM

*Nommer, c'est en un sens éterniser, c'est tirer la chose exprimée hors du chaos où tout se confond et du temps où tout se succède.
Gustave Thibon (- 2001)*

Le sigle français SGDT provient du terme anglo-saxon PDM (Product Data Management). On inclut généralement dans ce domaine les technologies proches des PDM comme : EDM/EDMS pour Engineering Data Management System, PIM pour Product Information Management, TDM pour Technical Data Management et enfin TIM pour Technical Information Management [MCI95]. Plus récemment, la société de conseil CIMdata a choisi de les appeler PLM (Product Lifecycle Management) pour exprimer le passage de la GDT statique au nouveau périmètre fonctionnel du cycle de vie du produit. Rebaptisée en cPDM (collaborative Product Definition management), cette gamme de produits, formant une "supply chain", veut étendre son périmètre pour permettre des échanges de données techniques entre l'entreprise et son environnement extérieur (ces fonctionnalités ne sont pas opérationnelles à ce jour).

CIMdata [CIM01] redéfinit donc cette technologie comme "a strategic business approach; applying a consistent set of business solutions that collaboratively manage the product definition lifecycle across the extended enterprise. These business solutions focus on the complete product definition, including all mechanical, electronic, software, and documentation components, and the processes that are used during the lifecycle".

Le domaine de la GDT n'a pas encore atteint sa maturité malgré la croissance de l'offre (Figure 1.2). Nous convenons d'appeler Système d'Information Technique (SIT) pour caractériser l'application de la GDT et les problématiques qui lui sont liées. Le périmètre des données gérées ainsi que les missions de ces systèmes font toujours l'objet de controverses (Encart 1.2). Néanmoins, en cette dernière décennie, les chercheurs du domaine de l'ingénierie des Systèmes d'Information (SI) ont tenté de rationaliser cette activité. Cependant, leurs résultats de recherches n'ont pas encore contribué concrètement à la mise en œuvre des SIT et des technologies de SGDT en entreprise. Les fournisseurs de ces progiciels, de leur côté, essaient de convaincre les clients de l'utilité et de l'opérabilité de leurs produits et les industriels tentent de cerner leurs besoins. Ces derniers faisaient depuis longtemps la GDT (mais sans peut-être le savoir !) et l'introduction des SGDT est d'autant plus délicate qu'il faut tenir compte de l'existant, de l'environnement culturel, organisationnel et informatique de l'entreprise.

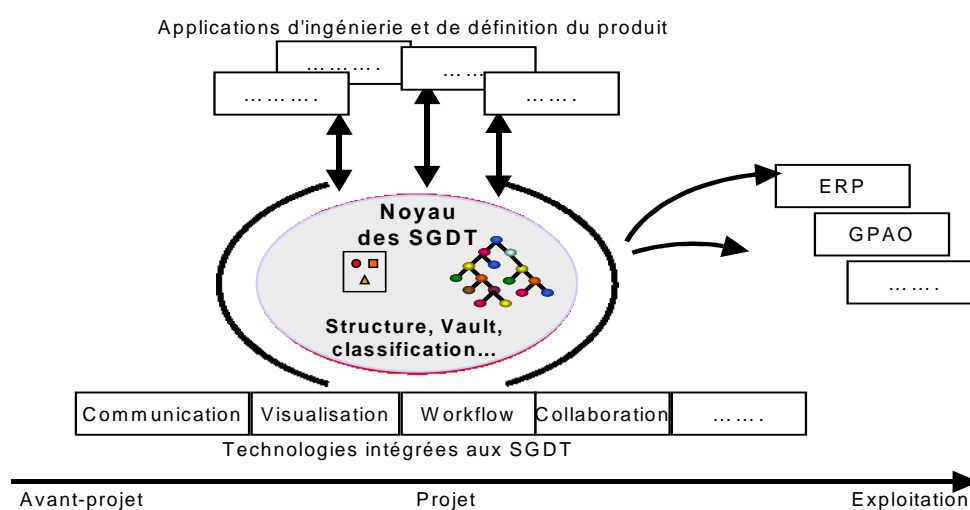


Figure 1.1 Position et fonctionnalités des SGDT en entreprise

Encart 1.2 Paradigme des Systèmes d'Information Techniques (SIT)

Nous nommons vrai un concept qui concorde avec le système général de tous nos concepts, vraie une perception qui ne contredit pas le système de nos perceptions ; la vérité est cohérence. Miguel de Unamuno (1864-1936)

La jeunesse inhérente du domaine de la GDT n'aide pas à fixer le périmètre et la couverture des Systèmes d'Information (SI). Appelé SI Produit par la communauté grenobloise [RIE99] ou encore SI Industriels (par quelques intégrateurs), nous convenons de les appeler SI Techniques par opposition aux SI de Gestion bien que la distinction soit déjà démodée. Les frontières entre eux s'effacent de plus en plus avec l'émergence de progiciels de type ERP et la technologie de Data Warehouse (Entrepôt de Données). Cette technologie se définit comme "une collection de données intégrées, orientées sujet, non volatiles, historisées, résumées et disponibles pour l'interrogation et l'analyse" [CAU01]. Elle intègre justement les données techniques et les données de gestion liées au produit, au projet voire à l'ensemble des données de l'entreprise.

Nous reprenons la définition de SIT proposée par L. Gzara [GZA00] que nous jugeons complète et concilie avec les autres propositions de la littérature : "Le système d'information produit est un dispositif organisationnel permettant de réguler la création, la circulation, l'utilisation et l'évolution du patrimoine informationnel de définition du produit, c'est-à-dire l'ensemble des informations qui définissent comment le produit est conçu, fabriqué et utilisé. Le support informatique de ce dispositif organisationnel est souvent réalisé à l'aide d'un Système de Gestion de Données Techniques (SGDT)".

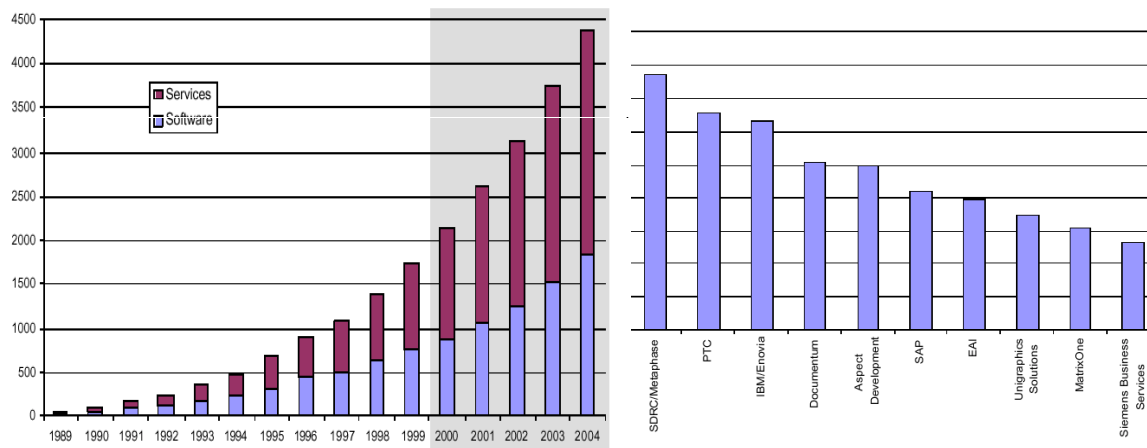


Figure 1.2 Croissance de l'offre en progiciels de SGDT (source CIMdata [CIM01])

II D'abord quelques constats...

Les constats ci-après sont fortement liés les uns aux autres. Ils sont établis sur la base de notre analyse du terrain chez l'industriel Renault, mais aussi, de l'évolution générale de l'environnement automobile. Nous résumons par la suite leurs conséquences sur la GDT.

III.1 De l'ingénierie du produit à l'ingénierie du process

Les années 80 sont marquées par l'émergence des outils de CAO et de calcul numérique, destinés aux BE pour optimiser la conception du produit. On parlait alors de l'ingénierie du produit. Au delà de ce périmètre, on parlait plutôt de l'industrialisation du produit et de son exploitation assurées par les outils de Gestion de Production (GPAO), de Maintenance (GMAO), de Qualité (GQAO) ou encore par les progiciels de type ERP. Les entreprises investissaient déjà dans les technologies 3D et les maquettes numériques avec lesquelles sont visualisées et analysées quelques fonctionnalités du produit. Les logiciels métier liés à l'ingénierie du produit ne cessaient donc de se diversifier. Avec eux, les informations circulant au travers de ces outils devenaient de plus en plus importantes et complexes à appréhender. Le besoin de gérer le "patrimoine informationnel" du produit s'est vite fait sentir et l'ambition de l'étudier tout au long de son cycle de vie était déjà annoncée comme cible pour les années à venir. Cependant, les entreprises ont jugé urgent de rationaliser d'abord les données circulant dans les BE, qui alimentent les logiciels de CAO et de calcul ou qui résultent des traitements dans ces outils. Les sites de production géraient bien leurs données avec leurs propres outils métier. Quant à l'activité des bureaux des méthodes en ingénierie, elle était jugée trop peu rentable pour que l'on se préoccupe de sa GDT ou des outils qui y sont déployés. Les acteurs du BE sont sans doute les premiers à être confrontés à la mise en œuvre des SIT et à bénéficier des technologies de SGDT dans leur vie quotidienne.

C'est seulement dans les années 90, que sont apparus sur le marché, une gamme de logiciels destinés à l'Ingénierie de Process appelés IPAO¹. L'IPAO est la combinaison d'outils logiciels et de méthodologies pour optimiser les travaux des méthodes sur l'ensemble du processus industriel. L'IPAO, vue comme complément naturel de la CFAO, permet de simuler, d'analyser, de valider et d'optimiser les systèmes de production et les processus associés. Cette activité se situe en amont de la fabrication du produit et en chevauchement des activités d'ingénierie de produit. Dans l'annexe B, sont présentées les activités d'ingénierie de process avec les problématiques essentielles qu'elles soulèvent. La Figure 1.3 positionne les principales familles d'outils logiciels évoquées jusque là par rapport au projet d'élaboration du produit manufacturier en entreprise.

Cette période est caractérisée par la volonté des industriels et des fournisseurs de logiciels XAO à faire coexister la conception du produit et du process. Les chercheurs également se lancent dans la création de méthodes et de techniques orientées produit/process, prenant diverses appellations telles que : DFA (Design For Assembly), DFM (Design for Manufacturing)...

Aujourd'hui, le cycle de vie du produit est lié aux préoccupations environnementales (phase de recyclage et de destruction du produit) et à celles de la satisfaction du client (avec les phases de service après-vente, maintenance...). Le fameux triptyque QCD (Qualité, Coût, Délai) du produit devient QC DPR (P pour Poids et R pour Recyclage).

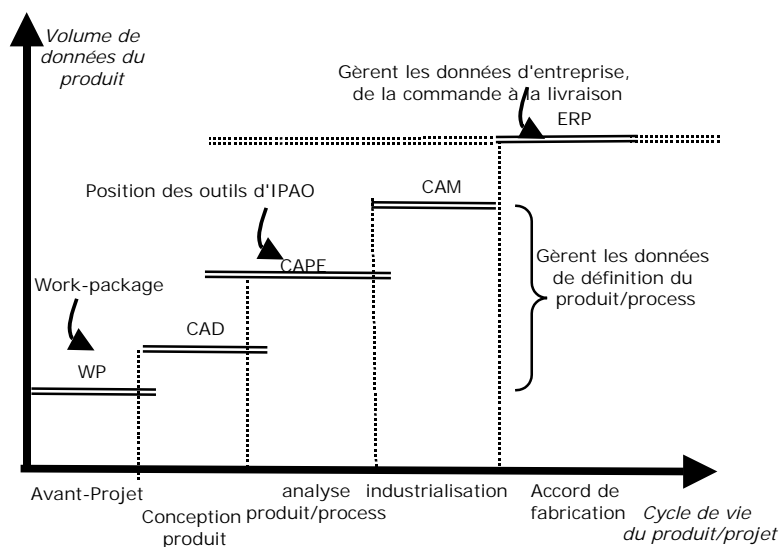


Figure 1.3 Position chronologique des outils d'ingénierie et volume des données

L'introduction de l'IPAO en entreprise fait sans doute rappeler les débuts de la CAO et de la GDT Produit. Nous nous sommes ainsi livrés à un jeu de comparaison que nous récapitulons dans le Tableau 1.1. Décrivons maintenant l'évolution de la gamme d'outils propre à l'IPAO.

¹ Traduction de l'abréviation CAPE : Computer Aided Process Engineering. Par ailleurs, Le terme anglo-saxon "process" est communément employé par l'industrie automobile pour désigner l'activité d'ingénierie de production, expression rationalisée par la communauté de la recherche.

	Logique Produit	Logique Process
Objet de conception	Produit avec ses différentes vues (as designed, as manufactured...)	Système industriel (Usine, Process et Ressources)
Clients créateurs	Acteurs du BE : designers, architectes, géomètres...	Bureaux de Méthodes : préparateurs, implanteurs, engageurs...
Nature de l'activité	Ingénierie du Produit : activités du BE liées à la définition du produit	Ingénierie de production : activités liées à la conception du process
Diversité /complexité	Configurations du produit : variantes et options du véhicule par exemple	Configurations du Process : hypothèses du système industriel
Maquette numérique	L'objet de visualisation est le produit et ses différentes composantes	L'objet de visualisation est le système de production, les flux...
Technologie cible	SGDT doté d'un environnement e-BOM ² (e pour engineering)	SGDT avec M-BOM (M pour Manufacturing)
Intégration avec outils	outils CAO et calcul en BE : simulation, cotation du produit...	Outils IPAO : simulation de flux et du process, implantation d'ateliers
Standardisation	spécifications du produit, ses différentes configurations...	Classification des opérations et des ressources, structure d'un site...

Tableau 1.1 Conception du produit et conception du process : éléments de comparaison

III.2 Conception verticale des outils d'IPAO

Qu'ils soient développés dans l'entreprise ou qu'ils proviennent du marché (Encart 1.3), les outils d'IPAO sont conçus d'une manière verticale : chaque outil, répondant à une fonction ou à un besoin du métier process, est alimenté par une Base de Données (BdD) locale où sont stockés également les résultats de ses traitements (le meilleur des cas). Une fonction du métier process peut apparaître dans plusieurs départements des bureaux des méthodes. De surcroît, elle est fortement liée aux autres fonctions qui participent ensemble à la réalisation du "long" processus d'industrialisation du véhicule. Prenons l'exemple de deux fonctions : la gestion documentaire et l'activité d'implantation des ateliers. Elles sont automatisées respectivement à l'aide de GED (Gestion Electronique des Documents) et de logiciels d'implantation d'ateliers (traduction libre du terme anglais "plant layout"). Ces applications sont différentes selon que l'on est dans les bureaux d'ingénierie de la direction mécanique ou dans les services de la direction carrosserie–montage. Elles peuvent aussi changer à chaque passage d'un jalon du projet véhicule à un autre. Pourtant, la nature des données et les fonctions qu'elles traitent sont très souvent similaires.

De même, pour réaliser des simulations de flux d'une ligne de production, des résultats issus d'autres outils (préparation, calcul logistique...) sont nécessaires. A l'heure actuelle, la plupart des offres de cette gamme de produits requièrent a priori une saisie fastidieuse de ces différentes données avant de pouvoir construire un modèle de simulation. Ces données sont stockées dans des fichiers de type "texte" qui rendent la mise à jour ou la distribution de ces résultats quasi-impossible, sous peine de passer du temps important dans la programmation interne du logiciel. Quand on dispose de BdD pour alimenter certains de ces outils, celles-ci ne sont pas optimisées et ne communiquent pas non plus entre elles. Par exemple, dans le métier de la tôlerie, nous trouvons plusieurs BdD décrivant les Liaisons dans les assemblés de pièces. Elles sont créées successivement par la CAO (nuage de points), le modeleur de gamme³ (définition des liaisons), le logiciel de simulation robotique (faisabilité du process) etc. Ces BdD décrivent beaucoup d'informations communes au concept de liaison.

² BOM (Bill Of Materials) : structure de graphe arborescente, concept expliqué dans la note de bas de page 6.

³ On appelle Modeleur de Gamme l'outil logiciel des préparateurs. Il permet, depuis l'ensemble des fonctions du véhicule définis par le BE, de construire les assemblés de pièces tels qu'ils seront réalisés sur site. Ils se traduiront par des opérations d'assemblage (du Process en général). Le cas échéant, il est possible aussi d'associer dans le même outil, les ressources nécessaires à la réalisation de ces opérations.

Il s'agit de simples exemples pour illustrer quelques conséquences sur la GDT du développement de l'IPAO en "Islands of automation" (Figure 1.4). De nombreux outils sont donc vus comme redondants, répondant partiellement aux spécifications initiales du Cahier des Charges (CdC) du produit et ne présentent presque pas d'intérêt lorsqu'ils ne communiquent pas avec l'ensemble des outils opérationnels.

La question de l'interopérabilité est toujours dans l'air du temps. Elle est d'autant plus difficile à réaliser lorsqu'il s'agit de connecter les applications développées en interne avec les logiciels du marché et ceci malgré la croissance et la diversification des technologies d'intégration.

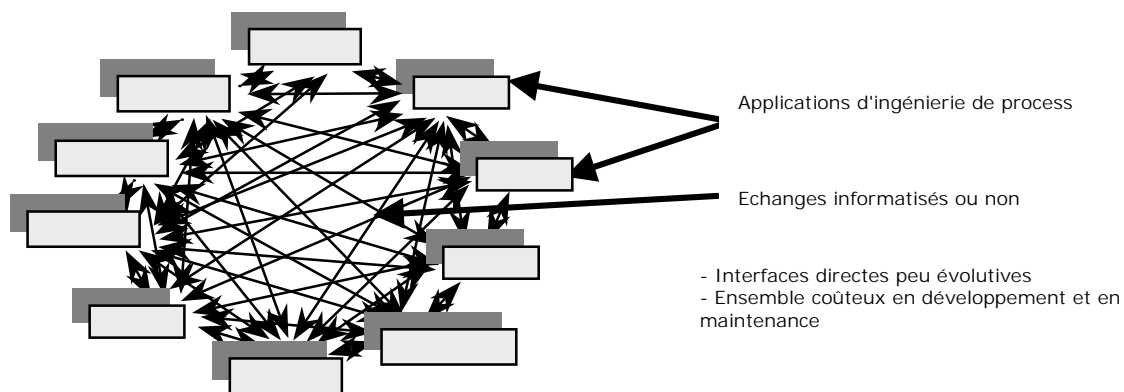


Figure 1.4 Conception verticale de l'IPAO : notion de "Islands of Automation"

Examinons un dernier constat sur l'architecture des outils d'IPAO. L'architecture, telle qu'elle est définie, impose une conception séquentielle du process. Prenons l'exemple de l'activité de montage et traduisons la logique de travail des bureaux des méthodes en ingénierie (Figure 1.5). Un préparateur ne peut commencer la construction des opérations de montage tant que les fonctions du véhicule ne sont pas complètement définies par le BE ; il ne peut construire les opérations sans les lier aux fonctions ou aux pièces du véhicule. Si le préparateur ne peut spécifier le détail des opérations qu'en présence des pièces à monter, peut-il cependant spécifier son process sous forme d'opérations suffisamment génériques pour traduire les grandes fonctions du véhicule à assembler ?

De même, l'engageur doit attendre que le préparateur ait fini de décrire complètement ses opérations et parfois que le logisticien ait étudié les emballages des pièces avant qu'il puisse effectuer un premier équilibrage⁴ de sa ligne. Peut-il réaliser des engagements macroscopiques sur les tronçons en attendant qu'il dispose des données de détail pour engager finement sa ligne, poste par poste ?

Le même retard de conception est constaté chez l'implanteur, qui est contraint de concevoir l'implantation avec des outils différents tout au long du projet véhicule. Il étudie en amont, les schémas d'allocation des surfaces de l'usine (allées, surfaces de stockage, lignes...) selon les hypothèses du site. Il bascule à un autre outil pour concevoir l'implantation des postes et des équipements. Cette activité est fortement liée aux surfaces dimensionnées dans le précédent outil mais aussi à l'activité de préparation et d'engagement. Des retours en arrière sont fréquents et les mises à jour sont toujours difficiles. Dans ce cas, est-il possible de concevoir un outil d'implantation intégrant ces différentes fonctionnalités nécessaires aux différents stades du projet ?

⁴ Equilibrage d'une ligne (Engagement) : activité d'affectation des opérations aux postes de travail d'une ligne de production, en respectant les contraintes métier produit/process.

Il est clair qu'à l'heure actuelle, l'avant-projet est dépourvu d'outils appropriés. Ces outils sont plutôt destinés aux phases aval et ne peuvent fonctionner qu'avec un nombre impressionnant de données de détail de conception indisponibles justement en avant-projet. Pour cette phase, la problématique porte sur la nature des hypothèses process à valider, sur les modèles de simulation à construire et sur les possibilités de conception du process avec un juste nécessaire d'informations. Cette problématique est liée au domaine de la préconception (conceptual design). La GDT est forcément amenée à évoluer pour faire face à cette nouvelle problématique.

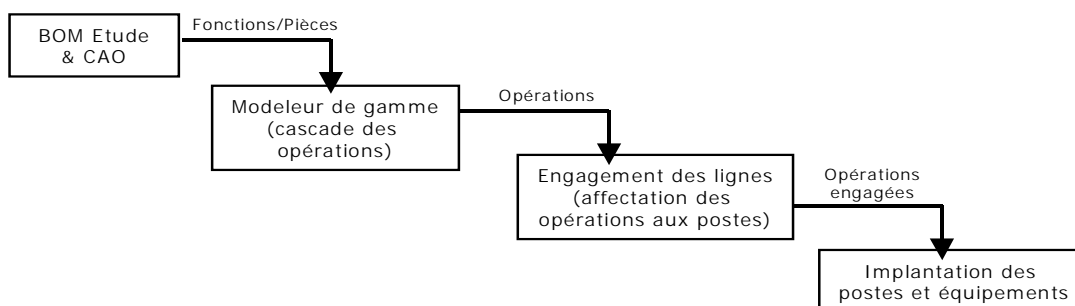


Figure 1.5 Architecture des outils en montage : conception séquentielle du process

Encart 1.3 Stratégie de développement des outils d'IPAO

Our Age of Anxiety is, in great part, the result of trying to do today's jobs with yesterday's tools. Marshall McLuhan (1911-1980)

Les entreprises, aujourd'hui, sont confrontées au problème d'externalisation (Outsourcing⁵) du développement de leurs logiciels métier. Deux courants majeurs s'opposent. Le premier encourage le développement en interne avec pour intentions de conserver le patrimoine d'informations et d'expertises métier de l'entreprise tout en remédiant aux problèmes de dépendance vis-à-vis de l'éditeur. Généralement, ce dernier (lorsqu'il a le monopole) "assiste" l'entreprise mais ne l'aide pas à faire croître ses compétences dans les phases de maintenance et d'exploitation des logiciels. Ainsi chaque évolution d'une application coûte beaucoup à l'entreprise en temps et en argent et ne répond pas forcément aux spécifications souhaitées. Le second courant stipule que le développement en interne risque de détourner l'entreprise de son objectif principal : produire des véhicules et non pas des logiciels métier. Ce courant vise cependant la réduction du nombre de fournisseurs de logiciels et l'implication de ces fournisseurs en établissant des relations de type "partenariat". La stratégie reste très controversée d'autant plus que les problèmes liés à l'intégration des SIT existants et à l'interopérabilité des outils ne sont toujours pas résolus. Serait-il plus sage d'adopter des décisions contextuelles en combinant ces deux politiques ?

III.3 Morcellement du long processus d'industrialisation du véhicule

La fabrication du véhicule passe par la succession des métiers principaux : l'emboutissage, la peinture, la tôlerie et le montage. De nombreuses activités transversales interviennent dans ces ateliers comme la logistique, le contrôle qualité, l'implantation, etc. Le processus de fabrication est illustré par la Figure 1.6.

L'ingénierie de ces métiers process se fait dans les bureaux des méthodes centralisés. Elle est poursuivie dans les Unités Élémentaires de Travail (UET) implantées physiquement en usines. Les bureaux des méthodes en ingénierie sont dispersés dans deux principales organisations : la direction Mécanique (où se fait l'étude produit/process des moteurs) et la direction Carrosserie-Montage. Cette dernière est divisée en différents départements qui étudient

⁵ "The act of contracting with an outside vendor to assume responsibility for one or more IT functions or services. In many cases, ownership of IT assets (including technology and employees) is transferred to the outsourcer"[WHI01].

exclusivement le produit et de nombreux autres départements orientés produit/process. Les études logistiques, de contrôle qualité... sont réalisées dans des départements indépendants. Les divisions à l'intérieur de tous ces départements sont en réorganisation permanente. Les problèmes d'échange et d'intégration des activités sont sans doute une des causes principales des mouvements de réorganisation. La plupart des conceptions process fait intervenir simultanément plusieurs métiers distancés géographiquement mais aussi séparés par des outils ne communiquant pas entre eux. Le processus d'industrialisation du véhicule transcende les divisions organisationnelles. Les études process débutent en effet dès l'avant-projet et ne se terminent pas avant l'accord de fabrication au moment de la délégation du suivi aux UET. Assurer une continuité entre les différents jalons du processus est nécessaire sous peine de vivre quotidiennement des dysfonctionnements, constitués principalement par les ressaisies et les itérations.

En effet, lorsqu'une même information circulant entre différentes unités organisationnelles, est saisie à nouveau ou transmise électroniquement, on peut considérer qu'une activité naturelle est fragmentée. Il serait plus logique en effet de saisir une seule fois l'information puis la partager en tant que produit fini au lieu de chercher à l'expédier au plus vite dans tous les sens. De même pour les itérations, si le résultat d'une activité sur un périmètre et à un jalon donné, se présente sous plusieurs formes, cela résulte le plus souvent d'une remontée insuffisante de l'information au sein d'un long processus morcelé. Que faut-il donc pour recoller les fragments du processus ? L'une des pistes à ce problème classique est l'application des techniques d'ingénierie des processus, répandues sous le nom de BPR (Business Process Re-engineering) et formalisées pour la première fois dans l'ouvrage de M. Hammer et al [HAM93]. Leurs applications soulèvent très souvent des dysfonctionnements perçus au niveau du SI supportant les processus opérationnels en entreprise.

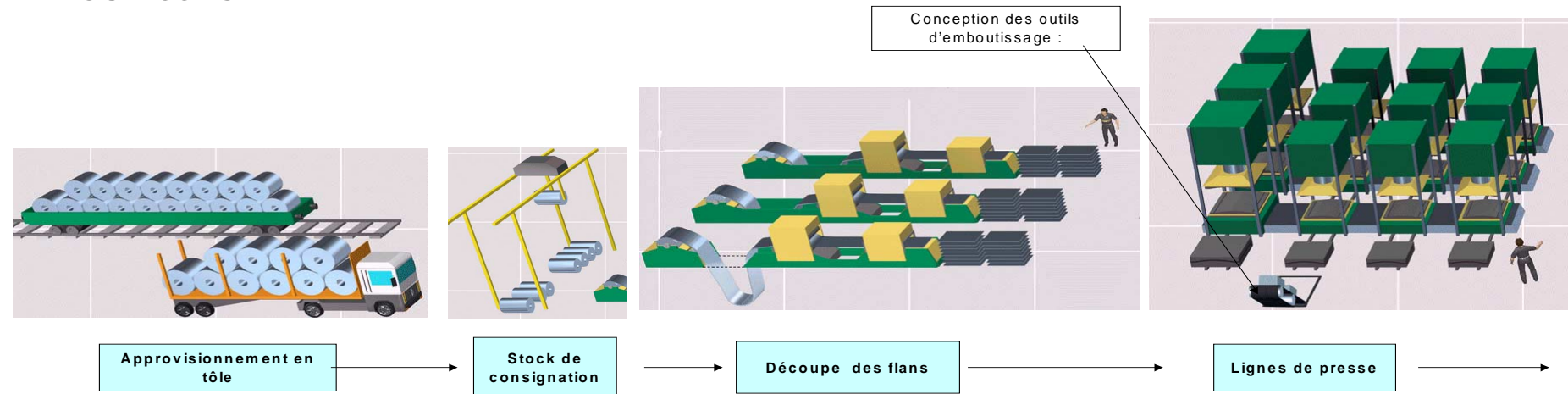
Dans ce qui suit, en s'appuyant sur l'historique de l'entreprise Renault, nous montrons comment les facteurs cités jusque là ont amené à faire évoluer la réflexion autour de la GDT et particulièrement sur le périmètre de l'ingénierie de production.

III.4 Les conséquences sur la GDT

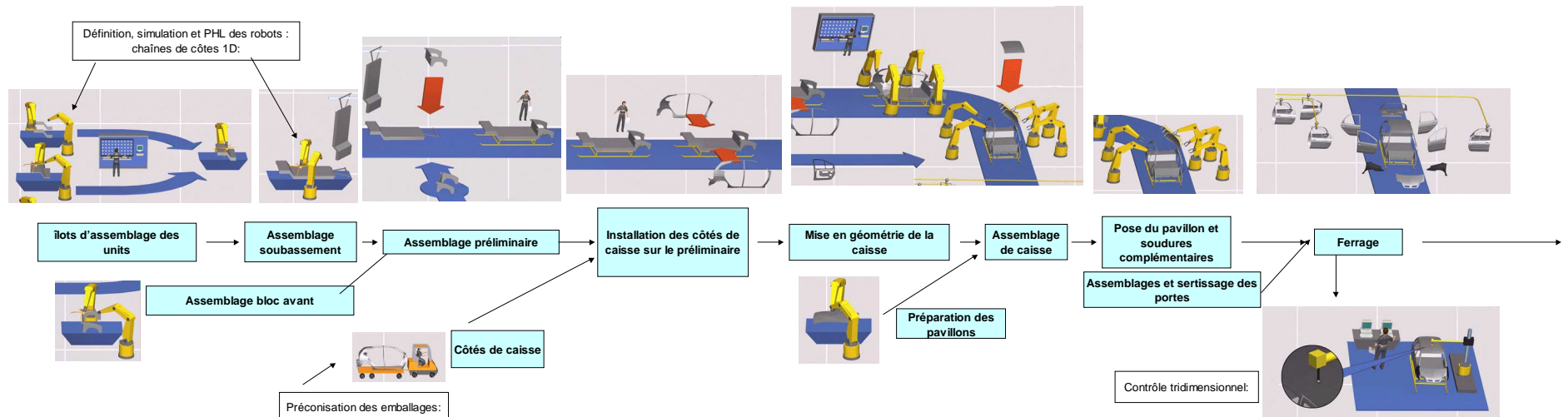
Les premières mises en œuvre de GDT chez Renault furent naturellement au service des designers. Les données résultantes ou servant à alimenter les outils de CAO et de calcul numérique (simulation, cotation du produit...) étaient donc prioritaires. Ainsi, les données géométriques des pièces du véhicule sont gérées par le logiciel du marché Métaphase en développant une couche IHM baptisée GDG (Gestion des Données Géométriques). Il venait de remplacer un ancien outil devenu obsolète. Quelques fonctions élémentaires de GDT ont été mises en place, telles que le versionnement et la maturité des pièces graphiques. GDG est désormais interfacé avec la CAO, principal outil des acteurs du BE. Mais très vite, les groupes chargés de mettre en place cette GDT ont su qu'il fallait aussi gérer les autres représentations du véhicule. De ce fait, ils ont lancé un projet ambitieux qui a donné naissance à une application appelée SIGNE (Système d'Information pour la Gestion des Nomenclatures de l'Entreprise). Conçue et développée en interne et implantée sur un SGBD, cette application vient naturellement remplacer l'ancien SI DOC-BE avec une volonté de lui assigner de nouvelles fonctionnalités et un périmètre de données plus large. En effet, comme son acronyme l'indique, SIGNE doit gérer l'ensemble des données techniques liées directement ou non au produit depuis sa définition jusqu'à sa fabrication. Bien entendu, un tel projet était trop ambitieux et il fallait encore délimiter le périmètre sous peine de voir échouer le projet.

Figure 1.6 Chaîne d'industrialisation du véhicule : métiers principaux (Source Renault)

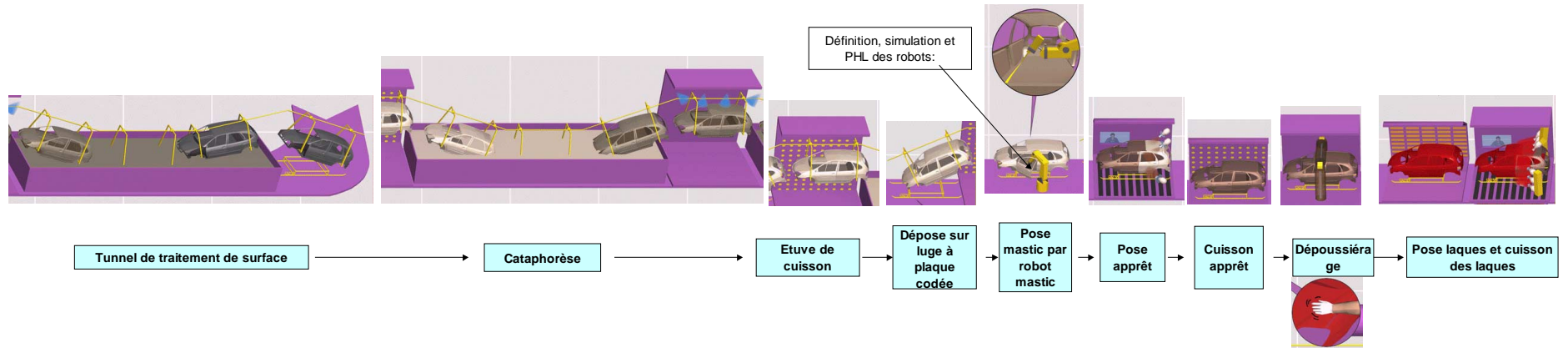
EMBOUTISSAGE



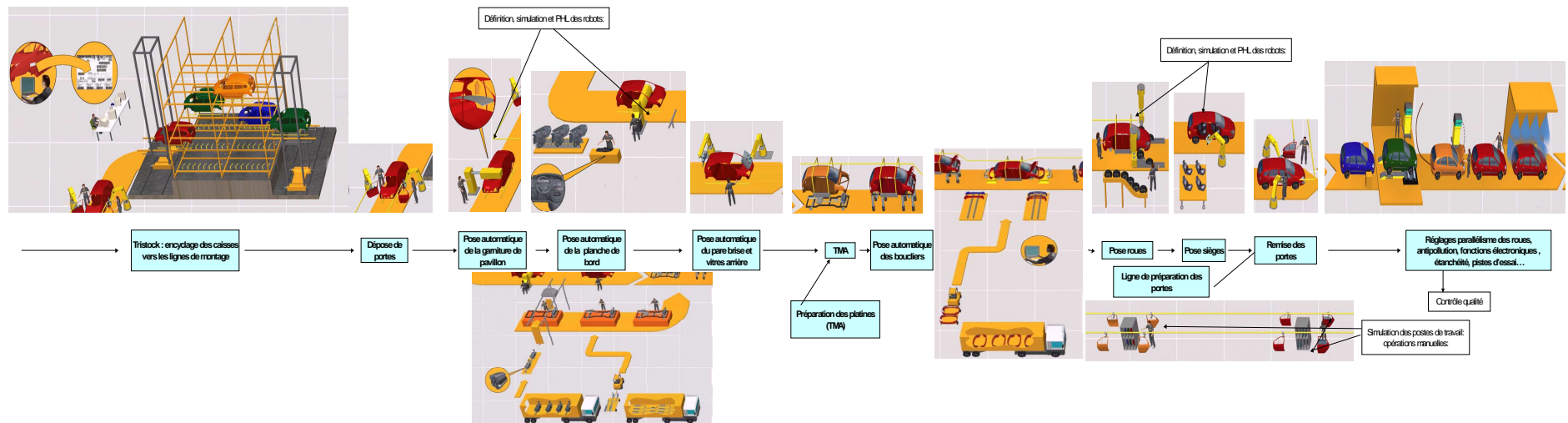
TOLERIE



PEINTURE



MONTAGE



Aujourd'hui, l'application SIGNE a réussi à intégrer les principaux SIT existants dans les BE. SIGNE décrit le véhicule par sa nomenclature Etude⁶ et les documents de définition et de description associés (CdC, normes, plans...). La gestion des configurations du produit a été possible grâce aux efforts des acteurs de l'entreprise en matière de standardisation des spécifications fonctionnelles du véhicule. Un langage appelé MTC (Mise en Titre des Critères) a été créé pour décrire l'ensemble des fonctions possibles du véhicule (avec variantes et options). Un logiciel appelé DOC-Véhicule est mis en place pour générer ces diversités possibles en cohérence avec le CdC initial. Des tentatives pour réaliser d'autres fonctions de GDT avec le même système SIGNE ont été étudiées. Ainsi, la documentation, la visualisation des pièces CAO initialement gérées dans GDG sont possibles. En revanche, réconcilier les différentes représentations du produit (essentiellement les objets graphiques de CAO et les données de la nomenclature Etude) et tenir compte de la totalité du cycle de vie du véhicule sont toujours des sujets d'actualité. L'environnement informatique ainsi que les autres problèmes classiques d'évolution et de maintenance des SI se posent aussi pour SIGNE. Néanmoins, cette application demeure la plus transversale en entreprise. Le nombre d'utilisateurs n'a cessé d'augmenter. Qu'ils soient en BE, dans les bureaux des méthodes en ingénierie ou en série, voire dans les services de gestion (Achats et Chiffrage), l'application reste accessible pour les opérations de consultation ou de mise à jour des pièces du véhicule. Elle mérite alors son label de "GDT d'entreprise" par opposition à "GDT locale". La Figure 1.7 illustre la position de ce SI et son intégration avec les applications métier.

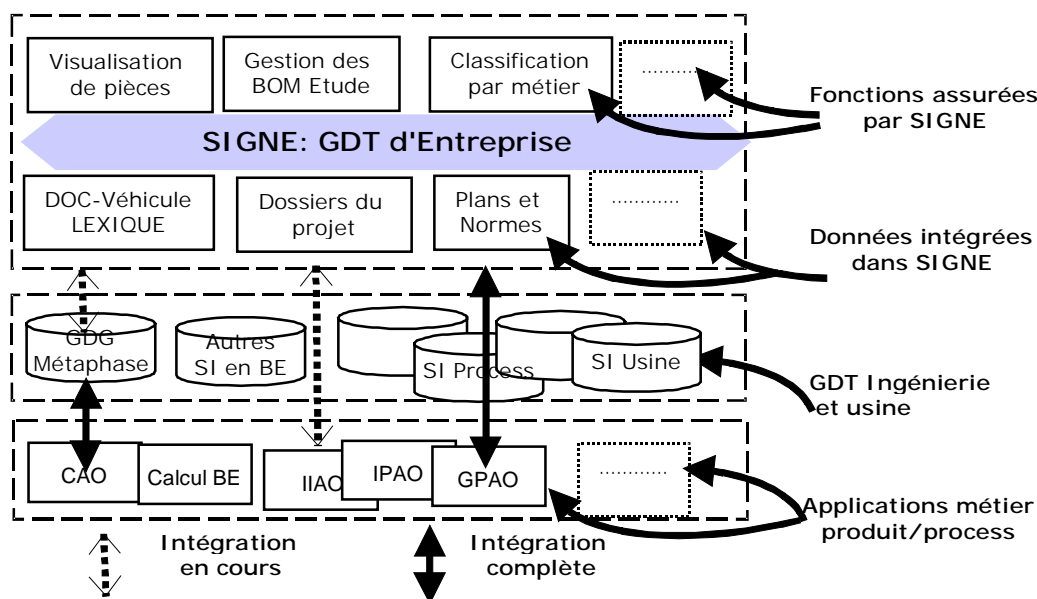


Figure 1.7 SIGNE : Application de GDT d'entreprise

Qu'en est-il de la GDT Process ?

Si la GDT du produit est maîtrisée et les SIT en exploitation (dans les sites de production) restent opérationnels malgré leur nombre important, il n'en est pas de même pour l'ingénierie de production. Cette activité, rappelons-le, située entre les études du produit et la fabrication, est caractérisée aujourd'hui par l'émergence de ses propres outils d'IPAO et d'un nombre

⁶ Nomenclature "Etude" (As Designed BOM) : est le nom que nous avons choisi pour désigner ce qui est appelé communément au sein de l'entreprise : Découpage Référentiel Fonctionnel du Produit. Il consiste à découper le véhicule en fonction des métiers du BE. Ainsi on retrouve les grandes fonctions du véhicule qui se déclinent plus tard en pièces physiques réalisant ces fonctions. Ce découpage est une "nomenclature" puisqu'il est représenté sous forme d'un arbre de nœuds liés entre eux avec la relation "composé de". Il est "Etude" puisqu'il couvre la vue des métiers en BE.

impressionnant de SIT (Figure 1.8). La conception verticale de ces outils, l'absence d'interopérabilité entre eux et le morcellement du processus d'industrialisation sont probablement les causes principales des dysfonctionnements récapitulés ici :

- *Difficulté d'accès et d'évolution des SIT existants* : les architectures informatiques ne peuvent plus supporter le nombre croissant des utilisateurs et le volume croissant des données. Les extractions d'informations deviennent des tâches fastidieuses et leur maintenance est quasi impossible.
- *Ressaisies et Itérations* : elles sont observées tout au long du projet véhicule, de passage de jalons et durant les échanges de données incontournables entre applications du métier process.
- *Redondance et perte d'informations* : lorsque des fonctions similaires se retrouvent dans plusieurs logiciels métier, les données sont-elles aussi dupliquées. La perte d'informations peut se produire au moindre transfert d'informations entre acteurs, conséquence de la non informatisation des échanges d'une part et de la rupture entre les outils d'ingénierie et d'exploitation d'autre part.
- *Absence de standardisation et de structuration des données techniques du process* : les terminologies employées pour décrire les mêmes objets process⁷ divergent d'un métier à l'autre. Ils ne sont pas structurés et leur évolution (cycle de vie et maturité) n'est pas gérée. De surcroît, l'activité d'ingénierie de process n'est pas tout à fait rationalisée.

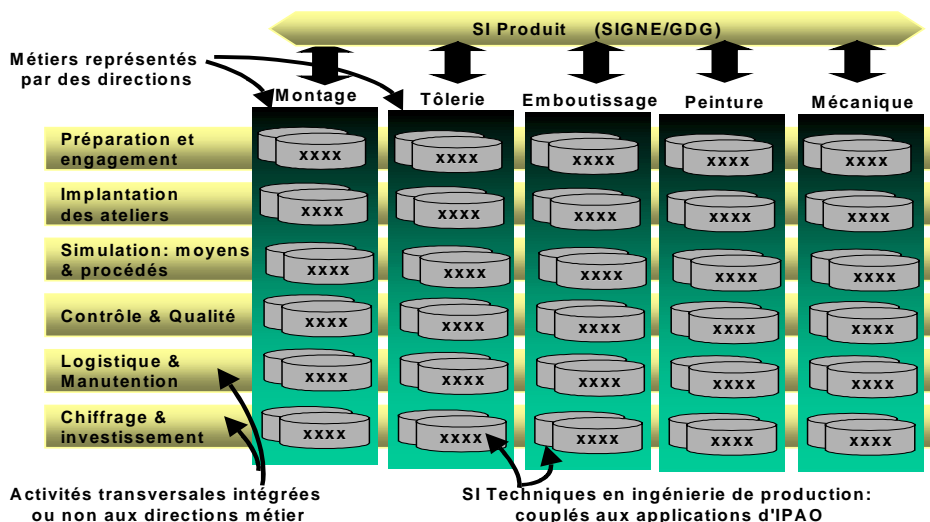


Figure 1.8 Etat actuel des SIT en ingénierie de production

Terminons avec un exemple qui permettra d'introduire les besoins en GDT process que nous formalisons ci-après. La conception du Bord de Chaîne⁸ (BdC) est complexe de par l'implication simultanée de plusieurs métiers process et la non rationalisation de son processus de conception et de ses règles de construction.

Le préparateur et l'engageur ont la charge d'optimiser la surface du BdC en fonction des contraintes liées à l'engagement des opérateurs affectés aux postes. Le logisticien préconise les emballages des pièces et les meubles où les stocker. Il a en plus la charge de dimensionner le stock de sécurité, de définir les fréquences et les lots d'approvisionnement.

⁷ Nous convenons d'appeler "objet process", les données manipulées en ingénierie de production. Elles concernent l'ensemble des caractéristiques du système industriel, des opérations, des ressources...

L'implanteur, quant à lui, a la charge de répartir les surfaces disponibles pour implanter les équipements appropriés. Plusieurs outils logiciels ont été développés à ces fins mais sans apporter de réelles satisfactions. Conçus pour une direction et un type d'acteurs donnés, ces logiciels couvrent partiellement le problème de conception du BdC et ne prennent pas en compte l'ensemble des contraintes et des informations juste nécessaires à la construction du BdC dans sa globalité. De plus, ces outils imposent a priori des saisies de données de détail indisponibles en avant-projet et ne sont pas non plus connectés aux systèmes documentaires...

La conception optimale du BdC (plus globalement du système industriel) ne nécessiterait-elle donc pas de passer par la fédération des SIT correspondants aujourd'hui "locaux" ?

III Besoins de GDT en entreprise : spécifications internes et externes

La partie précédente nous a permis de justifier le périmètre de notre étude qui est l'ingénierie de production, d'établir la liste des manques et dysfonctionnements principaux dans ce domaine. De ce fait, nous sommes maintenant plus aptes à formaliser les besoins en GDT, satisfaisant à la fois les clients finaux de la GDT process mais aussi les responsables de maintenance et d'évolution de celle-ci. Nous prenons soin de vérifier que ces besoins sont reconnus par la communauté des chercheurs et les industriels d'une manière générale.

Notons qu'il est difficile d'exprimer les besoins de l'entreprise en GDT sans emprunter les fonctionnalités des SGDT en général. Ces deux pôles évoluent en effet par ajustements réciproques. C'est grâce en partie aux fournisseurs de progiciels de SGDT que l'industriel a pu cerner et formaliser correctement ses besoins en matière de GDT. Sans l'industriel, l'éditeur n'aurait pas fait évoluer ses progiciels.

Les différents acteurs concernés par la GDT Process peuvent être classés en deux catégories. D'abord le client final (utilisateur) représenté par la maîtrise d'ouvrage et la direction informatique de l'entreprise (maîtrise d'œuvre) qui est en charge de spécifier et de maintenir la GDT d'entreprise. Le rôle de ces deux catégories d'acteurs a toujours fait l'objet de controverses et de définitions abusives. Nous fixons dans l'Encart 1.4, les missions de la maîtrise d'œuvre et d'ouvrage en cohérence avec la culture de l'entreprise.

Encart 1.4 Missions de la Maîtrise d'Ouvrage et de la Maîtrise d'Œuvre (Source Renault)

Ours is a world where people don't know what they want and are willing to go through hell to get it. Don Marquis (1878-1937)

Attributions de la Maîtrise d'Ouvrage (MOA) : structure d'orientation et de décision

La MOA est à l'origine du projet*. Elle doit garantir l'enjeu pour le métier (les utilisateurs et les clients finaux), l'entreprise ou des tiers. Elle est en charge des validations des jalons du projet et de son contenu fonctionnel. De plus, la MOA doit assumer le moindre changement dans toutes les dimensions du projet (stratégiques, commerciales, financières, humaines, juridiques, organisationnelles, ...).

Attributions du Maître d'œuvre (MOE) : structure de réalisation

La MOE est responsable de la conduite du projet. A ce titre, elle doit s'assurer de l'organisation, de la coordination et de l'animation de l'équipe projet. La conduite du projet va depuis les spécifications fonctionnelles du produit jusqu'à son déploiement en incluant le pilotage des prestations extérieures si besoin, la capitalisation des solutions innovantes et le retour d'expérience.

* Le projet au sens général. Il peut concerner l'élaboration d'un SI, d'une application logicielle ou encore le développement d'un véhicule.

⁸ Bord de Chaîne : surface de stockage et d'encyclage des pièces avec leurs emballages prêts à approvisionner les postes de travail. Les emballages sont stockés dans des meubles et différents supports et ils arrivent au Bord de Chaîne depuis le magasin selon des règles logistiques et de manutention.

En fonction de ces deux catégories de clients, nous classons les finalités de la GDT process en spécifications internes et externes. On entend par spécifications internes, l'ensemble des besoins liés à la structuration de l'information généralement perçus par la maîtrise d'œuvre. Les spécifications externes en revanche, traduisent les fonctionnalités résultantes de la mise en place d'une GDT, plutôt exprimées par le client final. Précisons seulement que toutes ces spécifications ne sont évidemment pas indépendantes les unes des autres comme nous allons le constater dans ce qui suit.

III.1 Spécifications externes : besoins de la maîtrise d'ouvrage

Nous résumons les spécifications externes en ces cinq points clés suivants.

- **"Zéro" ressaisie et "Zéro" rupture**

La conception verticale des outils et les frontières imposées par les directions sont à l'origine des ruptures au niveau du processus d'élaboration du produit. Cela conduit à des ressaisies manuelles d'informations et à la non intégration des différents métiers impliqués. Ainsi, les premiers efforts pour établir cette intégration sont focalisés sur le changement organisationnel au sein de l'entreprise. Sur une même plate-forme –désignée bureaux d'ingénierie pour l'étude produit/process"- que les designers des BE et les préparateurs des méthodes se réunissent désormais. Ils sont invités à valider conjointement les spécifications produit/process tout en mettant à leur disposition des outils orientés également produit/process. Cette période est suivie par une demande accrue d'impliquer les autres métiers de l'entreprise mais aussi les fournisseurs et former cette boucle du "supply chain". Cette chaîne n'est pas facile à réaliser car l'entreprise ne peut imposer le choix de l'utilisation des outils à ses partenaires. De surcroît, la question de l'interopérabilité se pose toujours malgré les efforts des actions normatives en matière d'échanges de données. L'intégration par les données est un des objectifs de la technologie de SGDT, sur laquelle peuvent être greffés les outils d'ingénierie. Aujourd'hui, la présence de SGDT en entreprise n'a pas chassé les ressaisies et les ruptures.

- **Gestion documentaire**

La Gestion Electronique de Document (GED) est probablement l'une des plus vieilles applications de la GDT. Les premières fonctions de gestion documentaire se résumaient à stocker, classer et accéder au document sans se soucier de son contenu. Ces fonctions ont évolué plus tard pour gérer également le contenu informationnel des documents et permettre ainsi leur création dynamique. La question de structurer les données potentiellement nécessaires à la gestion électronique des différents documents du projet n'a peut-être pas été soulevée a priori, causant la diversification et le nombre important d'applications logicielles dédiées à cette fonction. Par ailleurs, beaucoup de transmissions de documents continuent à se faire via le courrier ou les supports de Bureautique⁹. De plus, quelques documentations destinées à assister les opérateurs au niveau des postes de travail, sont souvent jugées trop techniques et peu parlantes (car il est encore possible de rencontrer des agents de production ne sachant pas lire). Sans doute, ces quelques dysfonctionnements seront vite dissipés à la mise en place d'une GED transversale s'appuyant sur une GDT d'entreprise commune.

⁹ Exemple : Les services techniques doivent fournir aux services de gestion les documents suivants : "Engagement des effectifs", "Avancement du plan d'action", "Bilan des effectifs qualité", "Evolution du temps opératoire", "les figurines représentant l'engagement des ateliers". Ces exemples de documents nous montrent encore une fois les frontières de plus en plus étroites entre les SI Techniques et les SI de gestion.

- **Usine numérique et revue de projet**

Concevoir en "tout numérique" était un des défis annoncés dans les années 90 par l'industriel. Le pari est presque tenu ; les études du produit à différents stades de sa vie, en atteignant des niveaux de complexité très hauts, se réalisent de plus en plus avec des prototypes virtuels. Les premières validations produit/process en numérique ont débuté sur des simulations d'un poste de travail (manuel ou automatisé) à l'aide des logiciels de simulation couplés à la CAO. L'idée est par la suite étendue pour concevoir tout le système industriel en numérique à différents niveaux d'agrégation (poste, ligne, atelier...) en évaluant sa performance technico-économique. Ce type de prototype appelé "Usine Numérique" deviendra sans doute l'outil approprié pour les revues de projet¹⁰ process. Ainsi, les différents interlocuteurs du projet (exemple : le responsable de chiffrage des investissements en équipements ou le préparateur chargé d'optimiser sa ligne de production) peuvent naviguer et visualiser le (ou une partie du) système industriel. Ces prototypes se réalisent à l'aide de composants de logiciels de simulation de flux couplés à des "Viewers". Cette dernière gamme de produit est amenée à disparaître car les SGDT intègrent de plus en plus cette fonction de visualisation appelée maintenant maquette numérique (Digital Mock-up). Hormis les contraintes techniques liées à la réalité virtuelle, la question cruciale soulevée est la disposition des données juste nécessaires permettant la construction de ces modèles numériques. Cette organisation des données process, nouveau socle fondateur de l'usine numérique, commence à être appelée DPM (Digital Product for Manufacturing).

- **Traitement de la diversité Produit/Process**

Nous avons vu plus haut que la diversité du produit est relativement maîtrisée grâce d'abord aux efforts de l'entreprise en matière de standardisation des spécifications fonctionnelles du véhicule. Les configurations du véhicule sont prises en compte dans l'ensemble des outils d'ingénierie de produit. On peut savoir par exemple le cas d'emploi d'une pièce dans un véhicule donné c'est-à-dire, trouver ses différentes utilisations dans l'ensemble du véhicule. Avec l'émergence des outils d'IPAO, le besoin de gérer également la diversité process s'est vite fait sentir. Encore une fois, c'est grâce aux développements internes en entreprise, que les premières diversités process ont été réalisées. Par exemple, une opération sur une ligne d'assemblage est configurée en fonction des caractéristiques du véhicule pour lequel elle est réalisée. Mais, la diversité process ne s'arrête pas là ; l'étude de l'affectation d'une même famille véhicule sur différents sites industriels en est un exemple. La gestion des configurations produit/process est encore une problématique d'actualité à laquelle les fournisseurs de progiciels ont répondu partiellement ou pas du tout. Cette fonction attendue dans les SGDT est appelée gestion des configurations (Configuration Management).

- **Rationalisation des procédures et des processus métier**

Reproduire d'une façon automatique les procédures métier, les faire partager et les enseigner est sûrement une des préoccupations majeures du client. La rationalisation permet entre autre de capitaliser et d'effectuer des retours d'expérience. Car le "mode d'emploi" n'est peut-être pas à réinventer complètement à chaque projet véhicule. Cependant, cette rationalisation ne doit pas empêcher la flexibilité et l'adaptation face aux contraintes environnementales de l'entreprise (Business Requirement) qui sont en permanente évolution. Cette fonction est communément désignée par "Workflow" et disponible désormais dans les SGDT.

¹⁰ Revue de projet : est l'acte de réunir les acteurs du projet autour d'un plateau pour faire valider les études et les décisions liées au projet. La validation marque la fin d'un jalon et le passage au jalon suivant qui représente un stade du cycle de vie de l'élaboration du produit. Les revues de projet ont débuté avec des outils "papier" et un nombre restreint d'acteurs, pour évoluer vers les maquettes virtuelles produit/process impliquant tous les acteurs du projet, voire les partenaires extérieurs.

III.2 Spécifications internes : besoins de la maîtrise d'œuvre

Les expressions suivantes ne sont peut-être pas perçues directement par la MOA qui représente le client final (utilisateur du SGGT). En revanche, c'est à la charge de la MOE de les spécifier. Voici quelques éléments clés.

▪ **Standardisation et référentiel des données process**

L'objet d'un référentiel process est de fournir une terminologie et une représentation commune des objets process que les différentes applications d'IPAO puissent ensemble comprendre et utiliser. La terminologie doit s'appuyer sur une sémantique précise et non ambiguë portant sur l'ensemble des objets process.

Evoqués déjà plus haut, les premiers référentiels de données techniques en entreprise concernent principalement les spécifications fonctionnelles du véhicule, comme le langage MTC communément utilisé et implanté dans les applications logicielles. Cependant, l'ingénierie de production ne dispose pas de référentiel commun des objets process manipulés. Voici des exemples où le besoin de standardiser se fait sentir.

- *Définition commune de l'objet "poste de travail"* : ce concept est souvent abstrait du fait qu'il n'est pas systématiquement représenté physiquement dans le site de production. Il est cependant essentiel dans la plupart des activités de conception de process (simulation, implantation, engagement...). De surcroît, nous constatons malgré ces définitions qui divergent selon les activités d'assemblage, d'usinage ou en mécanique, que cet objet peut être caractérisé avec des paramètres communs comme "le temps de cycle". Ce temps est nécessaire à la détermination du nombre d'opérations à regrouper dans le poste.
- *Découpage fonctionnel et structurel d'une implantation d'atelier* : par opposition au véhicule, objet de conception des designers, l'ensemble des acteurs du process (préparateurs, engageurs, implanteurs...) souhaitent avoir une description générique d'une implantation (leur objet de conception) sur laquelle ils peuvent s'appuyer dans leur travail quotidien de conception du process.

Sur la standardisation, plusieurs autres thèmes peuvent être développés : la typologie des opérations en carrosserie-montage, le langage de description des prestations générales etc.

Nous pensons que ce travail de standardisation doit se positionner en premier dans les projets de mise en place de SIT process. Car c'est cette représentation générique qui sera l'objet d'une implémentation informatique. La validation conjointe du lexique par la MOA et la MOE est une question cruciale qui se pose actuellement en entreprise.

Nous verrons également dans le chapitre suivant, que la standardisation des objets process est aussi une préoccupation de quelques importantes actions normatives, notamment PSL et MANDATE que nous aurons l'occasion d'exposer plus tard.

▪ **Représentation et classification des données**

Tout comme les données en BE, les données techniques circulant en ingénierie de production sont de natures différentes. Trois types peuvent être distingués. Tout d'abord, les données que l'on appellera plus tard nomenclature, sont aujourd'hui présentées "en vrac" -la granularité ou la généralité de la donnée n'est pas distinguée- et réparties dans bon nombre de SIT (gamme des opérations, des outillages...). La deuxième forme de ces données techniques est de nature graphique ou géométrique. Numérisés ou non, les modèles graphiques peuvent être des plans d'implantation (2D/3D), des ressources ou des process

manipulés dans les logiciels de simulation par exemple. Le dernier type concerne les dossiers et les documents qui accompagnent le projet de développement du véhicule, traitant particulièrement les spécifications process.

L'organisation de ces données sous forme d'arborescences, en conciliant les différentes représentations (Figure 1.9), est l'un des grands objectifs des SGDT. Les apports d'une telle structuration ont déjà été étudiés lors des mises en place de SIT produit en entreprise.

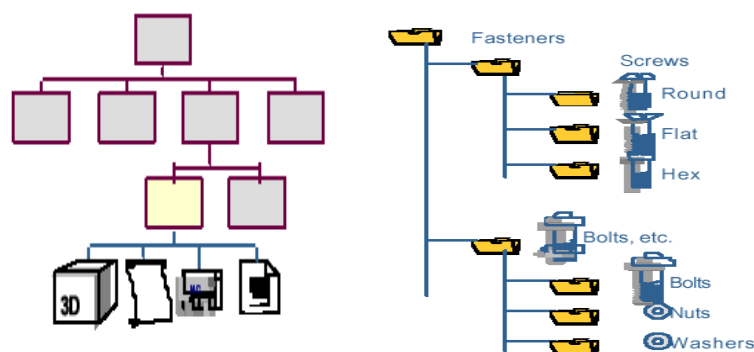


Figure 1.9 Classification et représentation des données

▪ **Formalisation des données en avant-projet**

Aujourd'hui, la conception du process s'inspire essentiellement des données d'exploitation en adoptant une approche "bottom-up" (Figure 1.10). Cette démarche est en effet bénéfique lorsqu'il s'agit de conceptions incrémentales¹¹. Le gain est considérable en empruntant les caractéristiques d'un projet passé capitalisé à travers les données d'exploitation en usine. Cependant, il arrive que le projet soit complètement innovant et il devient nécessaire de spécifier les caractéristiques du process en amont du projet sans devoir emprunter les données détaillées de l'usine qui s'avèrent inutiles. Ce type de conception appelé "top-down" exige la manipulation des données suffisamment génériques qui peuvent se traduire (ou s'éclater) éventuellement par la suite en données de détail.

Les difficultés principales pour concevoir en "top-down" sont liées au fait que les données macroscopiques sont indisponibles et non formalisées. Les activités d'ingénierie process qui sont concernées ne sont pas non plus rationalisées. Ajouté à cela, il manque des outils logiciels dédiés à ces phases très en amont du projet comme nous l'avons souligné (III.2).

Enfin, nous récapitulons dans le Tableau 1.2 les besoins de l'entreprise en GDT et les composants logiciels correspondants à ces besoins que nous allons décrire au chapitre suivant.

¹¹ Incrémentale : conception routinière (par opposition à la conception innovante) qui ne présente pas beaucoup d'innovations ou de changement dans les caractéristiques du véhicule ou du process précédents. Exemple d'une conception incrémentale : les améliorations mineures produit/process qui surviennent lors des premières montées en cadence du même projet véhicule.

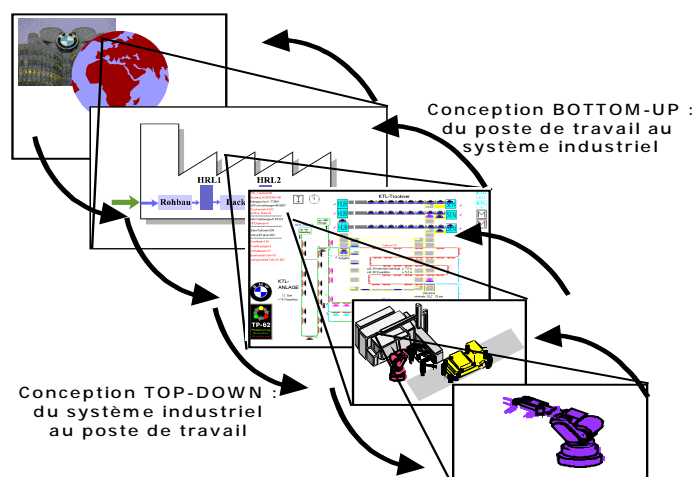


Figure 1.10 Conception bottom-up et top-down du processus d'industrialisation du véhicule

Besoins en GDT	Composant logiciel du SGDT & technologie
Consignation des données	Vault, Repository & Data Release Management
Gestion documentaire	Document Management
Rationalisation des procédures et processus métier	Workflow, Process & Change Management
Construction des nomenclatures et hiérarchisation	Structure (Bill Of Materials) Management
Diversité et configurations du produit/process	Configuration Management
Classification des bibliothèques de composant	Classification & Group Technology
Gestion de projet et gestion collaborative	Program Management & Work Breakdown Structure
Visualisation, Partage en entreprise étendue...	Digital Mock-Up, Web Technology, B to B, ...

Tableau 1.2 Synthèse des besoins en GDT et les composants SGDT correspondants

IV Goulets d'étranglement à la mise en œuvre des SIT en entreprise

Après avoir formalisé les besoins et les finalités de la GDT process en entreprise, nous soulevons dans cette partie les problèmes cruciaux rencontrés lors des mises en œuvre des SIT et des SGDT. Ces difficultés principales qui sont aussi dépendantes les unes des autres, une fois explicitées, nous permettent de cerner la problématique de notre thèse.

IV.1 CdC des besoins en GDT : modèles d'expression incomplets et informels

On dit "qu'on achète un système de CFAO, un traitement de texte, mais qu'on met en œuvre une GDT". Il ressort de ces propos, que les solutions de SGDT ne sont pas "prêtes à l'emploi". La mise en service d'un progiciel de SGDT en entreprise, livré généralement comme une coquille vide, passe impérativement par des phases d'analyse et de conception exprimées dans un CdC. Ce CdC doit traduire les attentes du client final représentées par la maîtrise d'ouvrage (spécifications externes). Le CdC doit également décrire la structure des objets du SIT projeté, travail plutôt assuré conjointement par la maîtrise d'œuvre de l'entreprise et le fournisseur du progiciel (spécifications internes). Venus des univers souvent différents, ayant des intérêts et des enjeux parfois divergents sur le projet en question, les acteurs concernés par la conduite du SIT, ont souvent du mal à livrer le CdC dans les délais impartis et avec la qualité exigée. Le CdC résultant est souvent incomplet de par la négligence de certaines spécifications qui peuvent aller au delà de la date de mise en service du produit, comme celles

caractérisant sa maintenance. Les spécifications sont rédigées d'une façon informelle et ne sont pas forcément comprises par tous. Ce qui accentue les difficultés d'échange et de communication et occasionne des retards de validations de ces spécifications.

Nous verrons dans le chapitre suivant quels sont les outils et les solutions techniques nés des efforts de la recherche en ingénierie des SI (et des SIT en particulier) afin de résoudre les difficultés liées à la formalisation des spécifications du CdC.

IV.2 Mode d'emploi : absent ou générateur de ruptures

Le projet de développement d'un SIT en entreprise débute avec la phase d'analyse des besoins exprimés par les différents clients en entreprise. Cette phase est suivie par la conception, la réalisation et enfin la mise en service du produit. Il faut également assurer les phases incontournables d'un SIT à savoir son évolution et sa maintenance.

Le processus de développement de ces SIT en entreprise n'est pas rationnel et présente des ruptures à travers les jalons du projet. Cela conduit à la lenteur globale du projet et à son inefficacité. L'incohérence des modèles de spécifications résultants des différents jalons du projet en est probablement une des causes principales. Ces livrables sont-ils seuls responsables des discontinuités et des ruptures dans le projet ? ou la démarche empruntée en est-elle aussi responsable ?

Qu'il soit soutenu ou pas par le fournisseur du progiciel de SGDT, un projet de mise en place d'un SIT met généralement plusieurs années avant que le premier livrable ne puisse voir le jour. L'expérience des SIT liés au BE (SIGNE évoqué en III.4) sont des exemples appropriés.

Par ailleurs, les développements théoriques du domaine des méthodes de développement des SI n'ont pas vraiment aidé l'industriel à mettre en place son SGDT. Ces méthodes sont jugées comme étant trop directives, empêchant la prise en compte des contraintes environnementales de l'entreprise ou comme étant trop génériques, donc n'aidant pas concrètement la conduite du projet. Nous pouvons néanmoins affirmer que le succès d'une démarche se mesure dans sa capacité à assurer un continuum entre les phases du projet, à pallier la lenteur habituelle de tels projets et aussi à traiter au mieux les "Business Requirements" de l'entreprise, expliqués ci-après.

IV.3 "Business Requirement" : difficulté d'appréhender sa dynamique

On entend par "Business Requirement", l'ensemble des contraintes environnementales de l'entreprise (informatiques, organisationnelles, juridiques, culturelles...). Ces contraintes peuvent être issues de nouveaux besoins qui s'ajustent avec l'évolution des stratégies et des politiques de l'entreprise. La dynamique de cette évolution (naturelle) est difficilement maîtrisée par les projets informatiques en général et les SIT en particulier. Il est en effet difficile d'anticiper sur les besoins et de faire des prévisions justes. Prenons deux exemples pour illustrer cette difficulté.

▪ Question de l'intégration ou non des SIT existants

Une étude récente effectuée par CIMdata à la demande de Renault, a recensé environ 600 applications métiers réparties dans les bureaux d'ingénierie et dans les sites de production. Ces applications¹², comme nous l'avons évoqué au début du chapitre, sont couplées à des BdD locales (structurées ou non). Elle font l'objet de problématiques controversées quant à la politique de leur développement (Encart 1.3). Le projet de mise en œuvre d'une "GDT process fédératrice" pose la question cruciale à laquelle nous n'avons pas de réponse : intégrer ces

¹²Appelées "Legacy Systems" par les anglo-saxons. Il est souvent associé au terme Legacy une connotation négative. Elle traduit la pénalité à payer pour les erreurs du passé (liées à l'architecture des SI...) comme le montre la définition suivante : "Older Information System applications that are crucial day-to-day operation of a business, but which may use technologies considered old or outdated by current standards" [WHI01].

SIT ou concevoir en rupture ? Il est clair que l'éditeur du SGDT sera favorable à une conception en rupture. Cependant, l'entreprise ne peut abandonner du jour au lendemain des applications opérationnelles. Serait-il possible de concilier les deux pôles ?

▪ **Question des changements organisationnels**

Pour illustrer les bouleversements qui opèrent sur la GDT, rappelons un événement majeur survenu en mars 1999 : l'alliance Renault avec Nissan. L'alliance fait que certains projets véhicule doivent se réaliser conjointement autour d'un même plateau. Il va donc falloir concilier les principaux SIT des deux entreprises, à savoir ceux qui gèrent les nomenclatures de pièces et les configurations du véhicule. Ce n'est pas tout, il est décidé aussi d'adopter la meilleure technique de travail des deux entreprises. Prenons l'exemple de la gestion documentaire pour les métiers d'ingénierie de production. Telle qu'elle est pratiquée chez Nissan, la fonction de documentation s'avère plus optimale. Cependant, la reproduire pour les bureaux des méthodes Renault n'est possible à moins de passer du temps à réorganiser les données techniques nécessaires à son élaboration, sans oublier les autres problèmes purement organisationnels et culturels qui surviendront suite à cette décision.

Il ne serait pas possible de développer davantage cette rubrique pour des raisons de confidentialité. Il faut toutefois retenir que le succès de mise en place d'une GDT en entreprise se mesure probablement dans sa capacité à prendre en compte des nouveaux besoins –pas forcément prévisibles- et à réagir facilement face aux évolutions permanentes des "Business Requirements" de l'entreprise.

IV.4 Jeunesse inhérente du domaine de la GDT Process

Nous terminons cet exposé des problèmes cruciaux par la jeunesse du domaine de la GDT. Bien que les premières applications de SGDT datent déjà des années 1990, il manque néanmoins une sensibilisation générale sur l'apport de telles technologies. Si l'industriel s'est finalement laissé convaincre de l'intérêt de la GDT du produit (tout au long de son cycle de vie), paradoxalement, la GDT process est négligée, voire jugée trop peu rentable pour s'en préoccuper. Les industriels restent parfois réticents face à l'introduction des SGDT, jugés "nouveau-nés" et incapables d'appréhender les réalités de la complexité industrielle.

A l'heure actuelle, il n'existe presque pas de travaux qui traitent par exemple des indicateurs de mesure (Metrics pour les anglo-saxons) qui permettent de juger de la performance et de la qualité du SIT durant les différents stades de son développement. De ce fait, les retours d'investissement ne sont toujours pas établis par les acteurs concernés.

Nous verrons au chapitre 2 de la thèse que ces problèmes préoccupent aussi bien les industriels que les chercheurs en ingénierie des SIT. Ils concernent aussi bien les organismes internationaux de normalisation que les éditeurs de logiciels de SGDT.

Le CdC de la thèse sera développé en détail au chapitre 3, après avoir étudié d'abord ce que la recherche en ingénierie des SI/SIT a pu apporter comme aide, ce que les actions normatives ont résolu comme problèmes et enfin ce que les éditeurs de SGDT et les expériences des industriels ont pu nous enseigner.

Toutefois, nous pouvons terminer ce chapitre avec ces mots clés résumant la problématique de la thèse :

- Expression des besoins et formalisation des CdC de la MOA/MOE en GDT Process.
- Proposition d'une démarche rationnelle de conduite de projet de SIT en entreprise, assurant un continuum entre les jalons du projet et une accélération de son processus.
- Dynamique du tout (formalismes, démarche) afin de répondre aux évolutions permanentes des business requirements de l'entreprise.

CHAPITRE DEUXIEME

*Quels outils et quelles méthodes pour l'ingénierie des
systèmes d'information techniques en entreprise ?
Etat de l'art*

*If men could learn from history, what lessons it might teach us! But passion and party blind our eyes, and the light
which experience gives is a lantern on the stern, which shines only on the waves behind us!*

Samuel Taylor COLERIDGE (1772-1834)

I Nouvelle phase de rupture pour les SI en entreprise

I.1 Introduction

Avant de rappeler la problématique de notre thèse, évoquons sommairement l'évolution des systèmes d'information (SI) et leurs nouveaux rôles dans les organisations.

Les premiers SI sont marqués par les paradigmes de type "*construire le SI, c'est construire sa base de données*". Les méthodes de développement de ces SI peuvent bien se résumer aux propos de Le Moigne "*l'analyse à la balayette*"[BAR96]. Ces SI sont conçus pour automatiser les fonctions récurrentes en entreprise, essentiellement de gestion, comme le souligne la définition de Buckingham et al.[BUC87] : "*a system which assembles, stores, processes, delivers information relevant to an organisation, in such a way that information is accessible and useful to those who wish to use it, including managers, staff, clients and citizens. An information system is a human activity (social) system which may or may not involve the use of computer systems*".

Aujourd'hui, de nouveaux rôles stratégiques sont additionnés à ces fonctions classiques. Ainsi, les SI couplés aux Technologies d'Information et de Communication (TIC) aident les entreprises à :

- Développer de nouveaux produits et services de meilleure qualité, dans les délais impartis et au moindre coût.
- Passer du mode d'échange au partage d'information.
- Travailler dans le cadre de l'entreprise étendue (en relation avec les fournisseurs, les clients et les différents partenaires).
- Favoriser les principes de l'ingénierie concourante et privilégier le travail collaboratif.
- Améliorer l'image de la firme et participer au service de l'avantage concurrentiel.

A ce titre, les SI constituent un enjeu stratégique pour les dirigeants des organisations. Leur rôle est passé d'une logique d'automatisation des fonctions vers une vision stratégique, favorisant les capacités d'adaptation de l'entreprise à son environnement. Ce nouveau rôle conduit en outre à modéliser conjointement les trois pôles : l'organisation, sa structure et ses SI, théorie exposée dans l'ouvrage de Tardieu [TAR91] (Figure 2.1).

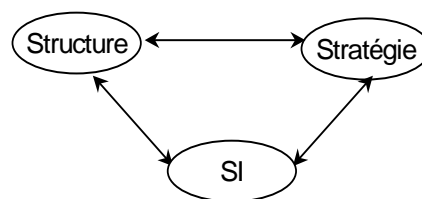


Figure 2.1 Le triangle stratégique : position du SI en organisation (source [TAR91])

Afin de répondre à ces nouvelles exigences, de nouveaux ingrédients sont associés aux SI. On parle alors :

- De relation récursive entre les SI et l'ingénierie des processus connue sous le nom commercial de BPR (Business Process Reengineering) et plus globalement avec l'ingénierie des systèmes et les théories du système général.
- Des supports automatisés au service du développement des SI (Ateliers de Génie Logiciel, outils de prototypages et de test...).

- Des techniques et langages de modélisation : de la modélisation des données aux récentes techniques objets.
- Du SDLC (System Development Life Cycle) : les nouveaux cycle de vie de développement et les méthodes d'ingénierie des SI.

Autant de nouveaux concepts que nous aurons l'occasion d'aborder tout au long de ce chapitre. Mais à présent, examinons d'abord ce qui caractérise les données techniques et les Systèmes d'Information Techniques (SIT) destinés à la productique.

I.2 Ce qui caractérise les Systèmes d'Information Techniques des autres SI...

A notre connaissance, il n'existe pas de texte explicitant formellement la distinction entre les SIT et les SI. Quelques écrits [BAR96;KAL97] rapportent que les SIT se distinguent par leurs nouveaux rôles des premiers SI classiques. Ces derniers, automatisant les fonctions de gestion (comptabilité, analyse financière), sont relativement simples et n'impliquent pas un changement organisationnel en entreprise.

Par ailleurs, les nombreuses discussions –informelles- que nous avons eues avec les chercheurs spécialisés du domaine [MOR94;RIE99...] nous amènent à la conclusion suivante : Un SIT ne devrait pas se caractériser par rapport à sa technologie d'implémentation (SGBD, SGDT, Data Warehouse –Decision Support System-, systèmes experts etc.) mais par la nature des données qu'il transforme en informations utiles, voire en connaissances. Nous identifions la nature de ces données techniques par les caractéristiques ci-après :

- Diversité des types de données : document, modèle graphique, vecteur, matrice, ensemble...
- Composition récursive des objets et notion de nomenclature : "robot" composé de {visseuse, pince...}. "visseuse" composée de {douille, adaptateur...}.
- Multiplicité des versions et évolution de la maturité des données liées au produit manufacturier : Opération "ordonnancée", "équilibrée", "analysée" ou document "soumis", "validé", "rejeté"...
- Hiérarchisation des structures de données : "emballage" est un moyen de conditionnement et qui appartient aux ressources logistiques.
- Classification des objets : données purement Produit, Process ou Ressource
- Diversité des relations entre objets : procédurales, temporelles, ...

Ces exemples nous donnent une idée de la complexité des données techniques et des problèmes de GDT relatés au chapitre 1. Nous avons voulu justement que le chapitre 1 soit le résultat de notre analyse du terrain sans faire référence à la littérature du domaine. Notre étude a ainsi débuté avec des constats liés à l'évolution du domaine de l'ingénierie de production, à la conception verticale des outils d'IPAO et au morcellement du processus d'industrialisation du véhicule. Nous avons conclu la première partie avec les conséquences de ces facteurs cités sur la GDT. La seconde partie est consacrée à l'expression des besoins et des problèmes liés à la GDT. Nous avons souligné dans la dernière partie les goulets d'étranglement liés à la mise en œuvre des SIT en entreprise. Ces goulets d'étranglement constituent la problématique de notre thèse. Nous les rappelons à la fin de ce chapitre après avoir étudié les contributions des recherches du domaine sur la résolution des problèmes cruciaux de l'industriel.

Ce domaine de recherche est large et pluridisciplinaire. Nous expliquons à l'Encart 2.1, comment nous avons structuré l'état de l'art et le plan de ce chapitre.

Encart 2.1 Orientation et plan du chapitre : interdépendance des thématiques

[...] *Un pareil plan est lui-même un demi-ouvrage. Il faut le laisser imparfait si on veut se plaire. Joseph Joubert (1754-1824)*

La structuration de ce chapitre a été difficile à réaliser, notamment en raison de l'interdépendance des travaux liés à l'ingénierie des SI/SIT. Les recherches peuvent en effet appartenir à la fois au domaine de l'ingénierie des SI, du génie logiciel et du génie industriel. Elles s'inspirent à la fois des sciences de la société (cognitives, des organisations) et des sciences de l'ingénieur. Ces recherches font participer aussi bien les industriels que les organismes de standardisation, les sociétés de conseil et les constructeurs de logiciels. Nous répartissons ces travaux en fonctions de leurs intentions en :

- *Technologies d'information et supports d'intégration*

Ces travaux ciblent essentiellement les modes d'échange, de standardisation et les plates-formes d'intégration des données techniques en entreprise. Nous examinons en premier les SGDT et les travaux liés aux fonctions intégrées à ces SGDT. Nous poursuivons avec les autres supports d'intégration issus pour la plupart par des actions normatives.

- *Outils et techniques de modélisation*

Nous présentons ici, les modèles de spécification proposés par la communauté de l'ingénierie des SI et les langages dédiés aux données techniques, nés des projets normatifs. Nous regroupons au second volet, les techniques de modélisation d'entreprise (Enterprise Modeling) pour lesquels les spécialistes de la productique ont beaucoup contribué. Enfin, nous étudions en dernier les caractéristiques des Ateliers de Génie Logiciel, principal outil informatique pour la conduite des développements des SI.

- *Méthodes d'ingénierie des SI et développement des SIT en entreprise*

Nous explorons d'abord les méthodes de l'ingénierie des SI, aussi bien académiques que pratiques. Nous nous spécialisons par la suite en traitant les développements des SIT en entreprise. Nous allons nous apercevoir justement que ces derniers travaux sont très récents et s'inspirent principalement des méthodes et techniques générales pour l'ingénierie des SI.

Nous concluons chaque thématique par un bilan personnel de l'apport et des limitations de ces recherches tout en se positionnant par rapport à notre problématique de thèse.

Signalons toutefois que ces parties ainsi structurées restent liées. Nous le montrons avec l'exemple de la norme STEP. Celle-ci appartient au thème 1 (en tant qu'action normative), au thème 2 (à travers le langage EXPRESS développé au sein de cette norme) et au thème 3 (une démarche est assignée à la construction des modèles normatifs STEP).

Enfin, au delà de cet état de l'art, nous pensons que pour mieux comprendre l'ingénierie des SI, il faut l'étudier de plusieurs points de vue : organisationnel, technique, humain... Le lecteur qui souhaite connaître la genèse de l'ingénierie des SI pourra consulter l'abondante littérature orientée technique [CAU01;WHI01;AVI00] ou encore philosophique [CHE90;HIR89].

II Technologies d'information et supports d'intégration

Les Technologies d'Information et de Communication (TIC) est une expression récente qui désigne la combinaison des systèmes informatiques (hard/Soft) et des technologies de communication (donnée, image, son, réseau...).

Cette dimension s'intéresse particulièrement à l'échange, au partage et à l'intégration des données techniques en entreprise. Ces travaux sont souvent nés des actions normatives et sous l'initiative conjointe des clients utilisateurs (industriels), des éditeurs de logiciels et des organismes de recherche et de standardisation. Nous introduisons d'abord la technologie la plus importante : les SGDT et quelques travaux liés au développement des fonctions désormais intégrées aux SGDT. Par la suite, nous décrivons quelques autres projets normatifs qui contribuent aussi à l'intégration et à la résolution de certains problèmes comme l'interopérabilité.

II.1 Les SGDT et les fonctions intégrées aux SGDT

Nous avons évoqué au chapitre 1, les facteurs clés d'apparition du domaine de la GDT et des progiciels de SGDT. Nous avons donné une définition complète de cette technologie et montré la croissance de l'offre sur le marché. Nous avons également présenté les besoins en GDT en entreprise indépendamment de l'offre progicielle.

A présent, examinons les fonctions majeures des SGDT, discutées dans de nombreux ouvrages [RAN95;MCI95] et papiers de recherche [HAM98]. Elles sont aussi rationalisées grâce aux cabinets de conseil dans la matière¹. Nous adoptons ci-après les noms anglo-saxons des fonctions de SGDT plus usités que les appellations françaises.

Data Vault & Document Management : le "coffre-fort" (traduction littérale de Vault ou Repository) est le cœur même du système. Il permet la consignation, le stockage et la sécurisation des données. A l'aide des fonctions de "Check-in" et de "Check-out", l'accès au coffre est contrôlé avec des droits assignés aux utilisateurs. Ces droits sont référencés sous l'acronyme CRUD (Create Read Update and Delete).

Du moment que les données juste nécessaires et les liens entre les données sont préalablement définis, il est possible alors de créer des documents d'une façon dynamique. Ces documents deviennent des objets en soi, stockés dans le coffre et subissent les même traitements que les données.

Structure & Configuration Management : le module appelé généralement PSM (Product Structure Management) a pour but de construire des structures de graphes composés de nœuds d'objets métiers (document, item, model). Ils ont des caractéristiques communes et sont reliés entre eux avec la relation de composition (part-whole, part-of). Ces structures de graphe appelées nomenclatures, BOM (Bill of Materials) ou encore "Part List", sont des compositions récursives d'objets pouvant décrire le produit à différents niveaux d'abstraction et selon des vues métier. Ainsi, un nœud peut-être un "item" dans les BOM Structurelles (organiques), une fonction dans les BOM fonctionnelles ou un "feature²" dans les BOM

¹ Citons : l'américain CIMdata (www.cimdata.com), le français MT Consulting (www.mtconsulting.fr), John Stark Associates (www.johnstark.com), PDM Information Center (www.pdmic.com) ou encore NKT services (www.nktservices.co.uk)

² En fonction des métiers d'ingénierie de produit/process, la définition de "feature" varie. Nous retenons pour sa clarté, l'explication de GAMA, groupement de chercheurs français affiliés à l'AFCEC. Un feature est donc la description géométrique d'une pièce (article élémentaire) pour laquelle un processus de réalisation est connu.

géométriques. Après le développement de ce module, le besoin s'est vite fait sentir pour gérer les différentes configurations du BOM. C'est le but de la création de la composante CM (Configuration Management)³. Elle permet d'obtenir des vues spécifiques du produit (fonctionnelle, structurelle...), de gérer sa maturité tout au long de sa vie ou de retrouver facilement un composant du produit dans les BOM.

Workflow & Process Management : le workflow est une représentation graphique d'entités et de relations. A chaque entité est attachée une classe et des attributs (responsables, durée de la tâche, coût, description, ressources requises...). Les fonctions attribuées aux workflows se résument à :

- Automatiser les procédures et les processus métier : la gestion de ces processus se fait selon des contraintes que l'on peut spécifier a priori. Elle peut porter sur la nature des tâches, leur enchaînement et la surveillance des coûts et des délais. Lorsque l'on ne connaît pas a priori le déroulement et l'exécution d'un processus, on a alors affaire à des workflows spécifiques "ad-hoc".
- Discipliner la création et le suivi des modifications : une création ou une modification d'une donnée est généralement référencée par un numéro de dossier de modification technique ECO (Engineering Change Order) et initialisée par un ECR (Engineering Change Request). La création ou la modification doit être validée par un responsable de l'ECO. Les approbations/désapprobations s'effectuent par courriers électroniques. Quand les étapes du processus de validation ont été franchies avec succès, les données modifiées, habillées d'une nouvelle révision, rejoignent le coffre-fort et suit alors une notification à l'ensemble des acteurs concernés.

Cependant, les workflows ne gèrent pas encore les procédures complexes (l'ordonnancement des ressources, l'optimisation des chemins critiques...) et ils sont souvent jugés par trop rigides, ce qui fait justement l'objet de quelques travaux de recherche (II.2).

Classification & Group Technology : le problème de classification a été traité, bien avant les outils de SGDT, par le domaine des Technologies de Groupe (TGAO). Le but de ce domaine est de classer les objets métier (pièces, plans, documents...) en fonction de leurs attributs communs afin de favoriser la réutilisation des composants. En effet, il se trouve que 80% des travaux d'ingénierie sont similaires aux travaux précédemment étudiés. D'où l'intérêt de construire des bibliothèques de composants standards, qu'il faudra faire évoluer et maintenir tout en facilitant la recherche et l'accès aux objets.

Program and Project Management : les SGDT peuvent également accompagner les projets grâce à cette composante. Elle permet en effet de planifier les tâches du projet avec des outils de type WBS (Work Breakdown Structure), des méthodes PERT et autres diagrammes de Gantt.

Utilitaires et autres fonctionnalités : les SGDT intègrent de plus en plus les différentes fonctions citées précédemment. Et pour les mener à bien, d'autres utilitaires (d'administration entre autres) sont également intégrés afin de gérer la communication, le transport, la traduction et la distribution des données. Signalons également que les échanges s'effectuent

³ CIMdata [CIM01] le définit ainsi : "The process of managing products, facilities, and processes by managing their requirements, including changes, and assuring conformance in each case".

parfois sous le respect de quelques normes comme CALS⁴ et les fonctionnalités s'ajustent avec le progrès des technologies d'informations. La technologie Web, les architectures Client/Serveur et les outils de visualisation (digital mock-up) sont désormais adoptés par de nombreux progiciels. Beaucoup de travaux de recherche s'intéressent à l'amélioration de ces fonctions comme le montre le paragraphe suivant.

II.2 Travaux de recherche autour des fonctions classiques de SGDT

Avec ou sans la collaboration des constructeurs de SGDT, de nombreux travaux du domaine de l'ingénierie des SIT concernent le développement des fonctions généralement connues comme composantes des SGDT. Citons deux fonctions : configurations du produit et workflow. Signalons que les travaux sur la structuration des données techniques et le développement des SIT en entreprise sont exposés dans la dernière partie de ce chapitre. Nous voulons d'abord présenter les recherches en ingénierie des SI en général, principale source d'inspiration pour les travaux liés aux SIT.

Configurations du produit : essentiellement basés sur les techniques d'Intelligence Artificielle, ces outils sont développés pour rationaliser la configuration de produits. Les constructeurs automobiles sont particulièrement intéressés par de tels travaux pour qu'ils puissent gérer la diversité de leurs véhicules et maîtriser la complexité de leurs spécialisations croissantes. En effet, on compte généralement pour un véhicule moyen environ 2000 fonctions, ajouté à cela les variantes et les options. On obtient souvent pour une combinaison, plusieurs solutions techniques (pièces génériques) qui à leur tour peuvent être traduites en plusieurs pièces physiques différenciées pour des critères comme : couleur de la pièce, nom du fournisseur... Cependant, la configuration du produit ne se limite pas à définir des véhicules "réalisables" du point de vue du CdC Fonctionnel de celui-ci. Il y a également une dimension qui, aujourd'hui, n'est pas vraiment considérée, à savoir la diversité process (introduite au chapitre 1, III.1). Sans chercher à être exhaustifs, quelques références des travaux dans ce domaine sont décrites dans l'Encart 2.2.

Encart 2.2 Quelques travaux dans le domaine de la gestion de configurations du produit

. L'équipe PDMG (Product data Management Group www.cs.hut.fi/~pdmg/) propose un langage de configuration à base de règles CRL (Configuration Rule Language) avec une sémantique déclarative pour définir d'une façon formelle les concepts en configurations de produit.

. B. Wesfechtel et al [WES98] ont examiné les similarités et les différences entre les logiciels de gestion de configurations CM et les SGDT, deux types de logiciels qui ont été développés pendant longtemps indépendamment.

. L'équipe de P. Schönsleben (Swiss Federal Institute of Technology, www.lim.ethz.ch/english/index.html) propose un modèle de configuration commun à toutes les phases du processus de configuration, visant à intégrer les différentes connaissances circulant durant les phases du processus. Ce travail particulier s'inscrit dans le cadre du projet GNOSIS sous l'initiative de l'IMS (Intelligent Manufacturing Systems).

Workflow : le standard d'échange de workflow WfMC (Workflow Management Coalition) [WFM96] le définit comme : *"the automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules"*. Autrement dit, l'objet de cette technologie est de

⁴ CALS (Continuous Acquisition and cycle Life Support) est un programme du "Department of Defense" (E.U). Il définit des normes pour les données techniques (création, transmission, utilisation) en vue de faciliter la composition des documents techniques des systèmes d'armes et plus généralement le dossier technique. Pour plus de détails, cf. l'ouvrage de V. Sandoval [SAN93].

rationaliser l'enchaînement des activités (procédures et processus métier) et le rôle des acteurs. Les premiers systèmes de workflow étaient plutôt rigides car ils préconisaient une connaissance a priori du déroulement des activités. Or, cela est souvent impossible lorsqu'il s'agit d'un processus d'ingénierie. Les rendre plus flexibles et adaptatifs est l'intention de nombreux travaux dans ce domaine (Encart 2.3).

Encart 2.3 Exemples de travaux dans le domaine des workflow

- . Projet ADEPT (Application Development based on Encapsulated Pre-modeled Templates) : l'objectif est de développer un workflow adaptatif supportant en temps réel les changements qui peuvent survenir pendant le déroulement des processus [HEN00].
- . Les travaux de M. Weske [WES99] proposent des workflows adaptatifs basés sur une affectation flexible de divers schémas et instances de workflow.
- . Projet CRISTAL (Cooperative Repositories & Information System for Tracking Assembly Lifecycle) exposé dans [KOV98].
- . Projet SIMNET (Workflow Management for SIMultaneous engineering NETworks) [ROU00] concernant l'intégration des données d'ingénierie au travers des workflow d'ingénierie.

Après avoir parlé de la principale technologie d'intégration et des travaux autour de ses fonctions, décrivons maintenant les quelques autres supports d'intégration.

II.3 Actions normatives et autres technologies d'intégration

Si la plupart des actions normatives sont rassemblées ici, c'est qu'elles se sont souvent intéressées aux thèmes de cette partie, à savoir : les standards d'échange, de partage et des plates-formes d'intégration. Décrivons, sans chercher à être exhaustifs, quelques travaux normatifs et projets internationaux du domaine.

STEP (STandard for Exchange of Product Model Data) : nous introduisons en premier la norme STEP parce qu'elle a été le moteur de plusieurs travaux de recherche. Répertoire sous la référence ISO10303 (AP214) et dédié aux constructeurs automobiles et à leurs équipementiers, le standard STEP est beaucoup plus qu'un format d'échange de données produit. Il possède un formalisme spécifique appelé EXPRESS. Il sert à normaliser les données automobiles avant de pouvoir les partager ou les échanger en suivant une méthodologie de développement spécifique à ce protocole. Nous ne nous attarderons pas sur cette norme largement abordée dans la littérature. Signalons seulement que STEP n'a pas résolu la question cruciale de l'interopérabilité et c'est la raison d'être des travaux qui vont suivre.

AIT-IP⁵ : initié par l'AIT, ce projet a pour objectif de spécifier une plate-forme d'intégration ouverte et interopérable pour la conception et la fabrication (CFAO) des industries aérospatiales et automobiles. Ces travaux ont fourni un ensemble d'outils d'intégration (essentiellement des composants de type service) permettant :

- L'interopérabilité des composants logiciels initialement développés séparément.
- La portabilité des composants logiciels développés sur différentes plates-formes matérielles et logicielles retenues par les partenaires [AIT98].

La plate-forme s'appuie sur STEP pour les modèles de données définissant le produit et la façon de les échanger. Elle est basée sur CORBA pour l'interopérabilité multi plate-forme

⁵ AIT-IP (Advanced Information Technology-Integrated Platform). Le projet est organisé en deux phases :
1) Projet ESPRIT 22148, durée : 05/96-05/97, chef de projet : Daimler-Benz
2) Projet ESPRIT 25285, durée : 09/97-11/98, chef de projet : Renault

indépendante des choix d'implémentation (langages de programmation) et des protocoles de communication. L'architecture du modèle baptisé ITRM (Information Technology Reference Model) et sa plate-forme d'intégration sont présentées sur la Figure 2.2.

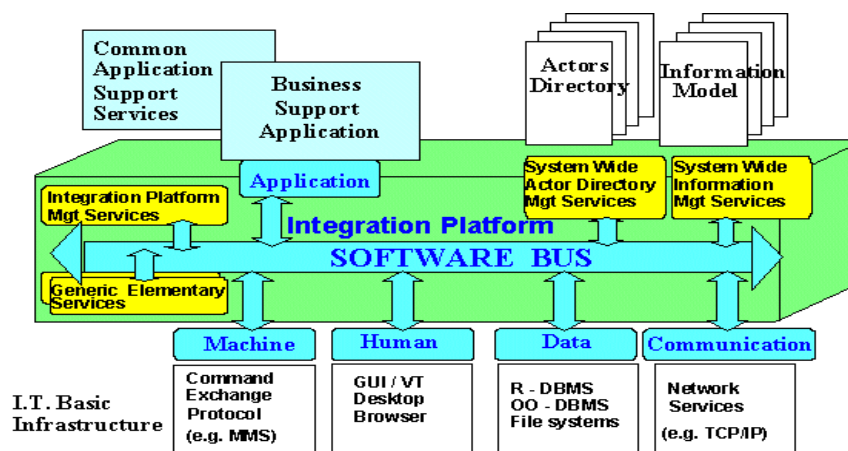


Figure 2.2 ITRM : Modèle de Référence de l'AIT-IP (Source [AIT98] ou www.ait.org.uk)

RISESTEP⁶ (Accès aux bases de données STEP distribuées) : ce projet de recherche cible la spécification et le développement d'une architecture distribuée pour l'échange et le partage d'information dans un contexte de maquette numérique. La pertinence de l'utilisation de BdD STEP distribuées a été testée. Ces BdD sont accessibles à travers un bus logiciel conforme au standard CORBA pour une utilisation en ingénierie concurrente. Pour l'entreprise, de tels travaux représentent un gain d'expérience sur l'utilisation des technologies objet (bus CORBA) et donnent des exemples d'usage des modèles STEP.

OPAL⁷ (Integrated Information and Process Management in Manufacturing Engineering) L'objet de ce projet est [OPA96] :

- Organiser les informations d'ingénierie incluant les données et les connaissances sur le produit/process, en utilisant les technologies modernes d'hyper-media.
- Fournir un accès uniforme à l'utilisateur et un système de navigation dans les informations d'ingénierie à travers l'intranet de l'entreprise.
- Fournir des méthodes définissant les opérations de workflow.
- Fournir des services en ligne de distribution pour accéder rapidement à l'information correcte par les professionnels (des départements d'ingénierie et de fabrication).

Ce projet apporte des concepts pour l'intégration de haut niveau des processus et de l'information, en ingénierie et en production via les nouvelles technologies d'information.

PDM Schema⁸ : cette norme est née d'un effort d'harmoniser les modèles de données issus des protocoles d'application de STEP. Elle est employée pour réaliser l'interopérabilité entre les protocoles AP203 et AP214. Précisons tout de même, qu'elle ne couvre pas la totalité des spécifications fonctionnelles des systèmes de SGDT.

PDM Enablers : est conçue aussi pour assurer l'interopérabilité entre systèmes SGDT/XAO. Des prototypes d'implémentation sont réalisés dans le cadre du projet européen Brite-Euram

⁶ RISESTEP (EnterPRISE wide standard access to STEP distributed databases) : Projet ESPRIT 20459, durée : 01/96-05/98, Chef de Projet : Aérospatiale, pour en savoir plus : www.ait.org.uk

⁷ Projet ESPRIT EP 20377, initiative de l'AIT.

⁸ Pour en savoir plus, cf. www.pdm-if.org/pdm_schema

AIT-SAVE⁹ pour assurer le lien entre les deux SGDT : VPM-Enovia de Dassault-Systèmes et Windchill de PTC.

TIMS (Testability for Interaction-Driven Manufacturing Systems) : est initiée par le NIST (National Institute of Standards and Technology). Son objectif est de rapprocher les deux normes PDM Enablers et PDM Schema, en établissant une correspondance entre les deux standards.

Enfin, comme nous l'avons vu, ces projets ont souvent bénéficié des technologies d'information modernes de type Corba, XML... Nous récapitulons les plus significatives dans le Tableau 2.1.

Technologie	Définition, cas d'utilisation	Intérêt
XML (Extensible Markup Language)	Format de spécification des données. Echange de contenu/contenant de type texte.	<ul style="list-style-type: none"> . Standard W3C, format adopté au Web . Large adoption par les éditeurs de logiciels . Indépendance entre la donnée et la représentation de la donnée grâce à XSL
CORBA (Common Object Request Broker Architecture)	Norme d'interopérabilité pour les applications distribuées. Communication synchrone entre applications	<ul style="list-style-type: none"> . Standard ISO/OMG . Services standards associés (ex: PDM Enablers) . Possibilité de laisser les données à leurs sources
EAI (Enterprise Application Integration) & Message Brokers	Message Broker : événements & messages envoyés par une application source, interceptés & lus par des applis abonnées. Communication synch& asynch	<ul style="list-style-type: none"> . Développement rapide quand les connecteurs sont disponibles. . Des références dans le monde des ERP (SAP, ...) . Possibilité de laisser les données à leurs sources
EJB (Enterprise Java Beans)	Environnement 100% Java, développement de portail Web, communication synchrone	<ul style="list-style-type: none"> . Environnement de déploiement intégré (Application Servers) . Sécurité des transactions
HDF (Hierarchical Data Format)	Solution pour les échanges de données volumineuses	<ul style="list-style-type: none"> . format binaire portable . API gratuit, Gestion rigoureuse des objets
API (Application Programming Interface)	Indépendance par rapport au format	<ul style="list-style-type: none"> . cacher la complexité des normes (ex: STEP) . Meilleure qualité des interfaces grâce à la factorisation
Portail Web Intranet	Diffusion de l'information au sein de l'entreprise	<ul style="list-style-type: none"> . présentation homogène sans besoin d'une formation . Accès synthétisé en temps réel ou différé
Portail Web Internet	Echange des informations avec les partenaires externes	<ul style="list-style-type: none"> . Accélération des communications

Tableau 2.1 Quelques autres technologies d'intégration

II.4 Synthèse : Apport des technologies d'intégration

L'intérêt de ces travaux est de fournir des modes d'échange, de partage et d'intégration des données techniques en entreprise. Nous pensons que les clients potentiels de ces contributions sont plutôt les éditeurs de logiciels (et les constructeurs de SGDT en particulier) que les industriels. Ces derniers, malgré leur forte implication dans bon nombre de projets normatifs, n'ont toujours pas résolu les problèmes cruciaux soulevés au chapitre 1. De surcroît, la question de l'interopérabilité, but central d'un large nombre de projets, subsiste toujours à tel point que l'on a vu fleurir des sociétés spécialisées en traduction de fichiers de CAO par exemple.

⁹ SAVE(STEP in A Virtual Entreprise) : son objectif est de fournir un modèle informationnel supportant l'entreprise virtuelle. Il est basé sur les résultats des projets RISESTEP, AIT-IP et des deux travaux normatifs PDM Schema et PDM Enablers. A ne pas confondre avec le programme SAVE (Simulation Assessment Validation Environment) lancé par "Lockheed Martin" et financé par "Joint Strike Fighter Program Office". Ce dernier s'est focalisé sur le développement et l'implémentation d'un environnement architectural ouvert pour l'intégration des outils de modélisation et de simulation. Cf. skipper.mar.external.lmco.com/save.

Il existe toutefois quelques actions normatives qui ne se sont pas focalisées sur ce seul problème. Deux d'entre elles retiennent particulièrement notre attention : MANDATE (ISO15531) et PSL (Process Specification Language), plutôt classées dans III et IV de ce chapitre et sont donc développées plus tard. En résumé, la Figure 2.3 illustre les rôles de ces différentes technologies en organisation.



Figure 2.3 Les clefs de la "design chain" (adaptée de : Simulog/Esprit Concept, www.simulog.fr)

III Techniques et outils de modélisation

Nous avons choisi de séparer les *techniques* de modélisation des *méthodes* de développement des SI qui sont exposées en IV. Celles-ci peuvent être décrites sans rentrer dans le détail des techniques qu'elles emploient. De plus, une technique est souvent adoptée et/ou adaptée dans plusieurs méthodes. Appelées aussi langages, formalismes, notations ou modèles de spécification, les techniques ont fait l'objet de plusieurs études pluridisciplinaires. Il est difficile de proposer une classification de ces techniques de par les interactions entre les disciplines. En effet, plus que jamais, les informaticiens ne conçoivent plus de SI sans faire appel aux gestionnaires des organisations et aux spécialistes en sciences cognitives. De même, pour une meilleure compréhension de l'entreprise moderne, les spécialistes en productique et du CIM (Computer Integrated Manufacturing) empruntent les techniques informatiques et les AGL (Ateliers de Génie Logiciel) pour modéliser les objets de l'entreprise.

En premier lieu, nous traitons les techniques de modélisation des données, des flux et des processus, résultats des recherches en ingénierie des SI.

Nous regroupons dans le second volet, les architectures de référence¹⁰ et les techniques de modélisation d'entreprise (Enterprise Modeling), travaux pour lesquels les spécialistes de productique et du génie industriel ont beaucoup contribué. En simplifiant, le premier thème englobe les diagrammes que l'on emploie durant un projet de développement d'un SI. En revanche, le deuxième propose des métamodèles sur lesquels on peut s'appuyer pour construire les modèles propres d'un SI, d'une entreprise ou plus généralement d'un système.

Nous examinons en dernier, les principales caractéristiques des ateliers de génie logiciel nommés CASE Tools (Computer-Aided Software Engineering) qui ont permis d'instrumenter quelques techniques et architectures de références. Nous terminons cette partie par un bilan où nous analysons l'apport des techniques exposées.

¹⁰ Ce domaine couvre les mots clés : Enterprise Framework, ISA (Information System Architecture) ou encore Enterprise Architecture...

Soulignons que notre but n'est pas de faire l'inventaire exhaustif des techniques mais d'aborder celles que nous jugeons potentielles pour répondre à notre problématique de thèse.

III.1 Modèles et langages de spécification

Dans le contexte de l'ingénierie des SI, un modèle est une représentation des concepts, défini avec une syntaxe et une sémantique. Il peut être modifié, validé et évalué pour sa robustesse et sa rigueur. Un modèle capitalise des informations et communique des idées [POPO0].

Les spécialistes français comme Cauvet et al [CAU01] ont recensé cinq modèles fondamentaux de spécification, jugés être à la base des nouvelles techniques d'aujourd'hui. Il s'agit des modèles : 1) Relationnel, 2) Entité-Association, 3) Flux d'information 4) Automates d'états finis et enfin 5) Réseaux de Pétri. Les deux premières techniques de modélisation statique de données ont dominé dans le monde professionnel jusqu'à l'émergence des techniques "Objet". Les trois dernières couvrent l'aspect dynamique de modélisation des entités et complètent ainsi les deux premières.

La communauté anglo-saxonne propose une classification peut-être un peu plus exhaustive que la précédente en mettant l'accent naturellement sur les techniques dont ils sont pionniers. Ainsi, nous retenons la classification proposée par Avison et al [AVI00] organisée chronologiquement par ordre d'utilisation dans les phases du développement d'un SI.

Citons d'abord les trois premières techniques, employées dans les phases très en amont de l'analyse de SI : dessins vulgarisés, définitions fondamentales et modèles conceptuels¹¹.

(1) *Rich pictures* : des diagrammes "caricaturaux" qui emploient des symboles clairs et compris. Les concepts modélisés sont généralement des éléments de structure du domaine (problème) que l'on souhaite étudier. Ce type de diagramme aide les acteurs à discuter leur propre rôle en entreprise, explicite les conflits et les aspects politiques du système, éléments difficiles à appréhender formellement. Les faits modélisés peuvent être subjectifs, voire jugés insignifiants. A l'instar des activités de brainstorming, nous ne devrions pas critiquer ces faits. La Figure 2.4 montre ce que l'on perçoit comme faits dans le processus de développement d'un SI en entreprise.

(2) *Root definition* : description verbale et concise des deux concepts vagues et complexes : problème et système, à l'aide des six items suivants : acteur, client, propriétaire, transformation, environnement et hypothèse¹². Ces items changent de sémantique en fonction de l'auteur qui énonce le problème comme le montre la Figure 2.5.

(3) *Conceptual models* : notons d'abord que ce terme a été abusivement employé dans certaines méthodes pour désigner les diagrammes Entité-Association (E-A). Les modèles conceptuels se construisent généralement depuis les deux techniques précédentes qui s'intéressent à la dimension structurelle du système. Les modèles conceptuels quant à eux, décrivent les processus et les liens logiques entre activités au sein du système. Ils se présentent comme outils de communication entre les designers du SI et les clients finaux. Sans se préoccuper toutefois de l'implantation du SI, les modèles conceptuels servent aussi comme instructions que le constructeur du SI doit suivre. Cela veut dire que les acteurs intervenant dans le développement du SI doivent comprendre ce type de modèles.

Les mécanismes de décomposition fonctionnelle et les niveaux d'abstraction dans la représentation peuvent être employés. La Figure 2.6 décrit un aspect de développement d'un

¹¹ Traduction libre de : rich pictures, root definition and conceptual models. Là aussi, nous garderons par la suite les appellations d'origine afin d'éviter toute ambiguïté.

¹² Hypothèse : traduction libre du mot allemand : Weltanschauung.

SI en entreprise en s'intéressant non aux activités traitées dans le SI mais à son propre déroulement.

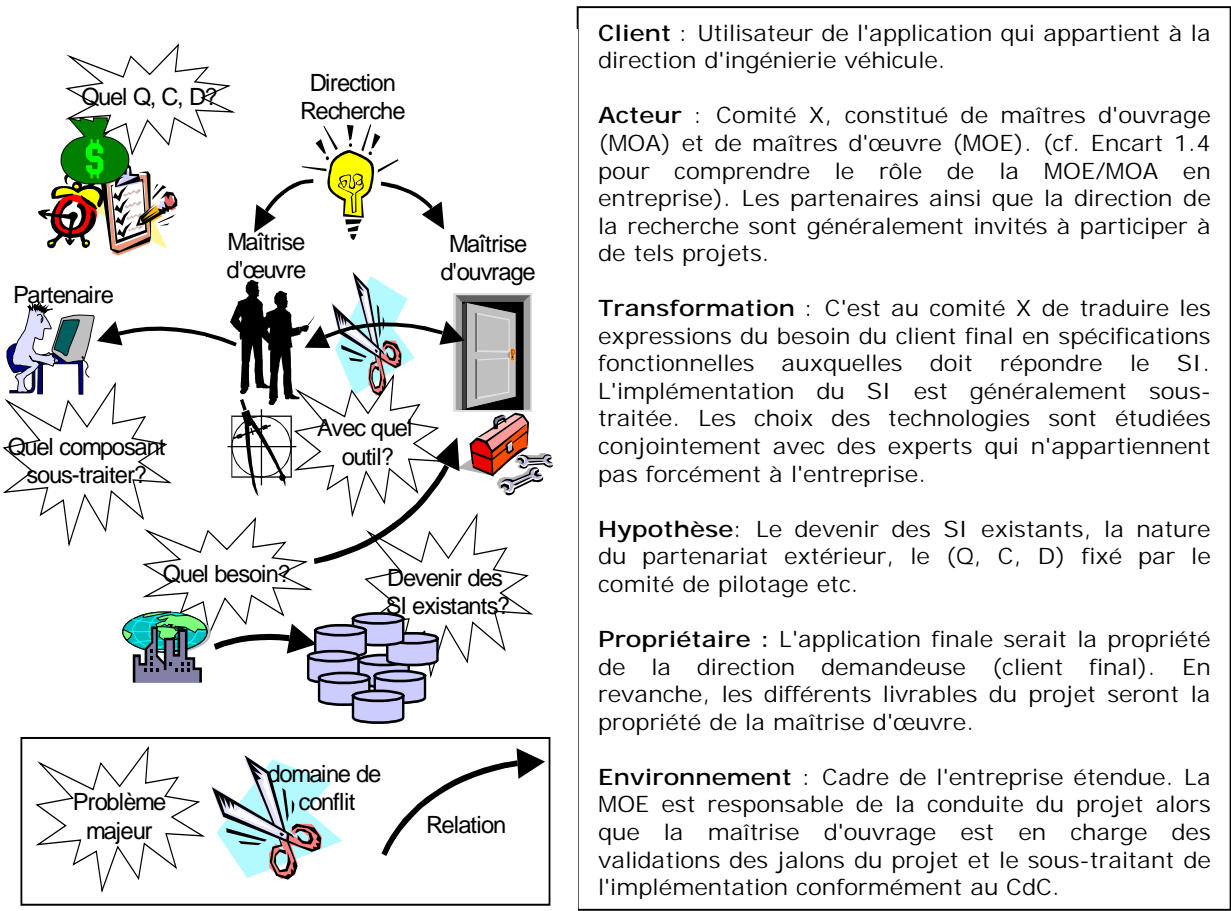


Figure 2.4 Contexte de développement d'un SI en entreprise. Description graphique à l'aide du diagramme "Rich Pictures" (à gauche)

Figure 2.5 Contexte de développement d'un SI en entreprise. Forme narrative à l'aide de la technique "Root Definition" (à droite)

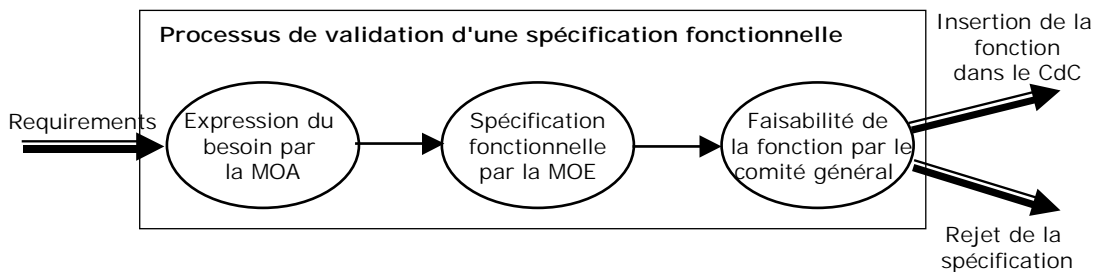


Figure 2.6 Processus de validation d'une spécification fonctionnelle d'un SI en entreprise. Diagramme "Conceptual model"

Les principales techniques suivantes sont plus populaires que les précédentes et dédiées aux phases d'analyse et de conception : modèles centrés données et modèles de processus/flux. Comme leur nom l'indiquent, les modèles de données relèvent du domaine de l'ingénierie de l'information¹³ où s'inscrivent les modèles relationnels, E-A et les normalités. Les modèles de processus sont issus du domaine de l'analyse structurée¹⁴.

Ces deux techniques sont fortement liées les unes aux autres. Nous verrons, en effet, que pour construire les diagrammes de processus, il est nécessaire de connaître les données requises pour étudier leurs flux et transformations. Il en est de même pour les techniques d'analyse de données qui nécessitent une compréhension au préalable des processus que les données supportent.

(4) *Modélisation des entités et normalisation* : basés sur la théorie des ensembles, les modèles relationnels sont connus surtout dans le domaine des Bases de Données (BdD)¹⁵. Ils ont pour objet de structurer les données d'une organisation sous forme de schémas conceptuels compréhensibles par les utilisateurs et se matérialisent en BdD relationnelles. Chaque BdD est constituée de tables (relations) et l'extraction des informations se fait à l'aide des langages assertionnels¹⁶.

Le processus de *normalisation* consiste en l'application d'un nombre de règles aux modèles relationnels pour les rendre faciles à comprendre et à manipuler mais aussi pour réduire les anomalies qui peuvent survenir dans les BdD relationnelles. Cela entraîne toutefois l'accroissement du nombre de tables. Nous ne nous attarderons pas dans la description des techniques de normalisation¹⁷ qui sont déjà utilisées dans bon nombre de méthodes aussi bien françaises (MERISE) qu'anglo-saxonnes (STRADIS, SSADM) (Encart 2.10).

Ces modèles sont relativement simples à réaliser mais l'expression d'associations entre ces relations est limitée aux contraintes référentielles (associations de type 1-n). Pour exprimer les associations de type n-p, il faut créer des relations (tables) supplémentaires, ce qui complique la maintenance des BdD relationnelles standards.

C'est à P. Chen [CHE76] que l'on doit la définition des modèles E-A en s'appuyant cette fois sur deux concepts : entité et relation. Ces modèles -qui ont fait l'objet des travaux significatifs en France- tentent de séparer la structure de donnée des traitements qu'elles peuvent subir. Ils facilitent l'expression des associations et pallient ainsi les difficultés rencontrées avec les modèles relationnels.

¹³ Traduction de Information Engineering (IE), domaine définit comme "A model driven and data-centered, but process-sensitive technique to plan, analyze, and design information systems"[WHI01].

¹⁴ Approche orientée processus employée en analyse des systèmes. Gane/Sarson et DeMarco/Yourdon sont les premiers inventeurs des "Diagramme de Flux de Données", modèles populaires depuis plus de 20 ans. L'approche aujourd'hui est revêtue de nouveaux concepts grâce aux techniques de BPR (Business Process Reengineering/Redesign). Hammer et al [HAM93], rappelons-le, sont les premiers à formaliser cette technique et la définissent comme "the fundamental rethinking and radical redesign of business processes to achieve dramatic improvements in critical, contemporary measures of performance, such as cost, quality, service and speed". L'optimisation des processus dont on parle peut se faire sans forcément avoir recours à leurs automatisations.

¹⁵ Parmi les nombreux spécialistes du domaine, citons le français G. Gardarin [GAR83] et l'anglo-saxon C.J. Date [DAT95].

¹⁶ Ils sont fondés sur les langages algébriques et prédicatifs et étaient à la base des langages opérationnels tels que : SQL (Structured Query Language) et QBE (Query By Example).

¹⁷ E.F. Codd, premier pionnier dans la matière, a défini un processus de normalisation à trois étapes introduit dans son célèbre papier [COD70]. Par la suite, des extensions sont proposées par W.A. Kent [KEN83] et dans la dernière édition révisée de l'ouvrage de C.J. Date [DAT95].

Nous résumons les principales actions des techniques de modélisation de données en ces points suivants (Figure 2.7) :

- Comprendre le domaine d'analyse, collecte d'informations
- Décliner les entités du domaine et les relations entre entités
- Etablir les clés qui vont identifier d'une façon unique ces entités
- Renseigner chaque entité avec des propriétés, appelées attributs
- Normaliser les schémas de données avec des règles de normalisation
- S'assurer que les événements (transactions) et les opérations (traitements) qui peuvent survenir sur le système soient supportés par le modèle.

Nous pouvons déjà retenir de ces techniques que :

- Le modèle de donnée peut être une *fin en soi* ou destiné à *la construction des BdD*. La partie persistante qui est le modèle physique se matérialise en BdD et la partie logique indépendante de son implantation, traduit le monde réel.
- Le processus de construction de ces modèles est *itératif* : dans le but de traduire le plus fidèlement la réalité, il est nécessaire d'optimiser le modèle de donnée en réexaminant les entités et les associations (type, cardinalité...).
- La modélisation des données n'est pas une science exacte ; on peut en effet obtenir pour une organisation donnée, plusieurs modèles cohérents qui la représentent.
- Dans certaines méthodes, des patrons de documentation sont utilisés afin de spécifier au mieux les éléments du modèle : entités, attributs, relations, événements ou opérations (Figure 2.8).
- Enfin, il est généralement difficile de séparer les données des traitements qu'elles peuvent subir. Ce qui explique probablement l'importance accordée à l'étape f). Nous verrons plus tard à l'aide d'exemples, que la structuration de données en organisation est fortement liée à l'analyse de ses processus et les modèles de données ne sont pas adéquats pour l'analyse des flux et des processus. Les diagrammes suivants sont les plus appropriés.

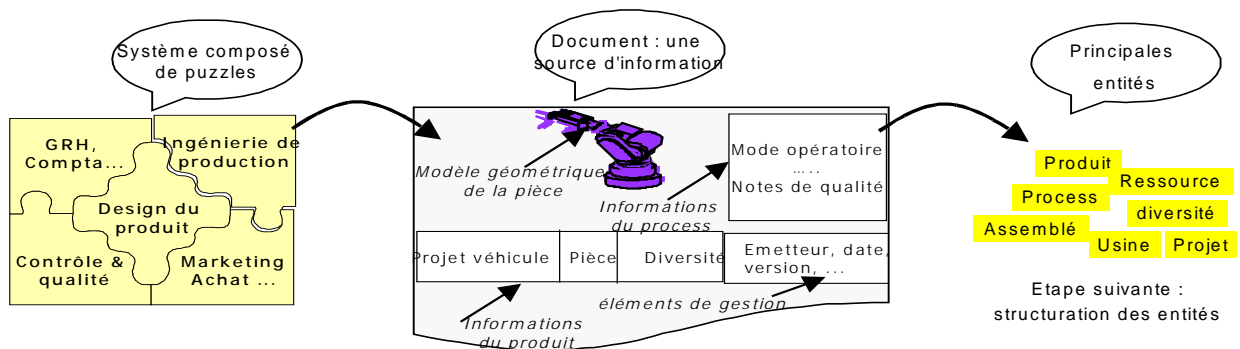


Figure 2.7 Techniques de modélisation des données : de l'analyse du système à la conception détaillée du modèle de donnée

Formulaire de spécification d'une entité	
Nom :	appellation standard de l'entité
Description :	description de l'entité
Synonymes :	autres noms qui désignent l'entité
Identifiant:	clé unique de l'entité
Date :	
Plage d'occurrences :	minimum, maximum
Utilisateurs :	profils pour créer, modifier... l'entité
Relation impliquée :	dans le diagramme E-R
Fonctions concernées :	qui requièrent cette entité
.....	
Commentaire :	

Figure 2.8 Patron de documentation d'une entité (adaptée de [AVI00])

(5) Diagrammes centrés processus

Dans le domaine de l'analyse structurée (cf. note de bas de page 14), un système peut être vu comme un processus opérant dans un environnement complexe. L'exécution du processus requiert des données d'entrées (Inputs) et produit des résultats (Outputs) (Figure 2.9). Quelques définitions du domaine de l'analyse des processus sont récapitulées à l'Encart 2.4.

Présentons maintenant quelques diagrammes les plus populaires en utilisant le contexte de l'ingénierie de production comme exemple.

Encart 2.4 Principes et termes clés dans la modélisation des processus

Nous retenons ici les définitions de Whitten et al [WHI01]. Nous introduisons la modélisation des processus, activité intervenant durant l'analyse du système. Nous nous penchons sur le principe de décomposition et enfin sur les termes "processus" et "fonction" pour les distinguer désormais.

Process modeling : est une technique pour organiser et documenter les flux de données à travers les processus du système. Elle permet de visualiser la logique, les politiques et les procédures implémentés dans ces processus.

Décomposition : est l'acte partitionner un système en sous-systèmes, processus et sous-processus. Chaque niveau d'abstraction indique plus ou moins de détails du système global ou d'un sous-ensemble de ce système.

Processus : est le travail effectué en réponse à une donnée d'entrée ou à un événement (Figure 2.9).

Fonction : est un ensemble d'activités connexes et continues d'un métier. Une fonction n'a pas de début ni de fin. Elle s'exécute simplement sans interruption et quand c'est nécessaire.

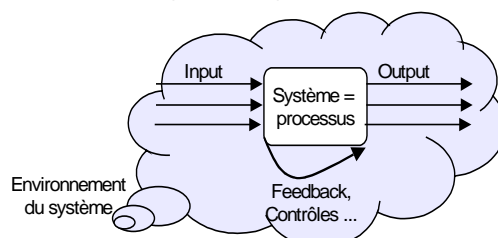


Figure 2.9 Modèle de processus classique

Diagramme de Flux de Données (DFD: Data Flow Diagrams) : décrit les flux des données et des processus exécutés par le système. Il était à l'origine des standards comme IDEF0 (SADT) [COL93] et de plusieurs autres modèles de processus. Un DFD permet de partitionner le système en unités indépendantes. Il décrit plus la partie logique du système que son implémentation physique. Autrement, il décrit ce que le système *va faire plutôt que comment il sera fait*. Grâce à sa représentation graphique et concise, il peut être un outil de communication entre l'analyste et le designer. La Figure 2.10 illustre un exemple d'utilisation d'un DFD en empruntant la notation de Gane/Sarson, plus connue que celle de DeMarco/Yourdon [POP00].

Arbres et tables décisionnels : sont utilisés en présence d'alternatives de décision dans le système. Les arbres décisionnels se construisent en déclinant les conditions (situations possibles) et les actions à prendre pour chaque point de décision (Figure 2.11). Les tables décisionnelles expriment sous forme de tableaux les conditions/actions d'un problème écrites sous une forme narrative.

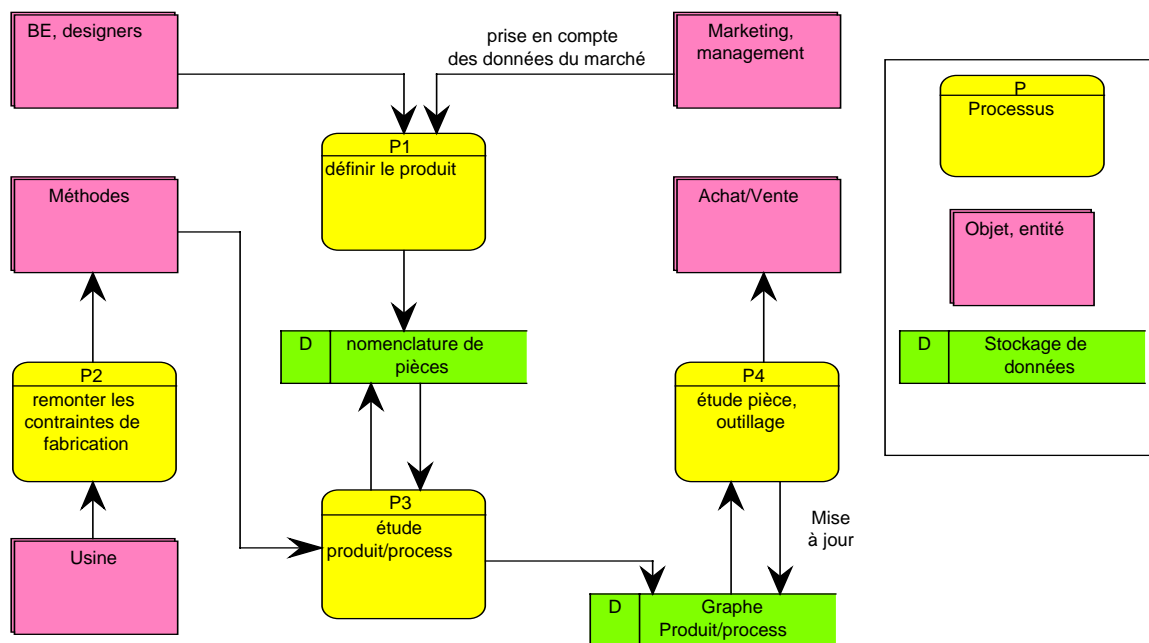


Figure 2.10 Conception Produit/Process en entreprise manufacturière. Diagramme DFD

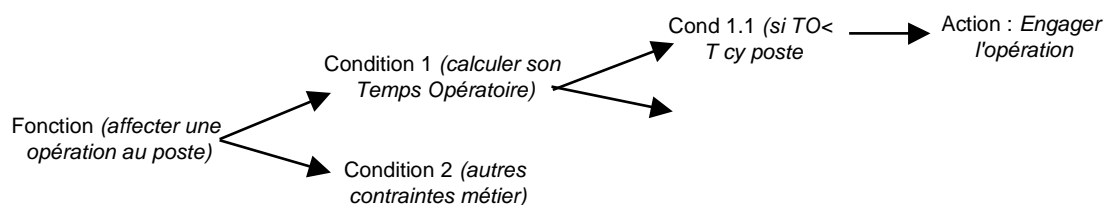


Figure 2.11 Alternatives de décision pour l'affectation d'une opération au poste. Arbre décisionnel

Cycle de vie des entités (Entity life cycle) : contrairement à son nom, il s'agit bien d'une technique d'analyse des processus et non pas des données. La technique montre les changements des états d'une entité, déclenchés par un événement ou un processus. Ces diagrammes décrivent l'aspect dynamique dans la modélisation, non abordé jusqu'à présent par les techniques précédentes. La Figure 2.13 montre un exemple des états possibles de l'entité "Opération" dans l'environnement d'ingénierie de production.

Matrices : utilisées à différents stades de développement d'un SI, les matrices aident à compléter et à documenter les relations entre entités (données, événements, processus, fonctions...). Les matrices peuvent alors jouer un rôle de synchronisation entre les modèles de données et les diagrammes de processus habituellement étanches entre eux. Les matrices de type CRUD (Create Read Update Delete) sont communément utilisées dans les méthodes de développement des SI. Elles peuvent décrire par exemple, les actions possibles (représentées avec CRUD) qu'une entité se voit assignée. Les entités peuvent être par exemple les applications logicielles et les données qui y sont manipulées comme l'illustre la Figure 2.12.

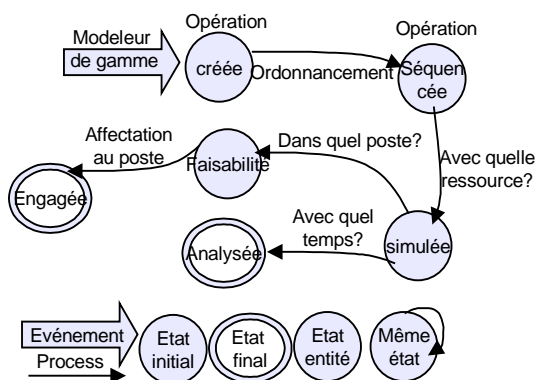


Figure 2.13 Cycle de vie de l'entité "opération" en ingénierie de production.

	Modeleur de gamme	Implantation d'atelier	Simulation de flux
Pièce	RUD	R	R
Opération	CRUD	R	RU
Ressource	CRUD	RUD	RU

Figure 2.12 Donnée-Application : matrice CRUD

Terminons l'investigation avec la technique la plus récente et la plus prometteuse : l'Objet

(6) *Les techniques Orientées-Objet (OO)* : l'originalité de ces techniques –qui réutilisent toutefois les principes des modèles existants- est leur intégration de presque toutes les dimensions abordées dans les techniques antérieures (1, 2, 3, 4 et 5) : la conceptualisation préliminaire, la modélisation des données et des processus. Les techniques sont largement discutées dans la littérature du domaine¹⁸. De ce fait elles ne seront pas détaillées ici. Il est néanmoins intéressant de noter quelques définitions de base dans l'Encart 2.5.

Par ailleurs, l'organisme international de standardisation OMG (Object Management Group) a unifié récemment ces techniques en notation universelle baptisée UML (Unified Modeling Language) [OMG99]. Cette notation est composée de neuf diagrammes (Figure 2.14) livrés sans "mode d'emploi". Nous verrons justement qu'il y a un nombre impressionnant de méthodes basée sur les techniques OO (Encart 2.9).

¹⁸ Citons par exemple les ouvrages [AVI00;BOU00;JAC95;HAR98;KET98;MOR00;OUS99], le site web <http://www.cetus-links.org/> et le forum de discussion "comp.object".

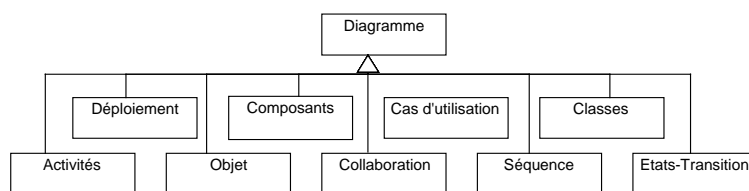


Figure 2.14 Les diagrammes UML

Encart 2.5 Définitions de base du domaine de l'objet

Work without hope draws nectar in a sieve, and hope without an object cannot live. Samuel Taylor Coleridge (1772-1834).

Les définitions suivantes sont empruntées aux références citées dans la note de bas de page 18.

Modélisation objet (Object Modeling) : technique d'identification des objets et des relations entre objets dans un système.

Objet Métier (Business Object): cœur du système, l'objet métier représente le métier de l'entreprise, indépendamment de son implantation technique ou de son utilisation.

Analyse orienté-objet (OO Analysis) : l'analyse objet consiste à étudier les objets existants pour voir s'ils peuvent être réutilisés ou adaptés pour de nouvelles utilisations. Elle permet en outre de définir de nouveaux objets pouvant être combinés avec les existants pour produire une application informatique métier.

Conception orienté-objet (OO Design) : consiste à affiner et à spécifier les objets identifiés dans la phase d'analyse. Cette technique peut être utilisée aussi pour la conception et l'implémentation des applications objet.

L'ensemble des modèles discutés dans cette partie sont classés dans le Tableau 2.2.

Stade du projet	Données et processus	Données	Processus
Stratégie	1) Rich pictures		
Analyse	1) Rich pictures 5) Matrice 6) Objet	4)Entité-Relation 4) Relationnel	5) DFD – Cycle de vie des entités 2) Root definition 3) Conceptual models
Conception logique	5) Matrice 6) Objet	4) Normalisation 4)Entité-Relation	5) Diagrammes de Processus
Implémentation	5) Matrice 6) Objet	4) Normalisation	5) Diagrammes de Processus

Tableau 2.2 Classification des techniques (source [AVI00])

Avant de refermer ce volet, jetons un coup d'œil sur les langages de modélisation des données techniques. Nous avons évoqué plus haut (II.3) le langage Express développé dans le cadre du standard STEP. Basé sur le modèle Entité-Relation Etendu, STEP autorise le mécanisme d'héritage (généralisation/spécialisation) et permet de traduire les diagrammes IDEF1x [BRA85] en programmes informatiques. Nous reprenons les arguments présentés dans la thèse de Gzara [GZA00] pour justifier de la non adéquation du formalisme EXPRESS à l'ingénierie des SIT. Par ailleurs, le langage PSL (Process Specification Language) a bien attiré notre attention. Développé par le NIST dans le cadre du projet SIMA¹⁹, PSL est un langage de

¹⁹ Systems Integration for Manufacturing Applications. Cf. <http://www.mel.nist.gov/psl>

spécification de process (process regroupe ici toutes les tâches de production : usinage, maintenance...) [GRU00].

Le projet de développement du langage PSL n'est qu'à ses débuts, néanmoins son CdC est clairement défini. Il doit être suffisamment générique pour être indépendant des applications logicielles de fabrication et robuste pour pouvoir représenter les informations du process circulant entre applications. Si l'implémentation informatique du langage nous semble non adaptée à notre problématique de thèse, son ontologie par contre peut vraisemblablement apporter de la valeur ajoutée à notre contexte. On en reparlera au chapitre 3 lors de l'exposition des choix de solutions et des orientations de la thèse.

III.2 Architectures des SI et cadres d'entreprise

Le terme Architecture de SI (ISA: Information System Architecture) peut-être défini comme un cadre unifié dans lequel les acteurs dotés de vues différentes peuvent organiser la construction des blocs fondamentaux (composants) du SI [WHI01].

J.A. Zachman [ZAC97] définit l'Architecture d'Entreprise (Enterprise Architecture) comme *"...The set of descriptive representations (ie, models) that are relevant for describing an Enterprise such that it can be produced to management's requirements (quality) and maintained over the period of its useful life (changed)"*.

Le cadre d'architecture d'entreprise (Enterprise Architecture Framework) est un schéma de classification de modèles pour la conception d'artefacts. Il permet à l'utilisateur de se concentrer sur des aspects choisis de l'entreprise sans pour autant perdre le sens global (holistique) du système. Un cadre divise l'énorme quantité de connaissances d'une entreprise en gros morceaux maniables²⁰.

Les définitions liées à la modélisation en entreprise (Enterprise Modeling) sont relativement stables. Retenons celle de F. Vernadat [VER96] : *"Enterprise modeling (EM) is the art of externalising enterprise knowledge which can be reused or add value to the enterprise... EM describes various aspects of enterprise structure, behaviour and organisation"*. Autrement dit, "Enterprise Modeling" a pour objet la construction de modèles d'une partie déterminée d'une entreprise pour en expliquer la structure et le fonctionnement ou pour analyser le comportement.

Le but des architectures de références des systèmes (SI, entreprises, systèmes de production...) est non seulement de prendre en compte toutes les dimensions (vues) du système dans la modélisation mais surtout de pouvoir assurer la meilleure intégration de ces modèles qui finalement constituent les puzzles du système.

Introduisons maintenant quelques cadres de modélisation que nous classons par thème de recherches.

Du monde de la productique et du CIM

GRAI-GIM : Doumeingts de l'université de Bordeaux est le père technique de GRAI [DOU93]. Il s'agit d'un cadre de modélisation d'entreprise, décomposant celle-ci en quatre sous-systèmes : informationnel, décisionnel, physique et opérationnel. Il utilise pour décrire ces composants les modèles de spécification IDEF0, modèles de la méthode MERISE et GRAI (formalisme inventé par la communauté pour représenter le système décisionnel). L'apport de

²⁰ Zachman le définit comme : "The framework as it applies to Enterprise is simply a logical structure for classifying and organizing the descriptive representations of an enterprise that are significant to the management of the enterprise as well as to the development of the Enterprise's systems. It was derived from analogous structures that are found in the older disciplines of Architecture/Construction and Engineering /Manufacturing that classify and organize the design artefacts created over the process of designing and producing complex physical products (e.g. building or airplanes)"

ce cadre réside essentiellement dans la prise en compte de la dimension décisionnelle en entreprise. Cependant, les modèles utilisés sont redondants et mal intégrés.

PERA : caractérisé par son architecture en couches, PERA s'intéresse particulièrement à l'ingénierie des systèmes industriels et tend à couvrir la totalité du cycle de vie du système. Seulement l'aspect fonctionnel et informationnel sont pris en compte dans ce cadre [WIL93].

CIMOSA (CIM Open System Architecture) : l'architecture CIMOSA est vraisemblablement complète [VER99]. Elle prend en compte les différentes vues de l'entreprise, à des niveaux d'abstraction ainsi que le cycle de vie de construction d'un modèle particulier. Cependant, en testant les quelques ateliers logiciels²¹ s'inspirant de ce cadre théorique, il s'avère que ces ateliers ne traduisent que partiellement les principes du cube CIMOSA (Figure 2.15).

GERAM²² (Generic Enterprise Reference Architecture and Methodology) : partant des résultats des travaux sur CIMOSA, GIM et PERA etc., le cadre GERAM se veut être un référentiel fournissant une terminologie, un environnement de modélisation cohérent et une méthodologie détaillée.

Du Génie Industriel

BASE-PTA : une norme issue des travaux du projet PTA (Poste de Travail de l'Automaticien). Elle propose un cadre général pour décrire les données manipulées par les outils XAO. Ce cadre sert de base pour élaborer des normes d'échange de données entre les outils XAO, en supportant l'évolution de données tout au long d'un projet d'entreprise [ND100].

Du Génie Logiciel

ARIS (ARchitecture for Integrated IS) : s'inspirant du cadre CIMOSA, l'architecture ARIS a pour objet de réaliser un *SI intégré* (plus particulièrement pour les systèmes de pilotage d'ateliers). De ce fait, nous classons cette architecture dans le génie logiciel et non pas en productique. Ses principes sont supportés par l'atelier logiciel ARISoolset (Figure 2.16). Il y a eu des tentatives de concilier les vues du SI en construisant une passerelle entre l'outil Rational Rose qui supporte la notation UML et ARISoolset. Cependant, on reproche souvent au cadre ARIS, la non flexibilité de la démarche imposée, qui convient plutôt pour le développement des applications logicielles et moins pour la modélisation d'entreprise.

²¹ Les ateliers logiciels testés sont CIMTool et Firststep (www.rgcp.com/pub/download/cimtool.zip & www.interfacing.com).

²² Développé au sein du groupe IFAC-IFIP Task Force, cf. leur livrable : GERAM, version 1.4, ISO TC184/SC4/WG5, N35, 1992.

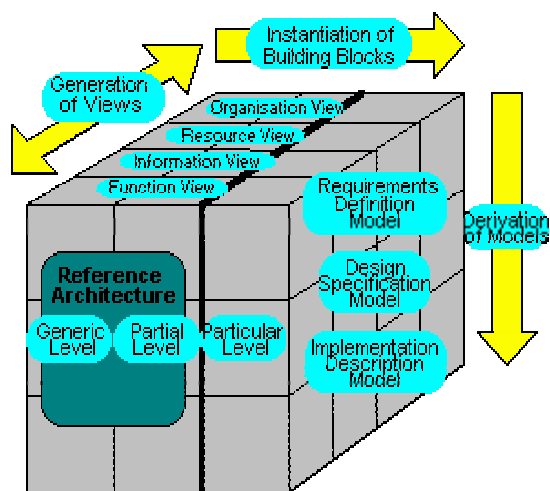


Figure 2.15 Cube CIMOSA (source : <http://cimosacnt.pl> à gauche)

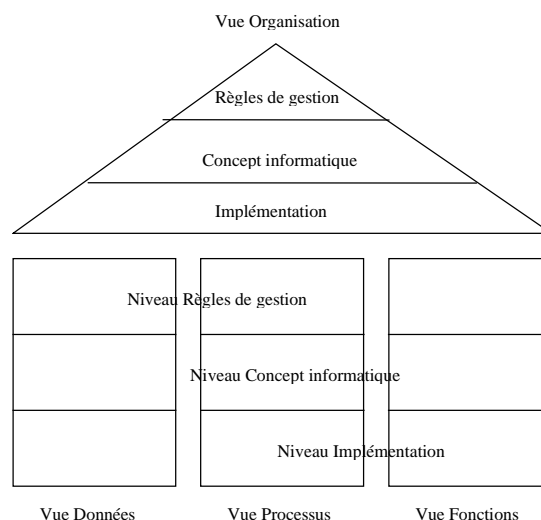


Figure 2.16 Maison ARIS (source : www.ids-scheer.de à droite)

Zachman : J.A. Zachman a publié pour la première fois les principes de son cadre : Zachman Framework en 1987 [ZAC87], mis à jour par la suite en 1992 [SOW92]. Il est parti du constat qu'il existe déjà une multitude de diagrammes de spécifications qui décrivent bien une dimension ou un aspect d'un système. Mais ces modèles existants ne communiquent pas entre eux et donc n'assurent pas une continuité entre les vues du système, dans les niveaux de granularité (générique, spécifique) ou encore dans le cycle de vie de son développement. Ce qui fait que l'on perd la compréhension globale du système et son développement est ralenti.

Il définit son cadre ainsi : *"is a true framework (that is, based on rules of classification) which hypothesizes a comprehensive set of descriptive representations of an enterprise. It is based on analogous descriptive representations of the physical products found in older disciplines of classical architecture and/or manufacturing. The logic construct is a matrix comprised of perspectives (planner, owner, designer, builder, subcontractor and user) versus interrogatives (what, how, where, who, when, and why)"* [ZAC97].

On comprend alors que le cadre de Zachman n'est pas une méthodologie mais un schéma de classification des modèles de spécification existants dans la matrice carrée de la Figure 2.17. Signalons que les usages de ce cadre ont démontré leur adaptabilité aussi bien pour développer des applications académiques qu'industrielles²³. Ce cadre est plus populaire dans le domaine de la conception et la construction des Data Warehouse [INM97].

L'Encart 2.6 explique plus de détails sur ce cadre.

²³ Une application en Industrie Aérospatiale est exposée dans : J. Karlin and D. Welch, A framework of Mission Operations System Models - 1996 - www.esoc.esa.de/external/mso/spaceOps/1-10/1-10.htm. Comme applications académiques, notons le programme de cours basé sur le cadre de Zachman de P.J.A. Van Vliet [VLI97], l'ouvrage de J.L. Whitten et al [WHI01] et celui de M. Cook [COO96].

Encart 2.6 Davantage de lectures sur le cadre de Zachman

*I keep six honest serving men (they taught me all I knew): Their names are What and Why and When and How and Where and Who.
Rudyard Kipling (1865-1936)*

Inmon et al [INM97] comparent le cadre de Zachman à la table des périodiques chimiques. Cette table aide les scientifiques à trouver facilement les propriétés des éléments chimiques alors que le cadre de Zachman aide à comprendre les usages potentiels des techniques et à connaître rapidement ce qu'une technique est capable d'assurer comme fonctions.

En plus du domaine de l'ingénierie des SI et du génie logiciel, le cadre peut éventuellement s'employer dans les autres professions d'ingénierie comme l'architecture, la conception mécanique etc. comme l'explique sa définition plus haut.

Ces techniques sont classées dans les cellules de la matrice carrée de la Figure 2.17. Chaque colonne représente une dimension de la modélisation du système (Focus) traduite par les six questions : "Quoi, Comment, Où, Qui, Quand et Pourquoi". Chaque ligne décrit un jalon du cycle de vie de la conduite du projet appelée perspective. Rappelons que le terme perspective est utilisé dans l'art. On parle de perspective lorsqu'on veut dessiner un objet selon la vue avec laquelle on perçoit l'objet : vue de dessus, dessous, de côté...

Définissons d'abord les dimensions du cadre :

1) Quoi : cette colonne concerne globalement les techniques de modélisation des données. De haut en bas, les cellules peuvent décrire respectivement les objets potentiels du système, le modèle conceptuel, logique, physique de données et enfin les livrables comme les BdD, les codes de programmes...

2) Comment : elle traduit les fonctions et les activités du système, supportant les entités de la colonne 1. Bien entendu, la représentation de ces activités varient en fonction des stades du développement du système (diagrammes de l'analyse structurée, Use Case de UML, ...)

3) Où : elle peut concerner l'endroit où se déroule une activité (bureau d'ingénierie, atelier...), ou est stockée une donnée (Serveur, PC...) ou encore où se trouve l'utilisateur final du système (Réseau...).

4) Qui : cette colonne traduit les aspects liés à l'organisation, aux acteurs impliqués, à leurs rôles et compétences...

5) Quand : la représentation du temps varie aussi à différents stades de développement du système. Le temps peut traduire simplement une séquence d'événements ou une transaction. C'est avec les techniques OO que la dimension temporelle est représentée désormais plus naturellement.

6) Pourquoi : cette colonne traduit ce que l'on a appelé jusqu'ici les business requirements du système : la raison d'être du système, ses contraintes environnementales, ses règles de fonctionnement etc.

Les perspectives du système se répartissent comme suit :

1) Contextuel : la ligne 1 contient des définitions et l'acteur concerné peut être un planificateur ou celui qui émet les nouvelles stratégies et politiques du système.

2) Conceptuel, 3) Logique, 4) Physique : les lignes 2, 3 et 4 sont représentées à l'aide des diagrammes de spécification. Les acteurs concernés sont, respectivement, le maître d'ouvrage (le propriétaire du système), le maître d'œuvre (concepteur) et le constructeur qui est en charge des détails de conception technique.

5) Livrable (out-of-context) et 6) "Functionning enterprise" : ces deux dernières perspectives contiennent les livrables et les composants réutilisables. Ces actions sont généralement déléguées au sous-traitant qui va assembler les modules (livrables) du système en respectant les contraintes d'implémentation imposées par le constructeur.

Par ailleurs, le cadre décrit des contrôles permettant de transformer une information se trouvant dans une cellule vers une autre ou de construire des modèles hiérarchiques et récursifs [SOW92;MAR99].

Enfin, l'institut Zachman (www.zifa.com) collabore avec beaucoup d'utilisateurs et des éditeurs de logiciels [POP00] afin de promouvoir l'implémentation rationnelle de ses concepts.































	DATA	How	FUNCTION	How	NETWORK	How	PEOPLE	How	TIME	How	INNOVATION	How	
SCOPE (CONTEXTUAL)		List of Things Important to the Business		List of Processes the Business Performs		List of Locations in Which the Business Operates		List of Organizations Important to the Business		List of Events Significant to the Business		List of Business Goals/Strat. Critical Success Factors	SCOPE (CONTEXTUAL)
Planner	Entity + View of Business Thing		Function + Class of Business Process		Node + Major Business Location		People + Major Organizations		Time + Major Business Event		End/Mean+Major Bus. Goal/Critical Success Factor		Planner
ENTERPRISE MODEL (CONCEPTUAL)		e.g. Semantic Model		e.g. Business Process Model		e.g. Business Logical System		e.g. Work Flow Model		e.g. Model Schedule		e.g. Business Plan	ENTERPRISE MODEL (CONCEPTUAL)
Owner	Ent + Business Entity Reln + Business Relationship		Proc + Business Process IO + Business Resource		Node + Business Location Link + Business Linkage		People + Organization Unit Work + Work Product		Time + Business Event Cycle + Business Cycle		End + Business Objective Means + Business Strategy		Owner
SYSTEM MODEL (LOGICAL)		e.g. Logical Data Model		e.g. Application Architecture		e.g. Distributed System Architecture		e.g. Human Interface Architecture		e.g. Processing Structure		e.g. Business Rule Model	SYSTEM MODEL (LOGICAL)
Designer	Ent + Data Entity Reln + Data Relationship		Proc + Application Function IO + User View		Node + IS Function IO + Resource Link + Line Characteristics		People + Role Work + Deliverable		Time + System Event Cycle + Processing Cycle		End + Structural Relation Means + Action Assesment		Designer
TECHNOLOGY MODEL (PHYSICAL)		e.g. Physical Data Model		e.g. System Design		e.g. Technology Architecture		e.g. Presentation Architecture		e.g. Control Structure		e.g. Rule Design	TECHNOLOGY MODEL (PHYSICAL)
Builder	Ent + Segment/Package Reln + Point-to-Point		Proc + Computer Function IO + Data Element Set		Node + Hardware/Software Link + Line Specifications		People + User Work + Software Product		Time + Execute Cycle + Component Cycle		End + Condition Means + Action		Builder
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)		e.g. Data Definition		e.g. Program		e.g. Network Architecture		e.g. Security Architecture		e.g. Timing Definition		e.g. Rule Specification	DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)
Sub-Constructor	Ent + Field Reln + Address		Proc + Language Stmt IO + Control Block		Node + Address Link + Protocol		People + Identity Work + Job		Time + Interupt Cycle + Machine Cycle		End + Sub-condition Means + Step		Sub-Constructor
FUNCTIONING ENTERPRISE	e.g. DATA		e.g. FUNCTION		e.g. NETWORKS		e.g. ORGANIZATION		e.g. SCHEDULE		e.g. STRATEGY		FUNCTIONING ENTERPRISE

Figure 2.17 Cadre de Zachman (source : www.zifa.com)

Ainsi nous terminons l'état de l'art sur les architectures de références. Le lecteur intéressé par des études comparatives sur les techniques de modélisation en entreprise, peut se référer aux travaux de K. Kosanke [KOS96] ou encore à l'ouvrage de F. Vernadat [VER99] qui a effectué une comparaison par rapport au cadre CIMOSA.

Enfin, nous caractérisons un cadre de référence dit "idéal" par ce qui suit [POP00] :

- Simple : facile à comprendre, il doit être plus logique que technique.
- Complet : en représentant l'entreprise (système) dans sa globalité.
- Un langage : aide à penser aux concepts les plus complexes et à les communiquer avec précision en peu de mots et surtout non techniques.
- Outil de planification et d'analyse : en améliorant la prise de décision...
- Outil de résolution des problèmes : permet de travailler avec des niveaux d'abstractions, de simplifier en isolant une variable sans toutefois perdre le sens global du système.
- Neutre : totalement indépendant des outils et des méthodologies. Donc n'importe quelle méthodologie peut être étudiée en fonction du cadre pour comprendre ce qu'elle couvre comme fonctionnalités.

A présent, examinons ce qui caractérise les outils qui supportent tous ces modèles de spécification et les architectures de référence.

III.3 Outillage des techniques : les ateliers de génie logiciel

Un Atelier de Génie Logiciel (CASE: Computer-Aided Software Engineering) peut être défini comme une application logicielle qui supporte les dessins et les analyses des modèles de systèmes²⁴. C'est une assez large définition qui fait que la communauté du génie logiciel ne s'est toujours entendue ni sur le périmètre ni sur les fonctions que doit assurer un AGL.

Il y a toutefois une distinction communément connue entre "Upper" et "Lower" CASE. La première catégorie concerne les phases très en amont du développement des applications (stratégie, analyse et conception logique). La seconde en revanche est destinée aux phases aval (conception physique, implémentation, programmation). Notons également que les vendeurs ont abandonné le mot CASE pour employer plutôt le terme "outil de modélisation". Ils sont aujourd'hui au nombre de 500 pour présenter environ 1000 outils dans le monde. Dans l'annexe C, nous exposons quelques éléments de l'étude comparative de ces outils effectuée sur la base des résultats de quelques cabinets de conseil dans la matière.

L'architecture "idéalisée" de ces logiciels est décrite par la Figure 2.18. Un AGL peut comprendre les composants suivants [AVI00;KET98;WHI01] :

- Dictionnaire de données (Repository ou encyclopédie) : c'est le noyau du CASE et la clé d'intégration. Le dictionnaire stocke et gère les modèles (définitions et graphiques) tout en maintenant la cohérence. Les modèles sont composés de symboles qui peuvent posséder des définitions. Nous présentons sur la Figure 2.19, un modèle objet simplifié de l'organisation d'une encyclopédie.
- Editeur graphique : sert pour la modélisation et la conception des diagrammes de spécification. Un outil peut couvrir ainsi une ou plusieurs catégories de diagrammes : la modélisation des données, l'orienté-objet, les analyses structurées et les modèles de processus etc., avec des notations plus ou moins standards. On trouve aussi des fonctions avancées comme la simulation des modèles obtenus, la navigation, le contrôle...
- Documentation automatique : ces tâches sont généralement coûteuses et fastidieuses. Automatiser l'extraction et la mise à jour des informations est bénéfique d'autant plus que la fonction de documentation est utilisée tout au long du projet.
- Génération de code, de BdD, Reverse Engineering : ces utilitaires permettent de lire du code existant ou d'en produire depuis les spécifications du système, de remonter des schémas physiques de BdD en modèles logiques pour les raffiner ou les réutiliser... Ce sont des exemples de tâches qui interviennent durant le processus de développement.
- Outils de gestion de besoin, audit sur les changements et tests : la gestion des besoins est affaire de tous ; elle commence par la prise en compte des besoins et se poursuit tout au long du cycle de vie du projet, d'où l'intérêt de fournir un outil de type "groupware" permettant le suivi des besoins. On dit aussi que les tests représentent 50% de l'effort global dans le développement d'une application logicielle et disposer d'outils de prototypages et de test est source de gains...

²⁴ "Software programs that automate or support the drawing and analysis of system models and provide for the translation of system models into application programs"[WHI01].

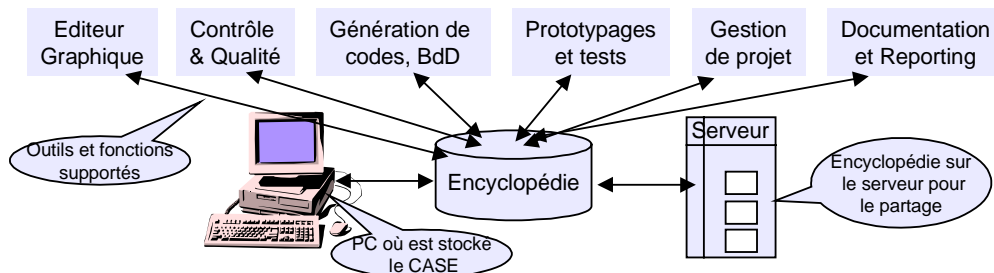


Figure 2.18 Architecture d'un CASE –AGL- (Adaptée de [WHI01])

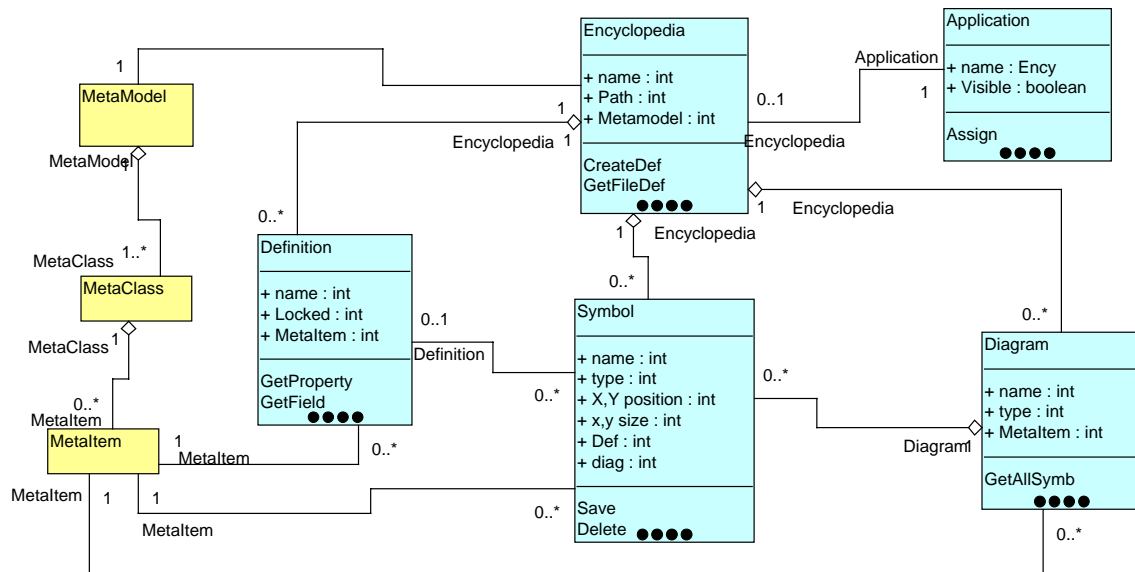


Figure 2.19 Organisation de l'Encyclopédie, noyau des AGL : Modèle objet simplifié

Au delà des composants, il y a des caractéristiques importantes mais souvent implicites sur lesquelles on peut se baser pour choisir un AGL. Nous citons deux caractéristiques : l'intégration et les métriques.

▪ L'intégration

On dit qu'un AGL est un outil intégré s'il comprend les modèles nécessaires pour les différentes phases du développement : planification, analyse, design, construction et implémentation. Cela ne suffit pas, le mot "intégration" en effet, veut dire beaucoup de choses. Ainsi Kurbel et al [KUR93] distinguent trois formes d'intégration. L'intégration horizontale consiste à assurer une cohérence et une intégration entre les techniques utilisées dans une phase particulière du développement de l'application. En effet, en phase d'analyse par exemple, il existe plusieurs techniques dédiées à cette phase : techniques d'analyse de données (E-R...), d'analyse structurée (DFD...) etc. Si l'information commune est partagée entre ces diagrammes à tel point de faire répercuter une modification d'un diagramme sur les autres, on parle alors d'intégration horizontale. L'intégration verticale quant à elle, concerne la cohérence entre modèles à différentes phases du projet. C'est-à-dire que les résultats d'une phase doivent être disponibles et réutilisables pour les phases suivantes. Enfin, la troisième forme d'intégration est liée aux fonctions de gestion de projet. Le développement d'applications logicielles et plus particulièrement des SI en entreprise est un projet de longue haleine impliquant beaucoup d'acteurs. Le choix d'un AGL est aussi conditionné par sa capacité à assurer l'intégration des activités au sein du projet. Il est clair qu'une telle

intégration -sous toutes ces formes- est une caractéristique rarement disponible dans les AGL d'aujourd'hui.

▪ **Les indicateurs de mesure et de qualité (Metrics)**

La métrique est un domaine très large et peut s'appliquer à différents niveaux en conduite de projets informatiques. Notre but ici n'est pas de lister les métriques possibles du domaine ou ceux qui sont implémentés dans les AGL en vue de mesurer la qualité des modèles. Nous nous contentons d'un exemple de métriques spécifiques avant de faire notre propre constat à ce sujet. R. J. Muller [MUL99] en expose quelques unes dédiées au domaine des modèles de données. Il classe ces indicateurs en cinq catégories :

Taille : concerne la taille du schéma de la BdD, le nombre de classes et d'objets, des indicateurs pour restreindre le nombre d'héritages hiérarchisés au sein d'un diagramme etc.

Complexité : deux notions se distinguent : la complexité du problème et de la solution. Comme les modèles de données décrivent généralement le problème, la complexité du problème est alors plus intéressante que la complexité de la solution dans ce cas. Nous savons par ailleurs qu'il y a différentes manières de modéliser un problème et plusieurs modèles cohérents peuvent être obtenus pour un problème donné (III.1). Il est alors intéressant qu'un AGL puisse disposer de métriques pour mesurer la complexité du problème à résoudre et estimer la façon de le modéliser.

Cohésion et cohérence : ici les métriques mesurent les cohérences (structurelles et abstraites) entre modèles du système. Elles sont aussi importantes si l'on veut veiller au continuum entre phases du développement d'un système au travers des modèles de spécification qui y sont déployés. Les techniques de normalisation discutées brièvement plus haut (note de bas de page 17) sont des exemples d'indicateurs qui améliorent la qualité des modèles résultants.

Accouplement (Coupling): Accouplement est le degré d'interdépendances entre modules. Les travaux dans le domaine des BdD démontrent qu'il est plus facile de réutiliser et de mettre à jour des modèles qui minimisent ce genre d'interdépendances. Ces dernières se manifestent de différentes façons et il devient parfois nécessaire de contrôler leur nombre et leur nature à l'aide des métriques.

Potentiel de réutilisation : ce type de métriques mesure la facilité à réutiliser un schéma de BdD, un modèle conceptuel ou n'importe quel autre composant.

Nous pensons que les indicateurs implantés dans les AGL, peuvent être tout aussi utiles que dérangeants. Ils sont utiles parce qu'ils améliorent d'une façon générale la qualité des modèles développés et le processus de développement, d'où l'intérêt de vérifier si un AGL est pourvu d'indicateurs au moins des plus classiques. Ils sont parfois dérangeants s'ils sont implantés d'une façon trop générique et ne répondent donc pas aux cas spécifiques de la réalité. Faut-il peut-être veiller à cette caractéristique avant de choisir un AGL ?

Durant cette partie -Techniques et outils de modélisation-, nous avons discuté d'abord des modèles de spécification en ingénierie des SI, suivis des techniques d'enterprise modeling et des architectures de références. Nous avons à la fin décrit les caractéristiques des AGL. Nous terminons cette partie avec un bilan qui complétera les bilans intermédiaires.

III.4 Bilan et analyse

Nous structurons ce bilan par ordre des thèmes abordés dans cette partie. Ce bilan présente des points clés cités dans la littérature ou qui relèvent de notre expérience sur le terrain.

Modèles de spécification des SI

- Dans les techniques d'analyse de données, l'étape préalable de collecte et de validation des données peut être réalisée de différentes façons : à l'aide d'interviews auprès des acteurs de l'organisation pour comprendre leurs besoins en données. Une autre façon est d'étudier les documents y circulant ou encore par l'observation de ses processus. Nous pouvons déjà affirmer qu'il faut intégrer ces différentes façons d'acquérir l'information. Car les documents seuls ne contiennent pas la totalité des données qui circulent dans l'organisation. Les acteurs peuvent émettre des informations subjectives et incomplètes et l'observation des processus opérationnels de l'entreprise n'est pas évidente à cause de leur complexité et de leur manque de rationalisation.
- L'erreur souvent commise dans les techniques de modélisation de données est que l'on décline les entités directement depuis les processus de l'entreprise. Ceci est valable pour construire un modèle d'activités, nommé "Business Model" par les anglo-saxons. Nous verrons plus tard qu'un business model ne peut servir directement pour produire une BdD.
- Il est difficile (long et coûteux) de fédérer l'ensemble des données d'une organisation dans un seul modèle de données. C'est l'une des raisons pour lesquelles, dans les pratiques industrielles, les BdD sont développées pour répondre à des fonctions locales exprimées par les acteurs d'une division donnée (pour en savoir plus, cf. chapitre 1, II.3).
- Si les modèles relationnels ont dominé dans le monde professionnel pendant longtemps, ils sont aujourd'hui remis en cause avec l'émergence de l'objet. Cependant, les chercheurs tout comme les industriels, savent pertinemment que l'on ne peut substituer si vite l'objet au relationnel, et qu'actuellement les seuls SGBD opérationnels en entreprise sont basés sur le mode relationnel. Plusieurs autres raisons ont amené la communauté de chercheurs [AMB99;MUL99] à réaliser des transformations (mapping en anglo-saxon) entre les modèles de données et les modèles objet.
- Les techniques d'analyse structurée sont longtemps utilisées en conjonction avec les modèles de données. Ses diagrammes montrent les flux de données manipulées par le système et les processus qui opèrent sur ces données. Par ailleurs, ces fonctions sont aujourd'hui incluses dans les techniques objet et de ce fait elles deviennent inutiles si nous retenons l'objet.
- UML, rappelons-le, n'est pas une méthode, mais une notation universelle et un langage fourni sans mode d'emploi. En effet, dans le manuel de spécification de la notation [OMG99], il n'est pas prescrit l'ordre d'utilisation des neuf diagrammes de UML, ni comment les adapter en fonction de l'application que l'on souhaite développer.
- Comme nous l'avons souligné plus haut (à la fin du III.1), les langages de modélisation des données techniques (EXPRESS, PSL), malgré leur apport dans la standardisation des référentiels techniques de l'entreprise manufacturière, sont inefficaces et impuissants dans l'expression des spécifications fonctionnelles que les acteurs doivent partager.
- Il existe bon nombre de problèmes que les techniques existantes n'ont pu résoudre. Décider du niveau de détail dans la modélisation ou fixer les frontières du modèle de données sont des exemples de décisions que le concepteur est amené à prendre souvent seul et intuitivement. Au chapitre 3, nous traitons plus en détail ces difficultés rencontrées et les solutions préconisées pour y remédier.

Cadres et architectures de références

- Les vues fonctionnelle et informationnelle sont prises en compte dans la plupart de ces cadres. En revanche, les autres aspects tels que la dimension décisionnelle, stratégique ou encore temporelle ne sont pas systématiquement représentées.
- Certains cadres de références offrent en même temps des démarches de modélisation. Celles-ci sont souvent jugées soit trop restrictives (comme ARIS par exemple) ou spécifiques à un domaine particulier (comme BASE-PTA). Par ailleurs, le cadre de Zachman qui n'impose pas a priori une méthode de modélisation a été adapté aussi bien dans le monde industriel qu'en recherche (note de bas de page 23).
- Les modèles d'entreprise résultants représentent une image idéale du système. S'ils capitalisent par exemple les contraintes du système, ils gèrent rarement leurs évolutions. Notons également que les cadres dédiés à la modélisation des systèmes de production ne se préoccupent pas des activités d'ingénierie de production.
- L'autre raison de la difficulté d'adaptation de ces cadres est que les ateliers logiciels qui les supportent ne traduisent que partiellement les concepts théoriques (comme CIMOSA). Nous verrons dans ce qui suit, les avantages et les limites justement de ces AGL.

Apports et limites de l'automatisation

En faisant usage des AGL, on peut observer des gains que l'on résume en ces points :

- *Amélioration de la productivité* : grâce à l'automatisation des tâches, en développant donc plus vite et avec moins d'hommes. Par conséquent, les coûts de développements sont réduits.
- *Amélioration de la qualité* : les objets de conception (les livrables) sont mieux spécifiés. Les contrôles et les métriques assurent la cohérence des modèles et la qualité du processus du développement.
- *Réduction du temps de la maintenance* : signalons que la notion de maintenance a évolué pour s'étendre aux phases amont de l'implémentation. Ainsi la maintenance peut concerner les diagrammes de spécification du maître d'ouvrage chargé d'exprimer le besoin...

Aujourd'hui l'intérêt est accordé plus aux phases amont qu'à l'implémentation et l'AGL favorise la réutilisation des composants (codes, schéma de BdD, diagrammes et définitions...).

En plus des caractéristiques fonctionnelles de l'AGL, le succès ou l'échec de son usage en entreprise est lié aux aspects de l'organisation dont :

- Maturité des départements informatiques de l'entreprise en technologies d'information.
- Historique des innovations et expérimentations dans le domaine des informatisations en entreprise.
- Degré de créativité des acteurs concernés, leur temps d'apprentissage, leur perception du travail d'équipe, les niveaux de satisfactions de travail, le rôle des sponsors et partenaires...

Enfin, il reste encore un point qui peut être à la fois fort et faible :

- Application des standards des méthodologies ou règles de développement : l'objectif est de modéliser formellement le processus de développement et d'appliquer strictement les règles définies dans le processus. C'est une question controversée à laquelle nous revenons à la fin du chapitre après avoir exposé les aspects méthodologiques liés à l'ingénierie des SI.

IV Méthodes d'ingénierie des SI et développement des SIT en entreprise

Après l'investigation des outils et des techniques de modélisation, nous exposons dans cette partie, en premier lieu, les méthodologies en ingénierie des SI. Celles-ci sont nées du même besoin commun avec la plupart des activités d'ingénierie (Encart 2.7).

Par ailleurs, et pour être concis, nous essayons de débattre ces travaux d'un point de vue plus pratique que théorique. Nous récapitulons toutefois dans l'Encart 2.8 les principales interrogations épistémologiques soulevées par cette communauté. L'investigation des méthodologies nous amène à analyser leurs apports et limites. Par la suite, nous discutons les travaux liés au développement des SIT en entreprise qui, souvent, adaptent/adoptent les méthodologies discutées auparavant. Nous terminons par un bilan sur ces développements.

Encart 2.7 Besoin de méthodologies pour les métiers d'ingénierie...

Quel besoin y a-t-il qu'un pont soit plus large que la rivière ? Le nécessaire est toujours la plus juste des concessions. Shakespeare (1564-1616)

Tout comme les autres métiers d'ingénierie, la présence des méthodologies en ingénierie des SI, peut se justifier par la volonté d'atteindre la cible classique : (1) Meilleur produit final, (2) Meilleur processus de développement et (3) Standardisation du processus.

Pour la qualité du résultat obtenu (dans ce cas, le produit final est le SI), il n'est pas facile de la mesurer d'autant plus que les critères de qualité n'ont pas encore été fixés par la communauté de l'ingénierie des SI. Néanmoins il existe quelques contributions du domaine comme par exemple celle de l'organisme IFIP (www.ifip.or.at).

Pour le processus de développement, certains pensent que l'on peut adopter les indicateurs de qualité des processus, populaires en industrie manufacturière pour les activités d'ingénierie de SI. Il s'agit principalement de la norme BS5750/ISO9001 qui contrôle plutôt le processus que le produit final attendu.

Enfin la standardisation de tout processus présente un intérêt tant pour faciliter la maintenance et l'évolution du système que pour améliorer ses spécifications fonctionnelles.

Encart 2.8 Interrogations épistémologiques autour des méthodologies

There are more things in heaven and earth, Horatio, than are dreamt of in your philosophy. Shakespeare, Hamlet I:V

La dimension philosophique est nécessaire à l'explication et à la compréhension des méthodologies. Si elle ne résout pas les problèmes, elle aide en revanche à les soulever et à se poser les bonnes interrogations.

Par exemple, la présence de méthodologies en soi suscite des débats au sein de la communauté. En effet, l'ingénierie des SI est une profession créative tout comme les autres activités d'ingénierie. Les méthodologies n'essaient-elles pas de réduire le degré de liberté dans le processus de développement ? Le processus de type "répétitif" ou "reproductible" peut-il s'appliquer au développement des SI (et plus particulièrement des SIT) ? Peut-on juger dès lors de la non adéquation des méthodologies dites "prescriptives" dans ce domaine ?

Par ailleurs, de nombreuses autres interrogations ont divisé l'opinion de la communauté. citons en quelques unes :

- . Y a-t-il une différence entre "méthode" et "méthodologie" ?
- . Une méthodologie doit-elle spécifier les outils et les techniques qui vont être utilisées ?
- . Doit-on obtenir les mêmes résultats à l'usage de la même méthodologie pour chaque application ?
- . Les outils et techniques de modélisation, peuvent-ils se suffire pour constituer une méthodologie ?

Le lecteur intéressé par les réflexions épistémologiques du domaine, pourra consulter par exemple l'ouvrage de P. Checkland [CHE90] ou le papier de R. Hirschheim et al [HIR89].

IV.1 Méthodologies en ingénierie des SI

Définitions : R. N. Maddison [MAD83] décrit une méthode comme "*a recommended collection of philosophies, phases, procedures, rules, techniques, tools, documentation, management, and training for developers of information systems*". Les définitions françaises les plus à jour rejoignent celle-ci. Ainsi, C. Rolland [ROL96;ROL00] caractérise une méthode par quatre composants : modèles, langages, démarche et outils définis comme suit.

Un modèle est un ensemble de concepts et de règles, destiné soit à expliquer et à construire la représentation des phénomènes organisationnels, soit à expliquer et représenter les éléments qui composent le SI et leurs relations. Un langage est un ensemble de constructions qui permettent de décrire formellement les spécifications du SI élaborées à différents stades du processus de conception en s'appuyant éventuellement sur les modèles. La démarche est le processus opératoire grâce auquel s'effectue le travail de modélisation, de description et de réalisation du SI. Enfin, les outils ou techniques pouvant concerner les trois composantes ci-dessus, ont pour objectif d'aider à leur mise en œuvre.

C. Rolland [ROL96] explique également qu'il y a deux dimensions à prendre en compte en développement des SI :

- *L'ingénierie des besoins (Requirement Engineering)* : consiste à acquérir et à représenter les connaissances dont les utilisateurs ont besoin. Cette activité se traduit par l'élaboration d'un schéma conceptuel du SI. Un chapitre entier a été consacré à ce domaine dans l'ouvrage collectif Cauvet et al [CAU01].
- *L'ingénierie des systèmes* : consiste à implanter le schéma conceptuel du SI sur un système technologique en employant les technologies d'information. Le système ainsi résultant facilite la gestion des connaissances dont les utilisateurs ont besoin. Nous pensons que les traductions anglophones possibles pour ce domaine sont : "Systems Design" ou "Physical Design", domaine présenté par exemple dans Whitten et al [WHI01].

Plus simplement, D. Rieu [RIE99] définit une méthode par deux éléments : des modèles de spécification (formalismes) et un processus de développement du SI (démarche).

Les formalismes permettent de représenter les systèmes sous différents aspects (statique, dynamique, fonctionnel) et à différents niveaux d'abstraction (expression des besoins, analyse, conception). Une démarche permet de guider les activités de développement. C'est une succession d'étapes permettant en particulier de passer d'une modélisation du système à une autre.

Evolution des méthodes : Les méthodes en ingénierie des SI qu'elles soient académiques ou pratiques n'ont cessé de fleurir à tel point que l'on parle de "jungle" de méthodes (Encart 2.9). Néanmoins, la communauté scientifique a retenu globalement quatre grandes familles dans lesquelles peuvent appartenir ces méthodes : méthodes analytiques, cartésiennes, systémiques et objet [AVI00;CAU01;WHI01].

Suivant la chronologie, C. Tessier [TES95] classe ces grands courants comme suit :

- Les années 60 avec les *méthodes d'analyse* par les données (analytique) ou par les fonctions (synthétique). Elles ont la particularité de se focaliser sur la réalisation du système avant sa conception.
- Les années 70 avec les *méthodes cartésiennes* : elles sont caractérisées par une approche fonctionnelle et une démarche descendante en conception du SI. Elles sont influencées par la programmation modulaire. Notons dans ce volet, les méthodes basées sur SADT.

- Les années 80 avec les *méthodes systémiques*. Elles sont basées sur une approche globale et conceptuelle du SI et sur les techniques de modélisation des données (relationnelles, E-A). Les contributions françaises sont considérables (Merise).
- Les années 90 avec les *méthodes objet* : elles s'appuient essentiellement sur les techniques OO et c'est la communauté anglo-saxonne qui reste leader en matière de développement de méthodes.

Par ailleurs, C. Rolland [ROL96] note trois principaux jalons dans l'évolution des méthodes (Figure 2.20) :

- *L'ingénierie des produits* : au début, les méthodes mettent l'accent sur la modélisation du produit final (SI résultant) en s'intéressant aux modèles de construction du SI.
- *L'ingénierie des processus* : l'intérêt est accordé par la suite, à la rationalisation des processus qui permettent l'obtention du produit final et garantissent la cohérence entre les modèles de produit.
- *L'ingénierie des méthodes* : la difficulté d'adaptation des méthodes qui sont de plus en plus nombreuses, a motivé ces derniers temps la communauté pour rechercher des manières pour construire, améliorer et faire évoluer les modèles de processus.

La thèse de C. Rolland a été bien confirmée par d'autres spécialistes en ingénierie de SI [BAR96;BOU00]. Nous pensons encore une fois que cette thèse peut bien aussi concerner les autres métiers d'ingénierie. Ne s'agit-il pas de la même évolution naturelle de méthodologies qu'en ingénierie de conception de produit manufacturier par exemple ?

Encart 2.9 On parle de jungle de méthodes...

The goal of Computer Science is to build something that will last at least until we've finished building it. Anonymous

N. Jayranta [JAY94] estimait, déjà en 1994, l'existence de plus de 1000 méthodes dans le monde. Les méthodes les plus connues sont celles qui se matérialisent en produits commerciaux et qui sont souvent des solutions "maison" (in-house pour les anglo-saxons). Elles sont généralement fournies avec des AGL (comme la méthode ARIS supportée par l'AGL ARISTOOLSET), des manuels d'utilisation et des supports de formation comme le cas de SSADM, SSM... Elles peuvent également proposer des patrons pour la construction de modèles et la conduite même de la méthode. Aujourd'hui, le marché est envahi par les méthodes OO. Elles sont promues pour la plupart par les gourous du domaine des techniques de programmation Objet et du génie logiciel. Le lecteur intéressé par ces méthodes pourra consulter par exemple l'ouvrage de Kettani et al [KET98], où sont décrites, en annexe, les principales méthodes Objet. Enfin, il existe tout de même des méthodes qui sont indépendantes du marché comme l'a été Merise.

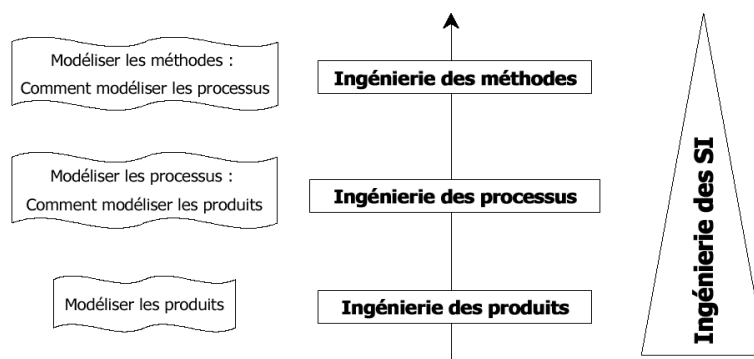


Figure 2.20 Evolution des méthodes en ingénierie des systèmes d'information (source [GZA00])

Sans être exhaustive, notre investigation des méthodes les plus populaires est récapitulée à l'Encart 2.10 dans lequel leur particularité est décrite. Une analyse plus approfondie des avantages et inconvénients des méthodologies est effectuée au IV.2. Il existe par ailleurs des études complètes, notamment des méthodes objet dans la littérature [AVI00;KET98].

Encart 2.10 Panorama de quelques méthodes d'ingénierie des SI

Art and science have their meeting point in method. Edward Bulwer Lytton (1803-1873)

MERISE : méthodologie française, elle est largement utilisée dans les pays francophones. Appliquée et testée en entreprise, elle est plus appropriée au développement des BdD et des SI de Gestion. L'approche s'appuie sur trois cycles : cycle décisionnel, cycle de vie et cycle d'abstraction. Dans le premier cycle, la prise des différentes décisions du projet est un processus qui concerne aussi bien l'utilisateur final que le maître d'œuvre du projet. Le cycle d'abstraction est la clé de la méthode. En séparant les données des traitements dès l'amont, le cycle d'abstraction est représenté à l'aide de modèles de spécification (relationnels et règles de normalisation) répartis en trois niveaux : conceptuel, logique (organisationnel) et physique (opérationnel). Ces niveaux traduisent bien une approche descendante en conception du SI. On lui reproche cependant sa démarche prescriptive [TAR91].

ETHICS (Effective Technical and Human Implementation of Computer-based Systems) : comme son nom l'indique, cette méthodologie tend à incorporer la position éthique. En s'appuyant sur une démarche participative et sur des aspects socio-techniques, la méthodologie préconise, que pour être efficace, la technologie doit s'ajuster étroitement avec les facteurs organisationnels du domaine d'application.

EUROMETHOD : résulte d'une initiative européenne, EUROMETHOD est plutôt un cadre de référence qu'une méthodologie. Il est dédié pour la gestion des services concernant l'investigation et le développement des SI. Il a pour souci de résoudre les problèmes liés à la diversité des méthodes.

ISAC (Information Systems work and Analysis of Change) : méthodologie scandinave, elle a pour enjeu d'identifier les causes fondamentales des problèmes des utilisateurs. La résolution de ces problèmes ne passe pas nécessairement par l'emploi des SI informatisés mais par l'analyse des activités et la reconfiguration des processus du domaine. C'est une approche vraisemblablement tournée vers les aspects humains et organisationnels.

RAD (Rapid Application Development) : cette méthode est née du besoin de développer rapidement les SI afin de répondre au mieux aux changements des business requirements du système. Elle est basée sur l'évolutivité, le prototypage et l'emploi des AGL. Les besoins des utilisateurs sont souvent déterminés à travers la méthode JAD (Joint Applications Development) [WHI01].

SSADM (Structured Systems Analysis and Design Method) : c'est le standard le plus utilisé au Royaume-Uni dans le développement des applications gouvernementales. La méthodologie combine les diagrammes de flux de données (DFD) et de cycle de vie des entités. Elle recommande également l'emploi des AGL.

SSM (Soft Systems Methodology) : proposée par P. Checkland [CHE90], la méthodologie est influencée par les concepts systémiques. Par opposition au paradigme scientifique qui, pour analyser un système complexe, décompose celui-ci en parties, les pensées systémiques revendiquent que les propriétés du système global ne sont pas entièrement expliquées à travers les propriétés des éléments le constituant. Cette méthodologie est plus appropriée pour traiter les situations floues, mal structurées etc. qui sont finalement fréquentes dans le domaine de développement des SI.

STRADIS (Structured Analysis and Design of Information Systems) : elle est proposée par Gane/Sarson. Les techniques principales employées sont orientées processus : décomposition fonctionnelle, diagrammes DFD et tables décisionnelles. Ces techniques mettent en avant la structure du système d'où l'origine de son appellation. Elle se veut être destinée pour n'importe quel développement de SI (indépendamment de sa taille ou de la nature de son livrable : automatisé ou non ...).

YSM (Yourdon Systems Method) : à ses débuts, YSM était similaire à STRADIS. Cependant, ses récentes versions se démarquent de STRADIS, en proposant pour l'analyse des processus, une approche basée sur le partitionnement des événements au lieu du principe de décomposition fonctionnelle [YOU92].

Cadres de comparaison des méthodologies : les premiers travaux sur les cadres de comparaison remontent aux conférences de l'IFIP des années 1980 regroupées dans T. W. Olle [OLL91]. Malgré leurs apports en matière de compréhension, ils étaient cependant critiqués à cause de leurs faiblesses à appréhender les réalités du terrain. Depuis, de nombreux autres travaux ont vu le jour, sous forme de check-list [MAD83], de cadres conceptuels [JAY94] ou de principes généraux [WHI01]. C. P. Catchpole [CAT87] a rassemblé dans sa thèse des propositions les plus significatives.

Avison et al [AVI00] ont repris un des cadres proposés par la littérature pour effectuer une étude exhaustive de comparaison des méthodologies existantes. Nous résumons sommairement le cadre en question qui est basé de sept critères pouvant être interdépendants.

(1) *Philosophie* : ce critère peut se traduire en ces quatre sous critères :

- *Paradigme* : deux courants majeurs se distinguent. Le premier est systémique, basé sur les principes holistiques. Ces principes sont bien explicités par les contributions françaises [BAR96;TAB93]. En revanche, le paradigme des sciences emprunte les principes de réductionnisme, de répétabilité et de réfutation pour résoudre la complexité des systèmes. Les méthodes basées sur le second paradigme ont eu plus de succès ces dernières décennies (les méthodes objet par exemple).
- *Objectif* : les objectifs des méthodologies peuvent être différents. Alors que certaines ont un but d'informatisation du SI et de développement d'applications logicielles, d'autres s'intéressent plutôt aux aspects d'analyse et de conception du SI.
- *Domaine* : ici "domaine" traduit les différentes vues que l'on peut avoir d'un système. Il faut toutes les prendre en compte de manière à avoir une meilleure compréhension du système.
- *Cible* : Certaines méthodologies sont spécialement dédiées à un type d'organisation et à des problèmes et à des environnements bien particuliers. Exemple : Merise convient aux grandes organisations. Elle est plus appropriée au développement des BdD et pour des environnements aux processus Batch et temps réel.

(2) *Modèles et (3) Techniques et outils* : Ils sont largement discutés en III.1. Nous pouvons constater toutefois que les méthodes d'une manière générale imposent le choix de modèles à utiliser. Ainsi la méthode Merise ne permet pas de remplacer les techniques relationnelles traditionnellement utilisées par les techniques récentes objet.

(4) *Contexte* : dans le Tableau 2.3, sont présentées les principales méthodes au regard des phases de développement de SI qu'elles couvrent.

(5) *Livrables* : il est intéressant de savoir ce qu'une méthode produit comme livrables à la fin de chaque phase du cycle de développement et plus particulièrement la nature de son dernier livrable (SI informatisé, documentation, ...).

(6) *Pratique* : le succès de l'exercice d'une méthodologie est fortement lié aux aspects environnementaux de l'organisation. Ainsi, il devient utile d'étudier les précédentes expériences et adaptations en organisation, de connaître les utilisateurs potentiels et d'évaluer leurs compétences...

(7) *Produit* : il s'agit de la forme de la méthodologie telle qu'elle est vendue sur le marché. Elle peut se présenter sous la forme de documentation académique et manuels (SSM,

SSADM) ou encapsulée dans un AGL (la méthode RUP et l'AGL Rational Rose). Elle est généralement fournie avec des services d'assistance et de formation.

Méthodologie	STRADI	YSM	IE	SSADM	Merise	JSD	OOA	ISAC	ETHICS	SSM	PI
Phase											
Stratégie											
Faisabilité											
Analyse											
Conception logique											
Conception physique											
Programmation											
Test											
Implémentation											
Evaluation											
Maintenance											

LEGENDE

Couverture quasi complète

Couverture partielle

Phase brièvement ou pas du tout mentionnée

Tableau 2.3 Couverture des méthodologies d'ingénierie des SI (source [AV100])

Terminons cet état de l'art sur les méthodologies en ingénierie des SI par le bilan suivant.

IV.2 Apports et limites des méthodes en ingénierie des SI

Notons que ce bilan ne cible pas une méthodologie particulière. Il se veut plutôt comme une analyse sur les méthodes en général. Là aussi, le bilan est effectué sur la base des conclusions des praticiens du domaine et de nos constats sur le terrain. Nous le structurons selon les thèmes que nous jugeons significatifs, c'est-à-dire liés à notre problématique de thèse.

Adaptabilité en organisation

Le succès ou l'échec de l'usage d'une méthode dépend de deux facteurs reliés entre eux :

- *Facteurs liés à la nature de la méthode* : le degré de *complexité* de la méthode, la nature de son *cycle de vie* ou de sa *démarche* (rigide, trop générique...) peuvent faciliter ou non son adaptation en organisation.
- *Facteurs liés à l'organisation* : généralement, c'est le département informatique de l'organisation qui est en charge du choix de la méthode qu'il personnalise souvent en fonction des contraintes environnementales. Pour cela, des cadres de comparaison comme celui décrit dans l'état de l'art paraissent utiles pour l'industriel qui n'est pas obligatoirement expert dans le domaine. Il est donc plus judicieux à ces personnes responsables, d'étudier d'abord leurs besoins et les usages potentiels de la méthode. Cette étude doit être aussi large pour identifier les futurs utilisateurs (qui n'appartiennent pas forcément à la direction informatique) et évaluer leurs compétences, tout en respectant les politiques de la firme et ses objectifs globaux (accélération du développement de SI, livrables au plus tôt...).

Tendances actuelles

Selon le penchant des méthodes à l'un ou l'autre paradigme, celles-ci peuvent être classées comme approches ascendantes ou descendantes.

- *Approches ascendantes (bottom-up)* : employant les techniques de rétroingénierie (reverse engineering²⁵), ces méthodes partent de l'analyse des composants produits du système afin de dégager les similarités et les différences entre objets qui caractérisent le système. Les méthodes de conception à base de composants et par réutilisation peuvent s'inscrire dans cette catégorie.
- *Approches descendantes (top-down)* : employant les techniques traditionnelles d'analyse de problèmes (problem analysis²⁶), ces approches s'appuient sur l'analyse globale du domaine et des décisions qui conduisent le processus de résolution de ces problèmes. Elles mettent en œuvre le problème pour lequel plusieurs SI peuvent être décrits. Les méthodes Merisiennes s'inscrivent ici.

A l'heure actuelle, nous pouvons affirmer qu'il n'y a presque plus de méthodes qui s'inspirent exclusivement de l'un ou l'autre paradigme. En effet, les méthodes descendantes qui empruntaient initialement les principes systémiques ne sont plus réticentes avec les solutions proposées par le paradigme des sciences (décomposition fonctionnelle, réductionnisme, réutilisation...). Elles sont de plus en plus outillées avec des AGL qui favorisent l'application de ces nouveaux principes. De même, les méthodes ascendantes comme beaucoup de méthodes objet et à base de composants accordent de plus en plus de l'intérêt à la dimension de l'organisation et à privilégier la vision globale du système.

Par ailleurs, l'histoire des méthodes de développement des SI n'a pas témoigné du succès absolu d'une méthode ou de la complétude et de la perfection d'une autre. Il n'est pas prévu non plus -à notre connaissance- que la communauté ait un désir de standardiser les méthodes de développement de SI ou de s'orienter vers une méthodologie commune.

Approches indépendantes des méthodes

Certains praticiens du domaine ont tendance à appliquer quelques approches pour soit compléter les méthodes utilisées ou pour démontrer leurs insuffisances. Celles qui nous intéressent sont :

- *Ad-hoc et contingence* : le premier mot latin et le second mot philosophique sont employés pour désigner et promouvoir les méthodes informelles, conçues probablement grâce aux expériences des praticiens. Les méthodes ici ne sont ni pérennes ni définitives du moment qu'elles évoluent parallèlement aux objectifs et aux contraintes de l'organisation, tout en s'adaptant à la nature du projet pour lequel la méthode serait utilisée.
- *Prototypage et test* : fonctions communes à d'autres activités d'ingénierie. Ainsi pour les mêmes raisons (réduction des risques et des coûts, satisfaction de besoins...) connues par exemple en construction automobile ou aéronautique que l'on privilégie, aussi dans ce domaine, le prototypage et les tests intermédiaires à différents stades du développement des SI.

²⁵ "Technology that reads the program code for an existing database, application program, and/or user interface and automatically generates the equivalent system model" [WHI01].

²⁶ "The activity of identifying the problem, understanding the problem (including causes and effects), and understanding any constraints that may limit the solution"[WHI01].

- *Développement incrémental (timeboxing²⁷)* : concept lié probablement à la philosophie du concurrent engineering pratiquée dans les professions d'ingénierie. Ce type de développement est caractérisé par un processus itératif (par opposition au processus séquentiel). Il présente des avantages tels que : la réduction du temps de développement, la prise en compte des besoins qui évoluent durant le développement et surtout une livraison plus rapide et plus fréquente des composants du SI.

Ces approches citées sont mieux reçues en entreprise que les méthodologies académiques. Les propos suivants de C. Garcia, qui était responsable du service informatique "qualité et méthodologie" de Renault, peuvent confirmer cette hypothèse : *"Merise n'est jamais entré en Régie, mais cela ne nous a pas empêché de suivre d'autres approches méthodologiques..."²⁸*. Nous nous spécialisons plus ci-après dans les travaux liés au développement des SIT.

IV.3 Développement des Systèmes d'Information Techniques en entreprise

Dans ce chapitre, nous avons abordé plusieurs thèmes autour des SIT, rappelons-les.

En introduction, nous avons souligné ce qui caractérise les SIT des SI et que le jeune domaine des SIT s'est appuyé essentiellement sur les travaux en ingénierie des SI en général.

Nous avons traité les recherches liées à la problématique d'intégration, des modes d'échanges et de standardisation des données techniques en entreprise (II). En abordant les solutions de SGDT, nous y avons intégré les travaux de recherche en Workflow et en gestion de configurations, fonctions faisant désormais partie des SGDT (II.2). D'autres travaux inscrits dans cette partie ciblent l'implantation des modèles normatifs (STEP, PDM Enablers...) et la configuration des technologies de SGDT(II.3).

Nous avons étudié par la suite (III) les modèles de spécification connus en ingénierie des SI. Nous avons constaté que la structuration des données techniques peut se faire naturellement à l'aide de ces outils, plus appropriés à notre problématique de thèse que ceux spécialement dédiés aux données techniques (EXPRESS, PSL).

Dans cette partie, nous avons traité les aspects méthodologiques liés à l'ingénierie des SI, source principale de solutions pour le développement des SIT en entreprise. Dans le bilan qui a suivi, nous avons discuté essentiellement des difficultés d'adaptation de ces méthodologies et rappelé quelques approches communément utilisées en entreprise sans pour autant adhérer à une méthodologie particulière.

Il reste néanmoins quelques autres travaux particuliers sur le développement des SIT et la mise en œuvre des SGDT en entreprise. Nous nous penchons maintenant sur ces travaux que l'on peut déjà classer en deux catégories : structuration des données techniques et aspects méthodologiques de conduite de ces projets. Enfin nous terminons par un bilan de l'apport et des limites de ces recherches.

Structuration des données techniques

Les travaux dans ce domaine, s'inscrivant dans des projets européens et internationaux, sont financés pour la plupart par des industries manufacturières. La forte implication de ces dernières est due au fait que les données techniques concernent naturellement le domaine de la productique et du génie industriel. Citons quelques-uns de ces travaux.

²⁷ "A technique that delivers information systems functionality and requirements through versioning. The development team selects the smallest subset of the system that, if fully implemented, will return immediate value to the system owners and users. That subset is developed, ideally with a time frame of six to nine months or less. Subsequently, value-added versions of the system are developed in similar time frame"[WHI01].

²⁸ Extrait de la revue : Le Monde Informatique, n°882, février 2001.

Computer Aided Engineering Group : de nombreux programmes de recherches²⁹ s'inscrivent dans ce groupe. Ils ciblent des thématiques aussi connexes que la modélisation du produit, du système industriel, de l'intégration des données du produit et du process tout en prenant en compte l'agent humain.

Les modèles informationnels sont à la base de la plupart de ces travaux, parmi ceux-ci citons le modèle de données du manufacturing, appelé MDM (Manufacturing Data Model) [ZHA99]. Cinq concepts composent ce modèle et décrivent les objets métier en entreprise manufacturière : Stratégie, Process, Ressource, Item et Flux (Figure 2.21). Harding et al [HAR99] se sont focalisés sur la problématique d'intégration de ces vues. Pour pallier les insuffisances des cadres de références génériques (III.2), un modèle d'entreprise baptisé FDM (Factory Data Model) a été construit sur la base du modèle initial MDM [YU00]. L'intérêt majeur de ce cadre est la réutilisation qui se traduit par l'acte d'instanciation des objets génériques définis dans le métamodèle pour une entreprise particulière. On parle alors de "Factory Design"³⁰. L'amélioration globale des performances de l'entreprise est un des gains estimés par la communauté en employant un tel cadre de référence. Soulignons également les travaux connexes [BUG98;CHE00;ZHA00] qui ont décliné les liens potentiels entre le modèle MDM de la Figure 2.21 et le modèle de produit construit dans les précédents programmes de recherche du groupe.

*Norme MANDATE*³¹ : les actions normatives s'intéressant exclusivement au système industriel (par opposition à la conception du produit) sont de plus en plus nombreuses. Elles sont récapitulées dans J. J. Michel [MIC00]. La plus significative est sans doute la récente norme MANDATE. Elle a un objectif triple qui est de : standardiser les échanges externes (clients et fournisseurs) au travers des protocoles EDI (Part 15531-2x series), contrôler les flux matériels dans les processus de production (Part 15531- 4x series) et enfin gérer les ressources de production (Part 15531-3x series). C'est cette dernière partie qui nous intéresse. En effet, la notion de "Ressource" en entreprise manufacturière n'est pas stable et les données techniques liées à ce concept sont stockées dans différents supports et gérées partiellement. De plus, les acteurs en entreprise ont rarement une compréhension commune de ces concepts (cf. chapitre 1 pour plus de détails). MANDATE n'est qu'à ses débuts et nous suivons de près ses évolutions. Retenons néanmoins sa volonté de ne pas se limiter aux ressources classiques étudiées dans les précédentes normes (outils de coupe, équipements d'usinage...). Elle se veut donc plus large pour inclure les ressources humaines, les programmes de commande numérique etc. Rappelons son architecture modulaire présentée sur la Figure 2.22. Les composants (numérotés de 1 à 6) sont définis comme suit [CUT00] :

- 1) Resource Hierarchy : la hiérarchisation des ressources peut être représentée par l'instanciation du "groupe de ressources" et de "ressource générique". La "ressource spécifique" est la spécialisation de la "ressource générique". La "ressource individuelle" n'est que la ressource physique disponible en entreprise.
- 2) Structure of Resource Characteristics : concerne les caractéristiques économiques (administration), des fonctions qu'elles assurent (capability/capacity) et de constitution.
- 3) Resource Status : propriété assignée à la ressource physique pour décrire son état (proposée, requise, réalisée...)

²⁹ Citons le projet MIM (Manufacturing Information Models) (période de 08/1997 à 07/2000). Il reprend les résultats du Projet MOSES (période 1992 à 1995). Notons également le projet MISSION, sous l'initiative du projet européen ESPRIT -29656. Cf. <http://info.lboro.ac.uk/departments/en/research/cae/index.html>

³⁰ voir les travaux connexes du groupe dans : www.lboro.ac.uk/departments/en/research/cae

³¹ Acronyme de MANufacturing DATa Exchange. Les spécifications fonctionnelles de la norme sont disponibles sur leur site officiel : <http://www.nist.gov/sc4/www/mandate.htm>

- 4) Resource Views : ce module dérive de l'agrégation de "resource characteristic" pour l'assigner à la "ressource générique".
- 5) Definition of Resource Characteristics : pour représenter les valeurs physiques de la ressource, sont distinguées deux valeurs : quantitative et qualitative.
- 6) Resource Configuration : doit décrire les différentes configurations de la "ressource spécifique".

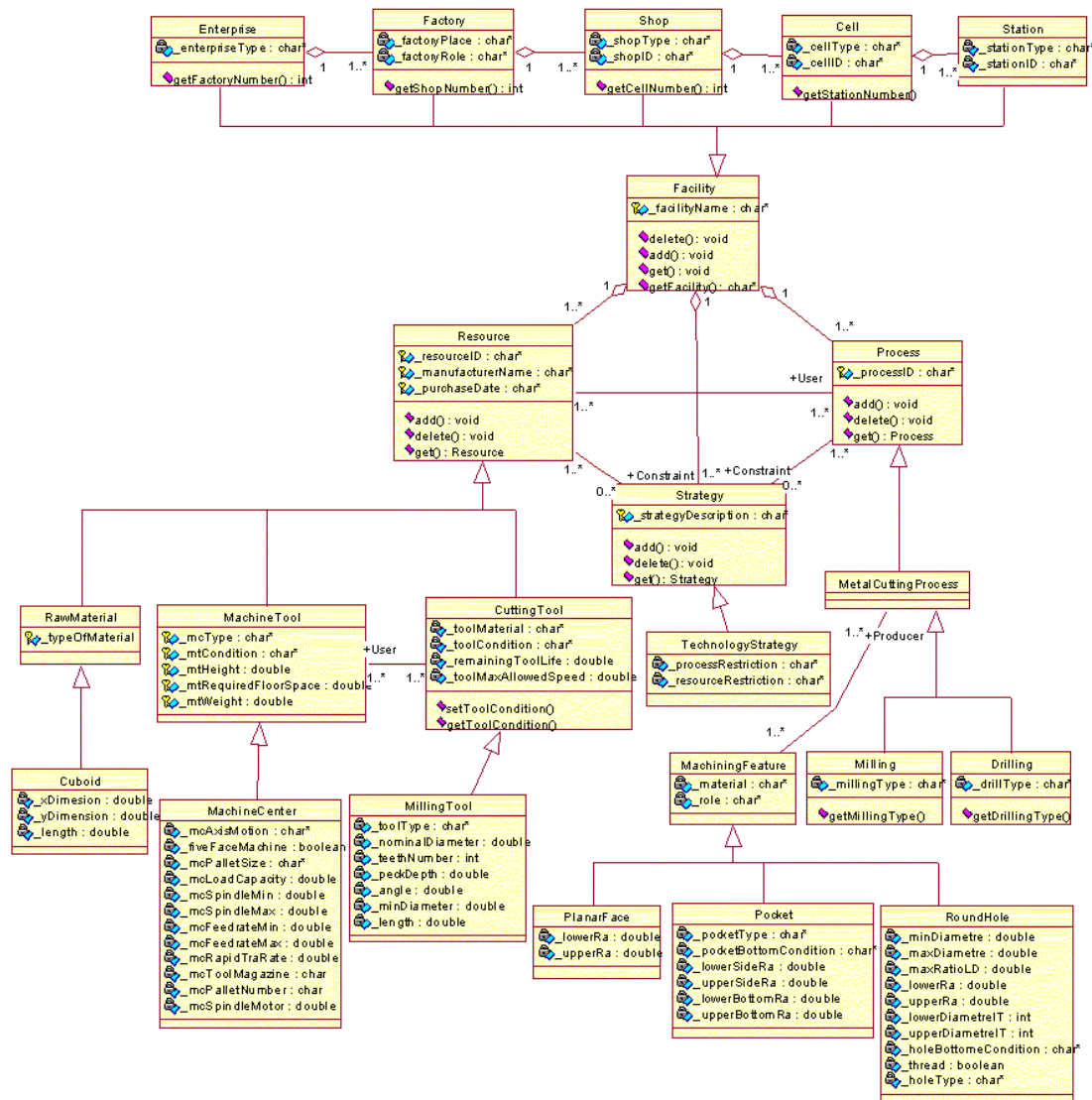


Figure 2.21 Le modèle MDM à l'aide du diagramme de classes UML
 (Source [ZHA00] ou www.lboro.ac.uk/departments/en/research/cae)

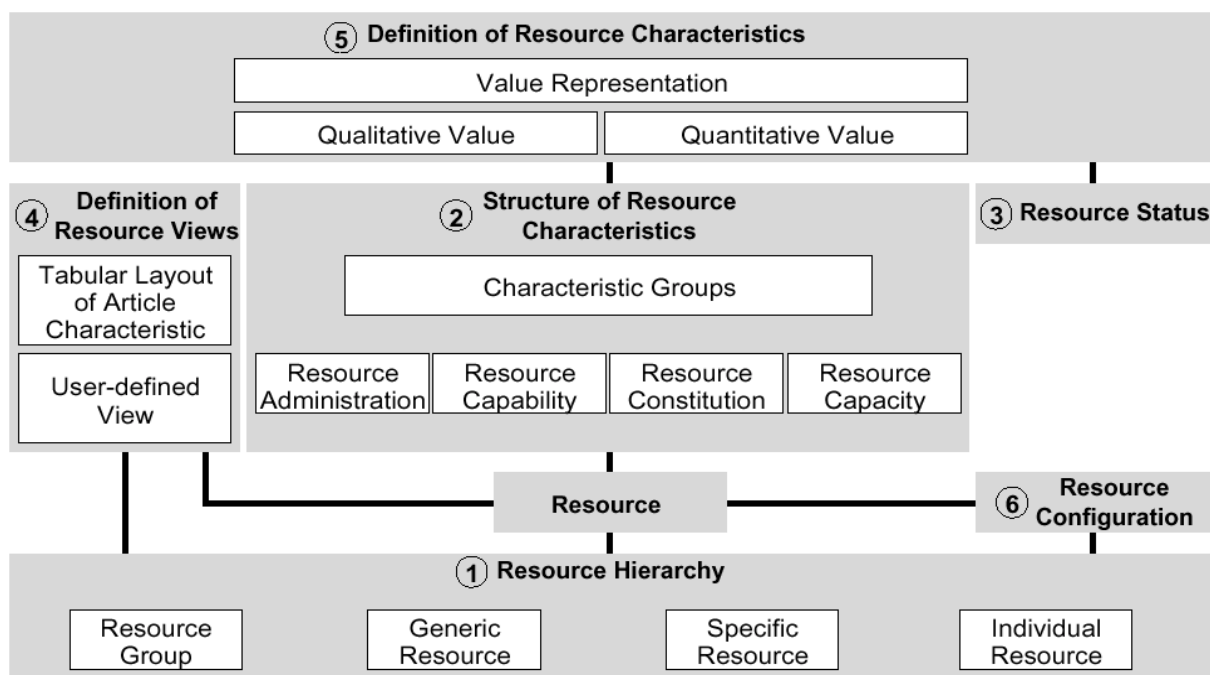


Figure 2.22 Structure générale du standard MANDATE
 (Source [CUT00] ou <http://www.nist.gov/sc4/www/mandate.htm>)

De nombreux autres travaux peuvent s'inscrire dans cette thématique de recherche. Les modèles de données résultants ont divers usages. Pour une meilleure gestion de données techniques, Bailey et al [BAI98] proposent une structuration de données basée sur la trilogie connue (Produit Process et Ressource). Les travaux de thèse de Million [MIL98] ont pour but de représenter les différentes vues des ressources en ateliers de production. D'autres structurations ont pour utilisation de construire des modèles de systèmes de production [KEL98], afin d'accroître la crédibilité des modèles de simulation et d'optimisation qui peuvent en être déduits. Un état de l'art sur l'évolution des architectures des systèmes de production est présenté dans la thèse de Ndiaye [NDI00]. Dans cette thèse a été développé un référentiel basé sur les techniques objet, ayant pour but d'aider la conception des systèmes automatisés de production.

D'autres travaux [DHA97] sont nés du besoin d'alimenter en données, les différents outils de conception et de fabrication de produit. Un schéma générique et une architecture d'intégration des BdD hétérogènes sont proposés dans ces travaux. Enfin, nous pouvons également noter les besoins ponctuels pour la construction de maquettes virtuelles en entreprise [ZHA'99].

Examinons maintenant les autres travaux liés à l'ingénierie même des SIT.

Aspects méthodologiques pour la conduite des projets SIT

Contrairement à la thématique précédente, il existe très peu de travaux s'intéressant aux aspects méthodologiques liés au développement des SIT en entreprise. S'il y a des actions développées à l'intérieur des organisations, elles restent cependant confidentielles et les livrables publics sont généralement superficiels, probablement de crainte de divulguer à la concurrence aussi bien les points forts que les points faibles perçus dans l'organisation. Ceux que nous avons enregistrés dans ce domaine sont naturellement soutenus par les vendeurs

de SGDT et les sociétés de conseil du domaine comme le montre le projet RapidPDM³². Il a été proposé une méthode générique pour accompagner la conduite de projets de mise en œuvre des SGDT en entreprise. La méthode encourage l'utilisation des supports informatiques (plus particulièrement des outils d'aide à la décision) tout au long du projet, aussi bien pour les phases amont d'ingénierie de besoins que pour les phases aval d'implémentation [PEL00]. L'accent est mis sur l'importance à impliquer les hauts responsables en entreprise dans un tel projet et à évaluer l'effet de la mise en place d'un SGDT sur la performance globale du processus de développement du produit [HEL00]. Ce phénomène a également été souligné par l'équipe du CERN [HAM98; HAM99].

Par ailleurs, Y.M. Chen et al [CHE97; CHE98] proposent une approche systématique pour l'implémentation des EDM (Engineering Data Management, technologie proche des PDM). Elle s'appuie sur cinq types de diagrammes couvrant les jalons principaux de mise en œuvre de cette technologie. Cette méthode peut vraisemblablement être classée descendante du fait que l'on analyse d'abord les activités de l'organisation avant de rentrer dans le détail de la conception technique à l'aide des techniques de modélisation de donnée/objet. Sous le même paradigme, s'inscrit la méthode OOMIS (Object-Oriented modeling methodology for Manufacturing Information System) [KIM93]. Comme son acronyme l'indique, la méthodologie a pour but de produire un modèle informationnel intégré des systèmes de production. Elle couvre les deux jalons d'analyse et de conception. Dans la première phase, on procède au principe de décomposition sur les fonctions du système de production en employant des modèles IDEF0. Les résultats de cette phase sont rangés dans des tables de fonctions et de données requises pour la réalisation de ces fonctions. Dans la seconde phase, ces résultats d'analyse sont transformés à l'aide des techniques objet en deux diagrammes de classes (les fonctions et les données sont modélisées à l'aide du même concept "objet"). Le diagramme des entités (qui traduit les données) peut être implanté dans les SGBD OO.

Il existe également des contributions d'ouvrages qui décrivent des guides pratiques de conduite de ces projets. Morley et al. [MOR00] empruntent exclusivement les diagrammes UML pour rédiger le CdC du maître d'ouvrage qui représente la phase d'analyse du SI projeté (informatisation du service de production).

Enfin nous concluons cet état de l'art en évoquant une nouvelle fois les travaux de la communauté grenobloise matérialisés par de nombreuses publications [RIE99], entre autres de la thèse de Gzara [GZA00] citée à plusieurs reprises dans notre mémoire. Sur la Figure 2.23, nous positionnons nos problématiques respectives en soulignant leurs complémentarité. Sa problématique concerne l'ingénierie de patrons³³, en modélisant des problèmes récurrents rencontrés dans un domaine et de les proposer à des fins de réutilisation (Processus 1). Notre problématique en revanche se penche sur l'ingénierie même des SIT (Processus 2) en ayant comme donnée d'entrée à la fois le CdC du maître d'ouvrage en entreprise à définir et les patrons d'analyse et de conception, résultats des travaux d'ingénierie de patrons. A notre tour, les nouveaux problèmes soulevés durant le processus de développement du SIT, s'ils

³² RapidPDM s'inscrit dans le cadre du projet ESPRIT 26892. De nombreux livrables intéressants sont disponibles sur leur site officiel : <http://www.rapidpdm.org>

³³ Traduction du mot technique "Pattern". Un pattern peut être défini à l'aide du triptyque (Problème, Solution, Contexte). Il propose une solution à un problème dans un contexte. P. Coad [COA92] définit un pattern orienté objet comme "an abstraction of a doublet, triplet, or other small grouping of classes that is likely to be helpful again and again in object-oriented development". Signalons aussi que le concept de pattern a été d'abord introduit dans d'autres domaines d'ingénierie (architecture, création vestimentaire...) avant le domaine du génie logiciel et de l'objet. En plus des références citées dans la bibliographie, de nombreux articles scientifiques concernant les patterns dans le domaine du génie logiciel sont disponibles sur les liens <http://hillside.net/patterns/papers> et http://www.cetus-links.org/oo_patterns.html

s'avèrent classiques et récurrents peuvent alors servir comme donnée d'entrée pour le domaine de l'ingénierie de patrons (flèche discontinue).

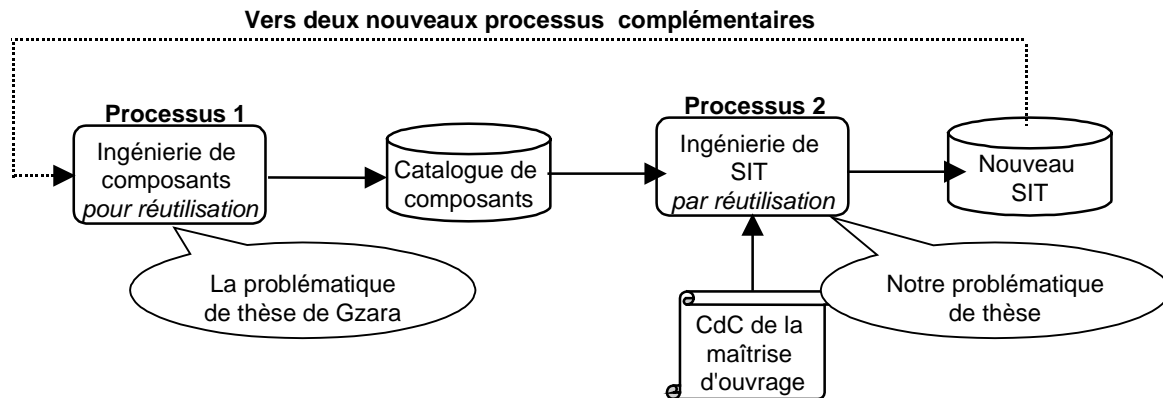


Figure 2.23 Processus pour et par la réutilisation (Adaptée de [GZA00])

IV.4 Ce que nous devons retenir des travaux liés aux développements des SIT

Structuration de données techniques

- Les techniques orientées-objet ont été privilégiées pour leur puissance à modéliser et à représenter les données techniques de nature composite, complexe et évolutive.
- Un intérêt particulier est donné à l'étude des liens potentiels entre les données du produit et du process. Cette volonté d'intégration contribue globalement à la meilleure gestion des données techniques et à l'intégration des SIT en entreprise.
- Concernant la norme MANDATE, comme nous l'avons souligné, malgré sa jeunesse inhérente, sa problématique de standardisation des ressources de production manufacturière nous intéresse de près. Cependant, nous pensons que leur penchant pour les langages standards de données techniques (l'emploi de EXPRESS et du formalisme graphique EXPRESS-G) ne devrait pas faciliter, à notre avis, les tests indispensables en entreprise pour une éventuelle validation de la norme. De plus, la norme n'a pas encore traité ces problèmes cruciaux que nous avons évoqués et n'a pas développé ces modules liés à la représentation de la ressource (hiérarchisation, maturité et état, cas d'utilisation).
- Les travaux de modélisation des systèmes de production améliorent la qualité des modèles d'évaluation des performances de ces systèmes (par simulation ou optimisation). Cependant, ils ne se sont pas préoccupés des activités d'ingénierie de production. La rationalisation de ces processus d'ingénierie est indispensable pour une meilleure structuration des données techniques du domaine (représentation dynamique des objets métier). Les données circulant en ingénierie sont pour la plupart informelles. Les représenter, gérer leur maturité tout au long du projet de développement de produits manufacturiers est un problème crucial soulevé au chapitre 1.

Aspects méthodologiques

- La méthode telle qu'elle est proposée par RapidPDM rejoint les tendances actuelles de "Gaining competitive advantage" en accordant de l'importance aux choix stratégiques survenant plus particulièrement dans les phases amont du projet (choix du SGDT, de la politique du développement etc.). Les méthodes sont nombreuses à introduire ces phases amont d'études stratégiques, tout en fournissant à cet effet des supports informatiques pour l'aide à la décision.

- La méthode proposée par Y. M. Chen [CHE98] couvre complètement les jalons du projet. Cependant, elle utilise des modèles de spécification non standards qui ne sont pas supportés par les AGL du marché.
- Dans la méthode OOMIS [KIM93], seules les vues : fonction et donnée sont prises en compte. Les dimensions temporelles, stratégiques etc. de l'organisation sont négligées.
- L'approche descendante qui prédomine dans ces deux méthodes est une bonne alternative pour la vision globale du système. En revanche, il existe des cas de figures en entreprise, qui rendent ces méthodes parfois insuffisantes. Lorsqu'il faut développer un SI depuis une version existante, il est judicieux de l'étudier en procédant à des techniques issues des approches ascendantes.
- Ces méthodes citées empruntent les nouvelles solutions connues sous le nom de "Filling the gaps". En effet, comme les techniques et les méthodes conventionnelles n'adressent pas la totalité du cycle de vie du projet, on procède alors à la combinaison des techniques existantes dans une seule méthode. Les combinaisons les plus usuelles concernent les diagrammes fonctionnels (IDEF0) ou de flux (DFD) avec les techniques de modélisation de données, comme l'ont montrés ces travaux.
- Les travaux d'ingénierie de patrons sont donc complémentaires au processus d'ingénierie des SIT (Figure 2.23). Les patrons capitalisent les problèmes récurrents rencontrés en développement des SI d'une part et proposent les solutions techniques pour spécifier ces besoins d'autre part. Ces patrons peuvent concerner aussi bien le produit (SI) que le processus de développement de ce produit. Il est donc utile d'étudier ces travaux (malheureusement récents et rares) en même temps que l'étude des besoins du maître d'ouvrage en entreprise. Cependant, le catalogue de patrons proposé par la communauté grenobloise [RIE99;GZA00] reste difficile à exploiter car il a été conçu selon les spécificités d'une entreprise (Schneider Electric) d'une part et les fragments du référentiel sont trop génériques à nos yeux. En revanche, nous avons constaté beaucoup de patrons d'abstraction et d'analyse facilement adaptables [FOW00;COA92].
- Enfin, il serait intéressant pour toute méthode, d'évaluer sa performance en introduisant les indicateurs classiques de temps, de coût et de qualité aussi bien pour les modèles qu'elle utilise que pour son processus de développement, activité peu abordée en ingénierie des SIT (cf. Encart 2.7 pour les indicateurs de qualité).

Par ailleurs, selon le caractère politique de l'organisation, celle-ci peut privilégier l'externalisation de l'activité d'ingénierie de SI (Encart 1.3). Dans ce cas, elle ne se préoccupe que du produit final (SI) et n'est pas demandeuse de méthodes pour la conduite de tels projets. Cependant, il arrive que d'autres choix politiques (question de confidentialité, protection du patrimoine informationnel) font que cette activité soit entièrement déléguée à son département informatique. Celui-ci est alors en charge d'exprimer le besoin de la maîtrise d'ouvrage, de traduire ces expressions de besoin en spécifications fonctionnelles, en détails de conception technique, voire en application logicielle qu'il faut également maintenir. Les services qualifiés du département informatique de l'entreprise devraient étudier la méthode appropriée pour le développement des SI (et des SIT). Si la méthode n'est pas créée au sein de l'organisation, elle est en revanche systématiquement personnalisée et adaptée à la culture d'entreprise, phase coûteuse mais indispensable. En effet, les industriels généralement revendiquent haut de n'avoir jamais utilisé les méthodologies académiques ou du moins sans qu'elles subissent des adaptations en fonction des contraintes et des objectifs de la firme.

La personnalisation consiste entre autres, à choisir les supports informatiques (AGL...), à préciser les livrables à différents stades du projet, à convaincre les acteurs concernés du déroulement du projet... Nous verrons justement au chapitre 3, le cycle de vie de

développement de SI chez Renault, accompagné de solutions que nous préconisons pour une meilleure conduite de ces projets.

Enfin, terminons ce bilan par la problématique toujours controversée concernant l'instrumentation des méthodologies dans les AGL (Encart 2.11).

Encart 2.11 Instrumentation des méthodologies dans les AGL...

An architect's most useful tools are an eraser at the drafting board, and a wrecking bar at the site. Frank Lloyd Wright (1868-1959)

Cette question divise encore l'opinion des chercheurs et des praticiens du domaine. La modélisation formelle du processus de développement et l'application stricte de ses règles peut paraître à la fois positive et négative.

E. Yourdon [YOU92], le parrain des méthodes structurées, disait *"a CASE Tool without methodology support is nothing more than a glorified drawing tool"*. En effet, une technique, présentée dans l'AGL sans un semblant de mode d'emploi ou de règles de construction est probablement difficile à utiliser. La notation UML représentée dans de nombreux AGL, reste difficile à employer si un minimum de guide méthodologique n'est pas proposé (par exemple : exprimer les cas d'utilisation des neuf diagrammes d'UML en fonction de la nature du projet).

Par ailleurs, les expériences antérieures en entreprises montrent à quel point l'utilisateur, ayant travaillé sur un AGL, se voit parfois contraint (sans même se rendre compte !) d'appliquer une méthode plutôt qu'une autre. Or, nous avons montré durant cette partie de l'incomplétude et de l'imperfection des méthodes et que très souvent, l'entreprise a recours à des ajustements, voire à des combinaisons de méthodes existantes.

Pour pallier ce problème, on parle aujourd'hui de métamodélisation dans les AGL. C'est une solution permettant à l'AGL de supporter un maximum de méthodes et de notations existantes. Le développement des "Meta Case" est issu probablement de ce besoin d'offrir un support de plusieurs méthodologies sans pour autant avoir un penchant pour une particulièrement. C'est le cas de l'AGL que nous avons choisi pour notre travail de thèse...

V Conclusion

Rappelons sommairement ce qui constitue la problématique de notre thèse. La maîtrise d'ouvrage chargée d'exprimer les besoins en GDT pour les métiers d'ingénierie de production, a du mal à les formaliser et à les expliquer à la maîtrise d'œuvre. Cette dernière a la même difficulté de traduction de ces besoins en spécifications fonctionnelles. En plus des traditionnels problèmes de communication en entreprise, ces derniers s'amplifient du fait que les modèles d'expression utilisés dans ces phases d'analyse et de spécification du CdC ne sont pas standards et ne sont pas compris par les nombreux acteurs impliqués dans le projet. Ajouté à cela, le manque de rationalisation des processus d'ingénierie de production ainsi que les données y circulant rend encore plus difficile l'ingénierie de ses SIT. De même, les échanges avec les partenaires du projet (éditeur/revendeur du SGDT, sociétés de prestations et de conseil) sont entachés de ruptures. Par conséquent, de tels projets durent généralement plusieurs années avant que le premier livrable ait vu le jour. Le livrable en question devient obsolète car il ne répond plus aux "business requirements" de l'entreprise qui sont en permanente évolution (exemple de l'alliance Renault avec Nissan). Par ailleurs, les acteurs du projet se voient confrontés à des questions controversées qui divisent leurs opinions. Le devenir des SIT existants en est un exemple ; faut-il concevoir en rupture -sans en tenir compte- ou devrait-on les intégrer dans le nouveau SIT ?

Sans toutefois citer les outils AGL homologués en entreprise, notons seulement que parfois les concepteurs du SI sont contraints d'obéir à une méthodologie (sans peut-être le savoir) qui

n'est pas forcément adéquate au projet en question, par la simple raison qu'elle est instrumentée dans l'AGL homologué.

L'objet de ce présent chapitre est d'effectuer une investigation dans la littérature des solutions potentielles à ces problèmes.

Nous avons identifié des thématiques aussi larges et pluridisciplinaires que les :

- Technologies d'information et supports d'intégration.
- Modèles de spécification pour les SI et langages dédiés aux données techniques.
- Architectures de références et techniques d'entreprise modeling.
- Ateliers de Génie Logiciel.
- Méthodes d'ingénierie des SI aussi bien académiques que pratiques.
- Développements des SIT en entreprise portant sur les travaux de structuration de données techniques et de quelques aspects méthodologiques.

Nous avons conclu la présentation de chacune de ces thématiques par des bilans évoquant leurs apports et limitations tout en se positionnant par rapport à notre problématique de thèse.

A l'issue de cette investigation, nous sommes en mesure de proposer les orientations et les choix de solutions que nous préconisons pour notre problématique de thèse, objet du troisième chapitre.

CHAPITRE TROISIEME

Orientations de la thèse et plan d'actions

La difficulté d'aboutir à un tout dans une architecture de complexités et de contradictions est due à l'insertion d'une multitude d'éléments divers dont les relations sont incompatibles ou qui font partie des catégories dont la perception est la plus ardue.

Robert VENTURI (1925-). De l'ambiguïté en architecture – La dure obligation du tout.

I Introduction

A la conclusion du chapitre précédent, nous avons rappelé les goulets d'étranglements à la mise en œuvre des SIT en entreprise et plus particulièrement en ingénierie de production, résultat de l'analyse décrite au chapitre 1. L'état de l'art effectué au chapitre 2 a pour but de rechercher les solutions potentielles à ces problèmes cruciaux. L'objectif de ce présent chapitre est de présenter les solutions retenues pour rédiger le CdC de la thèse.

I.1 Grandes orientations de notre thèse

Notre problématique de thèse traite particulièrement deux dimensions :

- Rédaction du CdC de la MOA/MOE en entreprise.
- Démarche pour l'élaboration du CdC et plus globalement pour la conduite des projets de mise en œuvre des SIT en entreprise.

Pour résoudre cette problématique, nous avons étudié dans le second chapitre les thématiques suivantes :

- Modes d'échange, de standardisation et d'intégration des données techniques en entreprise
- Modèles de spécification, techniques d'Enterprise Modeling et caractéristiques des AGL
- Méthodes en ingénierie des SI et développement des SIT en entreprise.

A la fin de la présentation de chacune de ces thématiques, nous avons établi des bilans personnels analysant les apports et les limitations de ces contributions vis à vis de notre problématique de thèse (chapitre 2, II.4, III.4, IV.2 et IV.4). Ces bilans nous conduisent naturellement à des orientations que nous exposons dans le présent chapitre.

Dans l'ordre de ces thématiques étudiées, nous présentons en premier le choix vers lequel s'orientent les constructeurs automobiles en matière de technologie d'intégration ; choix pour lequel notre problématique de thèse n'est pas affecté. Nous expliquons, par la suite, les modèles de spécification appropriés, le cadre de référence choisi dont le premier rôle est l'organisation des modèles de spécification retenus. Le choix de l'AGL suit naturellement et dépend beaucoup des solutions précédentes. Enfin, la dernière partie est consacrée à l'explication des aspects méthodologiques pour la construction du CdC. Nous mettons en avant les atouts et les faiblesses de la méthodologie homologuée chez Renault. Ces dernières nous empêchent de l'adopter. Nous préconisons également des solutions pour l'amélioration globale de la conduite des projets SIT.

Mais avant d'introduire toutes ces orientations, effectuons un autre constat personnel et délimitons notre périmètre d'étude dans ce qui suit.

I.2 Ingénierie véhicule – ingénierie SI : éléments de comparaison

A plusieurs reprises, nous avons noté des préoccupations communes à ces deux professions : l'ingénierie de développement du véhicule et l'ingénierie des SI. Nous pensons que ces deux activités peuvent bien s'ajuster et se compléter. Il est peut-être utile d'effectuer des études comparatives du point de vue des techniques déployées et des méthodes de conduite de ces projets. Le but est de pouvoir préconiser les meilleures solutions testées et d'éviter de répéter les mêmes erreurs. Nous nous sommes ainsi livrés à une petite analyse comparative en guise de démonstration de notre hypothèse (Figure 3.1). Quelques exemples le confirment déjà, comme c'est le cas pour les indicateurs de qualité des processus, populaires en entreprise

manufacturière et qui ont par la suite séduit la communauté de l'ingénierie des SI (Encart 2.7).

Le caractère de sous-traitance est de plus en plus la règle dans les deux professions. Ainsi, c'est généralement au fournisseur d'assembler les modules (livrables) du système (SI, ligne d'assemblage, véhicule...) en respectant les contraintes d'implémentation imposées par le constructeur. Celui-ci détaille la conception spécifiée par le designer. Ce dernier traduit les expressions du besoin du maître d'ouvrage en spécifications fonctionnelles tout en prenant en compte les nouveaux "business requirements" du système présentés par le planificateur...

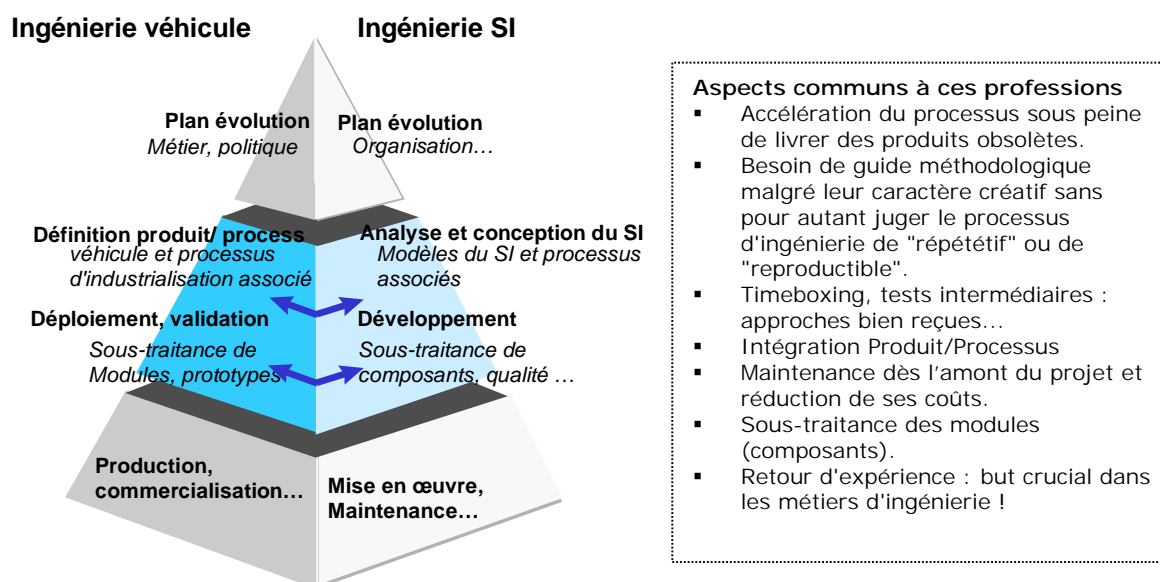


Figure 3.1 Ingénierie Véhicule – Ingénierie SI : Eléments de comparaison

II Périmètre de l'ingénierie de production

L'étude du terrain, effectuée chez le constructeur Renault, nous a permis de dégager le domaine de l'ingénierie (de système) de production comme un métier très concerné par les problèmes de la GDT et de l'ingénierie des SIT. L'ingénierie de production, rappelons-le, se situe en amont de la fabrication et chevauche les activités de design du produit. L'activité est communément appelée "Ingénierie de Process" ou "Process" tout court par les industriels. Nous avons cherché en vain une définition standard à ce terme. Nous nous sommes finalement résignés à réutiliser la définition de l'ingénierie de produit, en supposant que le produit, au sens général, peut être un artefact, un SI ou encore un système de production. Cependant, nous tenons à ce que la partie du problème à traiter soit tout aussi stratégique pour l'entreprise que pour la recherche. Pour simplifier, nous distinguons trois dimensions pour délimiter le périmètre de l'ingénierie de production (Figure 3.2), dimensions que nous détaillons ci-après.

II.1 Jalonnement du projet

Ce sont les phases amont du projet qui nous intéressent. Nous avons vu que les problèmes cruciaux en GDT se concentrent essentiellement à ce stade du projet où le processus d'ingénierie n'est pas rationalisé et les données y circulant sont mal identifiées. C'est la raison pour laquelle, les évaluations amont de performance (simulation, optimisation...) s'appuient sur les données d'exploitation par défaut. Notons aussi que ces stades amont du projet sont

particulièrement stratégiques en entreprise. Celle-ci ne les sous-traite d'ailleurs pas de par les conséquences des décisions prises à ces phases amont sur l'ensemble du processus d'élaboration du produit.

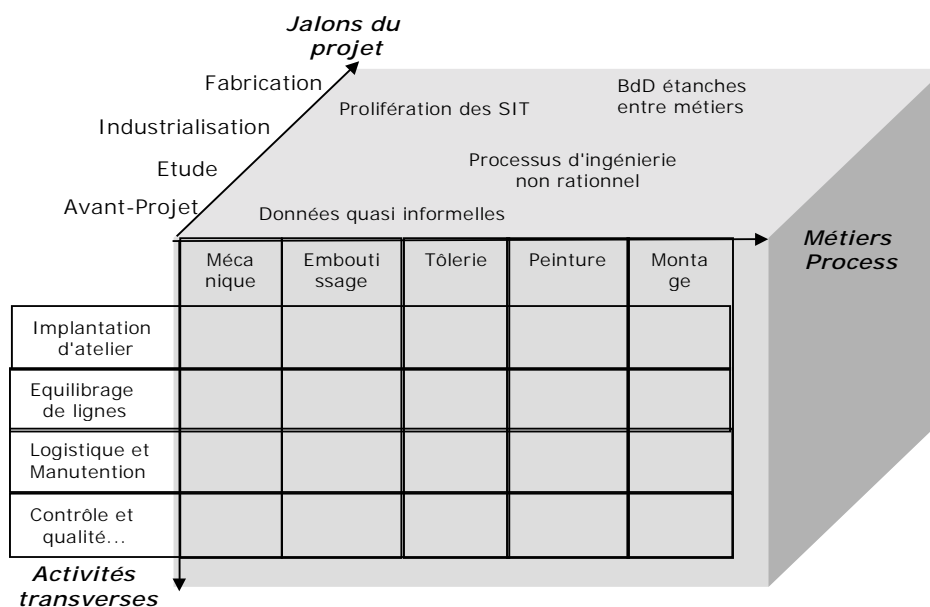


Figure 3.2 Périmètre d'étude : Ingénierie de production

II.2 Métier process

La direction de la recherche à laquelle nous sommes rattachés appartient hiérarchiquement à la direction Ingénierie Véhicule (IV). De ce fait, nous nous sommes intéressés naturellement sur les métiers de l'IV : emboutissage, peinture, tôlerie et montage. Cependant, malgré le cloisonnement entre les deux directions principales : l'IV et la Mécanique, nous retrouvons aisément des problématiques communes en matière de GDT¹. Par ailleurs, nous nous sommes vite aperçus de notre incapacité à appréhender l'IV dans sa globalité durant un travail de thèse. Nous avons alors décliné les deux métiers de tôlerie et de montage pour commencer l'étude et la généraliser par la suite à l'emboutissage puis enfin à la peinture, sans toutefois nous investir trop dans ce dernier métier. Cet ordre préférentiel se justifie par la présence des principaux problèmes cruciaux dans ces premiers métiers : prolifération des SIT, conception verticales des BdD, complexité du processus d'ingénierie, diversité produit/process etc. De plus, ces deux premiers métiers ont toujours suscité un intérêt général en industrie, en recherche ainsi que chez les organismes de standardisation.

II.3 Activités transversales

Chacun de ces métiers principaux évoqués ci-dessus est constitué de diverses activités tout aussi importantes en entreprise : activité de préparation (construction des opérations), d'implantation des ateliers, de définition des ressources... Certaines de ces activités sont représentées par des directions indépendantes comme la direction logistique, de qualité...

¹ En effet, après la validation de nos travaux par la maîtrise d'ouvrage de l'IV, ils ont été vite transférés à la direction Mécanique et bien reçus par ses responsables mandatés. A travers ces rencontres qui réunissaient à la fois les responsables de l'IV et de la Mécanique, nous pouvons confirmer que ces deux directions se rejoignent dans les problématiques essentielles en GDT (présentées tout au long du chapitre 1) et peuvent se différencier au niveau d'éléments secondaires relatifs aux métiers de conception et d'industrialisation des moteurs et des véhicules.

Par ailleurs, nous avons souligné au chapitre 1, que l'un des intérêts majeurs des réorganisations de ces différentes directions est de les rapprocher les unes des autres d'autant plus qu'elles sont concernées par le même processus opérationnel qui est l'ingénierie de production. Nous baptisons l'ensemble de ces activités de "transversales" du fait qu'une même activité intervient dans tous ces métiers process.

Encore une fois, l'immensité et la diversité de ces activités nous empêche de les détailler pendant la période de thèse. Cependant, nous avons pris le soin de prendre en compte les activités interdisciplinaires où sont confrontées plusieurs expertises et compétences du domaine. A travers plusieurs exemples (chapitre 1, II.4), nous avons illustré ces interactions.

Enfin, signalons que cette délimitation du domaine est très simplificatrice voire réductrice. Nous suggérons, pour qui s'intéresse à la genèse de cette activité et à l'apparition des problèmes de GDT, de revenir au chapitre 1 de la thèse mais aussi à l'annexe B où sont récapitulées les problématiques principales de l'ingénierie de production.

III Choix de la technologie d'intégration

Généralement, les entreprises manufacturières (et plus particulièrement les constructeurs automobiles) sous-traitent toute la phase d'implémentation du SIT voire son déploiement également. Cette tendance est renforcée par l'émergence des solutions intégrées sur le marché. C'est-à-dire, les éditeurs/revendeurs de logiciels fournissent le plus souvent à leurs clients des gammes de logiciels métiers accompagnés de leur propre SGDT comme technologie d'intégration. Nous caractérisons cette solution "d'architecture centralisée" (Figure 3.3). Ce fournisseur sera en charge de mettre en place ses gammes de produits tout en respectant les contraintes d'implémentation imposées par l'entreprise cliente. Il arrive en revanche que l'interopérabilité entre ces outils logiciels soit assurée par d'autres supports ou technologies d'intégration comme nous l'avons vu avec les travaux exposés dans la partie II.3 du chapitre précédent. Dans ce cas, l'implémentation est caractérisée par une architecture distribuée (Figure 3.3). Mais avant toute implémentation, l'expression du besoin et la rédaction du CdC est affaire du maître d'ouvrage et du maître d'œuvre appartenant à l'entreprise. La conduite de tels projets est aussi une responsabilité de l'entreprise cliente. Nous pensons que les orientations de celle-ci en matière de technologie d'implémentation du SIT n'affecte pas notre problématique de thèse. En effet, les problèmes liés à l'expression du besoin, aux spécifications internes du SIT projeté ainsi qu'à sa conduite restent indépendantes de la technologie de mise en œuvre. Seuls les détails de conception technique qui interviennent plus tard dans le projet et qui sont de plus en plus délégués à des prestations externes, en sont conditionnés. Aujourd'hui, ce type de contraintes sont considérablement réduites grâce aux nouvelles technologies d'information comme le Data Warehouse, Data Mart...

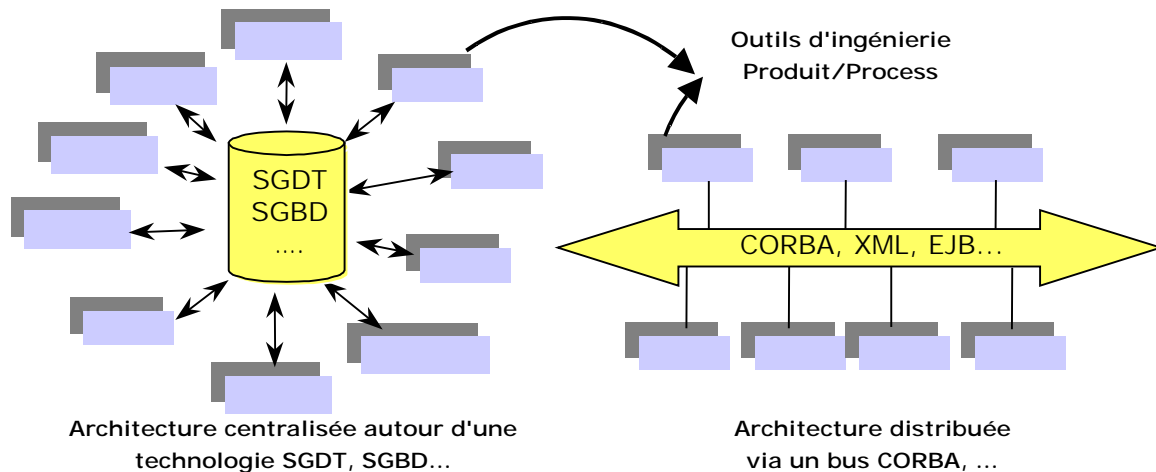


Figure 3.3 Technologie d'implémentation du SIT

IV Quels modèles de spécification ?

Après avoir éliminé les langages spécifiques aux données techniques (PSL², EXPRESS) et les techniques d'analyse structurée, les techniques potentiellement utiles sont celles issues du domaine de l'Information Engineering (IV.1), de modélisation des processus "Business Process Modeling" (IV.2) et bien entendu de la dernière notation des techniques Orientées-Objet (IV.3).

IV.1 Techniques de modélisation de données

Notre décision qui est de réutiliser les SIT existants en entreprise (choix méthodologique, cf. VII) nous amène à les étudier. Ces derniers, lorsqu'ils sont informatisés, sont conçus à l'aide des techniques de modélisation des données (E-R) et implantés dans des SGBD relationnels (comme ORACLE, versions antérieures à V.8).

Par ailleurs, dans le bilan consacré à ces techniques, nous avons constaté leur insuffisance à représenter la complexité des données techniques d'une part. D'autre part, pour que le modèle de données résultant soit aussi complet et représentatif du monde réel, l'analyse des processus du système concerné est indispensable. Cette activité est plutôt du ressort des diagrammes ci-après.

IV.2 Techniques de modélisation des processus (BPM)

La prolifération des techniques de modélisation des processus ne nous a pas empêché de discerner les techniques originales et de les présenter à l'aide d'exemples (chapitre 2, II.1). Ces techniques nous aident d'abord à comprendre deux concepts complexes : système et problème, indispensables en phase d'ingénierie du domaine. Par la suite, à l'aide de ces différents diagrammes, toutes les composantes d'un système (processus, fonction, organisation, application...) sont représentées. Le résultat de cette modélisation est appelé par les anglo-saxons "Business Model"³, dont la définition est plus large qu'un modèle de processus (chapitre 2, III.2). Certains chercheurs français désignent le SI informatisé, celui qui se matérialise en BdD, logiciel... et le SI organisationnel (et donc ce business model) partie non informatisable du SI. Ces deux parties, formant ainsi le SI global, sont considérées toutes deux comme des fins en soi.

² Si PSL n'est pas retenu en tant que langage, il demeure cependant utile pour rationaliser les concepts propres au domaine.

³ A ne pas confondre avec "modèle de domaine" dans la réutilisation en ingénierie des SI [CAU01; MAI94].

Nous avons retenu l'ensemble des diagrammes de CATALYST⁴ [CSC00], technique récente mais qui a l'avantage d'intégrer toutes ces dimensions du système. Dans les autres techniques, ces dimensions sont présentées séparément ou partiellement (comme la suite IDEF) ou encore véhiculant des méthodologies pas forcément avantageuses dans tous les cas de figure (comme ARIS).

Ainsi, comme les techniques de modélisation des données, nous retenons aussi les techniques de modélisation des processus (BPM) pour rédiger notre CdC.

IV.3 UML et patrons d'analyse et de conception

L'utilisation des techniques de modélisation des données, nous l'avons vu, est nécessaire mais non suffisante. Les caractéristiques des données techniques (chapitre 2, I.1) nous amènent indiscutablement au choix de la notation UML. Nous n'allons pas rappeler le succès de son utilisation en matière de structuration des données (et plus généralement des objets). Nous tenons cependant à préciser que UML ne se suffit pas pour décrire avec facilité et efficacité les différentes dimensions d'un système (processus, organisation,...) et particulièrement dans les phases amont de l'analyse. Nous verrons à l'exposition de notre travail dans les deux chapitres suivants que les diagrammes comme "UML Activity" ou "UML State" sont loin d'égaliser la richesse et la flexibilité des techniques de modélisation des processus (BPM).

Nous pensons en effet que le diagramme d'activité d'UML est plutôt destiné à représenter les processus automatisés. Le diagramme des états, plus formel, est mieux adapté aux processus qui donnent lieu à beaucoup d'interactions. Nous les employons toutefois, très tard dans le projet, pour décrire par exemple les traitements automatisés que l'on souhaite obtenir de l'application développée.

Néanmoins, quelques pionniers du domaine de l'objet [JAC95;MAR'00;MOR00] promeuvent l'emploi exclusif de la technologie objet (UML). Dans ce cas là, les diagrammes des cas d'utilisation sont créés dès les niveaux contextuels. Ils sont documentés par la suite avec le diagramme d'activité pour décrire les transitions et les conditions. Ils peuvent être complétés aussi à l'aide du diagramme de séquence ou de collaboration pour décrire les scénarios des cas d'utilisation.

Cela dit, nous restons persuadés qu'il faut beaucoup plus d'intelligence pour représenter les différents composants du système, à différents niveaux d'abstraction qu'avec comme seul concept "l'objet"...

Nous sommes tout aussi persuadés que l'une des créations spectaculaires en objet est le concept de "design pattern" [MAR00], introduit dans le chapitre précédent (IV.3, IV.4). Nous avons explicité l'intérêt des travaux d'ingénierie de patrons par rapport à notre problématique de thèse. Le principe de base d'un patron, rappelons-le, est de capitaliser un problème récurrent d'un domaine et sa solution de manière à faciliter sa réutilisation ou son adaptation. Nous nous sommes ainsi intéressés spécialement aux patrons d'analyse et de conception préliminaire⁵ que l'on peut retrouver dans des catalogues classés par domaine (commerce,

⁴ Créée par la société américaine Computer Sciences Corporation (voir le site de la filiale française : <http://www2.csc.com/france/Accueil/accueil.htm>). A l'heure actuelle, il n'existe toujours pas d'ouvrage officiel qui puisse l'évaluer ou la répertorier comme une méthodologie excepté les publications internes de son inventeur (Quelques-unes sont disponibles dans www.csc.com). Notons que le groupe pharmaceutique NOVARTIS a officiellement opté pour CATALYST pour l'ingénierie des processus de son organisation.

⁵ Par opposition aux *patrons d'implantation* qui décrivent comment implanter dans un langage spécifique, certaines caractéristiques définies en conception. Les technologies de SGDT par exemple, sont fournies avec des métamodèles (objet ou relationnel) sur lesquels seront raccrochés les modèles métier. Ils peuvent aussi servir pour réaliser l'interopérabilité du SGDT avec les outils d'ingénierie... Ces patrons d'implantation sont appelés aussi "patrons logiciel" et les patrons d'analyse et de conception "patrons métier" [GZA00;RIE99].

métriques, organisation...). Dans l'Encart 3.1, nous présentons à l'aide du diagramme de classe UML et d'une façon très simplifiée (sans formulaire de représentation⁶), des exemples de patrons que nous avons adaptés pour raffiner nos modèles d'expression ou pour représenter quelques problèmes de conception.

Ainsi, nous avons retenu les techniques de modélisation de données, de processus et l'objet. Examinons à présent comment nous allons les utiliser pour rédiger le CdC du MOA/MOE pour l'ingénierie de production.

Encart 3.1 Exemples de patrons d'abstraction et d'analyse

Imagination continually frustrates tradition; that is its function. John Pfeiffer

The Composite pattern diagram shows a Client using a Component interface. Component has methods Operation(), Add(Component), Remove(Component), and GetChild(int). It is implemented by Leaf (with Operation()) and Composite (with Operation(), Add(Component), Remove(Component), and GetChild(int)). Composite has a composition relationship with children of type Component. A note indicates a recursive call: 'for all g in children g.Operation();'. The Organisation Hierarchy pattern shows an Organization class with methods parent and children. It has a composition relationship with children of type Organization. A note says '{a company is the parent of divisions which are parents of departments}'. Organization is implemented by Company, Division, and Department. The Role pattern shows an Acteur class with a composition relationship with Role (multiplicity 0..*). Role is implemented by Role1, Role2, and Role3.

Patron "Composite" de E. Gamma [GAM95]

Il permet la construction de nomenclatures, compositions récursives d'objets. On distingue ici uniquement deux types d'objets : composite et simple (feuille de l'arbre).

Patron "Organisation Hierarchy" de M. Fowler [FOW00]

S'il n'existe pas de comportement spécifique entre les niveaux de l'organisation, une seule classe (organisation) peut suffire. Le cas ci-contre est utilisé lorsqu'on note des variations à chaque niveau de l'organisation. Parfois, il est préférable de remonter le lien de décomposition sur la classe super type.

Patron "Rôle" de P. Coad [COA92]

Il donne la possibilité à l'objet d'adhérer et/ou de perdre ses rôles. Il est utilisé lorsqu'un objet a des propriétés variables selon le rôle joué.

⁶ Formalismes standards de description de ces patrons. Les plus populaires sont : "Portland Form" (www.c2.com/ppr/about/portland.html), le formalisme de E. Gamma ; celui que la communauté grenobloise utilise [RIE99] ou encore de P. Coad : un autre pionnier du domaine [COA92].

V Cadre de Zachman : différents usages

Dans le bilan consacré aux architectures d'entreprise et des SI (chapitre 2, III.4), nous avons explicité les faiblesses des cadres orientés métier, incomplets (en vues ou en niveaux d'abstraction) et des cadres conceptuels qui n'ont pas été matérialisés en outils logiciels. Nous avons également présenté les raisons des difficultés d'adaptation de ces cadres lorsqu'ils supportent des méthodologies. Le choix est alors porté naturellement sur le cadre de Zachman, étayé dans l'encart 2.6. Pour notre problématique de thèse, ce choix peut être justifié à l'aide des arguments suivants.

Ne proposant ni une méthodologie ni de nouveaux modèles de spécification, le cadre de Zachman fournit cependant une structure logique pour organiser et classer les modèles de spécification de la littérature, notamment ceux que nous avons retenus pour rédiger notre CdC, parties grisées sur la Figure 3.4. Ainsi, de haut en bas, les techniques BPM couvrent les deux premières lignes. Les techniques de modélisation de données couvrent la colonne 1 (Donnée). L'objet, quant à lui, couvre les parties grisées des lignes 3, 4 et 5.

Plus qu'un schéma de classification des modèles, le cadre de Zachman répond aux questions typiques des acteurs en développement d'un système (SI, entreprise, artefact...), parmi lesquelles : quelle technique choisir et quand l'utiliser ? Qu'elle vue référence-t-elle et quel niveau d'abstraction traduit-elle ? etc.

Il propose également des règles de transformations logiques d'une information se trouvant dans une cellule vers une autre. Il permet des mécanismes de récursivité qui se traduisent par le versionnement et la décomposition hiérarchique dans les modèles...

Par ailleurs, les flèches descendante et ascendante de la Figure 3.4 désignent les contraintes du système. La flèche descendante exprime que certaines contraintes s'accumulent de haut en bas. En effet, à l'assemblage des livrables pour lesquels il est mandaté, le sous-traitant doit tenir compte des contraintes d'implémentation exprimées par le constructeur. Ce dernier doit respecter les spécifications du designer au développement du système. De même, le designer traduit les expressions du besoin en phase d'analyse en spécifications fonctionnelles. En revanche, les contraintes ascendantes, nous l'avons dit, sont considérablement réduites grâce aux nouvelles technologies d'information (comme les Data Warehouse), beaucoup moins contraignantes que les technologies traditionnelles de mise en œuvre des SIT (comme les SGBD relationnels).

Enfin, toutes ces caractéristiques favorisent l'obtention d'une continuité entre modèles de spécification et donc dans la conduite du projet. Cependant, ce cadre conceptuel ne suffit pas seul, il faut un AGL qui doit supporter tous ces principes...

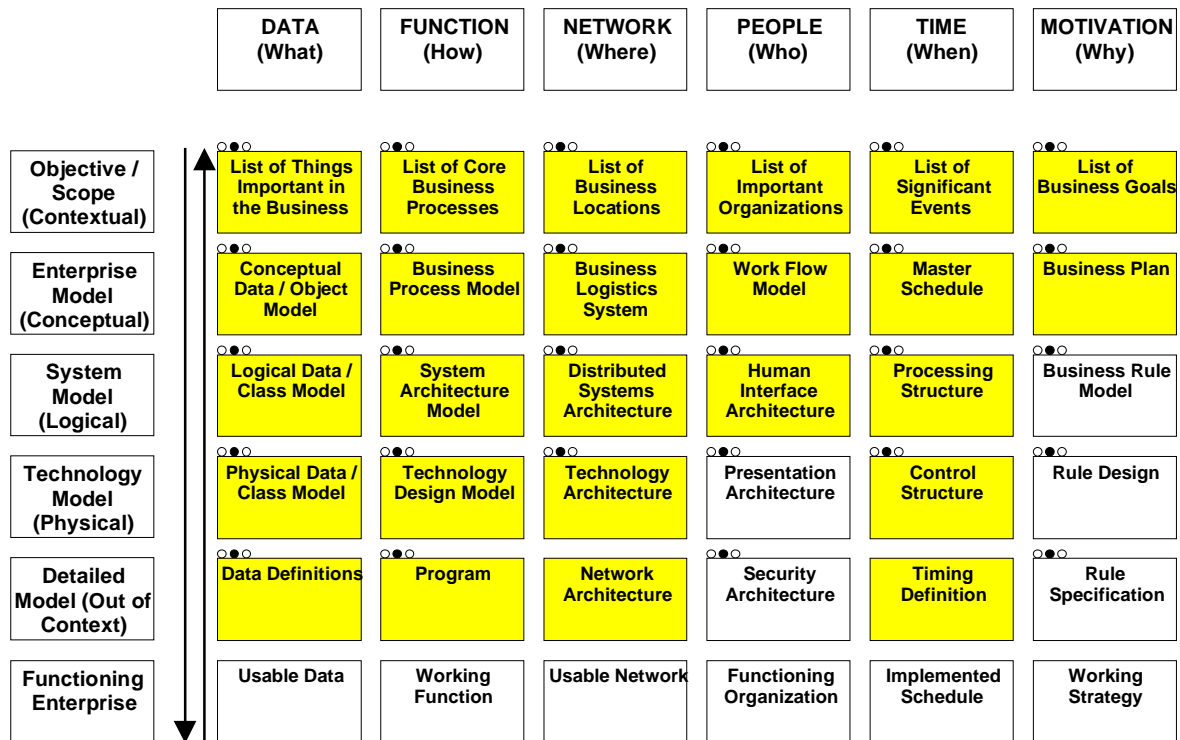


Figure 3.4 Cadre de Zachman : couverture des modèles retenus

VI L'atelier logiciel : System Architect de Popkin Software

Dans le chapitre précédent, les caractéristiques des AGL -perçues ou non par l'utilisateur (comme les aspects d'intégration et les métriques)- sont étudiées. L'architecture idéale d'un AGL est également décrite. En outre, la question controversée au sujet de l'instrumentation d'une méthodologie dans un AGL est débattue dans l'encart 2.11.

Il en ressort qu'il est plus intéressant qu'un AGL puisse supporter des notations et non pas des méthodologies, difficilement adaptables voire inadéquates pour certains projets. Pour toutes ces raisons et celles évoquées dans le chapitre précédent, les AGL homologués en entreprise ne nous satisfont pas. Nous nous sommes alors livrés à une évaluation des outils du marché⁷ pour choisir la solution qui puisse répondre au mieux à toutes nos exigences.

Les choix précédents en termes de techniques et d'architecture nous amènent à nous orienter vers cette gamme d'outils appelée "Meta Case" qui a la particularité de présenter un maximum de notations sans pour autant avoir un penchant pour une méthodologie particulière. Le choix est porté sur l'AGL System Architect de la société Popkin Software⁸.

SA2001 est plus qu'un outil de dessin ou de modélisation. Il est conçu autour d'une encyclopédie qui stocke à la fois les diagrammes, ses symboles ainsi que ses définitions, avec des possibilités d'extraction d'informations voire de communication entre encyclopédies. L'encyclopédie est partageable entre acteurs du projet en la rangeant par exemple dans un serveur (Figure 3.5).

⁷ Nous avons effectué cette étude au début de notre thèse. Les principaux résultats de cette étude sont résumés à l'annexe C du manuscrit.

⁸ Pour en savoir plus cf. www.popkin.com

De plus, l'outil supporte les notations et les techniques les plus populaires appartenant aux domaines de (Figure 3.6) :

- L'Information Engineering : techniques de modélisation de données (E-R, normalités...).
- L'analyse structurée : notations des pionniers comme Gane/Sarson et Yourdon/DeMarco.
- L'objet : des anciennes notations jusqu'à UML.
- Business Process Modeling : les techniques IDEF et CATALYST.
- Quelques autres diagrammes de spécifications répandus dans les systèmes commerciaux.

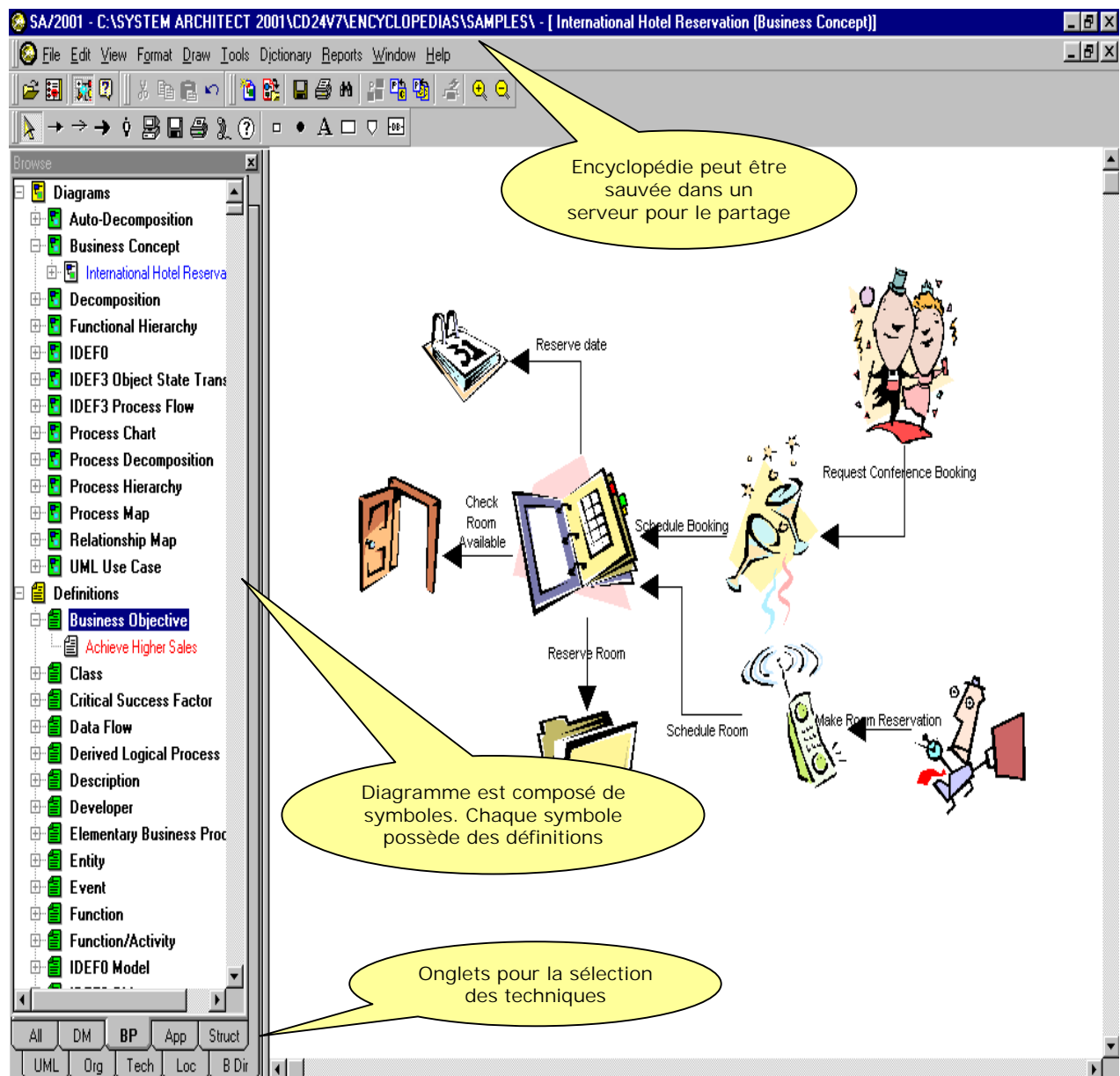


Figure 3.5 Environnement utilisateur de System Architect de Popkin Software

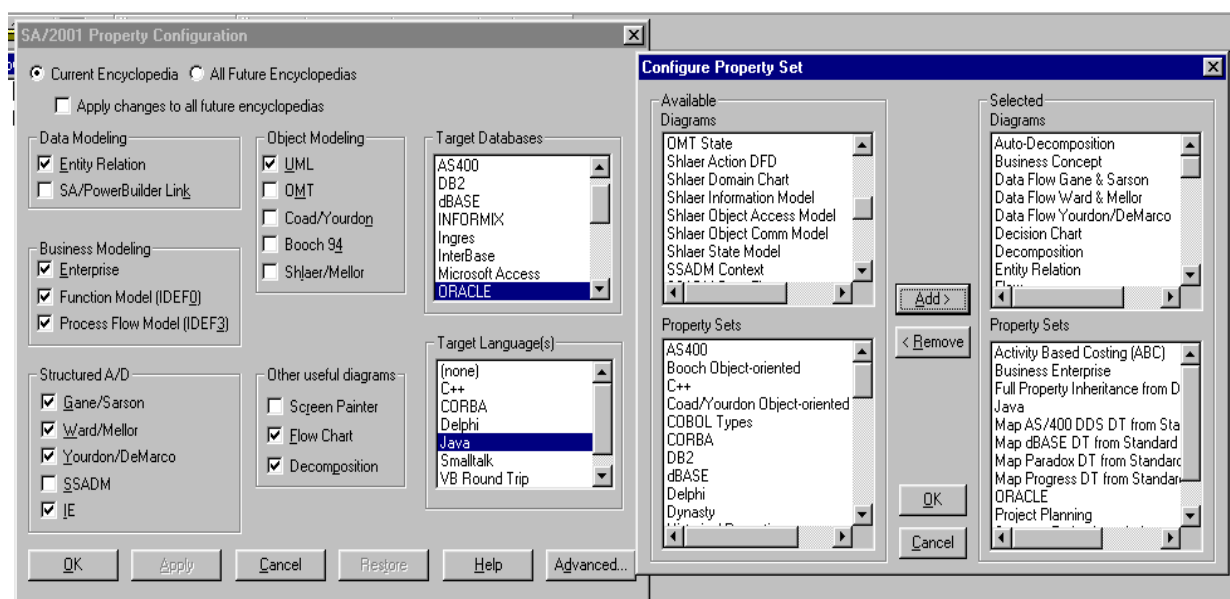


Figure 3.6 Notations et techniques supportées par l'AGL System Architect de Popkin Software

Le travail de modélisation s'effectue dans un environnement complètement intégré⁹ en assurant une continuité entre modèles mais aussi dans les jalons du projet.

En outre, il offre plusieurs fonctionnalités nécessaires à notre travail comme :

- Documentation : sur des supports Word de MS, HTML... en guise de livrables intermédiaires.
- Reverse Engineering et génération de code : possibilité de lire des BdD, du code... mais aussi de générer des schémas de BdD pour permettre les tests et les prototypes.
- Transformations bidirectionnelles entre les modèles objet (classe UML) et les modèles E-R.
- Présence de nombreuses règles de cohérence pour contrôler ou raffiner la construction des modèles de spécification.
- De surcroît, l'outil supporte le cadre de Zachman et les fonctionnalités qui en découlent.

A présent que les choix liés aux techniques et aux outils sont faits, il reste cependant une dimension nécessaire pour mener à bien notre travail ; il s'agit des aspects méthodologiques liés à la conduite d'un tel travail. Ce que nous allons présenter de suite...

VII Quelles approches méthodologiques ?

Dans cette partie, nous traitons tous les aspects méthodologiques liés à notre problématique de thèse. Nous avons besoin en effet d'un guide méthodologique pour construire notre CdC. Nous avons voulu aussi généraliser ces principes pour l'ingénierie même des SIT en entreprise. Tout d'abord, introduisons la méthodologie homologuée par la direction informatique chez Renault.

⁹ Voir chapitre 2, III.3 pour comprendre les différents types d'intégration que l'on peut associer dans un AGL, notamment les quelques références bibliographiques [ALS99; BRI93; HUR93].

VII.1 Méthodologie préconisée en entreprise : points forts et points durs

Nous avons abordé la conduite de ces projets au chapitre 1 sans introduire explicitement la méthode homologuée chez Renault nommée EXPERT. D'une part, parce qu'elle n'est pas systématiquement appliquée dans les projets d'entreprise ; seuls quelques fragments sont imposés (l'emploi des AGL homologués). D'autre part, nous avons voulu présenter d'abord les travaux de la littérature (chapitre 2, IV) pour nous imprégner des concepts liés au domaine.

Notons que cette section est à caractère confidentiel. Nous ne pouvons donc pas détailler les préconisations de EXPERT ; notamment la nature des livrables, l'objet de chaque jalon, les noms des AGL utilisés ou encore les paramètres économiques de la conduite de ces projets. Nous essayons néanmoins de décrire quelques points forts que nous avons intégrés dans le CdC de la thèse. Par la suite, nous justifions globalement de la non adéquation des autres principes pour notre problématique de thèse. Signalons aussi que les opinions émises au sujet de EXPERT n'engagent que nous.

En se basant sur la Figure 3.7, plusieurs points forts sont perçus :

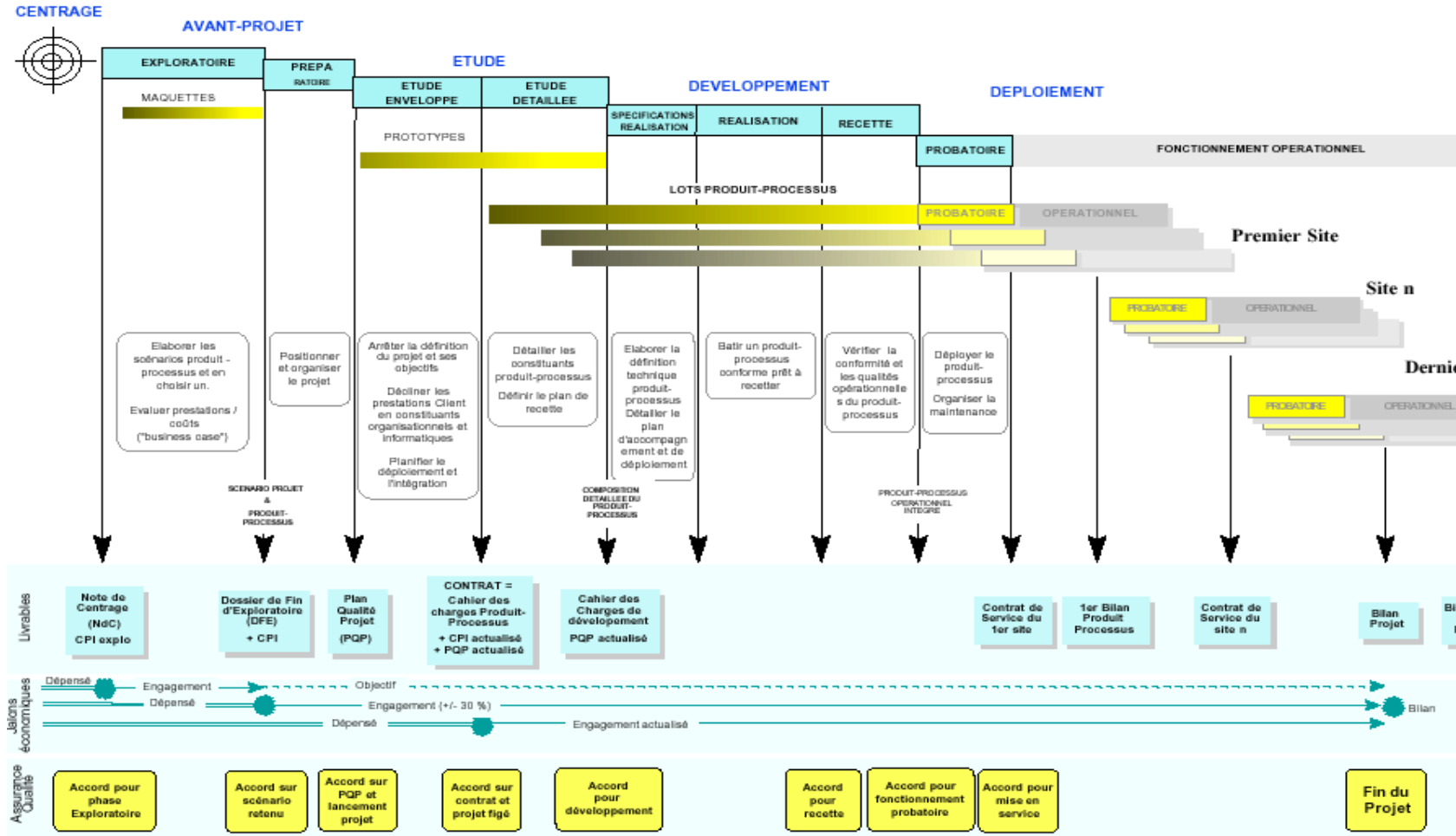
- *CdC conjoint MOA/MOE* : volonté accrue d'impliquer ces deux types d'acteurs pour construire un CdC commun, en précisant les rôles de chacun d'eux et sans privilégier l'un ou l'autre (voir encart 1.4 pour comprendre leurs missions respectives).
- *Intégration produit/processus* : interdépendance perçue entre les différents livrables (produit) et les processus (utilisateur, métier, ...) (voir chapitre 2, IV.4 pour comprendre tous les aspects d'intégration que l'on peut percevoir entre le produit et le processus).
- *Promouvoir le timeboxing et le processus itératif* : en pilotant par les délais, rapprochant les jalons tout en rendant visibles les livrables à l'ensemble des acteurs du projet. Ce sont des approches de mieux en mieux reçues en entreprises comme nous l'avons constaté dans l'état de l'art. Retenons aussi la volonté de remonter en amont la maintenance, phase se déroulant traditionnellement après la mise en service du produit.
- *Présence d'une chaîne de clients dans le processus de développement* : par opposition aux traditionnels projets pensés "mono-client" (utilisateur final du SI), la prise en compte des différents clients intermédiaires (comme la MOE vis-à-vis de la MOA) et externes (partenaires et prestataires) permet de prendre en considération leur satisfaction dans un souci d'amélioration globale de la conduite du projet.
- *Faire précéder tout projet d'un avant-projet* : cette phase est appelée "conceptual design" par les anglo-saxons. Elle consiste à formaliser les concepts de solutions à partir d'une première définition grossière des spécifications.

Par ailleurs, la phase "centrage" (Figure 3.7) créée plus récemment reste ambiguë et son CdC n'est pas clairement défini. C'est probablement la raison de la lenteur constatée pour traiter cette phase. Celle-ci a parfois nécessité plus d'un an, ce qui est particulièrement long.

La rupture flagrante constatée entre la phase d'analyse du domaine et de conception du SI est sans doute liée à deux facteurs. D'abord, le changement d'AGL, donc des notations mais aussi des règles formelles qui y sont présentes. D'autre part, l'absence d'une étude des liens potentiels entre ces deux jalons principaux. De plus dans EXPERT, les deux jalons sont présentés séparément dans deux manuels. Enfin, nous avons également constaté qu'il est des décisions stratégiques que seuls les hauts responsables peuvent prendre et assumer. La MOA/MOE, à notre avis, ne peut assumer une telle responsabilité.

Tout comme les AGL homologués en entreprise, nous n'avons pas non plus retenu la méthode homologuée EXPERT pour notre problématique de thèse.

Figure 3.7 Cycle de vie des projets informatiques chez Renault (source Renault)



VII.2 Démarche intégrée : stratégie ascendante et stratégie descendante

La démarche est le processus opératoire grâce auquel s'effectue le travail de modélisation, de description et de réalisation du SI [RIE99]. A travers les nombreux travaux d'ingénierie des SI/SIT en entreprise (chapitre 2, IV.2, IV.4), nous avons fait le constat de la complémentarité des approches top-down et bottom-up. Rappelons ce qui les caractérise :

Les approches descendantes partent de l'analyse globale du système et emploient les techniques traditionnelles de "problem analysis" pour produire des modèles orientés problème.

Les approches ascendantes empruntent les techniques de rétroingénierie. Elles analysent les composants du domaine (SI par exemple) pour produire des objets communs caractérisant ce domaine. Les modèles résultants sont orientés solution.

Dans la pratique, nous l'avons vu, ces deux stratégies ne s'opposent pas et peuvent s'employer en même temps. Nous avons alors décidé de les intégrer dans notre démarche avec les arguments suivants (Figure 3.8) :

Pour top-down : l'approche est plutôt classique, longtemps appliquée dans les méthodes de type Merise. Elle a l'avantage d'analyser le domaine dans sa globalité avant d'analyser les composants du système. Nous pensons que cette stratégie est indispensable si nous souhaitons intégrer les nouveaux business requirements de l'entreprise qui ne sont capitalisés dans aucun composant. Toutefois, cette analyse peut à notre avis s'effectuer en même temps que l'analyse ascendante suivante.

Pour bottom-up : nous pensons que cette stratégie peut s'appliquer indépendamment de la politique de développement des SIT en entreprise (Encart 1.3). Dans l'optique d'intégrer ou non les SIT existants dans le développement d'un nouveau SIT, il est utile d'analyser les composants existants du domaine et il serait plus efficace si cette analyse s'effectue en parallèle que la stratégie descendante. La stratégie ascendante a longtemps été utilisée dans les méthodes de conception à base d'objets et par ou pour la réutilisation. Parmi les approches d'ingénierie de composants réutilisables, deux méthodes ont attiré notre attention : analyse des similarités et recherche d'analogies [CAS92;CAU01]. Toutes deux se rejoignent dans l'étude des modèles de spécification : schémas conceptuels, physiques...

Au chapitre suivant, à l'analyse des SIT existants, nous verrons comment nous allons adapter quelques principes de comparaison utilisés dans les deux méthodes.

Discutons maintenant de la nature du livrable : CdC du MOA/MOE.

VII.3 Business et Data Model : deux modèles en interactions permanentes

Nous convenons d'appeler le CdC MOA/MOE référentiel du domaine. L'intention du mot "référentiel" est liée au fait que le CdC résultant est suffisamment générique pour constituer un cadre de référence pour le développement des futurs SIT appartenant au domaine de l'ingénierie de production (particulièrement en ingénierie de besoin).

Quant au terme "domaine"¹⁰, à notre connaissance, il n'est pas rationalisé en ingénierie de SI. Néanmoins, nous pouvons l'associer au terme le plus proche "problème" (par opposition à "système" qui est plus proche de "solution"). Ainsi "domaine" peut concerner le champ d'application : l'ingénierie de production et les problèmes de GDT perçus dans ce champ. Le "référentiel du domaine" serait alors orienté expression de besoins et spécifications de la GDT, réutilisable dans le développement des SIT du même champ d'application. Le chapitre 5 de la

¹⁰ Nous pensons que la définition du terme anglo-saxon "Scope" est réductrice comparée à l'expression de "domaine". En effet, le terme "scope" est défini comme : "the boundaries of a project-what part of the business is to be studied, analyzed, designed, constructed, implemented, and ultimately improved ?" [WHI01].

thèse, présentant les différents usages de ce livrable dans les projets d'entreprise, justifie la dénomination de "référentiel du domaine" que nous retenons pour la suite de la thèse.

A présent, décrivons ce qui compose le référentiel du domaine. Nous distinguons deux modèles constituant le référentiel : business model et data model (Figure 3.9). Cette composition rejoint les préconisations en recherche, à savoir de mettre en avant aussi bien le SI informatisé que le SI organisationnel [BAR96;TAB93]. Le SI informatisé est celui qui se matérialise en BdD, application logicielle... Tandis que le SI organisationnel est la partie non informatisable qui traduit les spécifications du domaine. En l'absence d'expression française standard, nous adoptons ces termes anglo-saxons pour la suite de la thèse. Voici ce qui caractérise les deux composants pour notre contexte :

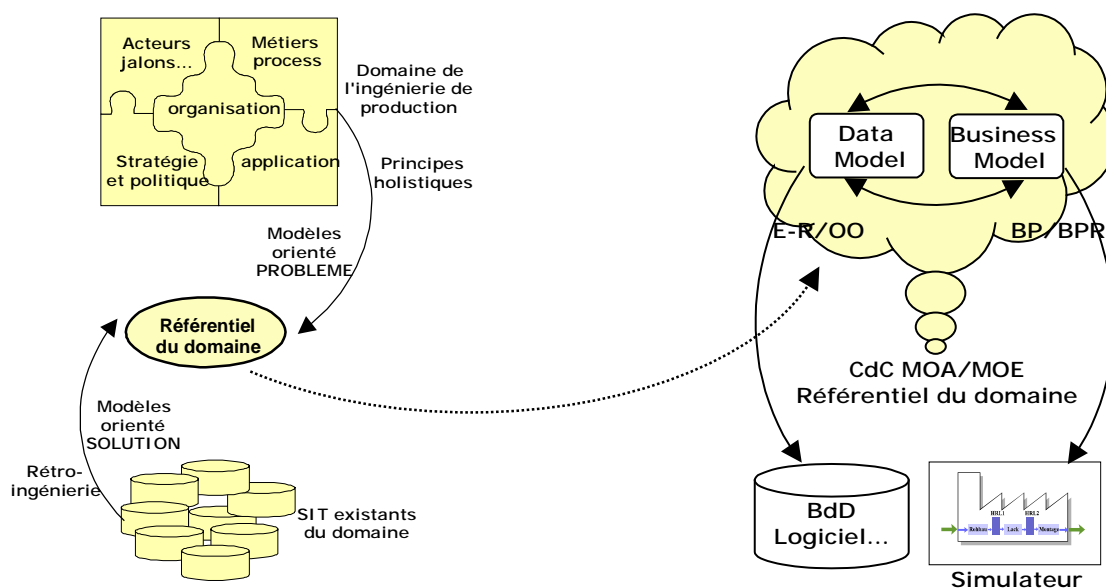


Figure 3.8 Notre démarche intégrée : ascendante et descendante (à gauche)

Figure 3.9 Business Model et Data model : Référentiel du domaine (à droite)

Business Model : nous l'avons introduit plus haut pour désigner le résultat de la modélisation avec les techniques BP. Plus concrètement, le périmètre de ce modèle peut couvrir les deux premières lignes du cadre de Zachman. Il doit traduire alors, à un haut niveau d'abstraction, le patrimoine informationnel de l'entreprise. Ainsi, le business model capitalise : des cadres terminologiques, des concepts liés à la taxonomie d'objets mais aussi des modèles contextuels et opérationnels pouvant traduire les stratégies et les politiques de l'entreprise. Pour cela, un tel travail n'est généralement pas sous-traité en entreprise en raison des aspects confidentiels qu'il peut contenir.

Data Model : comme son nom l'indique, il décrit à différents niveaux d'abstraction, les objets métiers à caractère commun et structurel. C'est ce modèle qui se matérialise, en fonction de la technologie d'implémentation, en SI informatisé (SGBD, SGDT, Data Warehouse...). Concrètement, il est réparti sur la colonne 1 (WHAT) du cadre de Zachman. Les niveaux d'abstraction du modèle traduisent les différents stades du projet où les hauts niveaux sont de la responsabilité de la MOA et les bas niveaux de la MOE et du sous-traitant. Ce dernier niveau marque les frontières de notre périmètre.

Ainsi, la construction de ces deux modèles peut s'effectuer en parallèle en s'ajustant conjointement. Il est possible de détecter par exemple pour chaque acteur ou chaque processus les SI qui le concernent et la manière de les développer ou de les améliorer pour modifier la situation actuelle. Inversement, pour chaque SI existant ou en projet, on peut déterminer tous les avantages stratégiques que l'on peut en tirer sans se limiter à une utilisation opérationnelle ou locale.

Nous discutons ci-après les sources potentielles d'acquisition des connaissances pour la construction de ce référentiel du domaine.

VII.4 Ingénierie de domaine : sources d'acquisition des connaissances

Nous préférons employer l'expression "ingénierie de domaine" que "étape préalable en ingénierie des SI" pour marquer une différence [CAU01]. En effet, la première expression met en avant la définition du problème pour lequel plusieurs SI différents peuvent être décrits. La seconde en revanche, est centrée sur la définition d'un SI vu comme une solution particulière répondant à un ensemble de besoins.

Par ailleurs, l'un des objectifs de l'ingénierie du domaine est la collecte, la structuration et l'abstraction des connaissances du domaine d'étude. En combinant la stratégie ascendante et descendante dans notre démarche (solution exposée en VII.2), nous intégrons aussi les techniques de rétroingénierie (ascendante) et d'analyse du domaine -problem analysis- (descendante). De ce fait, nous nous retrouvons face à une multitude de sources d'acquisition des connaissances que les anglo-saxons nomment "Fact-finding". Le terme désigne simplement le processus formel d'emploi des techniques variées pour la collecte d'information. Nous préconisons d'intégrer ces différentes sources et techniques pour les raisons suivantes (Figure 3.10) :

- *SIT existants* : cette source provient de la stratégie ascendante. Il est généralement difficile de comprendre ces SIT en l'absence des modèles de spécification amont qui les décrivent. Ceux-ci sont d'autant plus nécessaires lorsqu'il faut essentiellement concevoir un nouveau SI depuis sa version antérieure. Cependant, malgré l'intérêt de cette source d'information, elle reste insuffisante. Car il est des éléments que l'on ne voit pas matérialisés dans de tels supports et il faut procéder à la collecte d'information depuis les trois sources suivantes.
- *Interviews*¹¹ : comme nous l'avons mentionné plus haut (VII.1), il faut accorder de l'importance à toute la chaîne de clients impliqués dans le projet et pas uniquement à l'utilisateur final du produit. Leur accorder de l'importance signifie écouter leurs besoins et essayer de les satisfaire. Cependant, il faut tout de même se méfier de leurs opinions. Ces dernières peuvent être incomplètes, subjectives, voire locales (par opposition à une vision globale). De ce fait, il faut étudier en même temps les autres sources suivantes.
- *Modèles de processus* : Dans le chapitre précédent, nous avons déjà souligné l'interaction existante entre les données et les processus. D'ailleurs, avant l'apparition des concepts objet, les techniques d'analyse structurée s'employaient souvent en conjonction avec les techniques E-R illustrant ainsi cette interaction. En construisant les modèles de processus, que nous avons convenu d'appeler business model, des informations peuvent se créer pour traduire les nouveaux business requirements. L'observation des processus opérationnels de l'entreprise (avec tous les composants qui s'ensuivent) est indispensable

¹¹ "A fact-finding technique whereby the systems analysts collect information from individuals through face-to-face interaction" [WHI01].

mais pas aisée à cause de la complexité du système et du manque de rationalisation de ses processus comme c'était notre cas (domaine de l'ingénierie de production). Par conséquent, il ne faut pas négliger la source d'information qu'est la documentation.

- *Documentation produite* : ce sont les documents qui circulent dans le champ d'application de l'organisation. Nous avons vu au chapitre 1 (note de bas de page 9) que les documents peuvent refléter, en plus des données, la nature des processus qui les produisent. Dans cet exemple, nous avons montré l'interdépendance entre le processus d'industrialisation du véhicule et le processus de gestion financière par les documents qui s'y échangent. Tout comme les autres sources d'information, celle-ci n'est pas non plus complète car les documents seuls ne contiennent pas la totalité des informations qui circulent dans l'organisation. Cela dit, ces quatre sources différentes peuvent se compléter en les employant toutes.

C'est aussi à ce moment là (d'analyse du domaine) que l'on décide des frontières à fixer au périmètre d'étude, des limitations dans la fédération de données techniques ou encore du niveau de détail dans les modèles. Ces questions, à notre avis, sont contextuelles et les réponses peuvent varier d'un projet à l'autre comme va la montrer la solution suivante.

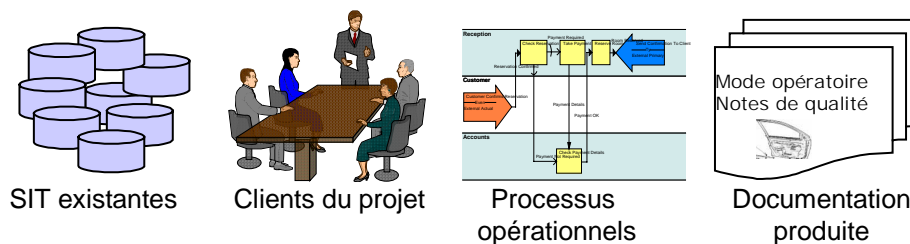


Figure 3.10 Ingénierie de domaine : sources d'acquisition des connaissances

VII.5 Approche contextuelle : triptyque clé (Objectif, Produit, Contrainte)

L'expression "approche contextuelle" est empruntée à la communauté française [ROL96; RIE99; CAU01]. La définition part du fait que les activités d'analyste et du concepteur reviennent souvent à un ensemble de prises de décisions sur des fragments du produit à développer. La décision dépend à la fois de la situation dans laquelle se trouve l'acteur et de l'intention qu'il a. De ce fait, nous pouvons résumer ce concept de contexte qui associe une décision à la situation dans laquelle elle peut être prise, au triptyque suivant (Objectif, Produit final, Contraintes environnementales)¹² :

- *Objectifs* que l'on veut atteindre issus d'une intention ou d'une motivation.
- *Produit final* à concevoir : il peut s'agir d'un fragment du modèle de produit mais aussi d'un modèle de processus.
- *Contraintes environnementales* : du fait que la décision doit tenir compte de la situation courante. Nous avons convenu de les traduire en "business requirement".

Pour illustrer notre rationalisation du concept contexte, donnons deux exemples contextuels. Le premier est lié à notre intention que les modèles de spécification soient compris par l'ensemble des acteurs du projet, notamment par la MOA qui généralement détient l'expertise

¹² Nous avons rencontré le triptyque (Purpose, Product, Constraints) dans l'ouvrage de H.W. Inmon et al. [INM97]. Il est employé comme version simplifiée du cadre de Zachman.

métier mais n'est pas familière avec les notations informatiques. Cela nous amène à chercher une manière de réduire la complexité qui se présente dans les modèles de spécification. La décision (solution) serait liée au triptyque :

- But : intention de partager nos modèles.
- Produit final : modèles résultants compréhensibles et cohérents.
- Contrainte environnementale : réduire la complexité des modèles.

Afin de prendre une telle décision, nous avons parcouru encore une fois la littérature et nous avons trouvé des éléments de réponse que nous résumons dans l'Encart 3.2.

Le second exemple est lié au concept de la réutilisation. La réutilisation peut s'appliquer à différents jalons du projet (analyse, conception, implémentation). Mais aussi, elle peut se présenter comme contrainte environnementale, une intention, ou encore comme une fin en soi. Quelques exemples du concept de réutilisation liés à notre problématique de thèse, sont résumés à l'Encart 3.3.

Encart 3.2 Comment réduire la complexité des diagrammes (modèles de spécification)

*Deux périls pour l'esprit : mésestimer les complexités de la nature, ou s'en laisser décourager au point qu'on se rabatte sur le surnaturel.
Jean Rostand (1894-1977)*

Brinkkemper et al [BRI93] distinguent deux types de complexité : la complexité du diagramme et la complexité dans les relations entre différents types de diagrammes.

Le premier cas de complexité peut être réduit en :

- Fournissant des vues : séparer les vues dans un même diagramme, permet de le comprendre facilement. Dans l'ouvrage, il donne l'exemple des diagrammes de type DFD dans lequel, on peut séparer la vue "flux de données" et la vue "contrôle d'information".
- Etant capable de montrer ou de cacher des informations : ce qui permet de lire facilement les diagrammes. Dans le diagramme de classe par exemple, nous pouvons montrer ou cacher les propriétés d'une classe : attributs, opérations (méthodes), responsabilités et contraintes.
- Supportant le principe de décomposition des diagrammes : tout en assurant la cohérence entre les niveaux.
- Colorant : par exemple tous les objets de même type auront la même couleur. Cela nous fait sans doute rappeler les travaux de P. Coad et al [COA99].

Le 2^{ème} cas de complexité appelé "modeling transparency" [BRI93], décrit les interdépendances entre types de diagrammes. On parle de cette interdépendance à l'usage d'un AGL. Le degré d'interdépendance est classé en niveaux :

- Niveau 0 : signifie que l'AGL en question ne possède pas de Repository et ne peut donc assurer aucun lien entre diagrammes.
- Niveau 1 : le Repository permet des dépendances entre diagrammes qui y sont stockés mais les connections ne peuvent s'effectuer qu'en fermant un diagramme et en chargeant un autre.
- Niveau 2 : en plus des caractéristiques du niveau 1, en chargeant plus qu'un diagramme à la fois, une modification sur un modèle peut se répercuter sur les autres diagrammes concernés d'une façon concourante.
- Niveau 3 : à ce niveau, il est possible pour l'utilisateur de définir des connections et des liens a posteriori en plus de ceux qui sont prédéfinis dans l'outil. Cette fonction est plutôt présente dans la gamme d'AGL appelés Meta Case.

Encart 3.3 Différents aspects de réutilisation en ingénierie des SIT

Most problems have already been solved in this world, we just keep forgetting the solutions. Unknown

Partant de l'idée de ne pas réinventer la roue à chaque nouveau développement, la réutilisation a pour but de construire un système à partir de composants existants préalablement décrits, mis en œuvre, testés et acceptés dans des projets antérieurs [RIE99].

Un modèle de composant (de réutilisation) vise d'une part, à capturer la connaissance du domaine, et d'autre part, à intégrer dans les composants des structures de conception et les décisions qui ont conduit à ces structures [CAU01].

Par ailleurs, la réutilisation peut s'appliquer à différents jalons du projet d'élaboration d'un SIT [GZA00] :

- En analyse : il s'agit de réutiliser les besoins et dysfonctionnements perçus lors des implantations passées des SIT en entreprise. L'expérience relatée au chapitre 1 concernant le projet SIGNE en est un exemple.
- En conception : reprendre par exemple les spécifications présentes dans les SIT existants (exemple : traitement de la maturité des objets métier) ou encore adapter les patrons d'analyse ou de conception de la littérature.
- En réalisation : cette partie est généralement sous-traitée. La réutilisation des composants logiciels est affaire du fournisseur du SGDT. Signalons tout de même que ce dernier type de réutilisation est le plus populaire.

Le but de cette introduction est de comprendre certains aspects de réutilisation que l'on peut assigner à notre problématique de thèse.

La réutilisation peut ainsi se présenter comme contrainte environnementale : réutilisation des BdD existantes dans le cas où nous voudrions que le nouveau SIT puisse intégrer les SIT existants.

La réutilisation peut être appliquée par une intention d'accélérer le développement en adaptant des composants existants sous toutes ses formes.

Enfin la réutilisation peut être une fin en soi. Ainsi, les usages du référentiel du domaine -comme nous le verrons au chapitre 5- expriment que le référentiel du domaine est un modèle de composant ou plus précisément un modèle d'abstraction du domaine.

VIII Synthèse

Nous avons exposé tout au long de ce chapitre les bases théoriques et les pistes de solutions pour notre problématique de thèse.

Tout d'abord, nous avons délimité le domaine de l'ingénierie de production, trop large pour être cerné dans sa globalité durant un travail de thèse. Nous avons tenu cependant à ce que le périmètre soit représentatif en vue de généraliser aux autres métiers du domaine et stratégique aussi bien pour l'entreprise que pour la recherche. Ainsi, nous retrouvons dans ce périmètre tous les problèmes cruciaux liés à la GDT (exposés au chapitre 1 de la thèse) qui se résument à : la prolifération des SIT, la non rationalisation du processus d'ingénierie, les données mal ou pas du tout identifiées, les BdD étanches entre filières, l'interdisciplinarité en conception du process etc.

Par la suite, nous avons souligné que la technologie d'implémentation du SIT en entreprise ne doit pas affecter notre problématique de thèse. En effet, le choix d'une architecture centrée (autour d'un SGDT par exemple) ou d'une architecture distribuée (via un bus CORBA par exemple), est indépendant des phases amont d'expression du besoin et de spécifications du CdC de la MOA/MOE. La conduite même de ces projets n'est pas non plus affectée par les

choix d'implémentation du SIT. Seuls les détails de conception technique prennent en compte les contraintes d'implémentation.

En tant que modèles de spécification nécessaires à la rédaction de notre CdC, nous avons retenu à la fois les techniques de modélisation de données, des processus (BPM), la notation UML ainsi que l'adaptation de quelques patrons abstraits et d'analyse. L'emploi de ces techniques est appréhendé aisément grâce au cadre de Zachman qui se distingue largement parmi les nombreuses architectures existantes. Aussi, nous avons choisi un AGL qui répond au mieux à nos exigences liées entre autres aux choix précédents en matière de techniques et d'architectures.

Dans la dernière partie de ce chapitre, nous avons présenté les aspects méthodologiques liés à notre problématique de thèse. Nous avons introduit en premier la méthodologie homologuée en entreprise. Nous avons décliné ses points forts que nous avons intégrés dans le CdC de la thèse. Cependant nous ne l'avons pas retenue pour les faiblesses qu'elle présente. Nous avons décidé en revanche d'intégrer dans notre démarche les deux stratégies courantes : ascendante et descendante pour leur complémentarité et leur nécessité dans ces projets. Nous avons également préconisé de construire conjointement le "business model" et le "data model", composantes complémentaires du CdC du MOA/MOE. Nous avons aussi convenu d'appeler ce livrable référentiel du domaine.

En combinant les deux stratégies -ascendante et descendante-, nous intégrons aussi les techniques qui y sont employées et par conséquent, les différentes sources d'acquisition de l'information en analyse du domaine.

Nous expliquons les raisons pour lesquelles nous qualifions notre approche de contextuelle. Aussi, nous montrons que les décisions que nous sommes amenés à prendre tout au long du travail de rédaction du référentiel sont fortement conditionnées par le triptyque (Intention, Produit final, Contrainte environnementale).

En choisissant ces pistes de solutions aussi bien pratiques que théoriques, nous construisons au chapitre 4, le référentiel du domaine et nous présentons au chapitre 5 ses différents usages dans les projets d'entreprise.

CHAPITRE QUATRIEME

Mise en œuvre du référentiel du domaine

C'est notre tendance à rechercher la régularité des occurrences et à prescrire de lois à la nature qui est à l'origine du phénomène psychologique de la pensée dogmatique ou plus généralement du comportement dogmatique...Ce dogmatisme est en partie nécessaire. C'est lui...qui rend possible la formulation progressive d'une bonne théorie, grâce à différentes approximations : dès lors que nous renonçons trop aisément, nous nous interdisons peut-être de voir que nous avons failli réussir.

*Karl R. POPPER (1902-1994). La science : conjecture et réfutations IN Conjecture et Réfutations –
La croissance du savoir scientifique, chapitre 1, Payot 1985*

Le présent chapitre est constitué de trois sections :

Section 1 – Construction du référentiel : ingrédients et mode d'emploi

Elle présente les ingrédients utiles à l'étape cruciale d'ingénierie de domaine. Le mode d'emploi est illustré à l'aide d'un scénario d'élaboration du référentiel répondant à un contexte. Celui-ci est défini à l'aide du triptyque : intention, produit final et contraintes environnementales.

Section 2 – Le business model

Elle est consacrée à la rédaction du business model. Celui-ci est basé sur trois types de modèles, construits chronologiquement et d'une façon contiguë : contextuels, opérationnels et modèles de transformation logique. Ce dernier type sert de passerelle pour accéder au data model.

Section 3 – Le data model

L'élaboration du data model se présente en deux étapes : le diagnostic de données dans lequel les techniques de rétroingénierie et d'analyse de domaine se confrontent pour produire des objets pertinents. Ces derniers sont structurés et affinés dans la seconde étape. Celle-ci est basée sur les techniques objets et sur l'adaptation des patrons abstraits et d'analyse de la littérature.

SECTION 1 – Construction du référentiel : ingrédients et mode d'emploi

Au chapitre 3, nous avons identifié les bases théoriques et pratiques pour réaliser le CdC de la thèse. Dans cette section, nous expliquons tout d'abord les ingrédients utiles pour l'étape cruciale de collecte des informations du domaine (I). Nous exposons par la suite un scénario d'élaboration du référentiel du domaine (II).

I Cadre de Zachman pour gérer les connaissances de l'entreprise

Nous avons auparavant expliqué les usages potentiels du cadre de Zachman (chapitre 3, V.) Nous illustrons ici sa contribution à organiser les connaissances du domaine, étape préliminaire en ingénierie de domaine (I.1). Sur la base de ces résultats, nous définissons le cadre du référentiel du domaine (I.2).

I.1 Cadres du système : interdépendances et récursivité logique

Dans le contexte de la problématique de thèse, nous distinguons trois cadres de systèmes différents mais pouvant être dépendants : cadre de produit, de SIT et d'entreprise. Pour décrire ces cadres simplement, nous réduisons les six dimensions du cadre de Zachman en trois variables dont la signification varie d'un cadre à l'autre.

Cadre de produit [structurelle, fonctionnelle, géométrique]

Ce cadre permet d'organiser l'ensemble des représentations du produit (véhicule) tout au long de son cycle de vie. Chaque ligne du cadre précise le jalon d'élaboration du produit et l'acteur responsable. Les colonnes en revanche décrivent les vues du produit les plus populaires. Ainsi, nous classons respectivement dans les colonnes 1, 2 et 3 les nomenclatures organiques (de pièces), le découpage fonctionnel du produit et les dessins géométriques de pièces (Tableau 4.1). Dans le contexte de notre problématique, la nomenclature telle qu'elle est

définie par le BE (décrite dans SIGNE¹) doit être intégrée dans la définition de notre référentiel. La nomenclature "as planned" doit traduire l'ordonnancement des assemblés de pièces tels qu'ils seront réalisés sur les sites de production. Cela veut dire, que la première nomenclature est élaborée selon les contraintes orientées produit de l'architecte en BE. Tandis que la seconde nomenclature prend en compte conjointement les contraintes produit/process.

Par ailleurs, nous devons penser que le produit peut être aussi le système de production. De ce fait, les acteurs et les vues du cadre de système de production varient naturellement. Quelques cellules du cadre devraient donc contenir des nomenclatures de Process² que nous devons créer tout en assurant une cohérence avec les données du produit. Enfin, ce type d'organisation nous aide à percevoir les concepts valables en entreprise qu'il faut intégrer et ce qu'il reste à faire.

	Structurelle	Fonctionnelle	Géométrique
Planificateur stratégique	Définitions / package de concepts		
Maîtrise d'ouvrage	Work structure	Spécifications fonctionnelles	Croquis de dessins
Designer (Architecte)	Engineering BOM (As designed)	Découpage fonctionnel (théorique)	CAO 2D, features maquette numérique 3D
Préparateur méthode	Mfg. Eng. BOM (As planned)	Découpage (pratique)	IPAO 2D (simulation) maquette numérique 3D
Fabricant sous-traitant	Part list (As built)	Fonctions (mises en service)	Dessins d'assemblés fabriqués...

Tableau 4.1 Cadre du produit : différentes représentations du véhicule industriel

Cadre de SIT [donnée, traitement, technologie]

Ce cadre regroupe les différents modèles d'un SIT. Les dimensions les plus populaires sont celles qui ont trait :

- Aux données (avec les modèles issus de l'ingénierie de l'information).
- Aux fonctions réalisées par ces données (à l'aide des diagrammes orientés processus).
- Et à la technologie qui spécifie où les modèles du SI sont produits ou maintenus. Cela peut être vraisemblablement l'architecture d'implémentation du SI : SGBD, SGDT, Réseau, Client/Serveur ...

Les contraintes d'implémentation d'un SIT ne concernent pas notre problématique. Par conséquent, la dernière dimension ne nous intéresse pas. Les deux autres dimensions en revanche doivent être prises en compte à la construction du référentiel.

Les vues du SIT, quant à elles, sont de trois niveaux d'abstraction : conceptuel, logique et physique.

Cadre d'entreprise [objets métier, processus opérationnels, espaces métier]

Discuté largement au chapitre 2, III.2., le cadre d'entreprise permet d'organiser les modèles d'investigation du domaine de l'ingénierie de production tout en le positionnant par rapport aux autres métiers d'entreprise. Nous rangeons les (fragments de) modèles d'entreprise en :

¹ SIGNE est le SI Produit développé en entreprise. Il porte le label de GDT d'entreprise (par opposition à GDT locale) par son utilisation à travers la quasi totalité des directions. Pour plus de détail, cf. Chapitre 1, II.4.

² Le terme Process couvre ici tous les concepts liés au manufacturing process : ressources, opérations de production, caractéristiques des ateliers de production...

- *Objets métier (business objects)* : entités génériques mais significatives ayant un rôle en entreprise. Cela peut être : acteur, jalon du projet, application logicielle, ressource etc.
- *Processus opérationnels (business processes)* : la dimension la plus populaire dont les principaux concepts sont expliqués à l'encart 2.4. Retenons seulement que les processus opérationnels en entreprise sont très souvent liés les uns aux autres et qu'ils transcendent les frontières des directions organisationnelles. (chapitre 1, II.3).
- *Espaces métier (business location)* : cette dimension spécifie l'endroit où se déroulent les processus opérationnels. Il peut s'agir des bureaux d'ingénierie, des surfaces de sites de production, des postes de travail...

De même, les niveaux de granularité sont représentés par les lignes du cadre. Ils correspondent de haut en bas aux : modèles contextuels, modèles d'entreprise (décrivant le problème) et aux modèles de système (décrivant les solutions, résultat d'une ré-ingénierie de des processus par exemple).

Il est courant que les trois cadres ainsi décrits soient fortement liés et cela peut compliquer l'étape d'ingénierie de domaine. De ce fait, nous n'imposons pas cette distinction tout au long de cette phase ; elle est juste utile en phase amont d'ingénierie de domaine.

Intérêt à distinguer ces différents cadres

Les trois principales activités d'ingénierie composant notre problématique de thèse : ingénierie de produit mécanique, ingénierie des SI et ingénierie d'entreprise (entreprise engineering) peuvent employer des techniques et des méthodes différentes (par exemple respectivement : l'analyse fonctionnelle, l'analyse de problème et la modélisation des processus). Comme nous l'avons montré, à travers la comparaison entre l'ingénierie de véhicules et l'ingénierie de SI (chapitre 3, I.1), ces métiers d'ingénierie peuvent être comparés par les techniques et les méthodes qui y sont employées.

Par ailleurs, sans utiliser ces différents cadres pour caractériser la description propre de ces systèmes complexes, il serait difficile d'avancer dans l'investigation. Le sens global du système n'est pas toutefois perdu. Car à tout moment, nous pouvons mettre en relation ces différents cadres. Prenons deux exemples pour illustrer ces interactions.

Liens potentiels entre cadres

(1) Liens : cadre de produit – cadre d'entreprise

Nous simplifions la description du processus d'élaboration d'un produit (véhicule) en entreprise (Figure 4.1). Ce processus peut être classé dans la cellule (2,2)³ du cadre d'entreprise. Ainsi, les différentes cellules du cadre de produit peuvent être les livrables de ce processus.

³ la cellule (2,2) traduit la *dimension* "processus opérationnels" et la *perspective (vue)* de la maîtrise d'ouvrage.

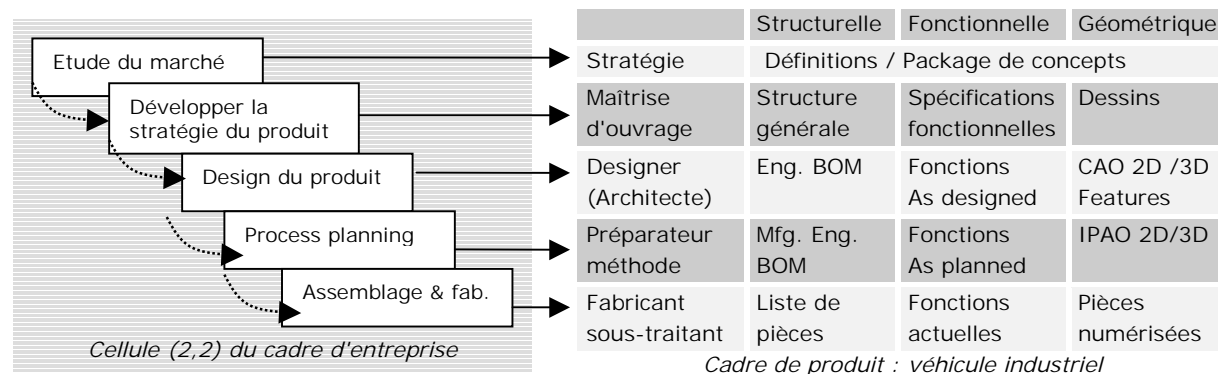


Figure 4.1 Interactions : cadre de produit – cadre d'entreprise

(2) Liens : cadre de SI – cadre d'entreprise

De même, nous simplifions le processus de développement d'un SI sans préciser sa nature (en spirale, en V...). Il doit être classé en cellule (2,2)⁴ du cadre du SI. Cette cellule peut bien avoir comme livrables les résultats du cadre d'entreprise (Figure 4.2).

Ces exemples d'interactions mettent en avant un concept fondamental dans la modélisation : la récursivité logique. Examinons comment elle se traduit dans notre contexte.

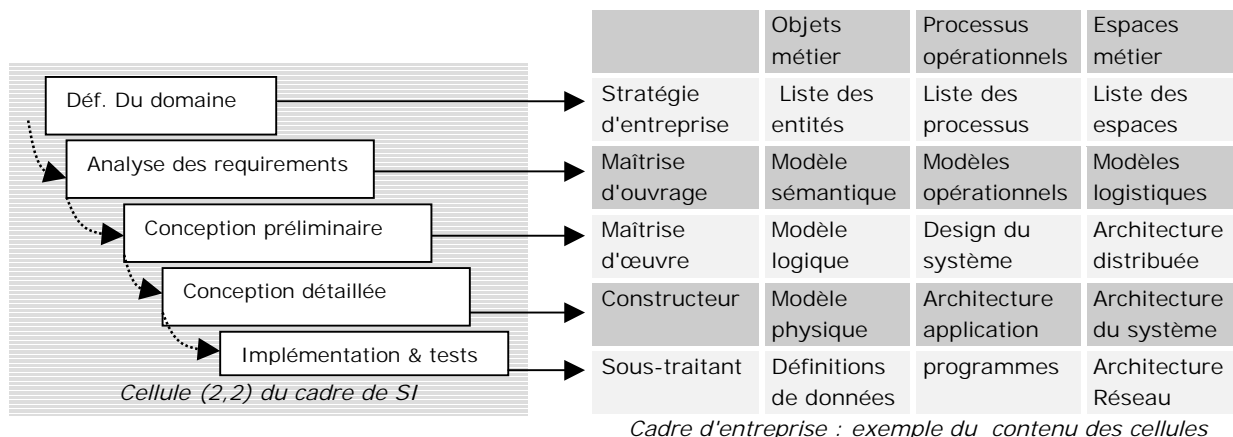


Figure 4.2 Liens potentiels : cadre de SI – cadre d'entreprise

Récursivité logique : versionnement et décomposition

La récursivité peut se traduire par le versionnement ou la décomposition (hiérarchisation). A l'aide du versionnement, il est possible de concevoir des variantes (de fragments) de modèles à partir des versions antérieures. Cette forme de réutilisation est facilitée par l'emploi d'AGL possédant une encyclopédie (chapitre 3, VI). Nous pensons par ailleurs que ce n'est pas une bonne stratégie de devoir suspendre la satisfaction des besoins à court terme tant que les différents modèles du cadre (différentes cellules de Zachman) ne sont pas complètement définis.

L'autre forme de récursivité, la hiérarchisation est le résultat du principe populaire de la décomposition fonctionnelle. On parle de partitionner un système en parties pour mieux le

⁴ la cellule (2,2) peut traduire la dimension fonctionnelle et la vue conceptuelle du SI.

maîtriser. Beaucoup d'objets peuvent subir cette décomposition : processus, temps, donnée... Nous verrons aux sections suivantes les décompositions effectuées sur ces objets.

En modélisation, la récursivité permet d'obtenir des niveaux de granularité des représentations. Cela nous amène à évoquer un autre concept aussi fondamental qui est la *métamodélisation*. Sans entrer dans le débat de la métamodélisation, signalons seulement que sa puissance à gérer les changements a longtemps été discuté, surtout dans le domaine du génie logiciel [NDI00; INM97].

Constatons simplement que les cadres de produit et de SI peuvent paraître comme des instances du cadre global d'entreprise. Celui-ci peut vraisemblablement être une instance d'un cadre plus générique que l'on peut appeler cadre d'ingénierie d'entreprise. Enfin, ce dernier doit se compléter à l'aide d'un repository contenant les connaissances les plus stables de l'entreprise. Ce processus de métamodélisation peut être une solution pour l'extensibilité et la flexibilité des modèles de connaissances. Ainsi, modifier n'importe quel modèle d'un cadre doit être aussi facile que d'ajouter une instance dans une base de donnée.

1.2 Périmètre du référentiel : dimensions et perspectives

Dans notre contexte, l'objet du référentiel, rappelons-le, est de fournir des modèles d'expression du besoin de la maîtrise d'ouvrage (MOA) et de spécifications fonctionnelles fréquemment posées par la maîtrise d'œuvre (MOE). Le référentiel doit s'appuyer sur une terminologie standard et commune des différents concepts gérés par les SIT dans le domaine de l'ingénierie de production. A ce titre, il est un cadre orienté problèmes de GDT, réutilisable dans le développement des futurs SIT du même champ d'application.

Rappelons également l'objet des deux composants de ce référentiel : business et data model. Le business model décrit le domaine de l'ingénierie de production en visualisant ses fonctions dans le monde : ce qu'il fait, comment et quand. Conçu pour répondre aux besoins d'acteurs (MOA/MOE...), le business model devrait leur fournir l'information juste nécessaire. La technique de modélisation des processus est la plus adéquate pour élaborer ce module (chapitre 3, IV.2).

Le data model, plus populaire que le précédent, est constitué d'objets métiers structurés à différents niveaux d'abstraction. C'est ce module qui se matérialise en bases de donnée. Nous avons vu également au chapitre 3 qu'il est recommandé d'employer les techniques de modélisation des données (VI.1), d'objet (VI.3) et d'adapter certains patrons décrivant les problèmes récurrents du domaine et préconisant les solutions à ces problèmes (Encart 3.1).

Le cadre du référentiel de domaine est alors une combinaison des trois cadres précédemment définis, adapté à la manière suivante (Tableau 4.1).

Nous le définissons à l'aide des six dimensions du cadre de Zachman et des trois perspectives suivantes : la vue stratégique, de la MOA et de la MOE. Dès que cette dernière entame la phase de sous-traitance de la conception détaillée du SIT, nous marquons les frontières du référentiel de domaine.

Le business model occupe les deux lignes du cadre avec comme perspectives : la stratégie et la vue de la MOA. Quant au data model, comme son nom l'indique, il couvre la première colonne du cadre.

Chaque perspective qui traduit un niveau d'abstraction est aussi représentée par un type d'acteurs potentiels : la vue stratégique devrait concerner tous les membres du comité du projet, du moment que chacun d'eux est sensé prendre en compte les business requirements de l'entreprise. Communément, on désigne pour chaque projet un acteur (service) qui se voit

assigner ce rôle. Il peut tout aussi appartenir à la direction informatique qu'à la direction de l'ingénierie véhicule.

Nous avons longtemps discuté des missions respectives de la MOA et de la MOE (Encart 1.4). Rappelons seulement que la MOA est sensée exprimer les besoins des métiers d'ingénierie de véhicule tandis que la MOE est en charge de traduire ces besoins en spécifications fonctionnelles pour lesquelles le SIT projeté doit répondre.

Notons toutefois que les cellules communes entre le data et le business model traduisent cette interaction permanente entre les deux modules (chapitre 3, VII.3).

Avec cette présentation, nous aurons répondu aux questions fréquentes posées par les acteurs d'un tel projet : quel modèle de spécification choisir ? et quand l'utiliser ? quel type de livrable et pour quel jalon ?

Néanmoins, certaines d'entre elles nécessitent des réponses contextuelles comme : qui est en charge de maintenir les connaissances du référentiel de domaine ?

Les réponses à ce type de questions sont apportées dans les sections suivantes.

	QUOI ?	COMMENT ?	OU ?	QUI ?	QUAND ?	POURQUOI ?
Stratégie	DATA MODEL OO/E-R	BUSINESS MODEL				
MOA		techniques de modélisation des processus				
MOE et ses						
Partenaires						

Tableau 4.2 Référentiel de domaine : vues et niveaux d'abstraction

II Méthodologie de construction du référentiel

Les différents aspects méthodologiques liés à la construction du référentiel et à la conduite des projets de développement des SIT sont déjà exposés (chapitre 3, VII). Nous illustrons ici un scénario de conception du référentiel, positionné par rapport au projet d'élaboration d'un nouveau SIT en entreprise.

Mais avant, décrivons quelques termes homonymes abusivement utilisés dans les deux composants du référentiel. Le but de l'explicitation de ces concepts est de pouvoir les employer plus tard librement sans toutefois créer des ambiguïtés pour le lecteur.

II.1 Business et data model : quelques concepts homonymes

Si le périmètre de chacun des deux modèles est clairement défini, il n'en est pas de même pour les concepts utilisés. Nous trouvons ainsi une multitude d'homonymes que nous récapitulons dans le tableau 4.3. Prenons le terme "cycle de vie" pour illustrer ce constat. En business model, il peut désigner les événements majeurs à l'origine du déclenchement d'actions ; ils peuvent être externes, internes ou temporels (jalons du projet véhicule, temps requis pour réaliser une opération etc.). Pour le data model, le cycle de vie peut être assigné aux objets métier. Il permet de gérer leur maturité, en mettant en œuvre de nouveaux concepts tel que le versionnement. Pour une meilleure compréhension des concepts employés tout au long de ce chapitre, nous suggérons au lecteur de revoir quelques définitions des chapitres précédents⁵.

⁵ Chapitre 1 : Missions de la MOA et de la MOE en entreprise (Encart 1.4) - Chapitre 2 : Processus (Encart 2.4) – Concepts Objet (Encart 2.5) - Chapitre 3 : Pattern (Encart 3.1, Réutilisation (Encart 3.3) etc.

Concepts	Business model	Data model
Acteur	Définit un ou plusieurs rôles qu'une entité (chose/personne) peut jouer dans un environnement métier	Définit un ou plusieurs rôles qu'une entité d'un environnement peut jouer en relation avec le SI. Un utilisateur est un exemple.
Cas d'utilisation (Use case)	Séquence de transactions dans un système dont la tâche est de donner un résultat de valeur mesurable à un acteur individuel du système.	Façon spécifique d'utiliser le SI. Il est composé d'une séquence d'actions déclenchée par un acteur externe et qui produit un résultat identifiable.
Objet métier (Business Object)	Représente des occurrences telles que : produit, livrable, document et autres entités manipulées dans le business.	Objet du domaine, décliné depuis l'analyse de ses processus par exemple. Cet objet est utilisé en étape d'analyse objet.
Objet de contrôle (Control Object)	Représente des tâches dans le business. Ces tâches devraient être accomplies par une ressource, qui est typiquement un spécialiste ou une routine, n'ayant pas affaire directement avec le client.	Un objet qui encapsule les spécifications fonctionnelles d'un ou de quelques cas d'utilisation. L'objet apparaît pendant les spécifications techniques et d'implémentation.
Objet d'interface (Interface Object)	Représente un ensemble d'opérations dans le business. Chacune d'elles devrait être exécutée par la même ressource. Cette tâche implique la communication avec l'environnement du business.	Un objet qui encapsule la fonctionnalité de gestion des interfaces du SI/application. Aussi, les objets d'interface apparaissent plus tard dans la conception technique.
Processus métier (Business Process)	Activités d'un métier dont l'objectif est de fournir un résultat observable et mesurable pour un utilisateur individuel du métier.	Procédures et traitements liés à l'évolution des objets ou aux règles métier pouvant être automatisés à l'aide de workflow ou dans le système informatique.
Requirement	Description des besoins pour un SI. Un requirement peut décrire des fonctions, des features (attributs) ou des contraintes.	Représentation des données des utilisateurs en terme d'entités, d'attributs et de relations. Un requirement est exprimé indépendamment de la technologie d'implémentation de la donnée.
Rétroingénierie (Reverse Engineering)	Elaboration d'un modèle abstrait d'un système existant et ses processus, avec l'objectif de gain de compréhension et de communication du business.	Possibilité de lecture d'un programme existant à l'aide d'un AGL, en le transformant en un modèle représentatif pouvant être édité et raffiné par un analyste.

Tableau 4.3 Business model et data model - Concepts homonymes (Adapté de [JAC95; KET98; MOR00; WHI01])

II.2 Principaux jalons d'élaboration du référentiel : un scénario de conception

Nous avons qualifié notre approche de contextuelle et nous avons convenu que le triptyque (intention, produit final, contrainte) pouvait bien la caractériser (chapitre 3, VII.5). Cela signifie que nous préconisons ici un scénario d'élaboration du référentiel répondant à un contexte donné. Par ailleurs, le processus d'élaboration du référentiel peut être associé à la conduite même de développement d'un nouveau SIT en entreprise. Examinons-les conjointement au travers du contexte défini à l'aide des paramètres ci-après (Figure 4.3).

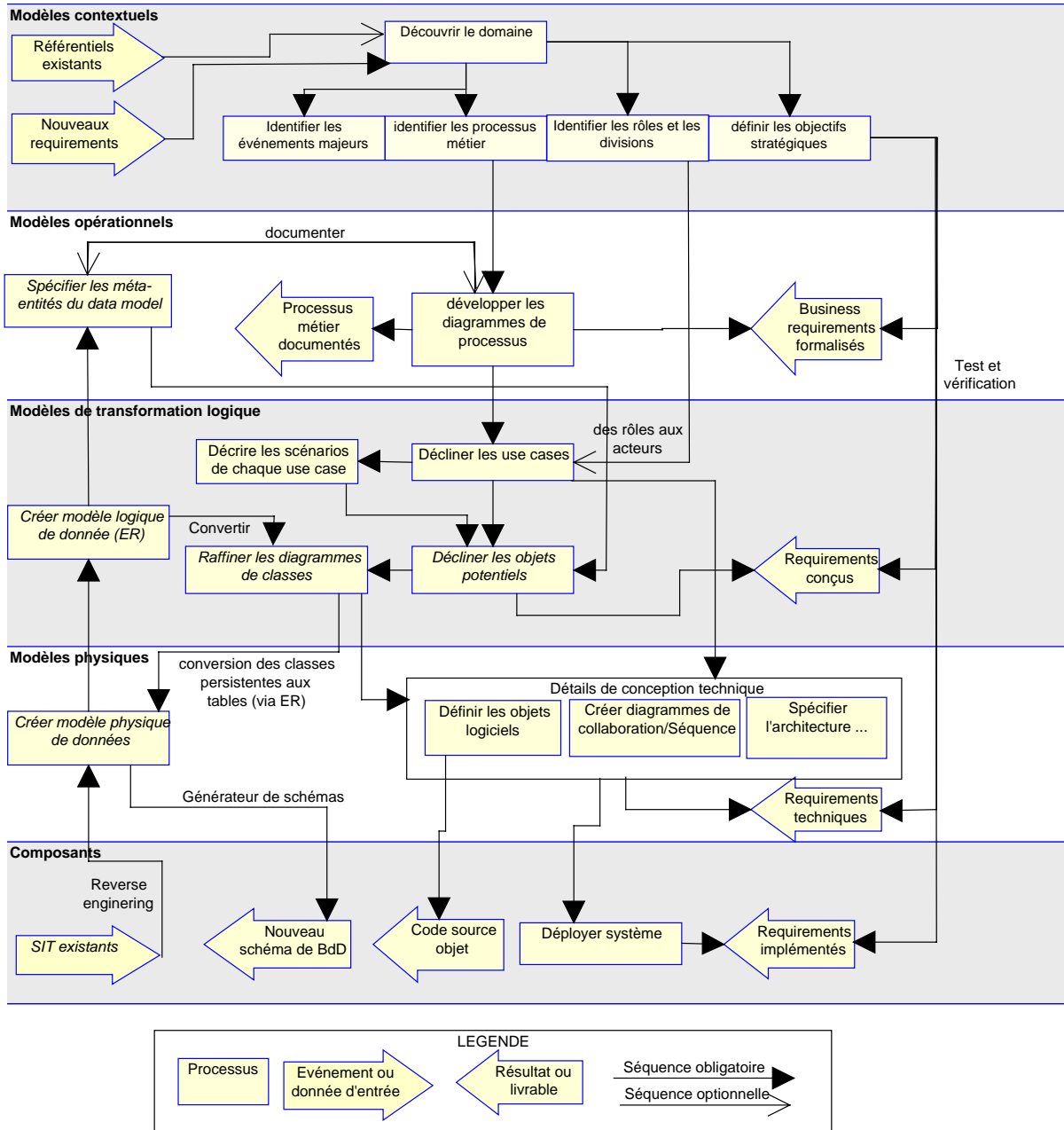


Figure 4.3 Scénario d'élaboration du référentiel et d'un nouveau SIT en entreprise

▪ **Intention et motivation**

Référentiel de domaine : sa raison d'être peut se justifier par le besoin de résoudre les goulots d'étranglement à la mise en œuvre des SIT du domaine de l'ingénierie de production. L'intention du business model n'est donc pas la ré-ingénierie des processus opérationnels ou leur amélioration (en évaluant leurs performances à l'aide de la simulation). De ce fait, nous nous focalisons seulement sur leur identification et leur documentation. Quant au data model, nous voulons qu'il décrive les objets métier (données du domaine⁶) à caractère commun et structurel. Le caractère commun exprime le fait que la donnée puisse être utile à l'exécution des activités du domaine et au fonctionnement des SIT existants. Le caractère structurel en

⁶ Tel que nous l'avons délimité avec les trois variables : métiers process, activités transversales et jalonnement du projet véhicule. Pour en savoir plus, cf. chapitre 3, II.

revanche exprime que la (valeur de la) donnée peut être créée ou modifiée à l'occasion d'un changement de la stratégie, de l'organisation, d'un processus ou encore d'un événement extérieur ayant un impact sur ses processus.

Nouveau SIT : le développement d'un nouveau SIT en entreprise a pour but de :

- Remplacer le ou les SIT existants devenant obsolètes.
- Concilier les tâches des acteurs, traditionnellement morcelées par l'emploi d'applications logicielles étanches et par les frontières entre directions organisationnelles.
- Plus globalement, fédérer les SIT existants autour d'une architecture centrée SGDT.

- **Produit final (nature des livrables)**

Référentiel de domaine : les trois hauts niveaux d'abstraction (contextuel, opérationnel et logique) forment le référentiel. Le livrable est ainsi composé de définitions et de modèles de spécifications, compris par l'ensemble des acteurs du projet. Une fois construit, le référentiel devient un modèle contextuel et pouvant servir en phase d'ingénierie de besoin pour le développement de nouveaux SIT. Il est présenté à l'aide de l'événement "référentiel existant" au niveau contextuel. Nous mettons aussi en "italique" les tâches concernées par le data model afin de les distinguer de celles du business model (Figure 4.3).

Nouveau SIT : le développement d'un nouveau SIT passe par tous ces jalons (contextuel, opérationnel, logique, physique et composants). La nature du dernier livrable dépend des choix technologiques et d'implémentation. Ainsi, selon que le choix est porté sur les techniques orientées objet (OO), relationnelles (E-R)... , le dernier livrable du SIT serait une application logicielle objet, un schéma de code ou encore une base de donnée relationnelle...

- **Contraintes et hypothèses**

Les choix précédents (objectifs, nature des livrables) génèrent de nouvelles contraintes et des hypothèses de travail. Nous avons par ailleurs illustré quelques contraintes au chapitre 3 (Encart 3.2, Encart 3.3). Il s'agit de l'aspect de la réutilisation des SIT existants et de la réduction de la complexité du référentiel. Ces deux contraintes répondent en partie aux objectifs respectifs : fédérer les SIT existants et produire un référentiel accessible et compris par tous les acteurs. Le choix de la réutilisation est illustré sur la Figure 4.3, par la présence des flèches ascendantes traduisant l'acte du reverse engineering et de la démarche ascendante. La réduction de la complexité se traduit en partie par la présence du référentiel à un haut niveau d'abstraction et par l'explicitation des vues séparément (stratégie, processus, organisation...).

Notons aussi un fait important en ingénierie du besoin (plus globalement en requirement engineering) : traditionnellement, cette discipline n'est associée qu'aux phases préliminaires d'un projet pour s'étendre aujourd'hui aux différentes phases du projet [SID96]. En effet, un requirement peut évoluer entre son identification (en phases amont) et son implémentation en aval. Cela signifie qu'il est naturel de le faire évoluer en cours de travail en le réexaminant à chaque étape (validation, design, implémentation). Cette préconisation est illustrée à travers les liens entre le processus "définir les objectifs stratégiques" au niveau contextuel et les livrables de bas niveaux.

Signalons enfin que nous pouvons commencer aussi bien par l'étape de rétroingénierie pour le data model que par l'étape d'investigation du domaine pour le business model, comme nous allons le constater aux sections suivantes.

SECTION 2 – Le Business Model

La construction du business model est descendante. A ce titre, les trois sources d'acquisition des connaissances : interviews auprès de la chaîne d'acteurs, observation des processus métier et documentation produite sont toutes appropriées (chapitre 3, VII.4). Par ailleurs, la colonne 1 du cadre du référentiel, commune au data model, met en avant une autre source de collecte d'information : la rétroingénierie que nous traitons à l'élaboration du data model, objet de la section 3.

Le business model est composé de trois types de modèles que nous construisons chronologiquement et d'une façon contiguë. Nous convenons de les appeler respectivement : modèles contextuels, modèles opérationnels et modèles de transformation logique. Le premier type de modèles est le résultat de l'investigation du domaine de l'ingénierie de production. L'investigation est effectuée selon quatre dimensions interdépendantes : stratégique, organisationnelle, temporelle et fonctionnelle. Les modèles opérationnels sont construits sur la base des résultats de l'investigation et à l'aide des diagrammes de spécification standards. Les modèles de transformation, quant à eux, jouent le rôle de passerelle pour accéder au data model.

Rappelons encore une fois que le business model traduit les caractères confidentiels de l'entreprise. De ce fait, nous limitons le nombre des diagrammes de spécification, en veillant toutefois à la bonne compréhension et à la qualité des modèles.

III Découverte du domaine : modèles contextuels

Au chapitre 1 de la thèse, l'ingénierie de production et ses problèmes de GDT sont dégagés comme domaine d'étude. Nous formalisons ici à un haut niveau d'abstraction les résultats de cette analyse : les objectifs stratégiques de la firme, ses nouveaux besoins, ses contraintes ainsi que sa future vision en terme de SIT.

La définition des requirements pour le business model (Tableau 4.3), génère deux types de requirements : fonctionnels et non fonctionnels.

Un requirement fonctionnel est une fonction ou une caractéristique (feature) qui doit être assurée par le SI projeté afin de satisfaire les utilisateurs du domaine. Il est facilement identifiable par l'emploi d'un verbe d'action dans son expression. Un requirement non fonctionnel en revanche, doit décrire les autres propriétés et les contraintes du système qui peuvent limiter les frontières de celui-ci. Des requirements liés au service, à la performance ou à la rentabilité du système sont des exemples.

La formalisation des requirements ne préconise pas l'emploi de diagrammes de spécification standards⁷. En revanche, les techniques de base "root definition", "rich picture", les tables matricielles... présentées au chapitre 2 sont particulièrement utiles.

La validation de ces requirements est un problème crucial. Car il s'agit de juger de l'exactitude, de la cohérence et de la complétude d'un requirement. Cette étape est affaire de tous (mais plus particulièrement de la MOA) et sujette à des évolutions tout au long du projet.

⁷ Il existe néanmoins des efforts de normalisation des formalisations des requirements. Par exemple, le standard MIL-STD-498 a pour objectif de définir l'ensemble des activités et la documentation adéquate dans le domaine de développement des systèmes d'armement et des SI automatisés pour le gouvernement des USA.

En s'inspirant des dimensions du cadre de Zachman et de celles de Catalyst⁸, nous effectuons notre propre investigation du domaine sur la base des quatre faits ci-après. Dès qu'on perçoit des concepts relativement standards ou récurrents, nous les capitalisons dans l'encyclopédie de l'AGL en vue de les réutiliser aux jalons suivants.

III.1 Vision stratégique

Comme tout projet, la mise en œuvre d'un nouveau SIT en entreprise est motivée par une intention de haut niveau, de nouveaux besoins et de nouvelles contraintes environnementales à prendre en compte.

Au chapitre 1, nous avons pu discerner les causes des problèmes et leurs effets (les symptômes des maladies). Ainsi, nous associons les itérations et les feed-back perçus dans le travail quotidien des acteurs au morcellement du processus d'industrialisation du véhicule et à la conception verticale des outils d'IPAO. Ou encore, la ressaisie des informations qui se manifeste tout au long du projet véhicule est causée par la non interopérabilité entre systèmes etc.

Par ailleurs, l'intention d'intégrer la nomenclature produit (présentée dans SIGNE), vue comme une contrainte, conditionnera la structure du référentiel du domaine que nous voulons construire. La Figure 4.4 illustre une manière de renseigner ces paramètres dont certains sont personnalisés afin de les adapter à la culture d'entreprise.

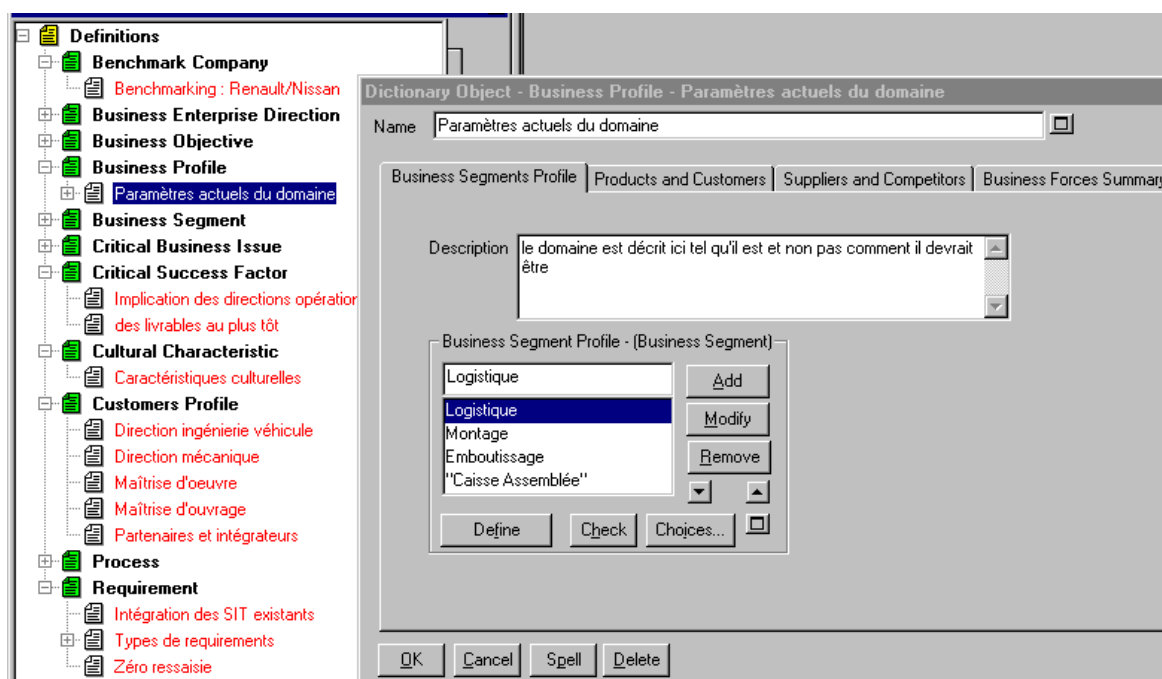


Figure 4.4 Objectifs stratégiques et nouveaux requirements de l'entreprise

⁸ Catalyst, rappelons-le, est une méthode développée récemment par CSC [CSC00] pour aider l'entreprise à la ré-ingénierie de ses processus. Le changement selon la méthode, peut s'opérer sur l'une des six perspectives suivantes baptisés "domains of change" : Application logicielle, Technologie, processus opérationnel, Lieu, Organisation et Donnée

III.2 Dimension temporelle

Généralement, le temps est associé aux concepts : événement et transaction. Il désigne par exemple les événements significatifs en entreprise comme les jalons du projet véhicule (Tableau 4.4). Ils sont à la base du déclenchement d'actions sur le processus et de l'accomplissement des livrables du projet. Nous verrons qu'avec le concept objet (UML), la dimension temporelle serait représentée naturellement.

Jalonnement documentaire	Création famille	100% fonctions	100% F/Variantes	Spécif. Produit	Spécifications process	Doc. d'exploitation
Jalons de la direction générale	Séminaire	RVE	RVP	RVA	RVD	RVS
Phases de développement	Exploratoire	Préparatoire	Etude enveloppe	Etude détaillée	Mise au point & réalisation	Mise en œuvre
Jalons de direction de projet	Intentions	Orientations	Pré-contrat	Contrat	ROP	AMPS
Jalons de synchronisation des métiers ...	Intentions par métier	Orientations par métier	Choix architecture	Choix design	Gel design programme	Réalisation outillages

Tableau 4.4 Principaux jalons significatifs en entreprise (Source Renault)

III.3 Vue organisationnelle

L'un des constats soulevés lors de notre analyse du terrain est que le processus d'industrialisation du véhicule transcende les divisions organisationnelles. Un référentiel du domaine de l'ingénierie de production peut concerner des clients se trouvant dans des directions différentes. Il s'agit de définir la structure organisationnelle du périmètre, ses acteurs, leurs rôles et compétences. L'endroit où se conduisent les activités du périmètre s'étend aux sites de production, prolongement des activités d'ingénierie mais aussi aux réseaux informatiques... A ce titre, la vue organisationnelle couvre les deux dimensions "who" et "where" du cadre de Zachman.

Par ailleurs, les acteurs potentiels intervenant dans l'ingénierie des SIT en entreprise, sont auparavant définis. Ils se résument à la MOA : représentant les métiers, la MOE informatique, le comité de pilotage et puis les prestataires développant les modules du SI pour lesquels ils sont mandatés.

Nous illustrons sur la Figure 4.5 et à l'aide de l'outil matriciel (Groupe de travail/Rôle), les acteurs du domaine et leurs rôles.

Work Group to Role											
		Ok		Cancel		Print					
Role	Work Group	"Analyste Organisation de Travail (AOT)"	"Chef d'UET (CU Projet)"	"Chef d'UET (CU Série)"	"Engageur en projet"	"Engageur en série"	"Logisticien en projet"	"Préparateur en projet"	"Prestataires & Partenaires"	Implanteur	"Responsable d'Affaire (R.A)"
"Affectation des opérateurs aux postes"			X		X						
"Affectation des opérations aux postes"			X		X						
"Affectation des opérations aux tronçons/UET"				X							
"Analyse des temps opératoires"	X										
"Calcul de flux logistiques et implantation des surfaces logistiques"						X				X	
"Conception des surfaces d'ateliers"										X	
"Conditionnement : pièce-emballage"						X	X				
"Définition et implantation du Bord de chaîne"						X	X	X			
"Evaluation de la qualité opératoire"		X									
"Formation des opérateurs"			X								
"Maquette d'effectifs et moyens d'approvisionnement"						X					
"Validation des composants à sous-traiter"								X		X	
"Validation des préconisations produit/process en"			X								X

Figure 4.5 Matrice "Groupe de travail/Rôle" – Aspect de la dimension organisationnelle

III.4 Décomposition du processus métier (business process)

La prolifération des définitions liées au terme processus et à sa décomposition (chapitre 2, III.1) est sans doute liée à la diversité des orientations en marketing et dans les champs disciplinaires (analyse structurée, entreprise modeling...). Pour rester neutre, nous avons retenu des définitions plutôt génériques liées au processus (Encart 2.4). Concernant sa décomposition, nous adaptons ce que l'on a trouvé dans la littérature à l'organisation de Renault de la manière suivante (Figure 4.6) :

- Processus primaire (Primary Process Group) : est un ensemble d'activités étroitement liées. Ces activités répondent à un ou plusieurs événements et produisent des résultats généralement interdépendants. Un processus primaire, le plus souvent, transcende plusieurs unités organisationnelles. Exemple : la logistique, le montage, l'emboutissage...
- Processus secondaire (Process Thread) : un processus primaire est très large. Pour le décrire, il est plus judicieux de le partitionner en processus secondaires. Le déroulement de ces derniers peut s'enchaîner ou non. Exemple : implantation, construction des opérations, simulation de flux d'ateliers...
- Processus élémentaire (Elementary Business Process) : comme son nom l'indique, c'est un processus qui ne se décompose plus. En revanche, pour l'exécuter et produire un résultat, il nécessite des fonctions (intemporelles) et un événement pour le déclencher. Nous le

distinguons des processus secondaires en le décrivant avec un verbe d'action (affecter, calculer, définir...).

Ce type de décomposition peut produire des nomenclatures récursives de type "arbre" ou "forêt". Un nœud appartenant à un arbre devrait avoir qu'un seul père. Un nœud appartenant à une forêt en revanche peut en avoir plusieurs. Nous choisissons cette flexibilité car on peut trouver un processus qui puisse être suffisamment générique ou standard pour qu'il participe à la réalisation de plusieurs processus pères. En effet, nous allons le constater à la construction des processus opérationnels, que certains processus élémentaires, comme "définir une gamme opératoire", peut vraisemblablement être le même aussi bien pour les métiers d'assemblage général que l'emboutissage par exemple. Cependant, ce même processus, même s'il a la même appellation, est différent selon qu'il est exécuté en ingénierie ou en exploitation à cause de la nature des données différentes qu'il requière.

Bien entendu, cette décomposition des processus peut ne plus être appropriée à l'extension du périmètre de l'ingénierie de production, qui n'est qu'un processus parmi tant d'autres en entreprise.

A ce niveau d'investigation, nous nous contentons de répertorier l'ensemble des activités appartenant au périmètre et de les classer en fonction de cette structuration. Dans la partie suivante (IV), nous procédons à leur modélisation à l'aide de diagrammes de spécification.

Quant à la dimension "donnée", nous avons préféré l'intégrer avec le data model, bien qu'elle soit perçue à travers les processus, l'organisation, les jalons etc. Par conséquent, elle peut être étudiée ici même.

Notons toutefois que ces dimensions ne sont pas étanches. Ainsi, pour répondre à un objectif stratégique de la firme, un processus est déclenché par un événement, impliquant un acteur se trouvant dans l'organisation.

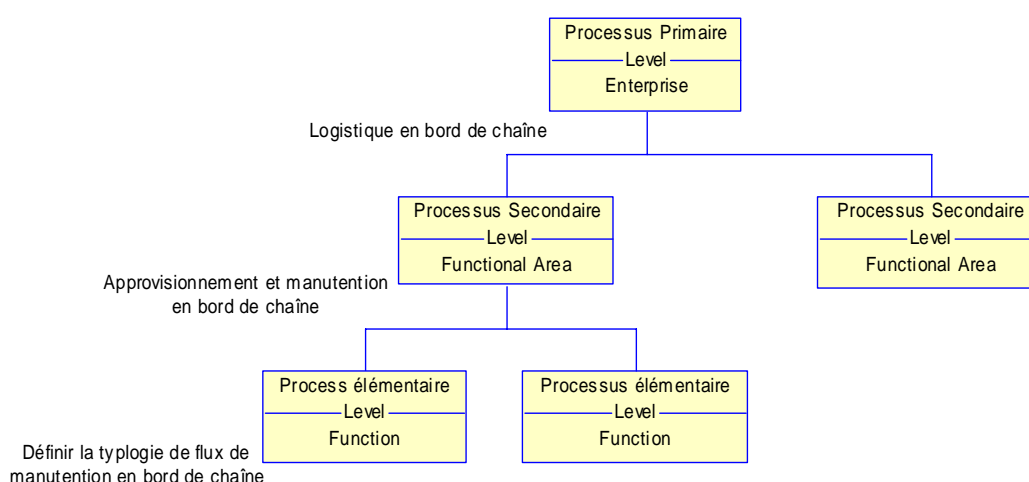


Figure 4.6 Décomposition des processus métier : des processus primaires aux processus élémentaires

IV Modèles opérationnels

La finalité de ces modèles n'est pas la reconfiguration des processus opérationnels de l'entreprise. Il faut alors juste identifier ces processus, les rationaliser si nécessaire et par la suite les renseigner. Cela nous permet d'alimenter le data model de nouveaux besoins en terme de données issus de l'analyse de ces processus. Nous pouvons par ailleurs établir des constats significatifs sur le fonctionnement général des processus.

IV.1 Construction des modèles de processus

Contrairement au processus de fabrication (manufacturing process), plutôt reproductible et répétitif, le processus d'ingénierie de production est d'autant plus difficile à rationaliser qu'il se situe en phases très en amont.

A titre d'illustration, énonçons quelques-uns de nos constats, basés sur l'analyse des processus du périmètre suivant :

- Les deux métiers : montage et tôlerie
- Les trois activités transversales : préparation, implantation et définition des moyens en bord de chaîne et au poste de travail.
- Jalon : avant-projet jusqu'à l'étude détaillée.

(1) La description partielle du processus de montage à la Figure 4.7 montre que la conception du process en amont est ascendante (Figure 1.10). En s'appuyant sur des données manipulées en exploitation d'un projet précédent, les préparateurs définissent les opérations et les ressources (uniquement les outillages) au niveau du poste de travail. Une fois que les postes sont définis un par un, des calculs d'agrégation et des simulations sont effectués afin d'évaluer les performances globales des lignes et des tronçons. Ce principe de conception, appliqué aussi dans les autres métier process, s'explique probablement par la nature des outils d'IPAO d'aujourd'hui. Ces derniers fonctionnent à l'aide des données microscopiques disponibles seulement en aval du projet. De ce fait, lorsqu'ils sont employés en amont, les données provenant des projets précédents sont réutilisées (chapitre 1, II). Pour le data model, devrions-nous imaginer des données macroscopiques conciliant les données de détail et pouvant alimenter les outils de demain conçus spécialement pour l'avant-projet ?

(2) Examinons maintenant un fragment de l'activité de tôlerie concernant la définition des "liaisons et des points de soudure"⁹ du véhicule (Figure 4.8). Avec l'avancement du projet, on s'aperçoit de l'évolution du concept de liaison (ce qui est naturel) mais aussi de la multiplicité des BdD stockant ces données (ce qui est un goulet d'étranglement). La notion de liaison prend des formes différentes : de modèles géométriques, à la liste des pièces de liaison, aux opérations traduisant la liaison (soudage, encollage, raboutage...). Ces mêmes opérations sont validées à l'aide de simulations. Elles sont analysées avec des temps opératoires et les ressources leur sont affectées. Nous constatons aisément la présence des données communes que l'on peut fédérer dans le data model.

⁹ Une liaison peut être traduite par un nœud du graphe produit/process ; l'espace où est réalisé l'empilage des pièces à l'aide des éléments de liaisons. Un point de soudure (PSR) est un élément de liaison au même titre qu'un rivet, goujon, écrou, cordon de soudure...

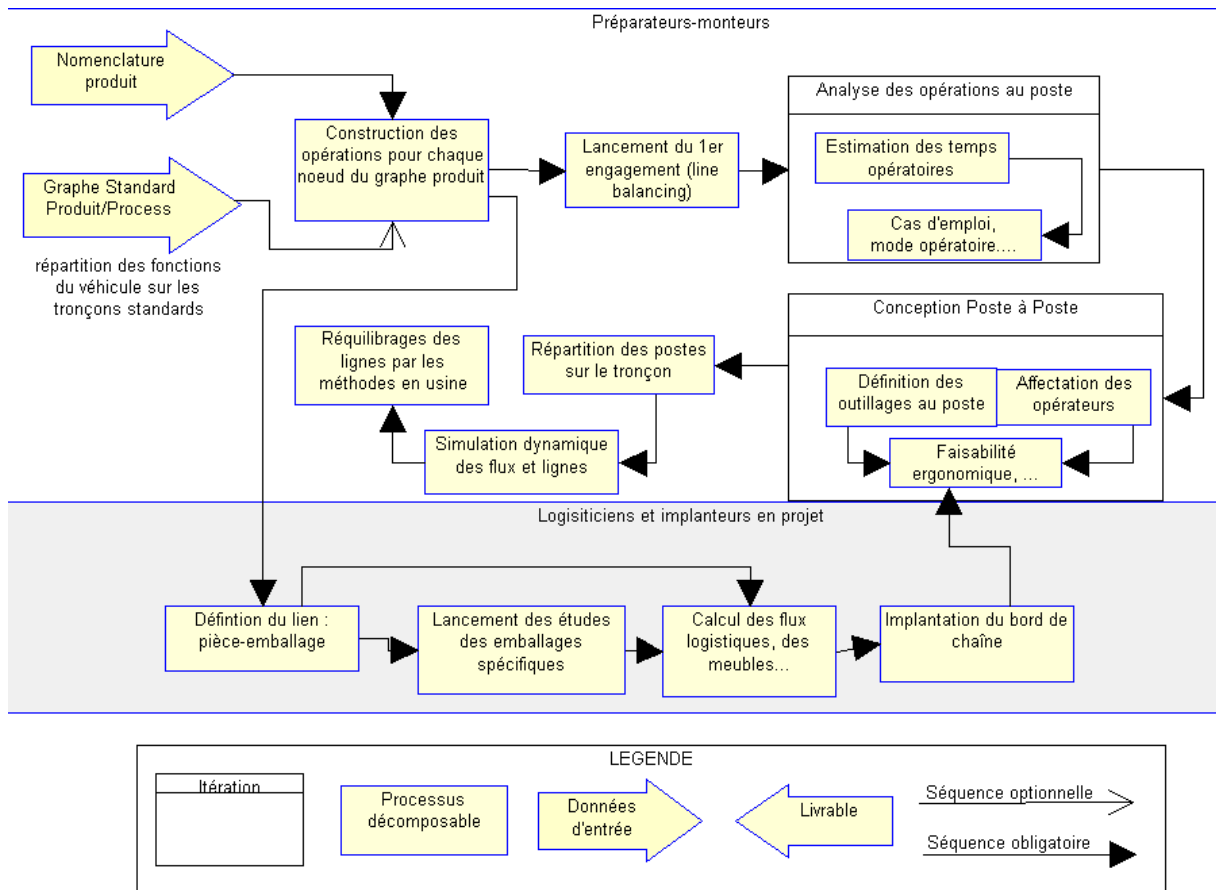


Figure 4.7 Activité de préparation en montage : conception en bottom-up

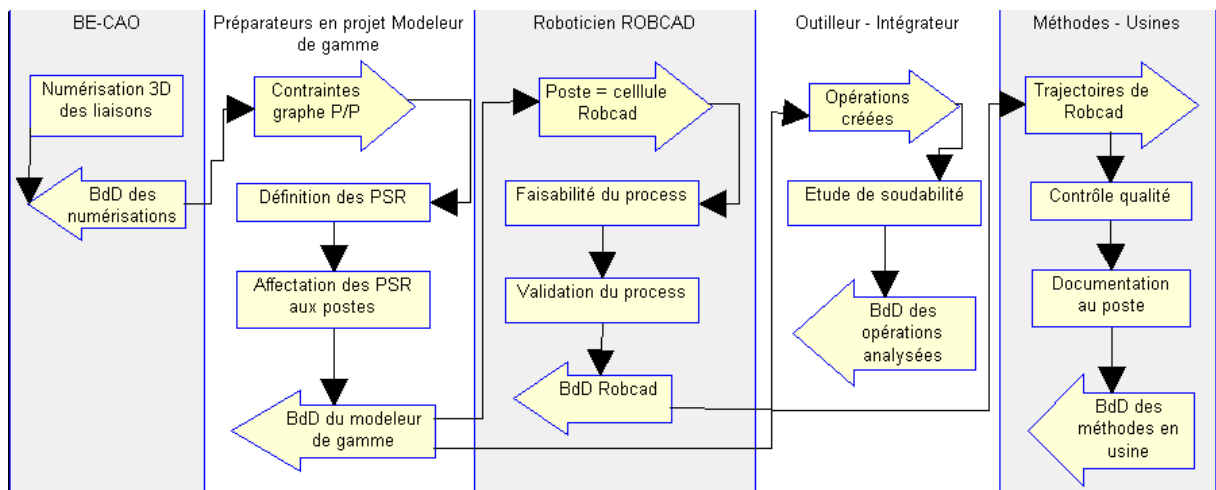


Figure 4.8 Processus de définition des liaisons en tôlerie : multiplicité des BdD

(3) Des discontinuités sont souvent perçues dans le déroulement des processus, à l'occasion d'un changement de jalon, de division organisationnelle ou encore d'outil IPAO. Pour remédier à ce dysfonctionnement, nous connectons les résultats des processus amont aux événements qui devraient déclencher les processus aval (Figure 4.9). Bien entendu, cette manipulation n'est cohérente que lorsque nous constatons que le processus aval n'est que le prolongement de son prédécesseur. Cela nous permet ainsi de créer des liens entre (fragments de) diagrammes et d'assurer par conséquent une vision globale du système.

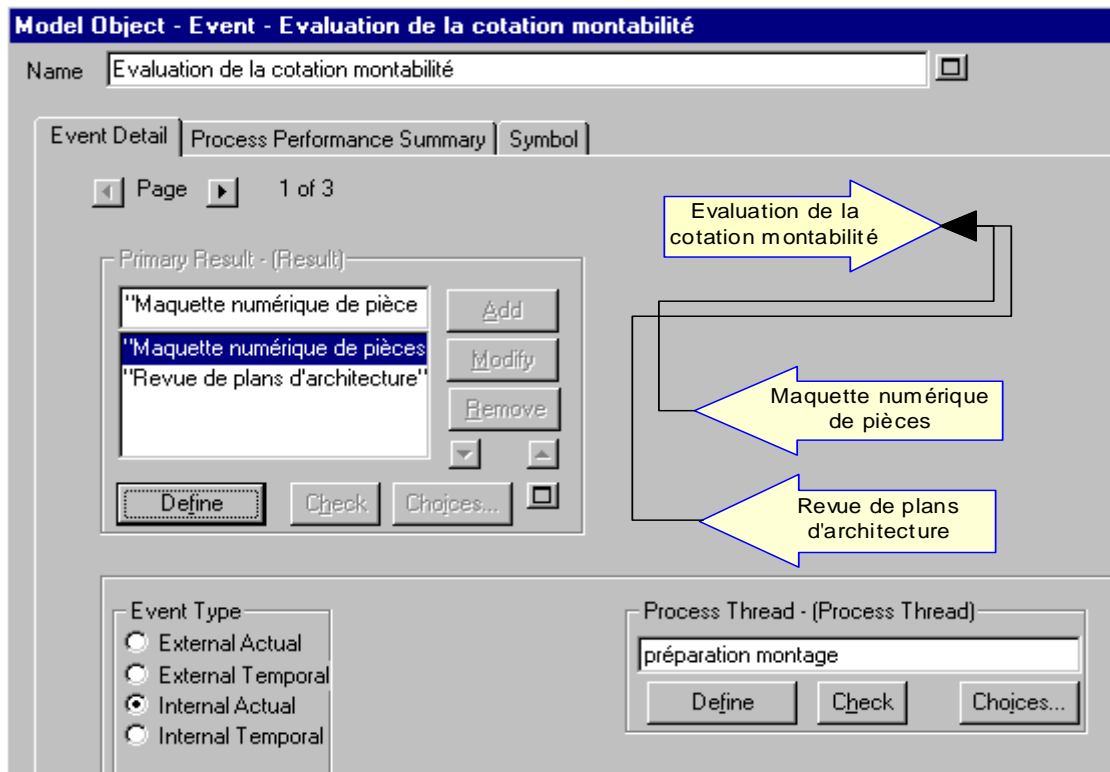


Figure 4.9 Connexions "Résultat - Événement" : recoller les processus morcelés

(4) En règle générale, la conception du process passe par les jalons suivants (Figure 4.10). Partant des spécifications du produit par le BE, les premières définitions produit/process sont réalisées conjointement par les architectes et les préparateurs. Ils produisent ensemble un graphe d'assemblage générique (découpage fonctionnel pour le montage et de la caisse ferrée pour la tôlerie). Le découpage est ordonné en fonction de sa réalisation sur site de production (Tronçons pour le montage et Units pour la tôlerie). Les spécifications purement process débutent avec la construction des opérations réalisant chaque nœud du graphe d'assemblage. Ces opérations sont affectées aux postes selon des règles métiers (line balancing). Les activités transversales de définition des ressources et d'implantation (répartition et calcul des surfaces au bord de chaîne) se font d'une manière concurrente. Elles produisent respectivement des catalogues de ressources et des plans détaillés d'implantation. A l'heure actuelle, seuls les graphes produit et produit/process peuvent être appelés nomenclatures. Là aussi, nous devons signaler le besoin de structurer ces opérations et ces ressources dans des nomenclatures afin d'améliorer globalement la gestion de ces données techniques.

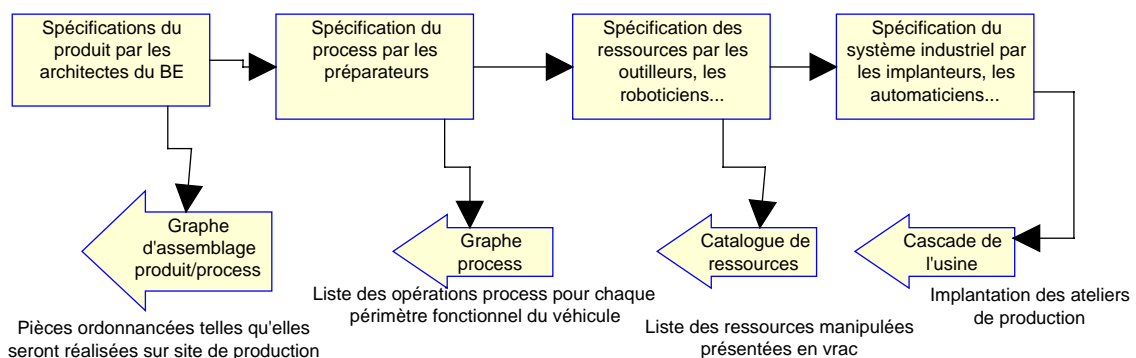


Figure 4.10 L'ingénierie process et ses principaux livrables

Cette analyse des processus, bien que pertinente, manque toutefois de précision. Pour obtenir une analyse détaillée des processus, documentons-les de près à travers leur composante la plus fine qui est le processus élémentaire.

IV.2 Documentation des processus élémentaires

Dans le souci de faire participer au mieux les acteurs du métier process, experts dans leur domaine, nous avons standardisé la manière de documenter les processus élémentaires. Leurs attributs tels qu'ils sont fournis par l'AGL ne correspondent pas à la culture d'entreprise. L'uniformité que nous avons donc imposée répond à deux objectifs : la complétude des attributs et la facilité de les renseigner. Pour la complétude, nous nous sommes inspirés là encore du cadre de Zachman pour les identifier. Pour la facilité, nous réduisons le travail fastidieux de saisie des informations en réutilisant les précédents résultats des modèles contextuels (jalons métier, applications logicielles utilisées...). Nous invitons alors les acteurs participant au projet (d'élaboration du référentiel ou de développement d'un nouveau SIT) à documenter l'onglet "Documenter le processus" illustré par la Figure 4.11 et la Figure 4.12.

Model Object - Elementary Business Process - Implanter les moyens au bord de chaîne

Name: Implanter les moyens au bord de chaîne

Unit Cost | Quality Cost | Performance Measurement | Function Detail | Documenter le processus | Symbol

Page 1 of 2

Quoi ? (données): Plan du bâtiment réalisé en avant-projet, Surfaces allouées des lignes de l'atelier, Synoptique des postes

Pourquoi ? (objectifs): Réception de nouveaux emballages de pièces

Qui ? (acteurs): Implanteur

Avec quoi ? (application): IPAM

Comment ? (processus): Positionner les ressources en fonction : des règles de flux logistiques et de manutention des résultats de calcul de l'engagement des lignes de la répartition des surfaces ...

OK | Cancel | Spell | Delete

Help texte 5 de Renault | Text | Len: 20

Figure 4.11 Documentation uniforme des processus élémentaires (1/2)

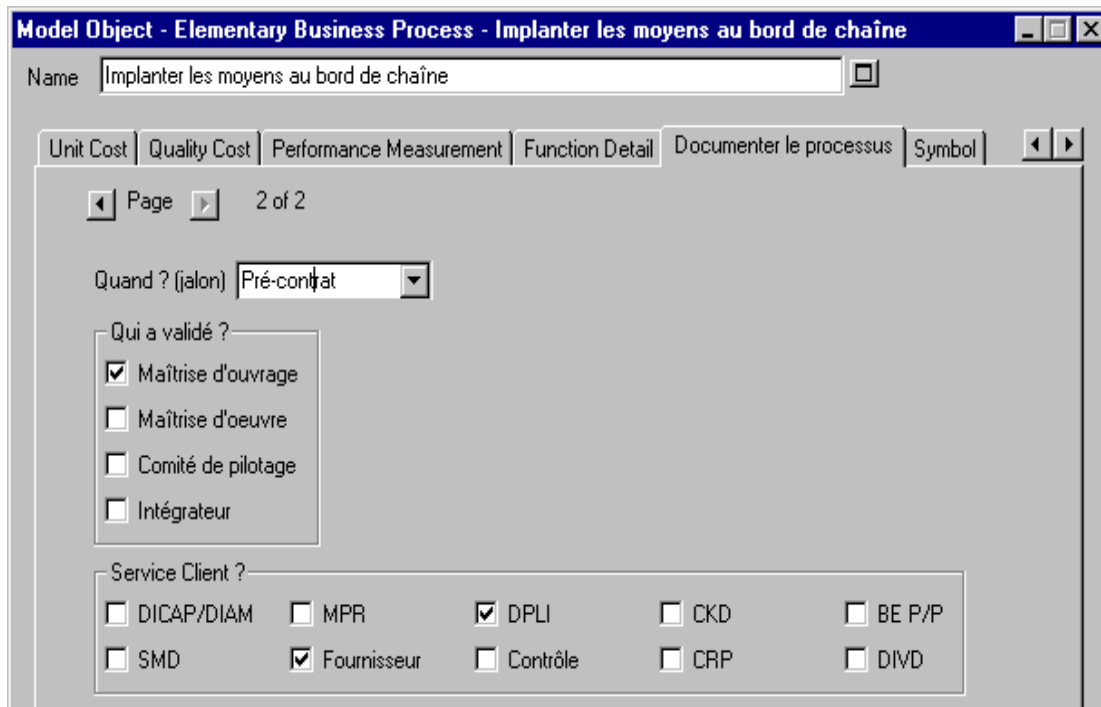


Figure 4.12 Documentation uniforme des processus élémentaires (2/2)

Les attributs "quoi, pourquoi et comment" sont volontairement définis comme des champs de texte. L'acteur peut ainsi rédiger librement les données requises, la finalité et l'objet de son activité. Pour assurer une continuité des processus métier à travers les unités organisationnelles, l'acteur peut cocher le ou les services clients de son activité. Pour une meilleure conduite du projet, il est utile de savoir à tout moment qui a validé les informations saisies dans chaque processus élémentaire. L'approbateur, ayant une vision locale du projet, valide partiellement ces attributs. De ce fait, Il devient utile de rendre visible ces champs en fonction de l'expertise de l'approbateur (Figure 4.13).

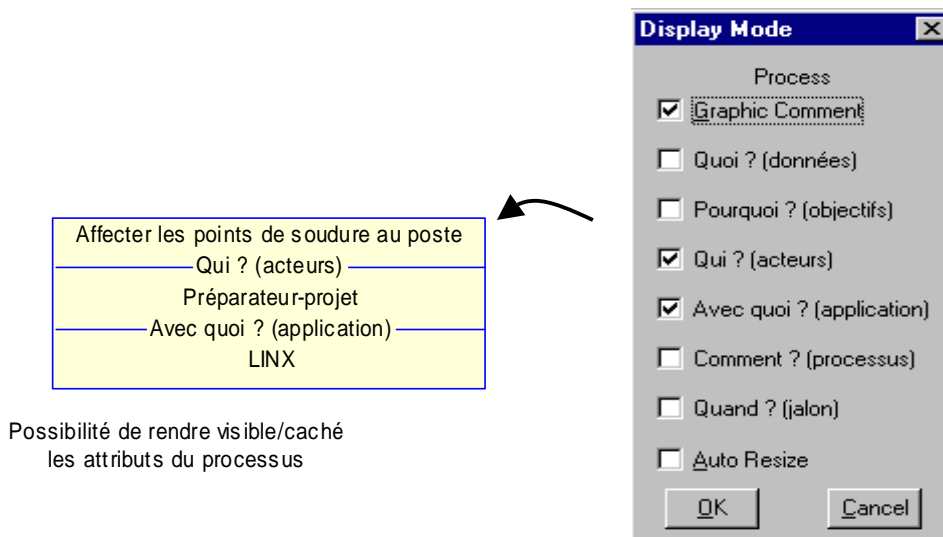


Figure 4.13 Réduction de la complexité des diagrammes

Par ailleurs, les acteurs du métier process maîtrisent mieux l'environnement de leur outil de travail (applications IPA0) que le déroulement même des processus pour lesquels ils sont

concernés. Pour cela, nous avons établi des diagrammes orientés application. Ces derniers, complétant les diagrammes de processus, permettent de capitaliser par exemple les flux d'informations circulant à travers les applications par ressaisies manuelles ou automatiques (Figure 4.14).

D'autres types de diagrammes sont créés en vue de cerner quelques aspects de l'organisation comme : l'architecture fonctionnelle des SIT existants, le processus des décisions... Nous garantissons la cohérence et l'intégration des vues du système en travaillant sur une même encyclopédie de l'AGL.

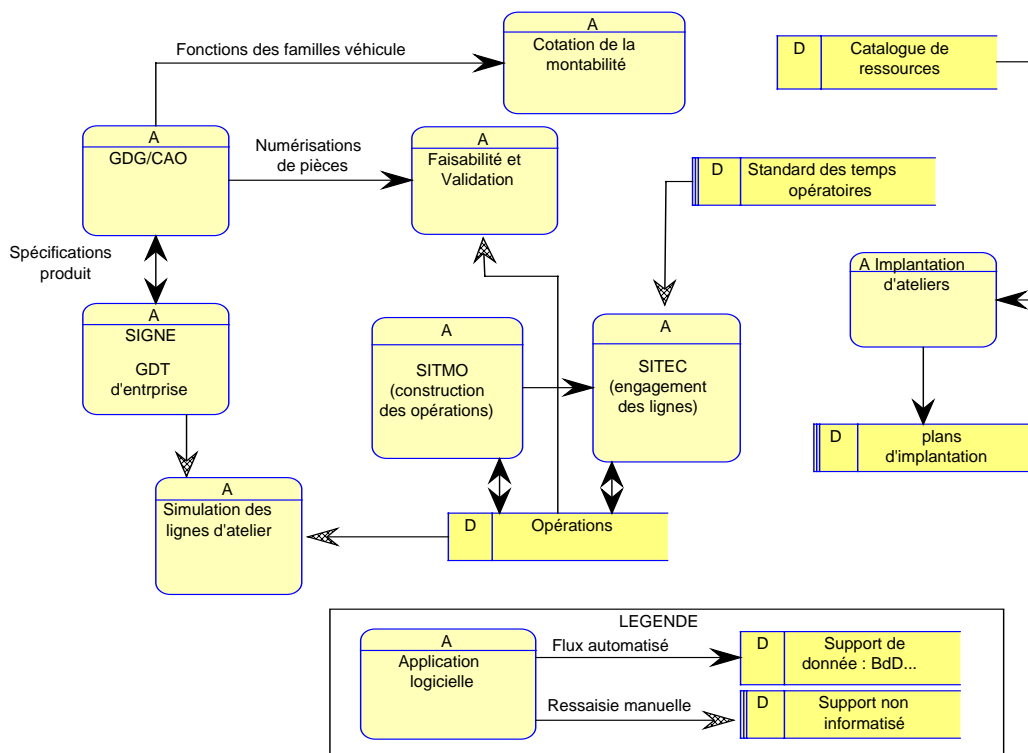


Figure 4.14 Quelques applications et flux en ingénierie de montage

Néanmoins, la finalité du business model n'est pas atteinte. Il est encore difficile de décliner les objets potentiels pour alimenter notre data model. L'objet de la partie suivante est de transformer justement ces présents diagrammes en modèles plus conviviaux, servant de passerelle pour accéder au data model.

V Mapping vers les données : modèles de transformation logique

Le problème de transfert et de communication vers le data model n'aurait pas existé si nous avions emprunté le procédure usuelle des utilisateurs de la seule technologie objet (UML) [JAC95;MAR'00;MOR00]. Notre choix de combiner les techniques de modélisation des processus, de données (E-R) et UML a déjà été argumenté (chapitre 3, IV).

Par ailleurs, l'état de l'art nous a montré que le processus d'extraction des objets métiers depuis l'analyse des processus est plus pratique que théorique ; il n'existe pas de règles de conduite de ce processus. Néanmoins, l'expérience des praticiens révèle que le chemin trivial serait de passer par le concept du cas d'utilisation métier, chemin montré sur la Figure 4.3 que nous expliquons dans ce qui suit.

V.1 Des processus aux cas d'utilisation métier

Avant d'illustrer le mécanisme de construction des modèles de transformation, décrivons d'abord le concept fondateur de ces transformations : le cas d'utilisation métier.

Un cas d'utilisation (use case) est une spécification d'une séquence d'actions, incluant des variantes que le système peut exécuter en interagissant avec les acteurs du système [KET98], définition complémentaire à celle notée dans le Tableau 4.3. Il s'agit donc d'une description du système privilégiant le point de vue de l'utilisateur.

Pour représenter ce concept, UML dispose du diagramme des cas d'utilisation. C'est un diagramme contextuel, montrant d'un haut niveau d'abstraction, les relations entre les acteurs et les cas d'utilisation. Il replace ainsi le système (particulièrement le SI) dans un contexte d'usage.

Les cas d'utilisation sont d'une utilité majeure aussi bien dans la conception de bases de données [MUL99] que dans l'élaboration de notre référentiel.

En effet, cette phase, contrairement aux apparences, n'est pas redondante vis-à-vis des modèles opérationnels précédemment construits. Grâce aux cas d'utilisation, il est plus facile de décliner les objets potentiels du domaine d'une part, de raffiner les processus précédents en réexaminant les requirements du système d'autre part. Il s'agit vraisemblablement d'un processus itératif, facilité par la réutilisation des concepts (symboles et définitions) stockés dans l'encyclopédie. Toutefois, nous ne préconisons pas de reconfigurer tous les modèles opérationnels en diagrammes de cas d'utilisation ; cette phase est appliquée seulement pour quelques (fragments de) processus secondaires –sous systèmes- jugés trop complexes.

Etapes de construction des diagrammes de cas d'utilisation

(1) Identification des acteurs : l'investigation des rôles dans le domaine (Figure 4.5) nous a permis de décliner les acteurs du métier. Un acteur peut participer à plusieurs processus. Dans un même processus, plusieurs rôles peuvent être joués. La Figure 4.15 montre une structuration hiérarchique des acteurs du métier. Dans ce contexte, l'héritage signifie que l'acteur fils hérite des rôles joués par l'acteur père. Autrement, tous les cas d'utilisation avec lesquels l'acteur fils communique, l'acteur père les utilise. Ce dernier peut communiquer exclusivement avec d'autres cas d'utilisation. La répartition hiérarchique des acteurs permet d'affiner la description des modèles de cas d'utilisation. Ainsi, dans certains contextes, il est plus judicieux d'employer l'acteur préparateur pour couvrir les rôles conjoints du préparateur projet et série. Cet acteur est employé aussi bien pour le montage que pour le métier de la tôlerie, dans lequel il est nommé responsable d'affaire. De même, il est parfois préférable d'employer implanteur au lieu des différentes spécialisations des implanteurs...

Par ailleurs, la définition du concept d'acteur (Tableau 4.3) exprime que tout ce qui est extérieur au système est un acteur. A ce titre, l'acteur des événements temporels par exemple serait le temps (III.2) puisqu'il est à l'origine du déclenchement de quelques actions...

(2) Identification des cas d'utilisations : il ressort des définitions plus haut qu'un cas d'utilisation s'obtient aisément depuis :

- Un processus élémentaire tel que nous l'avons identifié dans les modèles opérationnels.
- Un événement du système (business event).
- Un résultat d'analyse des flux sortants du système.
- Ou plus globalement, tout verbe d'action décrivant un requirement formalisé dans les modèles contextuels (III). Notons que celui-ci est la première version du cas d'utilisation appelé "requirement use case".

Le principe de décomposition fonctionnelle s'applique aussi sur les cas d'utilisation à travers les deux stéréotypes de liens "extends" et "includes" [OMG99] montrés sur la Figure 4.16 :

Le lien "includes" (appelé aussi "uses") veut dire que le cas d'utilisation "Etudier la faisabilité ergonomique au poste" inclut ou utilise les comportements décrits dans les autres cas d'utilisation. Ces derniers sont appelés abstraits lorsqu'ils sont référencés par un cas d'utilisation qui requiert leurs fonctionnalités ; ce qui est une forme de réutilisation.

Le lien "extends" est souvent employé dans des situations où se présentent des conditions ou des séquences d'actions à respecter. Sur la Figure 4.16, le cas d'utilisation d'extension "Déterminer le mix/volume du véhicule" peut être étendu avec des comportements additionnels définis dans le second cas d'utilisation si les conditions portées sur le lien "extends" sont satisfaites. Autrement dit, l'engagement d'une ligne (activité du "line balancing") doit tenir compte de la demande du marché formulée avec un langage de description des caractéristiques du véhicule à produire. Mais lorsqu'il s'agit de simulations en avant-projet, il suffit de calculer une diversité moyenne sur la base des données initiales.

L'affinement des cas d'utilisation d'analyse et des acteurs se poursuit d'une manière itérative en même temps que la construction des diagrammes suivants.

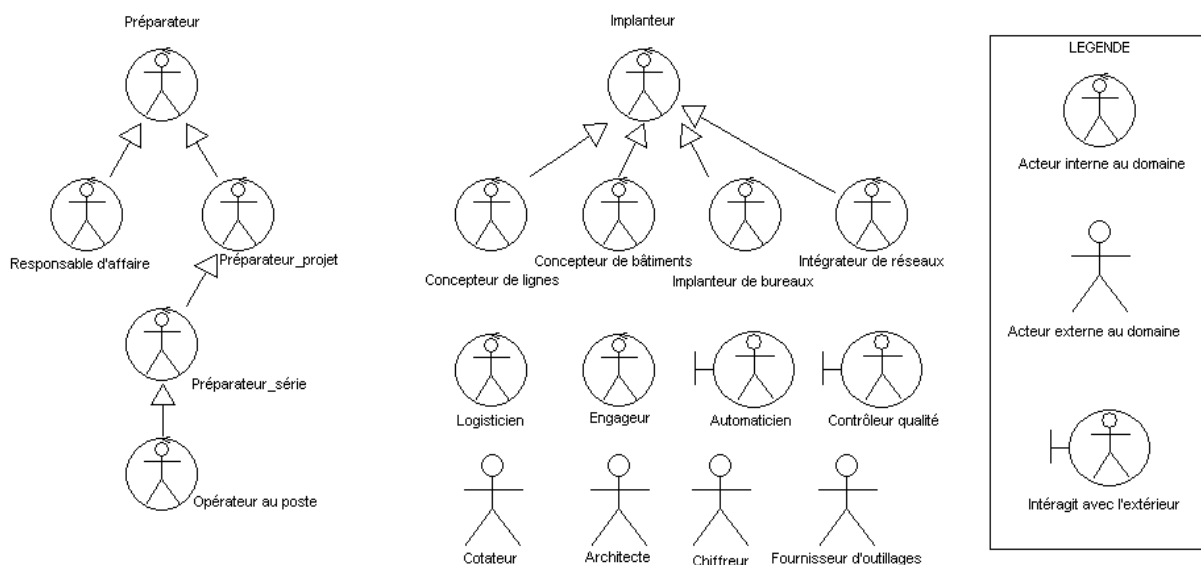


Figure 4.15 Des rôles aux acteurs : hiérarchisation des acteurs du métier

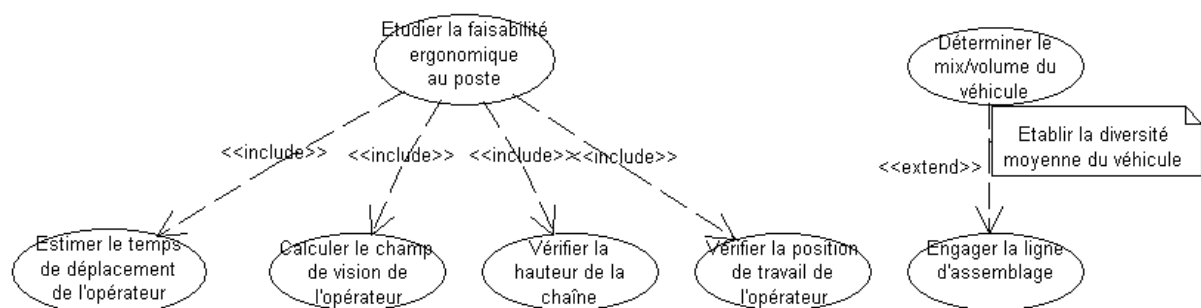


Figure 4.16 Les types de liens entre cas d'utilisation

(3) Construction des modèles de cas d'utilisation

La Figure 4.17 est un exemple d'un diagramme de cas d'utilisation. Il décrit, d'un haut niveau d'abstraction, les principaux acteurs et cas d'utilisation pour le processus de conception du bord de chaîne.

Chaque cas d'utilisation peut être documenté d'une manière itérative avec des informations détaillées que nous récapitulons dans un formulaire comme au Tableau 4.5.

Une fois les cas d'utilisation documentés, nous pouvons poursuivre le processus de déclinaison des objets potentiels. Mais, il arrive qu'un cas d'utilisation soit complexe et la représentation jusqu'à présent statique ne suffit pas, d'où l'intérêt des autres diagrammes dynamiques UML.

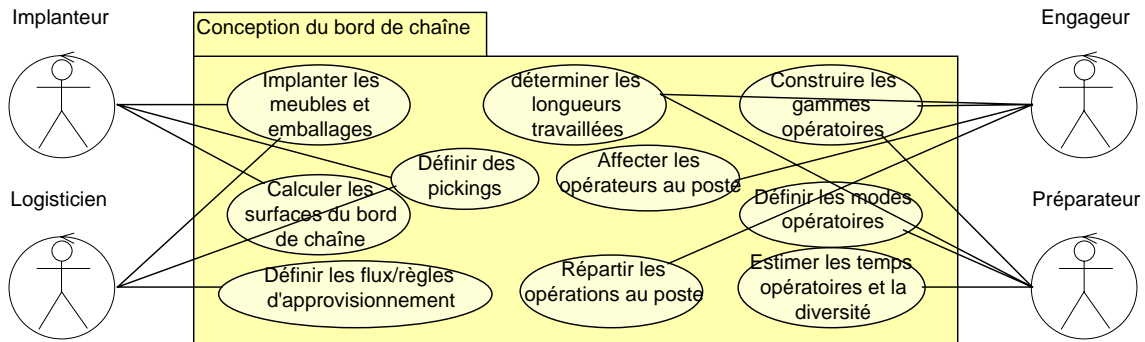


Figure 4.17 Conception du bord de chaîne. Diagramme de cas d'utilisation

Nom du cas d'utilisation	Définir des pickings	
Acteur(s) communiquant	Implanteur, Logisticien. Par la suite, Engageur, Préparateur et Pickeur	
Description	Il s'agit de construire des surfaces supplémentaires, appelées "picking" afin de stocker les emballages ne pouvant plus être contenus dans le bord de chaîne. Une activité de préparation et d'encyclage des pièces lui est associé	
Etapas de réalisation	Actions des acteurs (1) Initialisé par l'implanteur (3) Logisticien désigne les emballages à déplacer et envoie une proposition aux préparateurs (4) Ensemble, ils calculent les paramètres de rééquilibrages des postes concernés (6) La réalisation du picking conclut ce cas d'utilisation.	Réponses du système (2) Etude de faisabilité (à l'aide de simulations logicielles) (5) Validation de la conception du picking. Nouveau synoptique du bord de chaîne.
Pré-conditions	Débordement de la capacité de la surface du bord de chaîne	
Post-conditions	Respect des contraintes de performance (temps d'encyclage des pièces), ergonomiques (longueurs travaillées)...	
Hypothèses	Premier engagement de la ligne est déjà effectué	

Tableau 4.5 Description textuelle du cas d'utilisation "Définir des pickings"

Constatons qu'un cas d'utilisation peut comprendre plusieurs scénarios d'utilisation. Un scénario n'étant qu'une instance du processus, peut être construit à l'aide du diagramme de séquence ou de collaboration d'UML.

Le diagramme de séquence est souvent préféré au diagramme de collaboration, car il permet de mieux visualiser les transactions dans le temps comme l'illustre la Figure 4.18.

Par ailleurs, nous avons déjà souligné (chapitre 3, IV.3) que les autres diagrammes UML (Activité, Etat) ne sont pas très appropriés à ce contexte. Nous aurons toutefois l'occasion de les employer plus tard pour décrire certains processus automatisés.

A présent examinons le processus de déclinaison des objets potentiels.

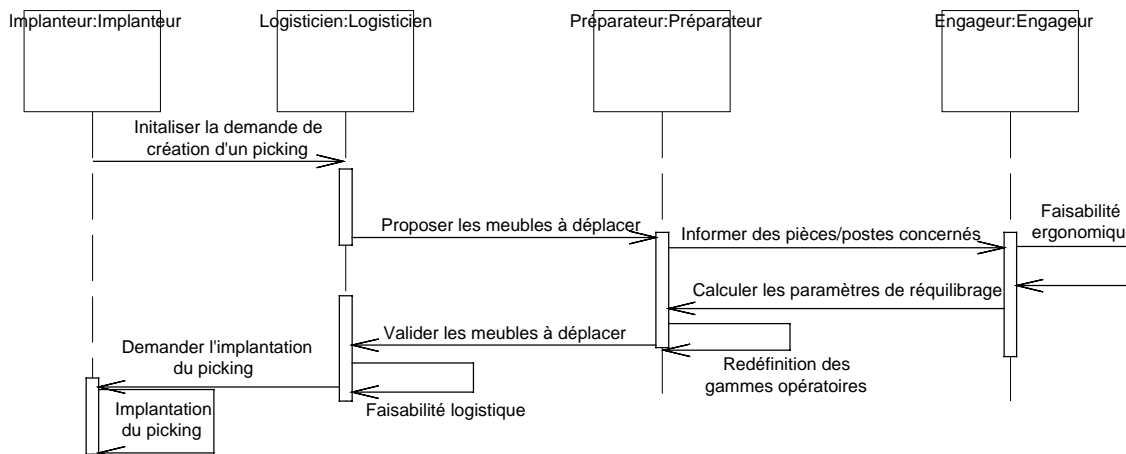


Figure 4.18 Scénario du cas d'utilisation "définir un picking".
Diagramme de séquence

V.2 Des cas d'utilisation métier aux objets potentiels

Nous arrivons à la frontière du data model. Néanmoins, nous pouvons transcrire ici quelques principes empruntés à la technologie objet avant de passer à la section suivante.

La Figure 4.19 situe nos objets métier vis-à-vis des objets d'un système.

Les objets métier sont au cœur du système, définis à plusieurs reprises dans la thèse (notamment dans le Tableau 4.3). Retenons seulement qu'ils sont parfois appelés objets du domaine ou objets persistants. Le mot persistant signifie que de tels objets sont pérennes et ce sont eux qui se matérialisent dans les BdD.

Certains auteurs classent ces objets métier en deux catégories [KET98] :

- (1) Objets du modèle qui encapsulent les méthodes, les informations et les règles de gestion liées à l'objet du monde réel.
- (2) Objets d'usage correspondent à une extraction d'une partie de l'objet du modèle ; ils sont manipulés par exemple par les utilisateurs au sein d'une architecture répartie. Les objets d'usage résultent de la conception technique des SI ; ce qui est exclu du périmètre de notre référentiel.

Les objets de l'application surviennent aussi pendant la conception objet.

Les objets d'interface pilotent le dialogue entre l'utilisateur et l'application, affichent des informations nécessaires et réagissent à ses sollicitations (exemple : le profil de l'utilisateur et son mot de passe tels qu'ils apparaissent sur l'interface de l'application).

Les objets de contrôle "régulent le trafic". Ils contiennent les règles métier leur permettant de coordonner les interactions entre objets. Un objet de contrôle est associé à un et un seul acteur.

Les dispositifs physiques sont les objets avec lesquels l'utilisateur interagit (exemple : souris). Comme leur nom l'indique, les objets d'implémentation représentent les aspects internes du système et n'apparaissent qu'en phase d'implémentation de l'application objet.

Pour mieux distinguer ces différents objets, UML propose des stéréotypes pour chacun d'eux. "Entity object" est le stéréotype des objets métier [OMG99].

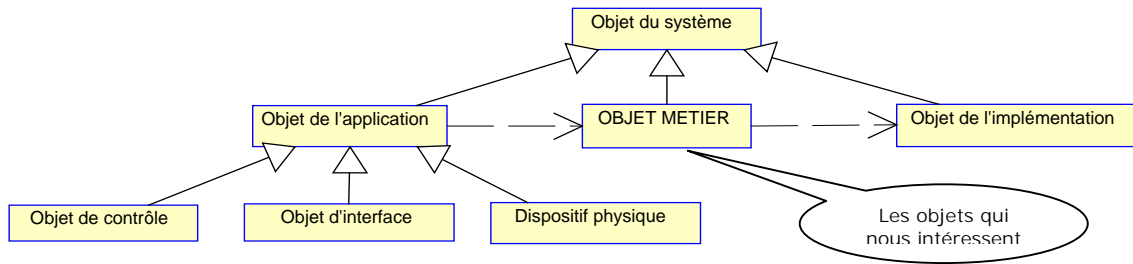


Figure 4.19 Typologie des objets (source [KET98])

Par ailleurs, la procédure standard de la technologie objet, stipulant que tout nom est objet, peut être appliquée ici en deux étapes :

(1) Surlignons tous les noms trouvés à la lecture des cas d'utilisation documentés (Tableau 4.5) et des processus élémentaires (Figure 4.12). Énonçons les constats suivants :

- Un processus est une classe d'objets dont les réalisations concrètes (réelles ou simulées) sont les instances.
- Les acteurs du métier et tout agent externe intervenant au processus d'ingénierie de production sont des objets.
- Les livrables des processus (documents, graphiques,...) et les événements (business event) sont aussi des objets...

(2) Analysons ces objets candidats, en procédant à la suppression des (Tableau 4.6) :

- Synonymes et noms redondants (chaîne et ligne de production, site et usine...).
- Objets sortant du périmètre de notre domaine (analyse financière) où leurs rôles sont externes au domaine (géométrie, contrôle qualité ...).
- Noms qui ne sont que des actions (initialisation), des messages (transmission), des propriétés d'autres objets (pas de chaîne) ou encore calculés (longueur travaillée).

Il est temps d'approfondir certains objets ambigus rencontrés ici, la granularité et l'abstraction des objets ainsi que les interactions entre eux.

Objets candidats		Raison de suppression ou d'approbation
Prix de revient, formation de l'opérateur	⊗	Externes à notre domaine
Champ de vision, pas de chaîne, ...	⊗	Propriétés d'objets à définir
Gamme opératoire, Assemblé, cascade...	✓	objets composites qu'il faut définir plus en détail
Initialisation, validation, transmission...	⊗	Transactions pouvant contenir toutefois des objets
Profil des utilisateurs, demande du marché, ticket d'entrée du projet ...	⊗	Pouvant apparaître plus tard en tant qu'objet d'interface ou de contrôle
Responsable d'affaire, responsable d'unit	⊗	Synonymes de préparateur
convoyeur, palette, meuble, emballage, lugette, car à fourche, balancelle...	✓	Types de moyens de manutention, de stockage, de conditionnement de pièces etc.
Processus primaire, secondaire, ...	✓	Concepts abstraits... à retenir
Hypothèse process/ligne, cas d'emploi...	✓	Définitions à approfondir... à retenir
Chef d'UET projet/série, chef ...	⊗	Appelons-les approbateurs par leur rôle principal
surface implantée, longueur travaillée...	⊗	Attributs calculés... Rechercher les originales

Tableau 4.6 Analyse des objets candidats : exemples

SECTION 3 – Le Data Model

La dernière section de ce chapitre est consacrée à la construction du data model suivant le scénario défini plus haut (II.2). Elle est scindée en deux parties principales. La première (VI) est consacrée au diagnostic des données dans laquelle les techniques de rétroingénierie et les résultats d'analyse du domaine (III, IV et V) se croisent. La partie suivante (VII) suit naturellement où sont structurés et affinés les objets métier en plusieurs étapes contiguës et itératives. Nous terminons avec une conclusion générale (VIII) dans laquelle quelques éléments de conception détaillée sont abordés.

VI Diagnostic des données

L'investigation des SIT existants par rétroingénierie est explorée en VI.1 ; elle vient compléter les résultats du business model. Nous regroupons en VI.2 les objets du domaine dans quatre méta entités dont les caractéristiques seront fixées.

Ces deux volets nous permettent de dégager, parmi les objets potentiels issus de l'analyse du domaine et des techniques de rétroingénierie, les classes de données pertinentes. La Figure 4.20 illustre ces chemins possibles pour arriver au diagramme de classes ; un diagramme de UML que nous utilisons pour structurer et raffiner notre data model dans VII.

Notons que les deux parties s'ajustent réciproquement. De ce fait, nous suggérons aux praticiens de les travailler simultanément.

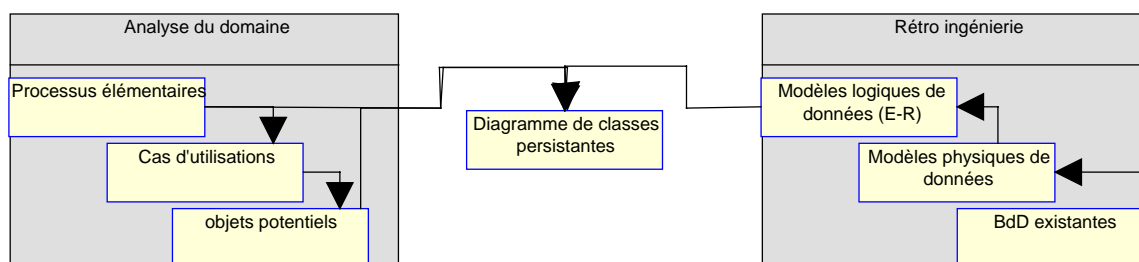


Figure 4.20 Chemins pour accéder au diagramme de classes

VI.1 Rétroingénierie : analyse des similarités et recherche d'analogies

Comme sa définition l'indique (Tableau 4.3), la rétroingénierie s'applique à n'importe quelle forme de SIT automatisé ou non : SGDT, BdD, application...

Prenons l'exemple des SGDT : en analysant quelques-uns de ces progiciels, nous avons constaté pour la plupart qu'ils fournissent comme méta-modèle trois objets natifs pour décrire le produit : "document, graphic & part". Ainsi toute nouvelle entité créée est systématiquement attachée à l'un des ces trois objets natifs. La distinction entre document et graphique est sans doute liée aux cas d'emploi de ces objets. En effet, les graphiques sont généralement rattachés à la nomenclature de pièce (construite depuis la composition récursive d'objets de type "part") alors que les documents peuvent être contrôlés avec des workflow sans les attacher spécialement à la nomenclature. Nous avons aussi constaté que l'objet "part" est loin d'encapsuler la structure et le comportement de toutes les entités du manufacturing.

Cependant, nous voulons que cette partie soit consacrée entièrement à la rétroingénierie des SIT automatisés chez Renault. Les autres types de rétroingénierie sont moins importants à nos yeux et peuvent être perçus en revanche d'une manière implicite.

Avant de décrire le processus même de rétroingénierie, récapitulons dans l'Encart 4.1 quelques principes fondamentaux en modélisation de données ; technique avec laquelle sont conçus les SIT existants.

Pour expliquer au mieux ces principes, nous les illustrons à l'aide de quelques extraits de modèles de données appartenant à Renault en se livrant parfois à de humbles comparaisons de formalismes avec UML.

Pour plus d'informations, nous suggérons au lecteur de revenir aux supports de cours dédiés au domaine de la modélisation de donnée -appelée aussi modélisation de BdD de par l'utilisation de la technologie de BdD en implémentation.

Encart 4.1 Eléments de modélisation des données - Quelques comparaisons avec UML

The excellency of every art is its intensity, capable of making all disagreeables evaporate, from their being in close relationship with beauty and truth. John Keats, January 13, 1818

Modèle logique : décrit le monde réel (comment il est ou ce qu'il fait) indépendamment de la technologie d'implémentation. Il est parfois appelé modèle conceptuel, modèle essentiel voire "business model". Un modèle logique contient des entités et des relations sémantiques, d'où l'appellation des diagrammes Entité-Relation. La Figure 4.21 est un exemple de diagramme E-R obtenu depuis le modèle physique.

Modèle physique : en plus de la fonction précédente, le modèle physique décrit également comment le système est implémenté en reflétant le choix de la technologie. Ses synonymes les plus courants sont : modèle technique et modèle d'implémentation. Dans les technologies relationnelles, le modèle physique est constitué de tables bi-dimensionnelles reliées entre elles par des liens appelés contraintes. Chaque table (appelée aussi fichier) est décrite à l'aide de colonnes (champs qui sont les attributs implémentés) et des lignes correspondant aux enregistrements (collection de champs ordonnés) (Figure 4.22).

La distinction des deux niveaux d'abstraction n'est pas perçue a priori dans UML. Celui-ci fournit le même formalisme basé sur ses neuf diagrammes pour modéliser les différentes vues d'un système ; UML laisse le soin au concepteur de présenter ses diagrammes suivant des préoccupations différentes. Nous pouvons toutefois considérer que le diagramme de classes est un modèle logique lorsque ses classes décrivent les objets métier et un modèle physique quand celles-ci décrivent des objets techniques.

Ce que les classes ont de plus que les entités ?

Les classes offrent des services décrits dans le compartiment opération. La méthode est l'implémentation concrète de l'opération. Les instances de classes communiquent grâce aux messages. UML formalise les contraintes entre sous-types avec le langage de contraintes OCL. Il propose la notion de stéréotype qui est une méta classification des classes...

De ce fait, la transformations d'un diagramme de classes en diagramme E-R est plus délicate que la transformation inverse.

Les règles de gestion dans UML sont ainsi rédigées en OCL ou décrites dans les méthodes des classes ou encore dans les diagrammes d'interactions. Dans la modélisation de données, ces règles sont décrites séparément et implémentées dans les BdD en tant que triggers et procédures.

Normalisation : technique d'analyse de donnée, elle permet d'épurer, en trois étapes, le modèle de donnée avant son implémentation pour qu'il soit simple, non redondant, stable et flexible. Ce processus est automatisé dans la plupart des AGL.

Il n'existe pas l'équivalent de la normalisation pour les diagrammes de classes. En revanche, beaucoup d'AGL fournissent quelques métriques en guise d'indicateurs de qualité des modèles de classes.

Processus de transformation du modèle logique en physique : qui est le processus de conception de BdD, est gouverné par les règles suivantes :

(1) Entités et attributs

- a. Chaque entité devient une table. Les noms sont parfois formatés en fonction des règles du SGBD.
- b. Chaque attribut devient un champ (colonne dans la table). Là aussi les noms peuvent être formatés.
- c. Chaque clé (primaire ou secondaire) devient un index dans la table.
- d. Chaque clé étrangère concrétise une relation possible entre instances de la table, permettant ainsi de réaliser les opérations de jointure. La clé étrangère est une clé primaire d'une entité parent qui se voit dupliquée dans une entité enfant.

A vrai dire, le concept de clé n'est plus utile en objet ; il suffit de spécifier dans chaque classe ses relations avec les autres pour permettre des navigations entre classes.

(2) Super types et sous types

Il peut arriver aux super et aux sous types les combinaisons possibles :

- a. Elles sont implémentées séparément dans des tables.
- b. L'entité sous type est intégrée à la super type pour former une seule entité.
- c. Les attributs de la super entité sont dupliqués dans chaque table représentant la sous type.

*L'héritage, concept existant dans la modélisation de donnée bien avant l'objet, consiste à regrouper les attributs communs à plusieurs types d'entités (**sous types**) dans une entité propre appelée **super type**. L'entité **sous type** hérite quelques attributs communs de l'entité super type et possède d'autres attributs la caractérisant. En plus de l'héritage des attributs, la sous type hérite également les relations de sa super type vers les autres entités (l'entité enfant dans la Figure 4.21 est entourée d'un double trait).*

Par contre, l'héritage multiple (possibilité d'avoir plusieurs parents) est une caractéristique de l'objet (UML) que l'on ne peut trouver dans les diagrammes E-R. cela dit, il est déconseillé d'employer celui-ci...

(3) Contraintes d'intégrité sont de trois types :

Intégrité des clés : s'assurer de l'unicité de la valeur de la clé primaire de chaque champ.

Intégrité du domaine : contrôler qu'aucun champ ne prend de valeurs n'appartenant pas au **domaine**.

Intégrité du référentiel : vérifier qu'aucune clé étrangère ne pointe vers une valeur de clé primaire inexistante.

Domaine : définit les valeurs possibles d'un attribut, concrétisé à l'aide de **type de donnée** et **valeurs par défaut**.

Type de donnée : les plus populaires sont : texte, mnémonique (texte libre), nombre, date, temps, booléen, liste de valeurs, image etc. Les types de données permettent de restreindre le domaine.

Valeur par défaut : c'est la valeur attribuée par défaut à l'attribut si l'utilisateur ne le spécifie pas.

UML propose la même démarche de structuration pour les attributs des classes.

*L'agrégation, concept existant aussi en E-R, se matérialise par l'entité faible. L'identification d'une **entité faible** (weak entity) dépend de l'existence de l'entité parent. Une entité sous type est une entité faible. La situation équivalente en UML est représentée avec le lien de composition forte. C'est-à-dire que la suppression de l'objet principal entraîne la suppression des objets qu'ils possèdent.*

Nous pouvons même raisonner que les attributs d'une entité sont une agrégation du moment qu'on peut dire : une entité est faite de ses attributs...

Les relations de type n-m (n>1, m>1) sont complexes. Elles sont souvent transformées soit en deux associations sémantiquement différentes de type (n-1) ou en créant une **entité associative**. Celle-ci hérite les clés des deux entités concernées et traduit une relation ternaire.

En UML, Les propriétés multi-valuées sont autorisées bien qu'il soit généralement préférable de les raffiner. Le rôle de l'entité associative peut être joué par l'attribut de lien (link attribute) en UML ou par une classe d'association.

Récapitulons quelques équivalences principales discutées jusque là entre les diagrammes E-R et de classes dans le Tableau 4.7.

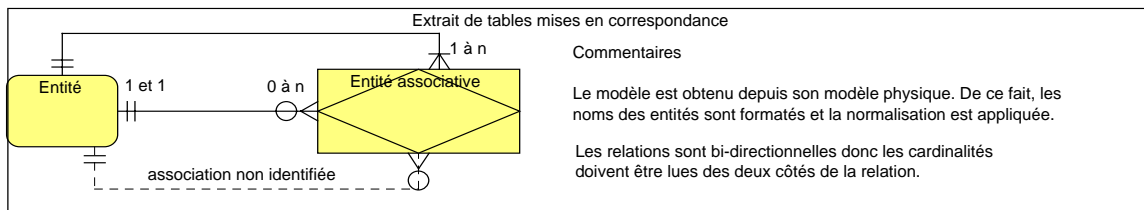
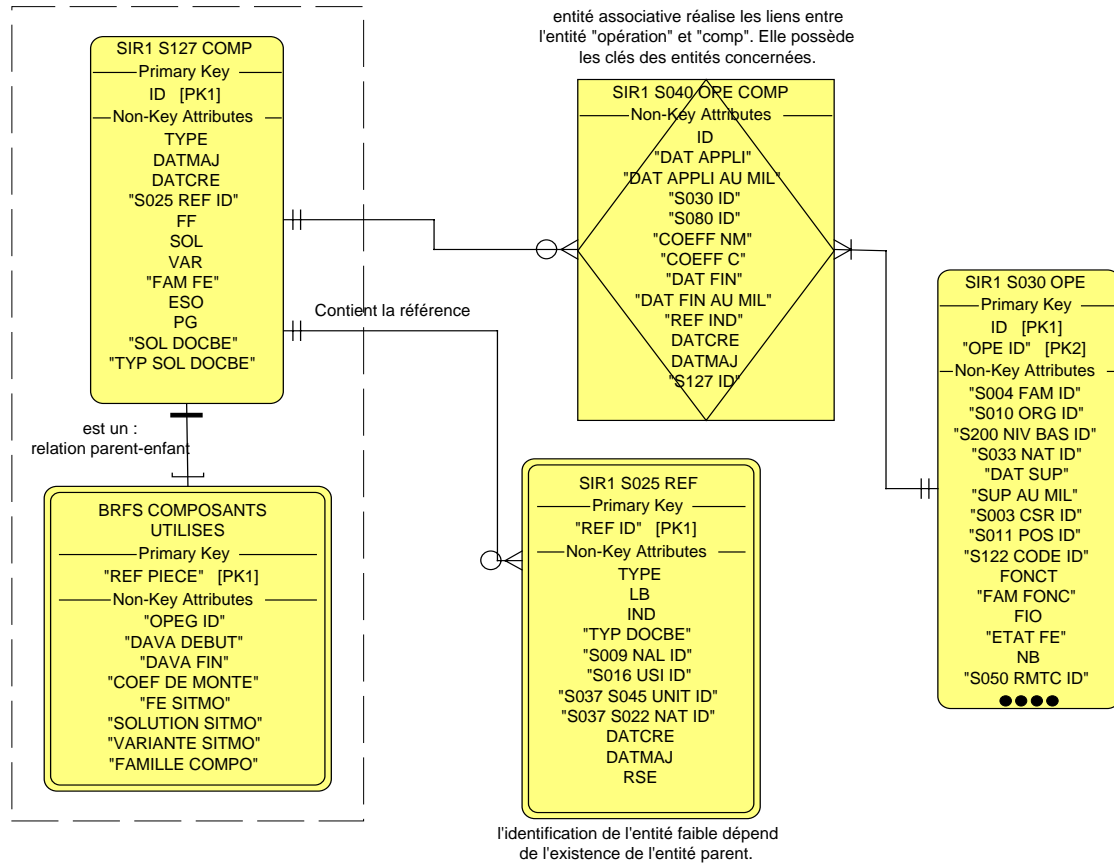


Figure 4.21 Fragment de modèle logique de données

Model Object - Table - 700_classe

Name: 700_classe

Columns: Table Info, Indexes, Symbol

	Name	PK	FK	Allow Null	Unique	Data Type	Q...	Default	Domain	Alt
1	classe_id	✓				CHAR	10			
2	model_number	✓				VARCHAR	10			
3	graphique			✓		character	6			
4	lib_vignette				✓	character	50			
5	vue					character	1			
6	lig_matrix					NUMERIC	15			
7	col_matrix					NUMERIC	15			
8	min_occurrence					REAL	5	0		

Insert Delete Define Choices...

Model Object - Constraint - <Unnamed>

Name:

Relation Line Info: Foreign Keys Symbol

Text Description: boîte de dialogue pour documenter les liens entre tables appelées contraintes

Line Style & Notation: Identifying Optional Parent

To Cardinality: Zero, One or Many

Referential Integrity: Parent Update: Cascade, Parent Delete: Restrict, Child Insert: Restrict, Child Update: Restrict

Figure 4.22 Tables et liens dans un modèle physique de donnée

E-R	UML
Diagramme E-R (schéma d'occurrences d'entités)	Diagramme de classes (diagramme d'objets)
Entité (occurrence d'une entité)	Classe (instance)
Association/relation (occurrence d'association)	Association (lien)
Cardinalité - Nom de rôle (verbe)	Multiplicité – Nom de rôle (nom)
Entité associative	Attribut de lien ou classe d'association
Entité faible	Composition forte (losange plein)
Notion de clé (identifiant)	Implicite, voire inutile dans les BdD objet
Normalisation	Pas d'équivalence pour le diagramme de classes
Procédures et triggers	Opération - Méthode – OCL
Types de données	Trois types : primitive, structure et énumération

Tableau 4.7 Quelques équivalences : E-R et UML

Processus de rétroingénierie

La transformation bidirectionnelle (du schéma de BdD au modèle logique et vice-versa) aujourd'hui est assurée automatiquement par de nombreux AGL, notamment celui que nous employons [POP00] avec pour avantage la possibilité de personnaliser chaque étape.

- De la BdD au modèle physique : l'import des SIT existants se fait selon plusieurs options en choisissant le SGBD, les éléments à importer etc. (partie gauche de la Figure 4.23).
- Du modèle physique au modèle logique : l'AGL propose une série d'ajustements mineurs, liés entre autres aux formatage des noms des tables et des champs. La partie droite de la Figure 4.23 montre l'organisation des modèles de données dans l'AGL.
- Du modèle logique au diagramme de classes : cette étape est facultative car nous pouvons entamer l'analyse sur les modèles logiques, voire physiques.

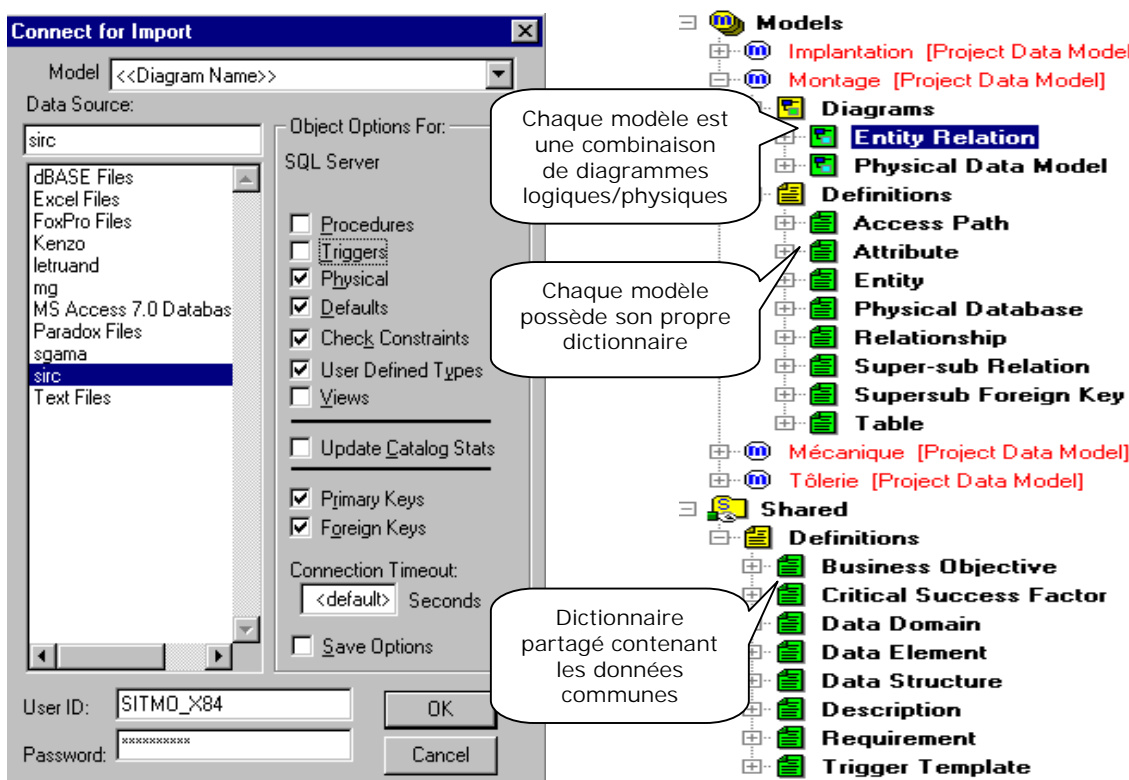


Figure 4.23 Import des bases de données existantes

Analyse des similarités et correspondances analogiques

Les deux méthodes que nous voulons adapter ici sont définies comme suit [CAU01] :

L'analyse des similarités, basée sur la comparaison d'éléments syntaxiques de systèmes existants, permet de concevoir des composants du même domaine que celui des éléments étudiés.

La *recherche d'analogies*, basée sur la comparaison des liens entre éléments et sur leur organisation, permet de concevoir des composants indépendants d'un domaine. Elle est réalisée à partir de l'étude des systèmes de domaines d'application divers.

Ces deux méthodes, rappelons-le, sont employées dans le domaine de la réutilisation en ingénierie des SI, pour identifier et fabriquer des composants réutilisables. Toutes deux, se rejoignent dans l'étude des modèles de spécification (schémas conceptuels, modèles physiques...) comme produits existants.

Pour notre contexte, nous essayons d'adapter simplement quelques *principes de comparaison* utilisés dans ces méthodes afin de *dégager les objets métiers communs et de discerner les duplications* présentes dans les SIT existants. Les deux méthodes emploient en effet des métriques et automatisent ces processus ; ce qui s'avère compliqué voire inutile vis-à-vis de la finalité de notre diagnostic.

Analyse des similarités

La comparaison s'effectue entre éléments devant provenir de différents systèmes d'un même domaine. A titre d'illustration, appliquons-la sur deux BdD alimentant respectivement les modeleurs de gammes du métier de montage et de la mécanique.

L'analyse des similarités met en œuvre deux mécanismes : la *recherche de similarités et la généralisation d'éléments similaires* [CAS92]. Nous les adaptons comme suit :

- **Nettoyage des schémas**

Les schémas sont d'abord épurés pour conserver uniquement les fragments comportant les entités candidates à la réutilisation. Les résultats du business model peuvent considérablement aider le processus de nettoyage. A gauche de la Figure 4.24 sont illustrées ces quelques entités candidates (pour alléger la figure, les attributs et les liens entre entités sont cachés).

- **Généralisation d'entités réutilisables**

La comparaison porte sur la nature des entités (fondamentale, faible, parent-enfant...), leur nombre et leur sémantique. Les attributs de chaque entité ainsi que les relations qu'elle possède avec les autres sont également étudiés. La partie droite de la Figure 4.24 montre un schéma E-R simplifié des objets métiers communs dégagés.

Recherche d'analogies

La correspondance analogique peut s'appliquer sur des systèmes de domaines différents pour produire des composants suffisamment génériques et indépendants. Prenons un exemple lié à l'expression de la diversité. Pour cela, récupérons les schémas E-R de SIT opérationnels appartenant à la direction mécanique et à la direction de l'ingénierie véhicule (montage et tôlerie).

La comparaison d'éléments dans cette méthode repose sur la satisfaction de trois types de contraintes : structurelle, sémantique et pragmatique [CAU01]. Essayons de les expliquer à travers des exemples :

- La similarité sémantique assure que les structures de connaissances en correspondance ont une signification similaire. Deux éléments sont dits similaires s'ils sont des instances

d'une même structure générique. Ainsi nous avons comparé le langage de description de la diversité des moteurs (employé dans la direction mécanique) à celui du véhicule (employé à la direction ingénierie véhicule). A travers les instances des tables décrivant ces diversités (partie gauche de la Figure 4.25), ces deux langages peuvent bien être représentés à l'aide d'une seule structure générique, présentée sur la partie droite de la Figure 4.25.

- La consistance structurelle privilégie les analogies qui maximisent les correspondances entre les relations des deux domaines. Afin de rationaliser le concept de la diversité process (voir chapitre 1 pour plus de détails sur cette problématique), nous nous appuyons sur la diversité produit exprimée par l'affectation d'une pièce à son cas d'emploi. Autrement, la diversité produit est traitée en sachant quelle pièce monter pour telle diversité véhicule. De même, une opération (d'assemblage par exemple) est réalisée avec des modes et des temps opératoires différents selon les caractéristiques du véhicule, ce que l'on appelle diversité process. Sans nous arrêter à l'affectation des opérations à la diversité véhicule, les outillages (de presses, de robots...) peuvent également être changés pour répondre à une diversité véhicule (c'est un cas rare dans la réalité) ; il s'agit du cas d'emploi de l'outillage. Enfin, il est également utile de connaître le cas d'emploi d'un poste ; cela est surtout valable pour les lignes non flexibles.
- Les contraintes pragmatiques impliquent que les éléments mis en correspondance répondent au but de l'ensemble des systèmes. Par exemple, les entités génériques produites doivent encapsuler (par leurs attributs et comportements) toutes les caractéristiques des composants qu'elles viennent de remplacer.

La Figure 4.25 illustre ainsi l'abstraction de plusieurs composants pour définir un problème général qui est le traitement de la diversité produit/process.

Les principes de comparaison de l'analyse des similarités et de la correspondance analogique sont ainsi appliqués pour générer les objets communs liés :

- Au concept d'engagement : celui-ci est plus populaire en montage dans l'activité d'équilibrage des lignes que dans les autres métiers. Mais il se trouve qu'il véhicule le même principe appelé par ailleurs hypothèse process, hypothèse ligne etc. L'engagement a donc pour finalité de fixer les paramètres des performances du système de production (cadence, production journalière...) en vue d'évaluer les solutions process en fonction de ces paramètres.
- Au principe d'évolution des objets produit/process (notion de versionnement), des dossiers (et documents) accompagnant le projet véhicule...
- A la description structurelle des sites, notamment l'unification de la notion du poste de travail.
- Aux ressources du manufacturing : nous retrouvons par exemple l'objet "emballage" dans SIGNE, dans les SIT d'implantation, de préparation mais aussi dans de nombreuses BdD opérationnelles servant en exploitation. Ces tables faisant référence à l'emballage sont indépendantes et documentent l'emballage en fonction des besoins métier donc nous les jugeons redondantes.

Autant d'exemples qui montrent que finalement le problème crucial dans l'ingénierie des SIT est la redondance des objets communs et la prolifération des sémantiques de ces objets. L'Encart 4.2 se focalise justement sur la question de la duplication à travers l'évolution de l'architecture des SIT en entreprise. La standardisation de ces concepts est présentée dans la partie suivante (VI.2).

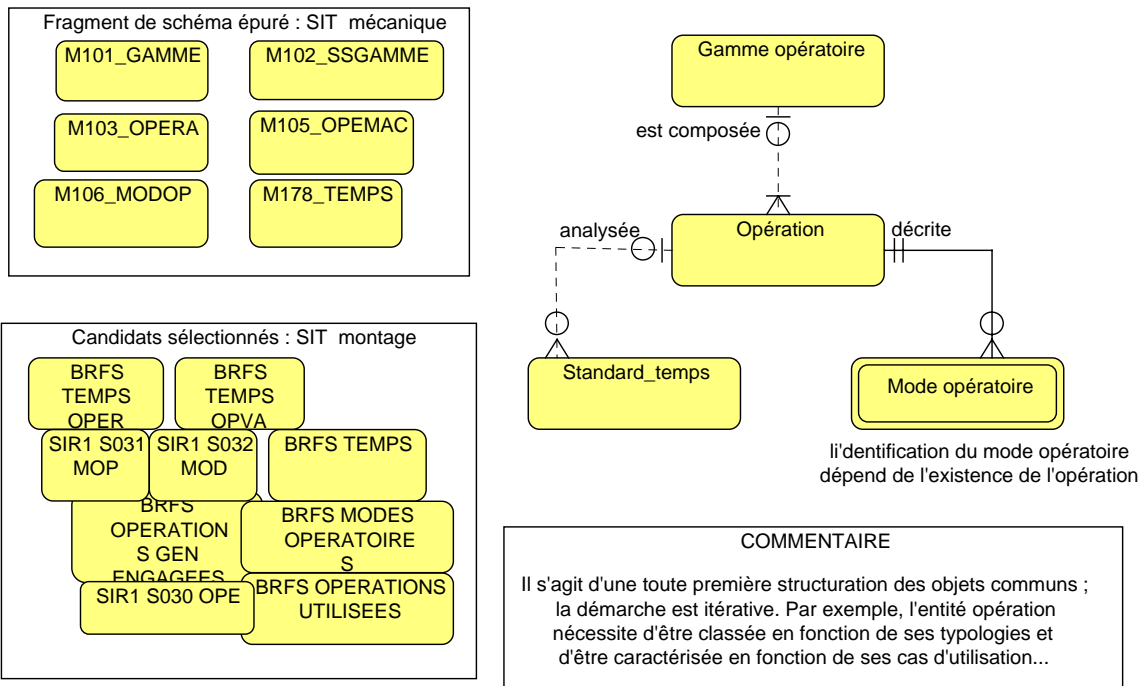


Figure 4.24 Analyse des similarités : généralisation de candidats réutilisables

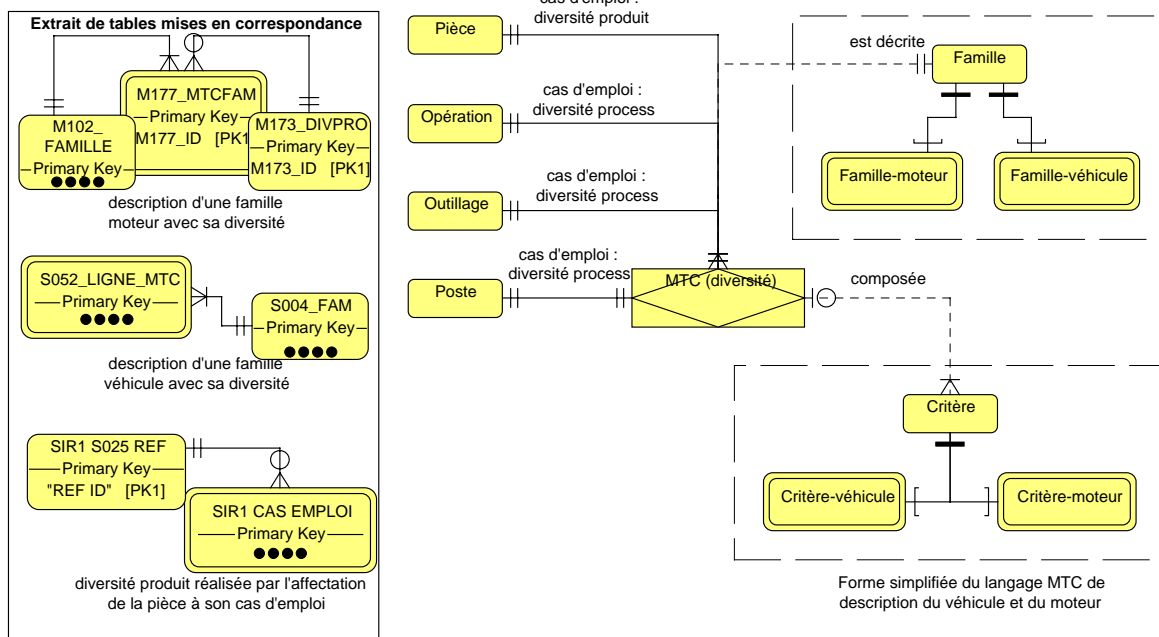


Figure 4.25 Correspondances analogiques : traitement de la diversité produit/process

Encart 4.2 Architecture des SIT et duplication des objets communs

Entities should not be multiplied unnecessarily. William of Ockham (1285-1347)

L'architecture des SIT opérationnels en entreprise a évolué progressivement depuis les systèmes à base de fichiers, tournés vers les applications métier, pour s'orienter vers les BdD comme technologie d'implémentation. La Figure 4.26 montre cette transition. Les systèmes à base de fichiers conventionnels sont faciles à développer ; ils sont destinés à des utilisations locales et répondent aux besoins ponctuels d'un service donné en entreprise. De plus, ils peuvent être optimisés -pour l'accès à l'application- et spécifiés en fonction des caractéristiques de l'application. Cependant, ils sont rarement optimisés pour une utilisation partagée entre applications. L'inconvénient principal des systèmes à base de fichiers est que l'information est dupliquée à cause de cette couverture verticale.

Or, la simple duplication d'une donnée dans deux fichiers implique la duplication :

- Des résultats utilisant cette donnée.
- Des endroits de stockage : sachant que les serveurs et autres supports sont aussi coûteux que leur maintenance.
- Des problèmes d'intégrité : c'est-à-dire, les tables peuvent afficher différentes valeurs pour le même item.

Un autre inconvénient majeur de ces systèmes est lié à leur inflexibilité. Ainsi, chaque nouveau besoin requiert une restructuration des tables concernées. Ces dernières, fortement couplées aux applications qu'elles alimentent (voir aussi Figure 4.14), vont conditionner ces applications. Et réciproquement, chaque évolution d'une application exige des changements au niveau des fichiers.

La technologie de BdD vise à réduire ces deux problèmes majeurs -la redondance et l'inflexibilité. Si une BdD est bien structurée, alors il est possible d'extraire des informations composites en combinant les données à l'aide de langages de BdD (SQL, QBE...). Se recentrer autour d'une "méga" BdD est pratiquement impossible à cause du coût de sa mise en œuvre -en temps et en argent. Néanmoins, les technologies ERP par exemple ont bien cette intention mais leur présence dans les entreprises n'a pas empêché la prolifération des BdD. A vrai dire, dans chaque entreprise, différents types de BdD peuvent exister (Figure 4.26). Les **BdD opérationnelles** (transactionnelles) sont développées pour supporter jour pour jour les activités métier. Ces BdD viennent remplacer progressivement les systèmes à base de fichiers. L'accès direct à ces BdD est généralement limité aux administrateurs de BdD en entreprise. L'utilisateur final accède plutôt aux applications métier (legacy systems pour les anglo-saxons, terme défini au chapitre 1).

La technologie **Data Warehouse** stocke normalement les données extraites des BdD opérationnelles et des fichiers conventionnels (à l'aide des techniques de "data mining"). L'analyse de ces données se fait à l'aide des langages de programmation de la quatrième génération et les outils de type "decision support tool". La technologie de BdD réseaux permet aux utilisateurs et aux différents services de développer des **BdD personnelles**. Les données peuvent être importées depuis tous ces supports cités. Les BdD personnelles sont souvent implémentées dans les technologies pour PC : Access de MS, FOXPRO...

Les SGBD gèrent l'architecture des BdD, en assurant la création, le contrôle, l'accès etc. Ce sont des logiciels achetés directement à des vendeurs comme Oracle, Microsoft, Sysbase, IBM... Les SGBD stockent généralement la donnée de l'utilisateur plus la méta donnée. Celle-ci est la structure définissant les données (entités, relations, règles de gestion...). Grâce aux nouvelles technologies à base du web, les BdD sont accessibles par Internet.

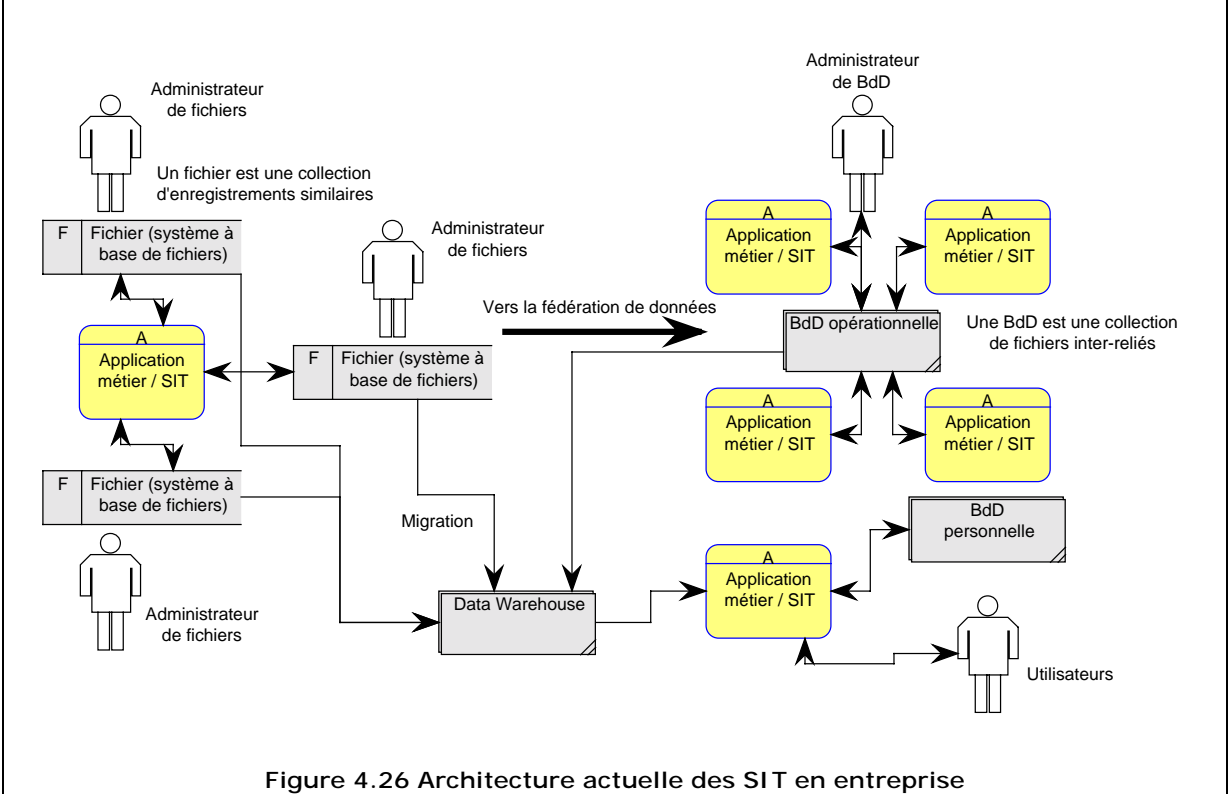
Enfin, terminons par la question suivante : **qu'est-ce qu'un bon modèle de donnée ?**

Pour cela, retenons ces critères universels ; un bon modèle doit être :

- **Simple**. Exemple : les entités ainsi que les attributs doivent porter des noms traduisant le monde réel.
- Essentiellement **non redondant**. Exemple : un attribut autre que la clé étrangère, doit décrire au plus une entité. Il est inutile de le multiplier dans les autres entités.

- **Flexible et adaptable** : plus le modèle est indépendant des applications qu'il doit alimenter, plus le caractère de flexibilité lui est attribué. Cependant, peut-on concevoir une BdD qui doit s'adapter facilement aux nouveaux besoins que l'on ne sait même pas prédire ?

La réponse à cette question est sans doute liée à la structuration même des objets métier : enjeu des paragraphes suivants.



VI.2 Méta entités et classification des objets métier

La Figure 4.10, résultat de l'analyse des processus d'ingénierie de production, met en avant quatre principaux livrables –découpage du produit, liste des opérations, catalogue de ressources et implantation des ateliers de production. Il ressort de ces livrables quatre concepts que nous nommons méta entités : *produit*, *process*, *ressource* et *système industriel*. La littérature abondante en structuration de données techniques [BAI98; HAR99] a souvent discerné le triptyque classique -produit, process et ressource- (voir chapitre 2, IV.4). Cependant, les définitions qui lui sont associées diffèrent d'un projet de recherche à l'autre. La rationalisation de ces concepts est sans doute l'un des objectifs de nombreuses normes récapitulées dans [MIC00]. La norme MANDATE [CUT00] s'est ainsi focalisée sur le concept de ressource alors que PSL [GRU00] vient la compléter avec un langage de spécification du process. En rajoutant la quatrième entité, nous assurons la complétude du domaine de l'ingénierie de production.

Le regroupement des objets métier -jusqu'ici rencontrés et qui vont s'enrichir au fur et à mesure de l'élaboration du data model- dans ces méta entités est d'un apport significatif. Le premier enjeu est de rationaliser les objets du domaine de l'ingénierie de production, en leur attribuant des définitions fixes tout en les consolidant à la culture Renault. D'autre part, cette première organisation de nos objets métier va aider considérablement leur structuration. Nous illustrons ces méta entités sur la Figure 4.27 que nous détaillons dans ce qui suit.

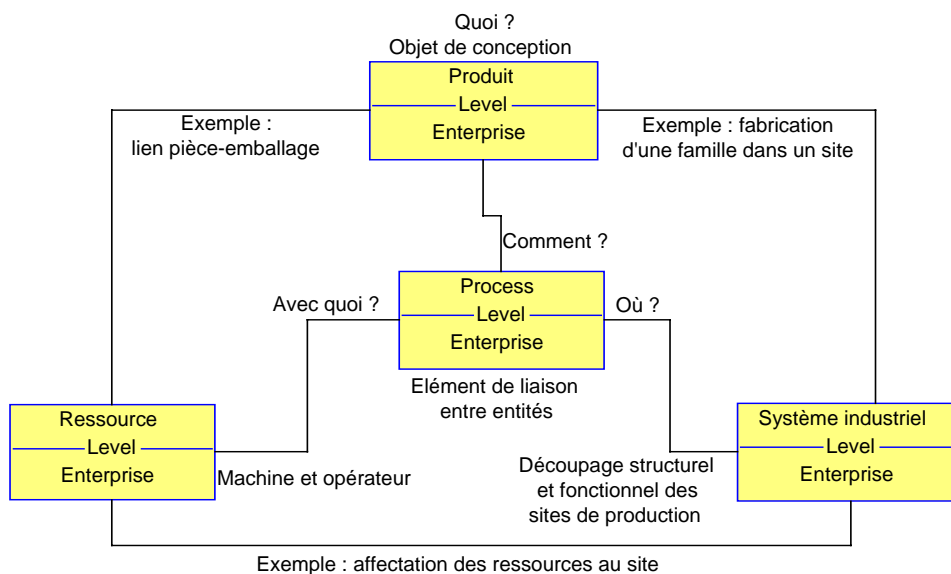


Figure 4.27 Les quatre méta entités du data model : liens centrés process

Produit

Comparés aux concepts propres à l'ingénierie de production, les définitions du produit sont relativement stables. Nous retenons ici que les concepts intervenant directement à la construction de notre référentiel. Ils sont empruntés à des normes et à des terminologies employées chez Renault. Comme nous l'avons convenu aussi, nous intégrons la nomenclature de produit définie dans SIGNE et le langage de description de la diversité du véhicule MTC (voir chapitre 1, II.4 pour plus de détails). En guise d'illustration, décrivons quelques-uns de ces objets.

Famille véhicule : ensemble de produits résultant d'une conception de base commune, mais comportant des types différents à l'intérieur desquels existent des versions et des variantes. Chaque famille véhicule est composée de plusieurs types et peut être classée dans une gamme –segment- de véhicules (classification propre au constructeur). Exemple : la famille véhicule "Mégane" codée X64, est composée de plusieurs types : J64 (Scénic), B64 (Berline 5 portes), K64 (Break) etc. Ces types de familles véhicules peuvent être développées en projet simultanément ou séquentiellement. Les conceptions incrémentales acquièrent généralement de nouveaux codes (exemple : l'ancienne Laguna : X56, la nouvelle variante s'appelle X74).

Découpage Référentiel Fonctionnel (DRF) du produit : il est, rappelons-le, à la base de la nomenclature SIGNE, fédératrice à toute famille véhicule et pour tout type de projet. Bien entendu, un tel référentiel est mis à jour régulièrement (avec de nouvelles fonctions) et mis à la disposition des projets véhicules ultérieurs.

Fonction : chaque véhicule est découpé fonctionnellement selon les critères des métiers du produit (moteur, soubassement, châssis ...). Ces fonctions, extraites du DRF, sont regroupées pour former des compositions récursives dont chaque nœud peut être un GFE¹⁰ ou une fonction élémentaire ; ce que l'on appelle communément *découpage métier*. Celui-ci peut être différent d'un projet à l'autre. A chaque fonction élémentaire sont associées des variantes et des options. Pour une variante est associée une solution technique, identification d'une

¹⁰ Groupe de Fonctions Élémentaires : sont assurées par des directions différentes orientées produit. Ainsi la superstructure est affaire de la DICAP (Direction Ingénierie Caisse Assemblée Peinte), l'habillage de la DIEC (Direction Ingénierie Equipement Carrosserie) etc.

réponse technique. Exemple : fonction projecteur, variantes : phare gauche, droit..., solution fournie par un fournisseur comme Valeo...

Pièce Générique (PG) : A chaque combinaison (fonction, variante, solution) correspond une pièce générique. Une fois la PG spécifiée se voit matérialisée en pièce physique.

Pièce physique : est appelée aussi "Part Number" par certains SGDT ou référence de pièce par le jargon Renault. Pour une PG, plusieurs références de pièces peuvent exister. Elles peuvent se distinguer par le nom du fournisseur, la couleur...

Assemblé : le véhicule peut être décrit aussi structurellement. Les compositions récursives de PG forment des assemblés.

Géométrie : partie ou totalité d'une pièce, caractérisée par une épaisseur constante. Une pièce n'ayant pas une épaisseur constante est découpée en géométries.

Nous verrons plus tard comment nous allons adapter SIGNE et l'ensemble de ces concepts dans des structures de données cohérentes.

Process

Le terme process a déjà été introduit dans les chapitres précédents. Les définitions suivantes sont principalement tirées de la norme PSL [GRU00] qui intègre à son tour les concepts standards issus d'autres normes [MIC00].

En toute généralité, le process est un ensemble d'opérations, d'actions, de procédés agissant sur un milieu extérieur (matières premières, constituants, documents, personnes...) pour en changer les caractéristiques. Le process a pour objectif de donner au produit ou à la personne, les caractéristiques de l'état final à partir des caractéristiques de l'état initial.

Ces caractéristiques peuvent être :

- Physico-chimiques (processus de transformation) : opérations d'usinage, de peinture, de maintenance...
- Géographiques (processus de déplacement, de transport) : déplacement d'un opérateur sur une ligne de production, convoyage d'une pièce, ...
- Psychologiques (processus de formation, de vente) : formation de l'opérateur à une technique de montage, lecture d'une fiche documentaire au poste...

La recherche des caractéristiques (états stables) à acquérir permet de mettre en place *une architecture du process*. Celle-ci, décrite plus tard avec une structure de donnée, peut jouer l'élément de liaison avec les autres méta entités comme l'illustre la Figure 4.27.

La recherche des *procédés* permettant d'acquérir ces états conduit, dans un contexte technico-économique donné, à déterminer l'ordre d'acquisition de ces mêmes états (*gamme opératoire*). Cela amène inévitablement un certain nombre d'états fugaces. Ceux-ci doivent être remis en cause afin de se rapprocher du juste nécessaire qui correspond aux seuls états stables et donc au processus idéal (on peut penser aux réengagements des lignes).

Examinons quelques objets déclinés jusqu'ici :

Processus primaire et processus secondaire : définis auparavant (III.4), les processus primaire et secondaire sont appelés respectivement par le jargon Renault métier et filière. Exemple : le processus primaire logistique peut être partitionné en conditionnement, évacuation de déchets, approvisionnement, réception et stockage de pièces etc.

Opération générique : description macroscopique et fonctionnelle d'un process. C'est l'expression dont nous avons convenu pour remplacer le terme *processus élémentaire*, issu de la décomposition des processus métier. Cet objet, résumant la finalité du process sans

spécifier son procédé, peut être employé très tôt dans le projet et se voit éclaté par la suite en opérations détaillées regroupées en gammes opératoires.

Gamme Opératoire (process plan) : une séquence d'opérations ordonnées permettant de traduire une opération générique.

Opération : l'entité la plus élémentaire que l'on ne décompose plus. Elle est documentée en fonction de ses cas d'utilisation : analysée avec un temps opératoire, engagée au poste, affectée à une ressource... Elle est décrite à l'aide d'un mode opératoire.

Mode opératoire (Operating procedure) : paquetage d'informations requises habilitées à réaliser l'opération. Le mode opératoire peut être standardisé ou laissé à la liberté du préparateur. Il est recommandé de le décrire avec un verbe d'action précis.

Pour fixer ces termes, prenons un exemple général. L'opération générique d'assemblage peut être décrite avec une ou plusieurs gammes opératoires. Une gamme peut être la séquence d'opérations agissant sur la pièce : chargement – mise en place – fixation – évacuation.

Il est possible d'obtenir autant de gammes en modifiant par exemple la technique d'assemblage (fixation, raccordement, serrage, encollage...). En fonction de la diversité du véhicule, chacune de ces opérations peut être documentée différemment avec un mode opératoire, des temps opératoires et des ressources requises.

Ressource

Nous adaptions ce que l'on a trouvé dans la littérature et principalement les concepts de MANDATE [CUT00] pour définir la ressource appartenant à notre univers de discours.

La ressource est tout périphérique, outil ou moyen qui contribue à la réalisation de biens ou des services du domaine de l'ingénierie de production. Une ressource est caractérisée par des propriétés ou des comportements mesurables tels que : le rendement, la fiabilité, l'engagement, la consommation etc. Ainsi, nous regroupons dans les ressources :

- Les machines (presses pour emboutir les pièces, équipements de manutention et de logistique...)
- Les outillages qui peuvent être des composants des gros équipements (comme pince qui appartient à un robot).
- Les programmes des machines à commandes numériques qui accompagnent par exemple les robots...
- Les effectifs impliqués directement dans le process qui sont la main d'œuvre directe. Ils sont appelés opérateurs ou agents de production. Ils sont répartis sur les lignes de production et dans les unités de stockages et de préparation (bord de chaîne, zones de préparation, pickings...). Ici, les responsables ayant pour rôle de définition ou d'approbation du process même se trouvant dans les surfaces des usines ne sont pas considérés comme ressources.

Les principaux objets pouvant alors organiser ces ressources sont :

Moyen générique : description macroscopique et fonctionnelle de la ressource sans spécification de sa structure, de son comportement, de son fournisseur... Sa granularité est au même titre que celle de l'opération générique définie dans le process. Un moyen générique peut se traduire plus tard en opérateur humain comme en équipement.

Opérateur : est documenté en fonction de l'activité d'ingénierie. Par exemple, la faisabilité ergonomique au poste requiert la connaissance du champ de vision et la taille de l'opérateur alors que l'activité d'équilibrage des lignes doit mesurer l'engagement de l'opérateur (en %),

les longueurs de ses déplacements etc. En revanche, des informations liées à la carrière d'un opérateur ou à ses heures de travail concernent plutôt les activités des ressources humaines.

Équipement : est une ressource physique. Elle se voit attribuée une référence et un nom de fournisseur. Elle est documentée en fonction de ses cas d'utilisation : implantée sur une surface de l'atelier, chiffrée avec des paramètres de consommation et de coût etc.

Outillage : généralement, la structure d'un équipement peut être décomposable. L'outillage est le composant le plus élémentaire. Le plus souvent, les outillages sont en contact direct avec la pièce donc leur emploi dépend fortement de la diversité du véhicule.

Système industriel

Indépendamment des entités précédentes, nous voulons regrouper les objets caractérisant les sites de fabrication pour la raison suivante : les concepteurs du process en avant-projet sont amenés à prendre des décisions liées directement aux sites de production -nouveaux ou existants. Les choix peuvent porter autant sur les paramètres d'implantation que sur les paramètres stratégique-économiques du projet :

- Répartition des surfaces des ateliers, définition des schémas de flux des bâtiments, prise en compte des installations existantes quand l'usine est déjà conçue...
- Définition de la maquette d'effectifs, des technologies d'investissements qui dépendent fortement du pays d'implantation...
- Définition de la cadence du site et la diversité véhicule qui peuvent changer au cours d'un projet en fonction des demandes du marché par exemple.

Il existe de nombreux concepts enclins à des confusions comme la définition du poste.

Poste (de travail) : nous convenons d'appeler poste l'espace où sont affectées les opérations avec les ressources associées. La répartition de celles-ci est conditionnée par :

- Le *temps de cycle* qui caractérise le poste. Le temps de cycle est fixe pour tous les postes d'une ligne et pour un engagement donné.
- Les caractéristiques d'implantation : sa position géographique, sa surface occupée...

A ce titre, un poste n'est ni une ressource ni un process mais un concept du système industriel. Les postes sont regroupés pour former des *tronçons* répartis par *pas de chaîne* (en montage), des *zones* et des *unit* (en tôlerie) etc. En fonction de l'activité de l'atelier, on parle de lignes de découpe de flancs, d'emboutissage, de ferrage, de montage...

Enfin, nous arrivons au terme de ce regroupement en méta entités. Ce travail introduit la notion de paquetage et le besoin de structuration des objets métiers ; ce que l'on va examiner dans ce qui suit.

VII Structuration et raffinement des objets métier

La structuration des objets métier s'effectue en plusieurs étapes contiguës et suivant un processus itératif.

Nous partons de l'organisation de ces objets en paquetages (VII.1) et des premiers diagrammes de classes construits sur la base des résultats précédents (VII.2). Les étapes suivantes se focalisent sur la construction des nomenclatures des objets métier (VII.3), des liens entre nomenclatures (VII.4) et des vues des objets métier (VII.5). Tout ce travail de structuration repose sur les techniques objet, notamment sur les solutions de patterns. Une conclusion générale (VIII) clôt ce chapitre où nous récapitulons l'apport de notre travail ainsi que les ouvertures de celui-ci.

VII.1 Organisation en paquetages

A vrai dire, la notion de paquetage est perçue dès le début de la construction du référentiel ; l'AGL offre la possibilité d'organiser les diagrammes ayant la même préoccupation -logique, physique, contextuelle...- dans des paquetages (voir par exemple la Figure 4.23). Dans ce cas, les paquetages jouent le même rôle que les cabinets.

A présent, introduisons une autre vision de paquetage¹¹ grâce à laquelle l'objet se distingue de l'E-R. En effet, la première technique pense le système comme une collection de paquetages alors que la seconde pense le système comme une collection d'entités, de granularité trop fine pour pouvoir prendre en compte aisément des modèles de grande taille comme le nôtre.

Un paquetage doit contenir des items sémantiquement proches, liés entre eux pour former un tout cohérent et suffisamment indépendant des autres composants. Ces items peuvent être des classes d'objets, des cas d'utilisation, d'autres paquetages, voire des diagrammes.

Le regroupement des objets métiers en paquetages a pour objectif de faciliter la maintenance et l'évolution du data model, tant au niveau des objets manipulés que des règles de gestion correspondantes.

UML ne propose pas un diagramme propre de paquetages mais il est possible de le construire à l'aide du diagramme de classes. La Figure 4.28 montre une version initiale d'un diagramme de paquetages construit sur la base des principes suivants.

Critères de découpage

Un bon découpage doit tenir compte de plusieurs critères. Ces critères constituent le fondement même du concept de paquetage, à savoir : les paquetages sont un moyen de structuration, d'organisation des acteurs du projet, de gestion des configurations et un moyen de réutilisation [KET98].

Sans entrer dans les débats de cette définition parfois controversée, dans les faits, ces critères sont simples à comprendre et à mettre en œuvre. Particulièrement dans notre cas, car l'analyse faite jusqu'ici facilite cette démarche. Nous retrouvons ainsi des paquetages :

- Regroupant les méta entités : produit, process, ressource et système industriel.
- Décrivant les types d'objets métiers : document, graphique...
- Résultant des cas d'utilisation étudiés : engagement, répartition des surfaces...
- Classant les objets par niveaux de granularités : niveau macroscopique et microscopique des objets métier...

Relations entre paquetages

La contenance stricte : un paquetage contient des paquetages ; une imbrication qui constitue une hiérarchie.

L'utilisation : relation la plus courante, se traduit avec le lien de dépendance du diagramme de classes. Ce lien résume toutes les relations qui existent entre les éléments de modélisation des paquetages respectifs ; elle est unique. Le lien de dépendance traduit qu'un changement dans un paquetage cible implique des modifications au niveau du paquetage source.

Par ailleurs, il n'est pas interdit d'employer d'autres types de relations communément connues entre classes (héritage, "Realizes", ...).

Cependant, lorsque la finalité du modèle objet est de développer une application logicielle, des règles plus rigides doivent être respectées comme :

¹¹ Package: a general purpose mechanism for organizing elements into groups. Packages may be nested within other packages [OMG99].

- Eviter les couplages forts entre paquetages surtout les dépendances circulaires, difficiles à faire évoluer.
- Une classe appartenant à un paquetage ne peut être la propriété d'un autre paquetage ; elle peut être référencée seulement dans un autre paquetage.

En résumé, les paquetages doivent être :

- Homogènes vis-à-vis des finalités (correspondant à des attentes uniques des acteurs).
- Homogènes vis-à-vis des évolutions.
- Homogènes vis-à-vis de la couverture organisationnelle et géographique.
- Homogènes vis-à-vis du cycle de vie des objets contenus.

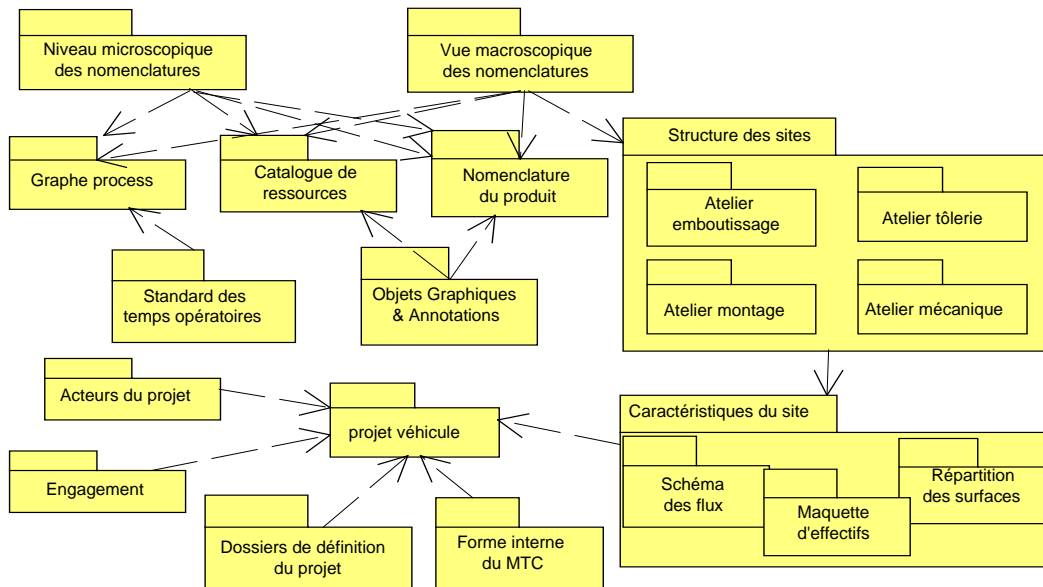


Figure 4.28 Diagramme initial des paquetages

VII.2 Des objets potentiels aux classes d'objets persistantes

Comme c'est mentionné sur la Figure 4.20, la construction des diagrammes de classes repose sur les résultats d'analyse du domaine et de rétroingénierie. Poursuivons la structuration par :

- La transformation en diagrammes de classes des modèles E-R issus des techniques de rétroingénierie et jugés pertinents (Figure 4.24 et Figure 4.25).
- L'examen des objets potentiels issus de l'analyse du domaine (Tableau 4.6) afin de dégager les classes d'objets pertinentes.
- Le regroupement des classes d'objets obtenues dans des paquetages (Figure 4.28). Cette démarche nous permet d'élaborer notre data model en plusieurs diagrammes, tout en formant une structure de graphe hiérarchisée.

A titre d'illustration, choisissons le paquetage "nomenclature produit". Nous devons y regrouper le DRF de SIGNE mais aussi tous les objets liés au produit. Cela nous amène à construire des sous paquetages liés entre eux, pouvant traduire par exemple les vues des métiers process sur le produit. Par exemple, un préparateur en montage raisonne en termes de fonctions (l'habillage du véhicule) alors que le préparateur en tôlerie conçoit des assemblés de pièces (la caisse du véhicule étant décrite structurellement). La Figure 4.29 est un diagramme de classes initial représentant ce dernier cas. Il est construit suivant les étapes ci-après, généralement communes à plusieurs méthodes objet et proposant des astuces valables pour tout le processus de structuration [OMG99] :

- Identifier les associations et les multiplicités qui gouvernent ces associations. Rappelons qu'une association entre classes d'objets traduit ce qu'une classe a besoin de savoir sur une autre. La multiplicité, valeur définie dans chaque côté de l'association, détermine combien d'instances de la classe peuvent être associées à une instance de l'autre classe.
- Déterminer les généralisations et les agrégations. Le lien d'héritage est plus délicat à spécifier et à implémenter que l'agrégation. Pour ce faire, il existe des principes simples expliquant quand il faut employer l'héritage à l'agrégation ; le principe de substitution de Liskov¹² en est un exemple.
- Représenter, si nécessaire, la dynamique de ces classes d'objets à l'aide du diagramme des états ou d'activité. Ainsi, à la classe "assemblé" est rattaché un diagramme des états présenté sur la Figure 4.30. Il décrit les états que la classe "assemblé" se voit attribuée depuis sa création jusqu'à sa validation et cela tout au long de son cycle de vie en ingénierie¹³.
- Décliner la nature de la classe -abstraite, dérivée, normalisée. Une classe est abstraite si elle ne possède pas d'instances. Un élément est dérivé s'il est déduit depuis un autre élément. La normalisation est liée à l'E-R. Par ailleurs, toutes nos classes sont en principe persistantes¹⁴ (par opposition à transitoires) ; ce sont elles qui se matérialisent dans les Bdd.
- Documenter les compartiments de chaque classe : attributs, opérations, contraintes, responsabilités... Les contraintes peuvent être rédigées en langage naturel ; l'emploi du langage OCL n'est pas impératif au début de l'analyse.
- Introduire les contraintes également sur les associations et les multiplicités pour une meilleure précision.
- Insérer des notes de commentaires pour une meilleure clarté des diagrammes.

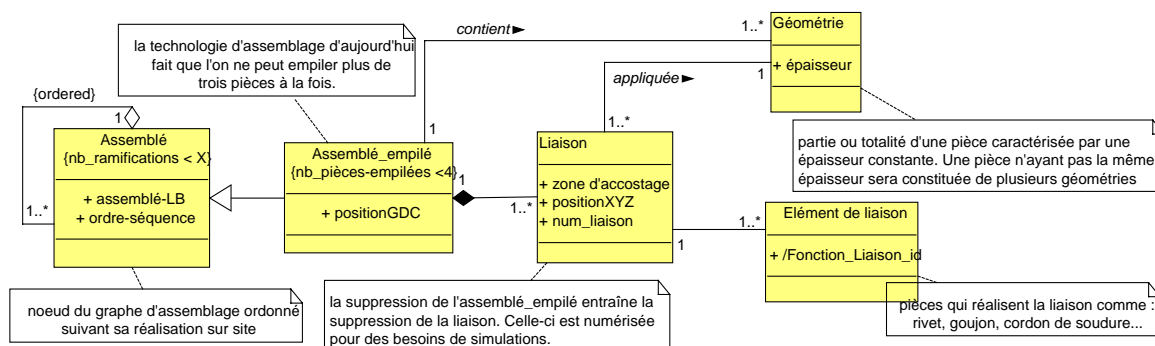


Figure 4.29 Assemblé de pièces vu par le préparateur tôlerie. Diagramme initial de classes

¹² Cf. www.objectmentor.com/publications/lsp.pdf ou <http://c2.com/cgi/wiki?LiskovSubstitutionPrinciple>. Il existe par ailleurs de nombreux principes OO ayant un apport sur le développement d'applications logicielles ; les plus populaires sont récapitulés dans <http://www.design-up.com/design/principesoo/index.html>.

¹³ Notons qu'en exploitation, l'assemblé subit des traitements différents de ceux subis en ingénierie. Il peut être livré (en tant que pièce détachée), vendu (aux concessionnaires et garages), stocké (dans les magasins des usines), réparé (durant un processus de maintenance)...

¹⁴ La persistance des objets se justifie par le besoin de les mémoriser pour une utilisation ultérieure : opérationnelle ou statique. Cependant, la gestion de la persistance de ces objets s'inscrit dans les détails de conception technique car elle dépend de la technologie d'implémentation et sort ainsi du périmètre de notre référentiel. Néanmoins, nous pouvons noter que cette tâche est l'affaire du spécialiste (administrateur de Bdd par exemple).

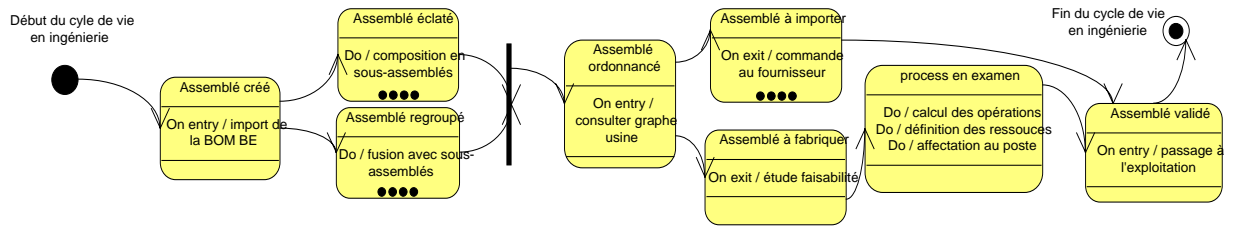


Figure 4.30 Cycle de vie de la classe "assemblé" en ingénierie. Diagramme des états

Cependant, à ce niveau de structuration, le diagramme n'est pas affiné. Il n'est pas non plus associé aux autres objets produit/process appartenant au même paquetage ou non ; c'est justement l'enjeu de ce qui suit.

VII.3 Les nomenclatures

Plus généralement, une nomenclature –notion déjà introduite- est une composition récursive d'items (fonctions, organes, géométries...) formant une structure de graphe hiérarchique, comme l'explique la définition suivante : *"A listing of the parts used in a manufacturing process along with the relation of one product to another insofar as assembly of the final product is concerned. The bill of material is a classical recursive structure"* [INM97].

Nous avons également exprimé notre intention de structurer ainsi nos quatre méta-entités. Pour ce faire, nous adaptions quelques patrons abstraits et d'analyse décrits dans l'encart 3.1. Pour alléger les figures suivantes, les compartiments des classes ainsi que le détail des associations sont cachés ; notamment la sémantique des liens de composition spécifiée avec OCL qui conditionne la nature de la nomenclature (de type arbre, forêt, ...).

Produit

Nous construisons la nomenclature de produit, représentée sur la Figure 4.31.

Le découpage SIGNE est intégré. Il propose une description générique valable pour toute famille véhicule à l'aide de la hiérarchie de classes : niveau supérieur, niveau intermédiaire et niveau de base. Cette structuration est l'instanciation du patron "Organisation Hierarchy" de M. Fowler [FOW00]. Exemple, "équipements extérieurs" (niveau supérieur) est composé de trois niveaux intermédiaires : enjoliveurs, accessoires et protection. Le niveau intermédiaire "protection" est divisé en cinq niveaux de base : protection dessous de caisse, bouclier avant, antiprojections, bouclier arrière et protection latérale et arrière.

Dans la réalité quotidienne, les fonctions se substituent à la classe niveau de base. De ce fait, nous rendons celle-ci abstraite pour encapsuler la composition récursive des fonctions relatives à une famille véhicule. Cette structure est l'instanciation du patron abstrait "Composite" de E. Gamma [GAM95].

Suivant la variante et la solution de chaque fonction élémentaire, il correspond une ou plusieurs pièces génériques. La pièce générique peut être un assemblé de pièces ou une (référence de) pièce élémentaire¹⁵. Cette structure est également représentée à l'aide du patron "Composite"[GAM95].

¹⁵ Il faut toutefois noter qu'une pièce élémentaire peut être un assemblé aux yeux du fournisseur. Pour un constructeur qui importe environ 70% des pièces du véhicule, celui-ci ne doit pas démanteler les modules importés ; ils sont montés tels qu'ils sont et donc leur décomposition est inutile.

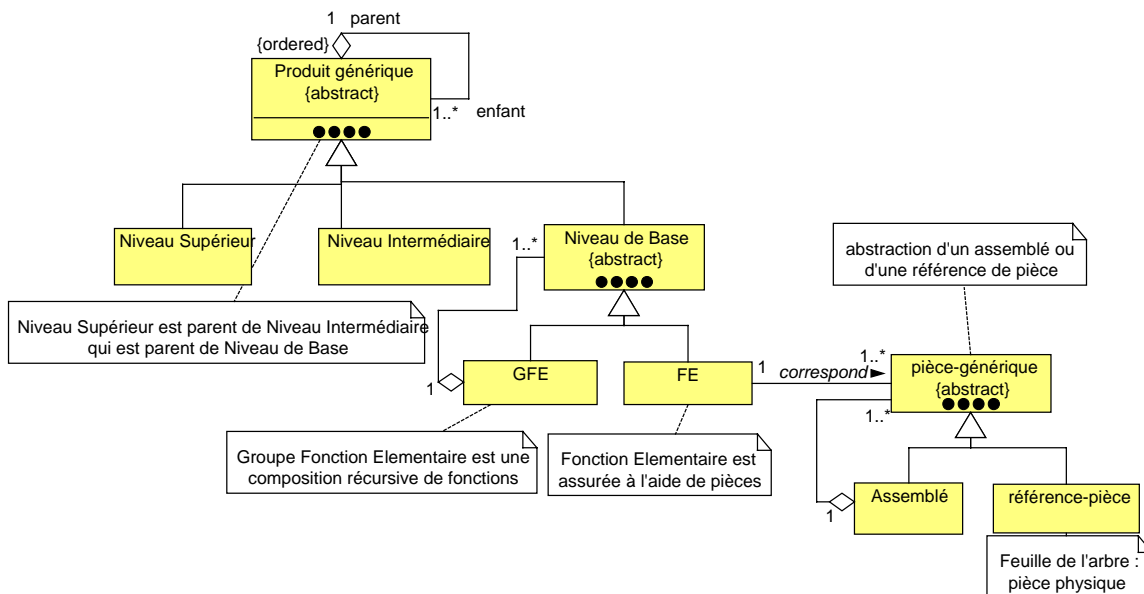


Figure 4.31 Composition fonctionnelle et structurelle du produit

Process

En s'inspirant des niveaux de la nomenclature produit (granularité et nombre), nous construisons une structure similaire pour le process caractérisée comme suit (Figure 4.32).

La classe abstraite "processus métier", divise le métier d'ingénierie de production en processus primaire, secondaire et élémentaire. Nous avons convenu d'appeler cette dernière classe opération générique (VI.2). Cette structure, reposant sur le patron "Organisation Hierarchy" de M. Fowler [FOW00], possède les mêmes rôle et niveau de granularité que sa partie homologue en produit.

Nous avons par ailleurs souligné que les instances de l'opération générique peuvent être substituées par celles des classes gamme opératoire et opération. Nous formons à l'aide de ces trois classes une structure composant-composite [GAM95]. La gamme opératoire est une composition récursive des opérations. La classe opération, étant la feuille de l'arbre, est documentée avec un mode opératoire, un standard des temps opératoires etc. La partie droite de la Figure 4.32 montre un exemple d'instanciation du process à travers cette structure.

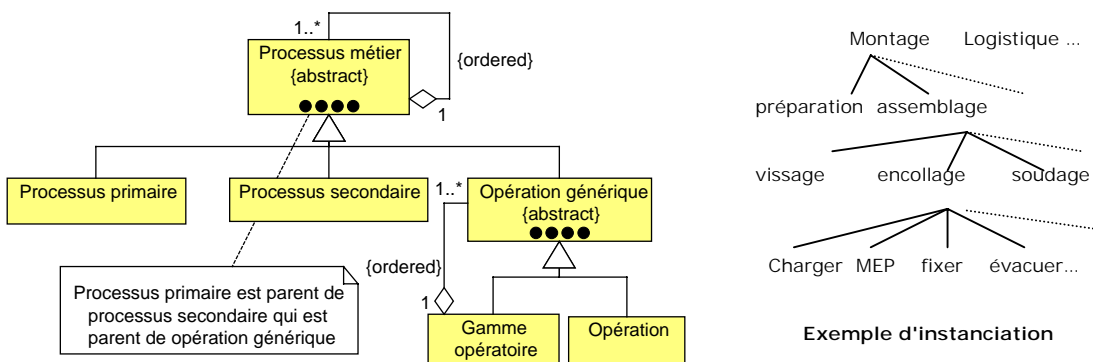


Figure 4.32 Graphe du process

Ressource

Partant des définitions liées à la ressource (VI.2), sa nomenclature peut être décrite à l'aide des composants suivants (Figure 4.33).

Nous rattachons à l'organisation hiérarchique des métiers d'ingénierie de production la classe abstraite moyen générique. Celle-ci forme, avec la classe composite équipement et la classe composant outillage, une structure récursive [GAM95]. De par le choix d'inclure les ressources humaines dans la nomenclature, la classe Opérateur se voit hériter de la classe moyen générique.

La partie droite de la Figure 4.33 est un exemple d'instanciation suivant cette structure.

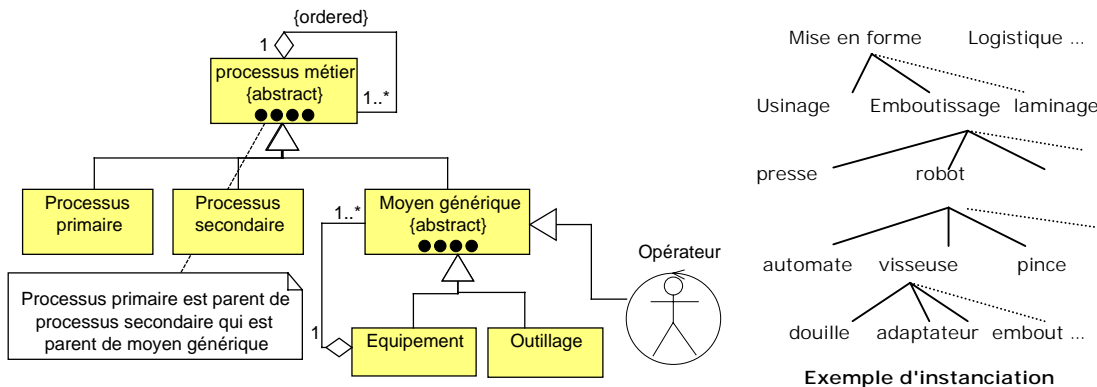


Figure 4.33 Nomenclature de ressources

Système industriel

La Figure 4.34 présente une structure générique du système industriel. La classe abstraite système industriel divise le site en ateliers en empruntant le patron "Organisation Hierarchy" [FOW00]. Nous distinguons différents types d'ateliers suivant ses activités. Les découpages des ateliers se distinguent par les propriétés et les comportements des classes les constituant.

Bien entendu, tout comme les trois précédentes structures, celle du système industriel ne présente que l'infrastructure du data model. Il reste cependant à déterminer les liens potentiels entre nomenclatures (VII.4) et à extraire les différentes vues de ces nomenclatures issues des métiers d'entreprise (VII.5).

VII.4 Liens entre nomenclatures

Les liens potentiels illustrés sur la Figure 4.27 introduisent la notion de granularité. Par ailleurs, le choix de fixation des niveaux des nomenclatures n'est pas arbitraire ; cela nous permet de réaliser des liens homogènes –en granularité- entre éléments appartenant à ces nomenclatures. Parcourons-les de haut en bas et identifions deux types de liens :

- Liens macroscopiques : ce sont les liens qui unissent les classes génériques ou abstraites, se trouvant dans les niveaux supérieurs des quatre structures. Ces classes sont relativement stables et peuvent être communes à tous les projets véhicule. Cependant, elles sont plutôt manipulées dans les phases avant-projet lorsqu'on ne dispose pas de données de détail. Il s'agit de lier les découpages génériques du produit (niveau supérieur, intermédiaire) à ceux du process/ressource (processus primaire, secondaire). Un tel couplage peut correspondre aux éléments supérieurs du découpage des sites de

production. Les classes abstraites descendantes sont également liées entre elles et par lesquelles sont déclinées les classes persistantes.

- Liens microscopiques : traduisent les liens entre les classes fondamentales se trouvant dans les bas niveaux des arborescences. La Figure 4.35 illustre un fragment de liens microscopiques existants entre les éléments des nomenclatures.

Dans les deux cas, il n'est pas nécessaire de surcharger les diagrammes de liens dérivés car il suffit de spécifier les mécanismes de navigation entre classes dans les éléments concernés.

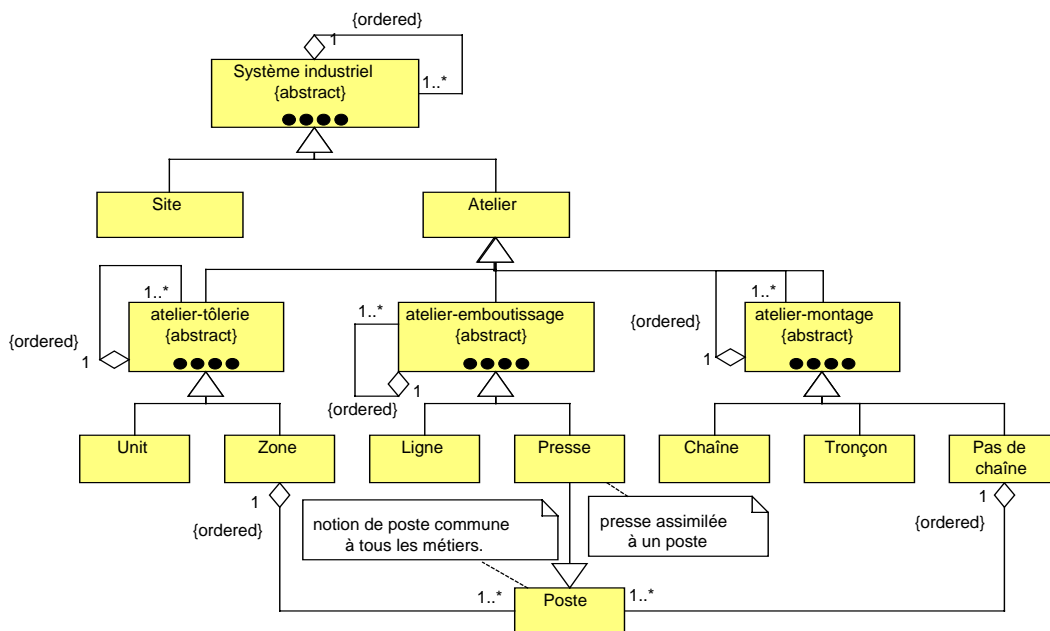


Figure 4.34 Découpage structurel du système industriel

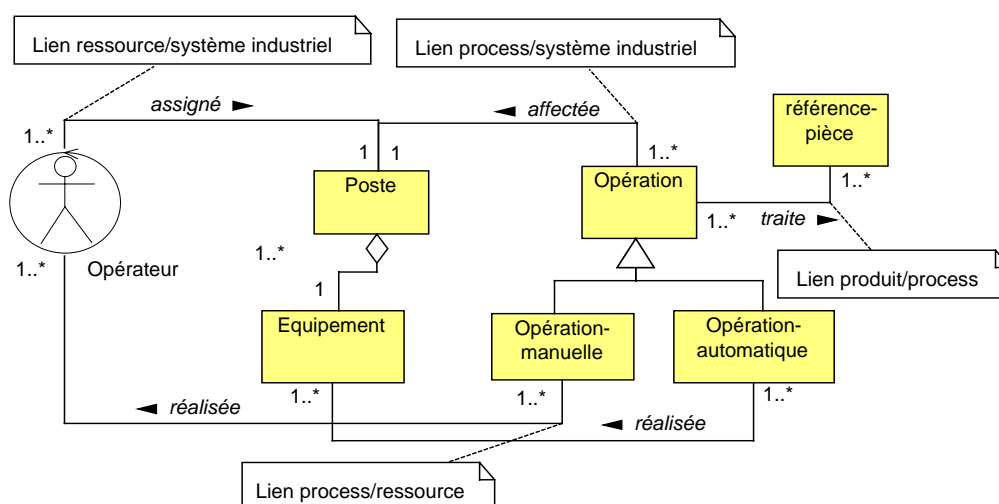


Figure 4.35 Liens entre éléments de nomenclatures : vue microscopique

VII.5 Vues des objets métiers

La notion de vue peut être définie comme "... a selective perception of the enterprise, which accentuate very much the special aspects and neglect others" [AIT98].

Concrètement, les cas d'utilisation métier peuvent être à l'origine de la vue. Ils sont issus des activités étudiées (préparation, implantation, engagement...) et génèrent de nouveaux objets. Ces objets sont répartis dans les paquetages illustrés sur la Figure 4.28. Les paquetages, tout en étant suffisamment autonomes, peuvent dépendre des autres paquetages ; ils complètent notamment l'infrastructure du data model basée sur les quatre nomenclatures –produit, process, ressource et système industriel. Présentons trois exemples.

Rôles des ressources en ingénierie de production

La ressource, en fonction de son niveau de granularité (moyen générique, équipement ou outillage), est documentée différemment. De plus elle est caractérisée par des propriétés (attributs) variables selon son rôle joué dans le domaine.

Ainsi par exemple, l'objet équipement possède des propriétés intrinsèques telles que son identifiant, ses composants (outillages), sa fonction principale etc. En revanche, suivant le processus métier dans lequel il est employé (chiffrage, implantation, simulation...), l'équipement se voit assigné dans un contexte particulier (hypothèses du projet véhicule), plusieurs rôles dépendants des activités d'ingénierie de production ; l'équipement peut être :

- Chiffré dans le cas d'études d'investissements.
- Numérisé à des fins des études de faisabilité du process.
- Implanté sur les sites, occupant ainsi des surfaces d'ateliers.
- Simulé en lui attribuant des propriétés sur la fiabilité, le rendement etc.

Cette expression de rôles de la ressource est représentée sur la Figure 4.36 en adaptant le patron "Rôle" de P. Coad [COA92].

Nous appliquons également ce dernier pour exprimer les rôles potentiels des autres objets métier : opérateur, opération, site etc. (voir les définitions des ces objets dans VI.2).

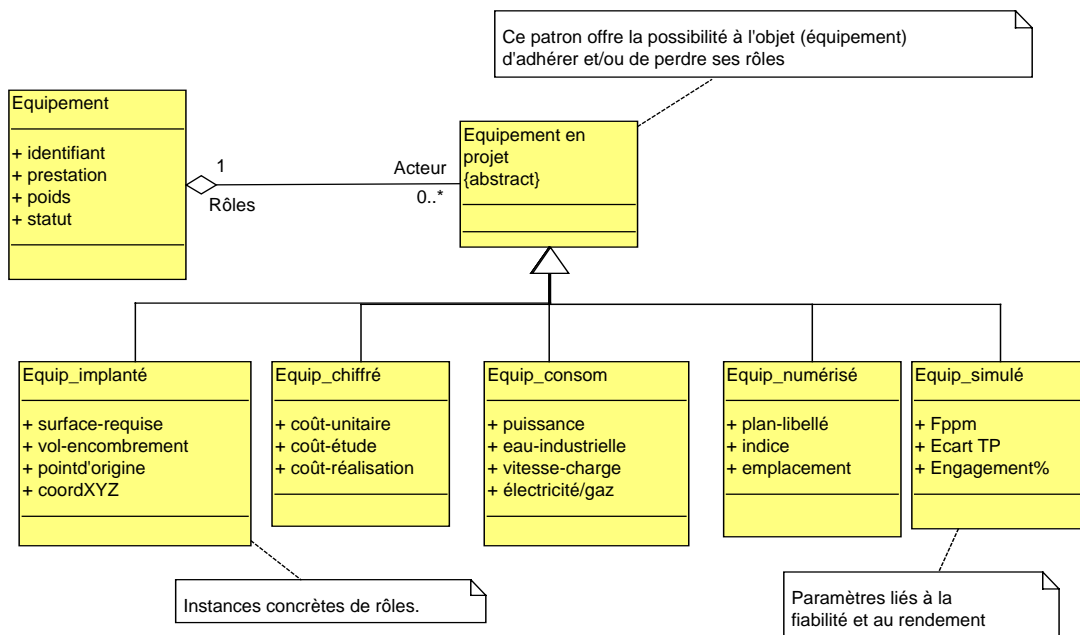


Figure 4.36 Rôles de la ressource en ingénierie de production

Structure fédératrice de l'objet document

Sans entrer dans le débat des typologies de documents, constatons seulement qu'il existe divers documents¹⁶ (CdC, fiches au postes de travail, résultats des essais et de contrôle, comptes-rendus...) se distinguant finalement de très peu en propriétés ou en comportements.

En effet, ces documents, souvent regroupés dans une imbrication de dossiers peuvent être :

- Gérés manuellement ou automatiquement à l'aide d'outils spécialisés : GED, SGDT...
- Créés en ingénierie, en exploitation ou évoluant au rythme de l'avancement du projet véhicule.
- Dépendants ou autonomes, tout en complétant la description du projet véhicule ou des nomenclatures.
- Pilotés par des propriétaires changeants ou non...

Tous ces paramètres, caractérisant l'objet document, peuvent être pris en compte à l'aide d'une seule représentation suffisamment générique et non encombrante. Elle est issue de l'adaptation du patron populaire "pattern document" [MUL99] comme l'illustre la Figure 4.38.

L'enjeu de cette structure est de retrouver le document contenant les informations recherchées. Elle fournit également une classification des documents à l'aide de la classe "TypeDocument". La classe "Copie" avec laquelle l'emplacement du document est identifié, est l'instance réelle de l'abstraction de "Document". Pour le versionnement, il lui est attribué aussi une jungle de définitions. Le mieux est de le décrire suivant la logique métier. Ainsi nous distinguons version mineure pour exprimer les révisions intermédiaires des documents et version majeure pour marquer le passage du document au jalon suivant. Enfin, un document peut ne pas concerner de sujets ; cela permet d'y accéder sans passer par l'index des sujets.

Au-delà de l'indexation et de la classification des documents, le rôle et la maturité de ces derniers peuvent être gérés à l'aide des systèmes de workflow. Nous nous arrêtons là, car le cas d'application décrit au chapitre 5, porte justement sur l'élaboration d'une application pour la gestion dynamique des documents.

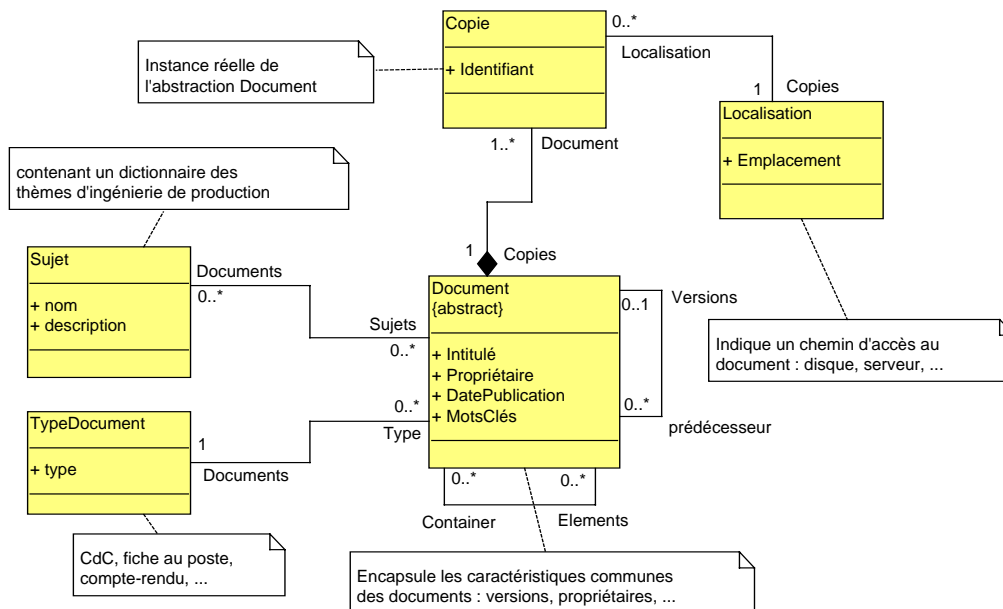


Figure 4.38 Structure générique du document

¹⁶ Nous avons toutefois choisi de structurer séparément les objets graphiques des autres documents. Les objets graphiques se distinguent en effet par leur forte dépendance par rapport aux quatre nomenclatures.

Nous voici au terme de l'élaboration du référentiel. Sans chercher à être exhaustifs dans la description, nous avons tenu à représenter les aspects pertinents d'analyse et de modélisation. Ces derniers concernent également les premières phases d'ingénierie des SIT en entreprise. Cependant cette dernière activité requiert des études supplémentaires que nous mentionnons dans la conclusion qui suit.

VIII Conclusion et ouvertures

Le référentiel obtenu est orienté expression des besoins en GDT du domaine de l'ingénierie de production. Il est de ce fait réutilisable en phase amont d'analyse, dans le développement de nouveaux SIT en entreprise.

Les modules composant le référentiel -business model et data model- sont construits à l'aide de diagrammes de spécification standards. Ces diagrammes sont suffisamment formels pour mieux représenter les besoins de la maîtrise d'ouvrage et les spécifications fonctionnelles de la maîtrise d'œuvre. En outre, ils sont suffisamment compréhensibles et facilitent par conséquent la communication entre ces principaux acteurs.

Le business model est constitué d'un ensemble de modèles -contextuels, opérationnels et modèles de transformation logique- conçus d'une façon contiguë et chronologiquement et ce, suivant les vues stratégique, organisationnelle, temporelle et fonctionnelle.

Ces différentes vues sont certes populaires dans le monde de l'entreprise modeling. Cependant, elles demeurent partiellement prises en compte dans les SI organisationnels. De plus, la cohérence entre ces vues n'y est pas complètement assurée.

La data model regroupe les objets métiers communs aux activités d'ingénierie de production. Ces objets métiers sont structurés et affinés pour devenir des objets persistants. La structure de données résultante est fédératrice ; elle est formée d'objets macroscopiques jusqu'alors inexistantes dans les SIT de l'entreprise, liés aux objets microscopiques, souvent redondants dans les SIT opérationnels et sujets à des sémantiques diverses.

Si les travaux de structuration des données techniques sont nombreux, ils se focalisent toutefois sur les domaines de l'ingénierie de produit ou de la production stricte. Le domaine de l'ingénierie de production est resté méconnu jusqu'à l'émergence, récemment, des outils d'IPAO.

L'élaboration du référentiel repose sur un guide méthodologique dont les atouts peuvent se résumer à :

- L'intégration des approches ascendante et descendante : traditionnellement, elles sont déployées l'une ou l'autre dans les développements des SIT. Nous avons alors montré la complémentarité des techniques appartenant à ces approches -la rétroingénierie et l'analyse des problèmes.
- L'interaction permanente entre le business model et le data model : grâce notamment aux modèles de transformation logique qui jouent le rôle de passerelle entre les deux composants. Ces modèles traduisent le processus d'extraction des objets métiers depuis l'analyse du domaine. Soulignons que ce processus a toujours été exécuté d'une façon informelle. Cette rationalisation, à notre avis, doit contribuer à l'intégration des deux composants du SI, à savoir : le SI organisationnel et le SI informatisé.
- L'utilisation du cadre de Zachman pour organiser les connaissances de l'entreprise : celui-ci nous a permis de rationaliser l'étape cruciale d'analyse du domaine où nous sommes confrontés à plusieurs métiers complexes d'ingénierie. En outre, il nous a aidé à gérer la

diversité des techniques employées : E-R, techniques de modélisation des processus et UML. Le cadre de Zachman a déjà fait ses preuves dans le monde anglo-saxon. Mais, à notre connaissance, il n'est pas encore familier pour la communauté francophone.

- La prise de décisions en respectant le triptyque : produit final attendu, intentions du projet et contraintes de celui-ci. Ainsi, les étapes du processus d'élaboration du référentiel sont guidées par ce triptyque plutôt flexible et convivial comparé aux règles rigides des méthodologies du domaine de l'ingénierie des SI.

Nous avons expliqué par ailleurs que notre cadre méthodologique peut s'appliquer aussi pour l'ingénierie même des SIT. Mais nous sommes conscients qu'il y a plusieurs éléments de conception qu'il faut prendre en compte très tôt. Très sommairement, citons quelques-uns que nous jugeons utiles d'aborder ici.

- **Prototypes et génération de BdD/codes**

La démarche itérative que nous préconisons peut se traduire également en effectuant des tests intermédiaires de prototypes. La fonction de génération automatique de schémas de BdD et de codes est assurée par la plupart des AGL. Cette fonction est d'autant plus intéressante lorsque le SIT informatisé est une BdD. Du code SQL peut être généré facilement et exporté vers un SGBD pour la compilation. Nous pouvons ainsi vérifier la cohérence et la qualité de certains composants du data model. Les composants du business model sont facilement simulés grâce au simulateur intégré à notre AGL. Cela dit, cette fonction est peu utile car nous ne prétendons pas faire de la ré-ingénierie des processus.

- **Transformations entre modèles objets et modèles relationnels**

La future génération de BdD est certainement tournée vers l'objet. Mais aujourd'hui, la seule technologie vraiment opérationnelle est le mode relationnel. Il n'existe pas non plus d'efforts de standardisation dans le domaine de la construction des schémas de BdD orientés objet (OO) ou hybrides : Objet-Relationnel (OR). En revanche, les transformations bi-directionnelles entre les diagrammes de classes et les modèles logiques de données peuvent s'effectuer à l'aide des AGL. Bien entendu, le sens de transformation le plus délicat est celui qui va des diagrammes de classes vers les schémas E-R ; ces derniers sont relativement pauvres pour encapsuler toutes les informations provenant des diagrammes de classes.

- **Distribution et réplication des données**

Les décisions liées à la distribution et à la réplication des données doivent être prises avant même la conception de la BdD/SIT. Chaque BdD doit être représentée avec son propre modèle logique. Dans la réalité, les SIT sont rarement centralisés (c'est-à-dire implémentés dans un seul serveur). Les technologies web, réseau et client/serveur permettent une distribution (horizontale ou verticale) des systèmes les rendant accessibles à travers le monde.

La réplication concerne la duplication physique des tables dans des localisations multiples. L'avantage de la réplication est que le trafic réseau peut être réduit. En revanche, la réplication complique les contraintes d'intégrité des données et requiert plus de capacité de stockage physique.

Par ailleurs, durant l'élaboration du référentiel, nous avons été confrontés à des choix de décisions difficiles. Citons deux d'entre eux sous formes d'interrogations :

- **Quel niveau de détail dans la modélisation ?**

Illustrons-le à l'aide d'un simple exemple. L'objet opérateur est renseigné grâce aux cas d'utilisation métier étudiés (simulation ergonomique, engagement, gestion documentaire...). Pour le premier cas, nous avons besoin de sa taille moyenne et de son champ de vision et pour le second, de son rendement -sans avoir besoin de connaître son sexe- et de la longueur de ses déplacements. Le troisième cas demande de connaître la langue homologuée caractérisant le site -mais pas la nationalité d'origine de l'opérateur- et le degré de formation de celui-ci -sans vouloir pour autant gérer sa carrière.

Ce cas est facile pour identifier les besoins en informations afin de renseigner un objet ; celui-ci est simple et les cas d'utilisation métier concernés sont relativement stables. Cela n'est souvent pas le cas et il devient alors difficile d'anticiper sur de tels besoins.

▪ **Quelles frontières fixer dans la fédération des données ?**

Ce problème n'est pas d'actualité mais il n'est toujours pas résolu malgré l'émergence des technologies de type Data Warehouse en entreprise. Si nous avons délimité par ailleurs notre domaine d'ingénierie de production, il demeure cependant difficile de dégager clairement les frontières de celui-ci à cause des nombreuses données communes avec les métiers connexes d'ingénierie produit/process.

Les réponses à ce type de questions sont vraisemblablement contextuelles, liées aux caractéristiques des projets. Le chapitre suivant démontre ce constat à travers un cas d'application industriel.

CHAPITRE CINQUIEME

Usages du référentiel du domaine

Le problème de Hume est le suivant : [...] sommes-nous justifiés à raisonner à partir de cas (répétés) dont nous avons l'expérience sur d'autres cas (les conclusions) dont nous n'avons pas l'expérience ? La réponse de Hume est : non, si grand que soit le nombre de répétitions.

*Karl R. POPPER (1902-1994). La connaissance objective, chapitre 1 -
La connaissance conjecturale : ma solution aux problèmes de l'induction.*

Pour ce dernier chapitre, nous sélectionnons un cas industriel qui nous a permis de rendre compte des usages potentiels de notre référentiel mais aussi des perspectives de notre travail de thèse tant au niveau du référentiel qu'au niveau du cadre méthodologique.

Nous introduisons d'abord le contexte de l'application industrielle (I), défini à l'aide des trois paramètres habituels : intention, produit final et contraintes. Nous décrivons par la suite l'étape de rédaction du CdC des besoins, basée sur l'adaptation du référentiel (II). Enfin, nous illustrons, à l'aide des copies d'écran du produit final obtenu, les fonctionnalités acquises (III). Nous terminons ce chapitre par une conclusion portant sur les difficultés rencontrées et les ouvertures possibles de l'application (IV).

I Contexte du cas industriel

Les constructeurs automobiles éprouvent le besoin de concevoir continuellement leurs systèmes industriels¹ suivant les orientations stratégiques et les singularités de chaque entreprise (marché, implantations, relations sociales, passé de la firme...).

Renault s'engage, peu de temps après son alliance avec Nissan, à reconcevoir son propre système industriel nommé SPR (Système de Production Renault). Celui-ci cherche à transposer les points forts de Nissan² sur Renault tout en conservant sa culture et ses atouts.

L'enjeu du SPR est de rendre plus performant l'ensemble du système industriel, en diminuant le poids des dysfonctionnements liés à la qualité, aux pannes, aux modifications de produits mal validées etc. Il rassemble tous les acteurs qui concourent à la fabrication (équipes de conception produit/process, acheteurs, fournisseurs...) en leur faisant partager les mêmes principes et les mêmes valeurs exprimés à travers les cibles du SPR.

La définition d'un nouveau système documentaire pour le process figure parmi les nombreuses actions engagées du SPR. Ce travail soulève plusieurs interrogations tant au niveau stratégique, organisationnel que technique. L'aspect technique constitue notre problématique que nous décrivons ci-après.

I.1 Intention et motivation : nouveau système documentaire

La documentation actuelle est dispersée, différente suivant les métiers, difficile à capitaliser et fournit un surplus d'informations. Elle manipule des données de détail qu'il n'est pas possible de décrire en phases amont d'ingénierie. L'élaboration d'un nouveau système documentaire a donc pour but de résoudre ces dysfonctionnements. De plus, il se veut être un support de communication adapté au contexte de l'internationalisation. A travers cette nouvelle documentation, des spécifications process justes et nécessaires devraient être transmises ; ce qui facilitera la capitalisation des expériences passées.

Il existe quatre types de documents³ représentés sommairement sur la [Figure 5.1](#) ~~Figure 5.4~~. Les trois premiers sont créés en ingénierie et ont un rôle de préconisations pour le fabricant. Celui-ci construit le dernier type de document en s'appuyant sur les recommandations de

¹ Citons comme systèmes : DC (Daimler Chrysler Operating Principles), Ford (Ford Production system), Nissan (Nissan Production Way), Toyota (Toyota Production System), Valeo (Valeo 5000)... Ces systèmes ne sont pas diffusés de par leur caractère confidentiel. Mais globalement, ils peuvent avoir les mêmes objectifs avec comme différence la priorisation de leurs principes. Ainsi, Daimler a fait de l'innovation l'un de ses quatre principes de base, Toyota met en avant la réduction des coûts. Quant à Renault, le facteur humain reste au cœur de ses préoccupations.

² Malgré une situation difficile pour Nissan, son système industriel est très performant. Les usines Nissan au Japon ont un niveau de performance comparable à celui de Toyota. L'usine Nissan de Smyrna est en tête aux USA et celle de Sunderland est la meilleure en Europe.

³ La signification des acronymes est Process Control Chart pour PCC, Feuille Opération Process pour les documents FOP A ou B et Feuille Opération Standard pour les documents FOS.

l'ingénierie. Le processus de capitalisation des expériences repose sur la comparaison du process réellement effectué par rapport à celui préconisé.

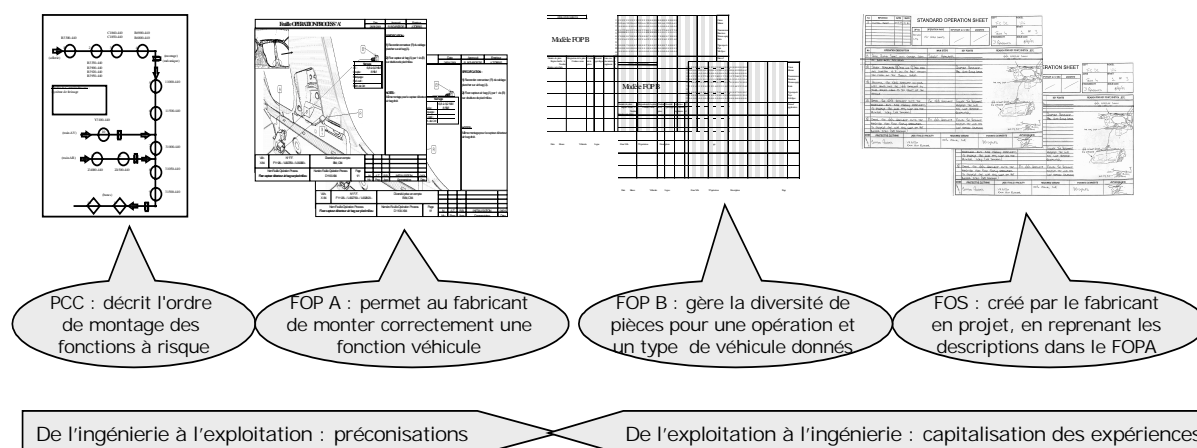


Figure 5.1 Nouveau système documentaire à concevoir

Les premières interrogations soulevées suite à ce projet sont :

- Quel est le nouveau contenu informationnel des documents ?
- Y a-t-il des liens potentiels entre ces documents ?
- Peut-on assurer une continuité entre l'ingénierie et l'exploitation à travers ce système documentaire ?

Ces questions sont bien entendu inter-reliées. Il faut sans doute commencer par l'investigation du contenu de chaque document afin d'y répondre.

I.2 Produit final attendu : application informatique de grande taille

Les intentions du projet conditionnent la nature du produit final à obtenir. Celui-ci doit être utilisé par une grande population qui participe de près ou de loin à la conception du système industriel. Ces participants selon qu'ils soient en ingénierie ou en exploitation, ont des enjeux et des profils différents. Par ailleurs, la quantité de documents à produire pour chaque projet véhicule –tout métier confondu- est estimée à des milliers : une centaine de PCC, environ 2000 de FOP A ou B et 2000*5 de FOS. Un tel système documentaire d'aussi grande taille nécessite une gestion plus rigoureuse, sous peine de provoquer des goulots d'étranglement.

Pour des raisons d'allègement mais aussi de confidentialité, nous ne détaillerons qu'un seul type de document : FOP A en montage. Sur la [Figure 5.2](#), le document FOP A - élaboré avec Excel de MS- montre une diversité d'informations produit/process qu'il faut également gérer.

Ces différents paramètres orientent notre choix de technologie vers les applications de GED (Gestion Electronique de Documents) devant être opérationnelles à grande échelle. Le terme générique GED désigne les SI intégrant les différents supports optiques et assurant les fonctions de stockage-archivage et de consultation-diffusion.

L'application GED est d'abord un progiciel dont la structure d'accueil⁴ est générique mais suffisamment ouverte pour la personnaliser suivant les clients utilisateurs. Les questions qui se posent immédiatement sont :

- Quelle structure d'accueil pour l'application ?
- Comment va-t-on organiser l'architecture des données dans la GED ?

Il est clair qu'avant de pouvoir y répondre, il faut recueillir l'ensemble des besoins et des contraintes environnementales que nous présentons ci-après.

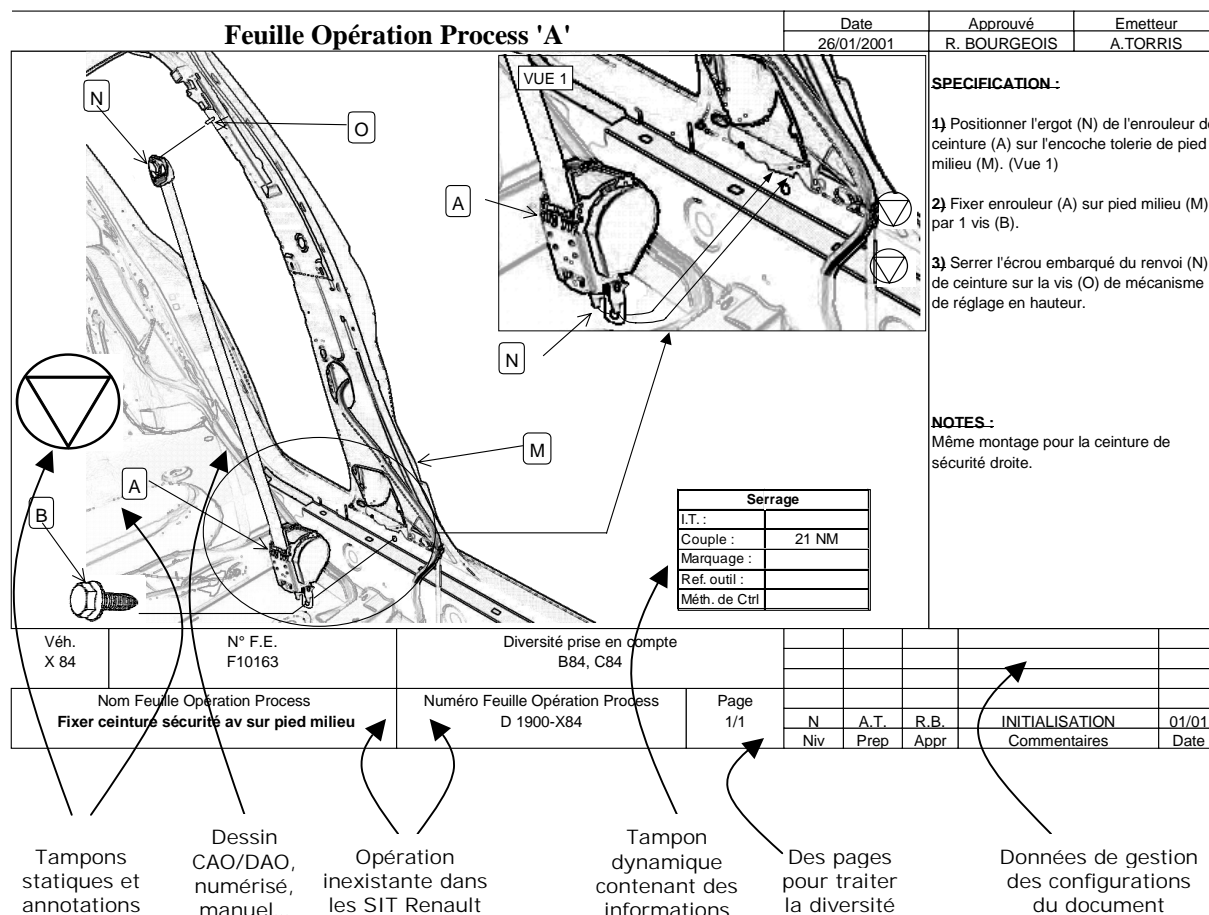


Figure 5.22 Contenu informationnel du document FOP A

I.3 Contraintes environnementales : intégration des SIT existants

En étudiant le contenu informationnel des documents (Figure 5.2), nous distinguons :

- De nouvelles données issues de l'application des principes du SPR : comme l'opération décrite dans le FOP A ; elle est macroscopique, ayant un identifiant et un mode opératoire différents de ceux caractérisant les opérations manipulées dans les SIT opérationnels.
- Des données appartenant à la nomenclature de produit Nissan : il s'agit du découpage du véhicule orienté produit/process appelé zone véhicule. La référence de chaque zone véhicule précède l'identifiant de l'opération macroscopique.

⁴ Nous convenons d'appeler structure d'accueil, l'environnement de l'application visible à l'utilisateur : le gestionnaire de cabinets se trouvant à gauche de toute application de GED, la liste des boutons et des icônes...

- Des données existantes comme : les fonctions du véhicule dans la GDT d'entreprise SIGNE, les opérations détaillées -manipulées dans les FOS- se trouvant dans les modeleurs de gammes, les outillages répartis dans plusieurs BdD etc.
- Des dessins géométriques de pièces pouvant provenir des outils de CAO/DAO. Mais, il arrive parfois que ces dessins soient conçus à la main, récupérés en scannant des photos numériques...
- Un ensemble d'annotations complétant le dessin ; ils peuvent être statiques (des symboles non changeants) ou dynamiques (lorsqu'ils véhiculent des informations changeantes).
- Données liées à la gestion de configuration du document. Nous devons savoir les versions du document, les propriétaires (approbateur et émetteur)...

En décrivant les informations contenues dans les documents, des contraintes se dégagent d'elles même. Résumons-les à l'intégration des SIT opérationnels mais aussi à la prise en compte de nouvelles données auparavant inexistantes. La satisfaction de ces contraintes passe nécessairement par la compréhension des éventuels liens entre ces différentes données dont certaines sont déjà structurées dans des paquetages (Figure 4.28). Mais beaucoup d'entre elles doivent être à nouveau étudiées, notamment la nomenclature de produit Nissan qu'il faut comparer à celle de Renault. Autant de questions qui seront résolues à l'élaboration du CdC de l'application.

II Adaptation du référentiel et CdC du projet

La rédaction du CdC du projet passe par plusieurs étapes. Nous commençons par sélectionner les paquetages du référentiel pertinents pour ce contexte industriel (II.1). Ces derniers sont réexaminés suivant les quelques nouveaux cas d'utilisation métier (II.2). Il devient alors possible de redéfinir une nouvelle structure de données pouvant supporter la création et la gestion dynamiques des documents SPR (II.3). Nous traitons en dernier la phase de mise en œuvre sous-traitée à deux sociétés de prestations (II.4).

II.1 Des paquetages aux composants : architecture générale des données

Le passage de la notion de paquetages en composants -toutes deux existantes dans UML-, permet de marquer le passage de la phase d'analyse à la conception. Les composants sont des groupements d'objets qui travaillent étroitement ensemble. Les composants peuvent être classés par leur type, définis à l'aide des stéréotypes : application, document, librairie, table etc. Dans la programmation objet, certains composants existent dans l'environnement de compilation, d'édition de liens ou d'exécution. Dans notre cas, ces composants représentent les modules de l'application que l'on va développer ; ils contiennent essentiellement les classes d'objets persistants qui vont se matérialiser dans la BdD permettant la création des documents SPR.

Tout comme les paquetages, les composants peuvent être liés entre eux avec :

- Le lien de contenance stricte, formant ainsi des imbrications de composants.
- Le lien de dépendance : un composant source dépend d'un composant cible si un changement dans ce dernier implique des modifications dans le composant source.
- Le lien "realizes" : ce lien, utilisé entre deux types de symboles (classes, composant, paquetage...), exprime que l'élément source supporte au moins toutes les opérations définies dans l'élément destination mais sans nécessairement supporter sa structure de données (attributs et liens). Ce lien est le plus souvent utilisé pour associer un composant à une classe d'interface.

La ~~Figure 5.3~~ ~~Figure 5.3~~ montre les principaux composants suivants :

- Documents SPR : tous les documents créés dans l'application peuvent être rangés dans des supports de stockage différents de ceux de leurs contenus. Ce composant regroupe les classes d'objets représentant ces documents mais aussi leurs comportements et les liens potentiels entre documents.
- Bibliothèque de composants graphiques et annotations : ce composant regroupe les classes décrivant les objets graphiques conçus à l'aide des logiciels de CAO/DAO ou d'autres moyens de numérisation. Un cas particulier de graphique s'appelle annotation ; il renseigne les dessins d'assemblés de pièces. Ces annotations sont représentées comme des symboles statiques ou dynamiques (appelés parfois tampons par le jargon Renault). Les objets graphiques sont appelés intelligents lorsqu'ils sont décrits à l'aide des définitions prises en compte dans le modèle de donnée. Ce dernier repose sur l'infrastructure définie à l'aide des nomenclatures suivantes.
- Nomenclatures : nous reprenons les paquetages décrivant les structures de données basées sur les quatre entités –produit, process, ressource et système industriel. Seuls les niveaux de nomenclatures servant à la création de documents y sont extraits. Depuis les nomenclatures, sont également extraits quelques cas d'utilisation métier représentés dans le composant suivant.
- Vues des objets métier : comme nous l'avons évoqué au chapitre 4, les cas d'utilisation métier peuvent être à l'origine de la vue obtenue des objets métier mais aussi de la création de nouveaux objets métier. Certaines vues sont donc déjà étudiées dans notre référentiel et d'autres méritent d'être examinées à nouveau ; ce que l'on va voir par la suite.

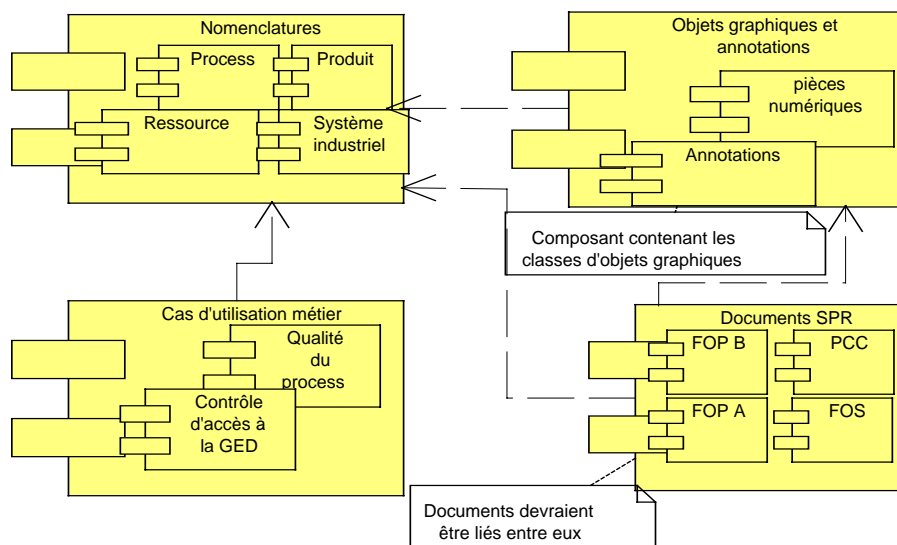


Figure 5.3 Architecture générale des données. Diagramme de composants

II.2 Traitement des procédures et des cas d'utilisation métier

Les procédures et les règles métier doivent être rationalisées afin de les automatiser à l'aide du système workflow intégré à la GED. Une des actions amont consiste à définir les droits des utilisateurs de l'application de la GED. La ~~Figure 5.4~~ ~~Figure 5.4~~ illustre ces droits à l'aide de la

matrice CRUD⁵ dont les colonnes désignent les profils (rôles) et les lignes représentent les fonctionnalités de la GED en ligne.

Fonctions de la GED	Rôles	Préparateur	Approbateur	Consultant	Administrateur	Fournisseur
Document FOP A		CRUD	RU	R	CRUD	R
Structure d'accueil générale		R	R	R	CRUD	R
Requêtes de recherches		RU	RU	R	CRUD	R
Edition de rapports		CRUD	CRUD	R	CRUD	R
Workflow		R	R	R	CRUD	R
Liens entre documents		CRUD	RUD	R	CRUD	R

Figure 5.4+ Contrôle d'accès et de modifications des utilisateurs. Matrice CRUD

Nous devons signaler par ailleurs que le composant décrivant les documents est aussi une extraction d'une vue métier. Dans ce composant nous pouvons rattacher à la classe document de type FOP A, un diagramme des états décrivant les transitions du document FOP A en respectant certaines procédures métier⁶, comme le montre la [Figure 5.5](#).

En examinant les documents SPR, de nouveaux cas d'utilisation métier apparaissent comme la qualité du process, activité non étudiée à l'élaboration de notre référentiel. La prise en compte de ces cas d'utilisation génère de nouvelles classes d'objets. Nous devons les rattacher à l'infrastructure du modèle de donnée basée sur les quatre nomenclatures que nous décrivons partiellement dans ce qui suit.

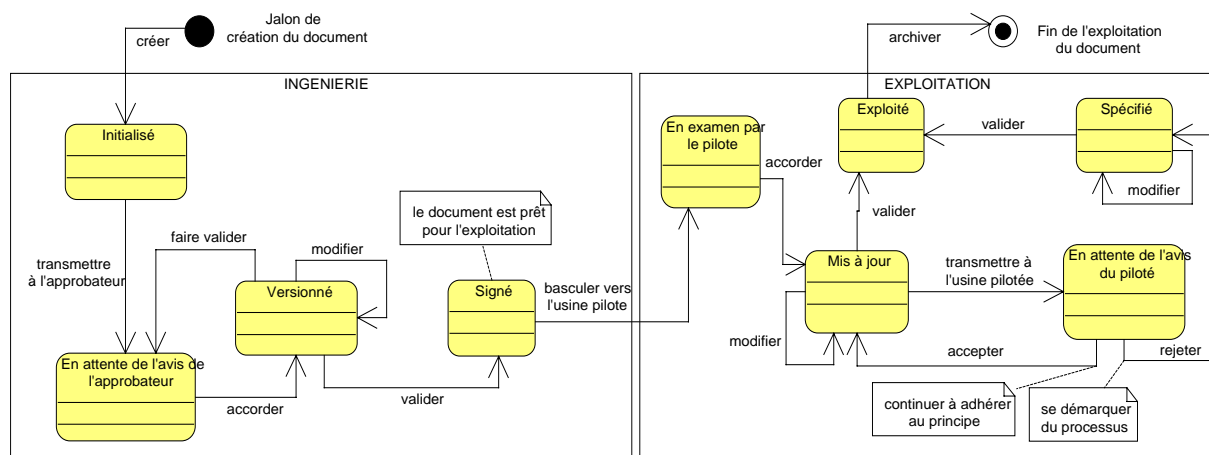


Figure 5.55 Cycle de vie du document FOP A. Diagramme des états

⁵ Acronyme de Create, Read, Update et Delete. La matrice CRUD a été présentée auparavant comme technique de modélisation.

⁶ Un exemple de ces procédures est le principe pilote/piloté. Pour des raisons de confidentialité, nous ne pouvons l'expliquer.

II.3 Structure des données : produit, process, ressource et système industriel

Par opposition aux gestions traditionnelles de documents, la gestion des données contenues dans les documents, permet leur création et mise à jour dynamiques.

Par ailleurs, la structuration des quatre entités dans des nomenclatures est étudiée lors de l'élaboration de notre référentiel. Cependant, l'examen des documents SPR ainsi que celui des cas d'utilisation métier liés au processus de gestion documentaire, nous incitent à réétudier à nouveau notre ancien référentiel. Ainsi, nous devons extraire depuis le référentiel, uniquement les composants intervenant directement à la construction des documents SPR. Nous retenons par exemple les paquetages décrivant le projet véhicule, la structure générique de l'objet document, les niveaux de bases des nomenclatures...

Nous devons également créer de nouveaux diagrammes de classes issus de nouveaux besoins (nouveaux types de documents avec leurs comportements spécifiques, vues liées à la qualité des opérations...). La [Figure 5.6](#) montre un fragment du modèle de donnée déployé et la position des objets de type "document".

La démarche de structuration des objets est semblable à celle décrite au chapitre 4 concernant l'affinement du modèle de donnée. L'étape suivante consiste à traduire le CdC du projet décrit jusque'ici et à mettre en œuvre l'application de GED.

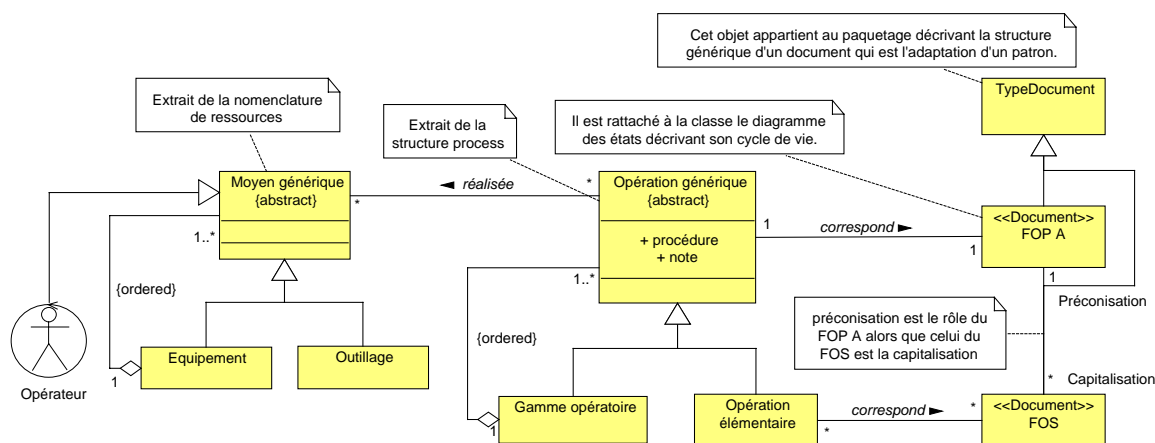


Figure 5.66 Fragment du modèle fédérateur de données

II.4 Mise en œuvre de l'application

Le choix du progiciel Documentum⁷ a été effectué sur la base d'une étude comparative des progiciels du marché. Sans nous attarder sur la définition des critères d'évaluation ; ceux-ci sont techniques, fonctionnels voire stratégiques.

Documentum est une gamme de logiciels de gestion de documents qui permet aux grandes entreprises d'accélérer et d'automatiser la création, l'actualisation, la diffusion et la réutilisation des documents et des données critiques en environnement de travail coopératif⁸.

Il intègre de nombreuses fonctionnalités exigées pour notre projet, en incluant les technologies telles que :

- Le Web.
- Le Groupware et le travail collaboratif.

⁷ Pour en savoir plus : www.documentum.com

⁸ Extrait de la revue "Guide d'achat 2000" éditée par Archimag. Pour en savoir plus : www.archimag.presse.fr

- Le Workflow documentaire permettant par exemple de gérer automatiquement le cycle de vie des documents, depuis la création jusqu'à l'archivage en passant par la validation et la diffusion.
- Le viewer (visualisateur) permettant la visualisation des formats de documents sans disposer nécessairement du logiciel qui a servi à les créer. Avec la fonction d'import d'objets graphiques, il devient possible de créer les documents SPR sans exiger le type de l'outil CAO/DAO ; il suffit simplement de spécifier les formats appropriés.

Cependant, comme tout progiciel, le déploiement de Documentum ne pouvait s'effectuer sans l'intervention de la société Documentum créatrice du produit. Tout le travail de paramétrage du progiciel suivant notre CdC a été confié à Euriware⁹, société de prestation et partenaire officiel de la société Documentum.

Décrivons le processus d'ingénierie logicielle de ses éléments essentiels, sans entrer dans les débats qui lui sont liés :

Equipe de projet

La ~~Figure 5.7~~ ~~Figure 5.7~~ montre les principaux rôles des acteurs de ce projet. Nous représentons la maîtrise d'ouvrage, car la tâche d'expression du besoin et de spécification fonctionnelle nous a été confiée, tâche décrite précédemment (II.1, II.2 et II.3). Les représentants des deux sociétés (Euriware et Documentum) sont les maîtres d'œuvre; ils traduisent nos spécifications avec lesquelles le progiciel est configuré. Il a fallu pour cela fabriquer des modules de code écrit avec l'interpréteur de commandes de Documentum. Il a fallu également manipuler les langages de manipulation de bases de données relationnelles après avoir installé le SGBD Oracle8, supporté par Documentum. Nous avons ainsi pu vérifier la cohérence et la qualité des composants de notre modèle de données.

La Direction Ingénierie Véhicule qui nous a sollicités pour ce travail est notre client final ainsi que les futurs utilisateurs se trouvant dans les sites de production. Le client final est au centre de la démarche ; il est consulté à chaque validation conjointe d'une spécification entre la maîtrise d'ouvrage et la maîtrise d'œuvre. Examinons dans ce qui suit ces jalons intermédiaires.

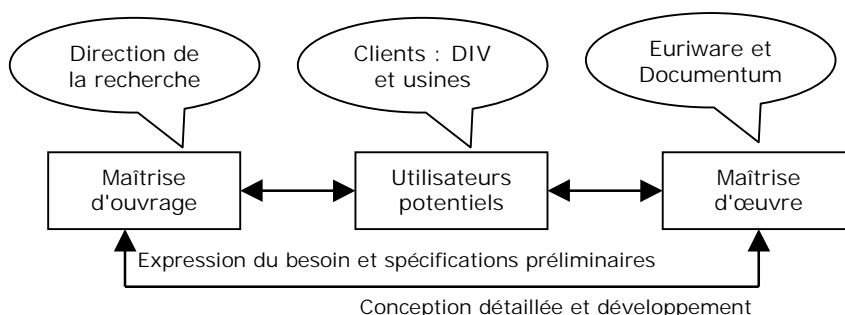


Figure 5.7 Les rôles dans le projet de développement de l'application de GED

⁹ Pour en savoir plus : www.euriware.fr

Durée et phases de mise en œuvre de l'application

La durée de prestation est étendue à neuf semaines. Elle est subdivisée en phases dont les achèvements sont marqués par des réunions de toute l'équipe du projet. Ces revues de projet nous permettent de présenter les fonctionnalités obtenues au client final et d'évaluer sa satisfaction.

Elles nous offrent également la possibilité de formaliser les échanges entre acteurs du projet, de vérifier parmi les tâches pour lesquelles la maîtrise d'œuvre est mandatée, celles réalisées, reportées ou éliminées lorsqu'il est impossible de traduire nos spécifications. La [Figure 5.8](#) montre ces différentes phases définies par leurs fonctionnalités majeures.

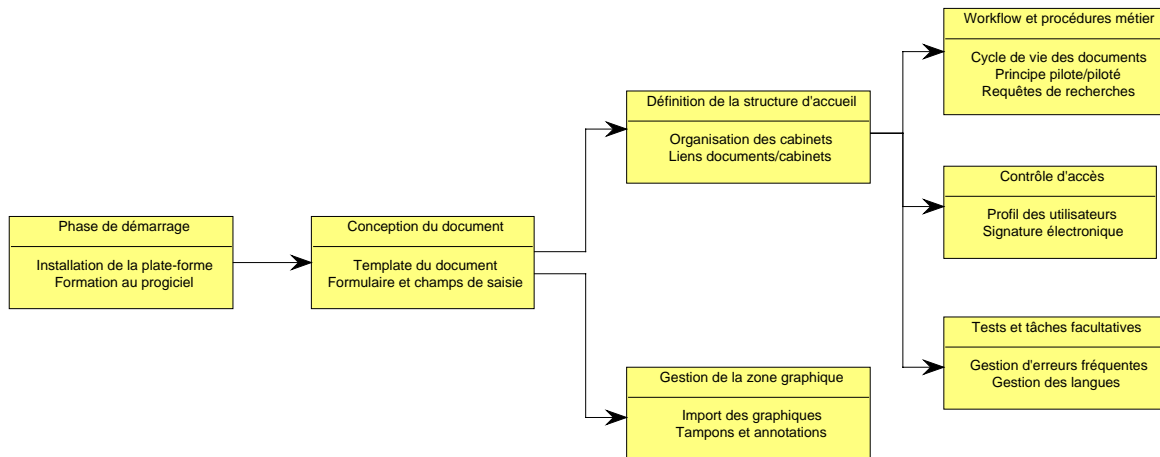


Figure 5.8 Phases de mise en œuvre de l'application de GED

Livrables

En phase d'analyse, les premiers livrables se résument aux diagrammes de spécification dont certains sont illustrés plus haut. Cependant, ces diagrammes restent insuffisants vis-à-vis du client final souhaitant valider les spécifications métier sur des prototypes réels. Ainsi les revues de projets qui ont suivi la phase d'analyse sont accompagnées de présentations des fragments du progiciel configuré. Ces présentations peuvent concerner le processus de création d'un document, la tâche d'administration des requêtes de recherche, de mise en place d'un workflow etc.

L'utilisation des diagrammes de spécification est toutefois poursuivie en phase de conception et de mise en œuvre du progiciel. Ils représentent en effet un moyen d'échange formel avec la maîtrise d'œuvre. La [Figure 5.9](#) montre l'architecture informatique de l'application GED modélisée à l'aide du diagramme de déploiement de UML. Une telle spécification est plutôt invisible pour l'utilisateur final, seuls les maîtres d'œuvre et d'ouvrage sont directement impliqués. L'application finale de GED est accompagnée d'un manuel d'utilisation de l'application logicielle rédigé conjointement au service du client final.

A présent que la rédaction du CdC du projet est décrite, examinons à présent les fonctionnalités obtenues à l'issue de la configuration du progiciel.

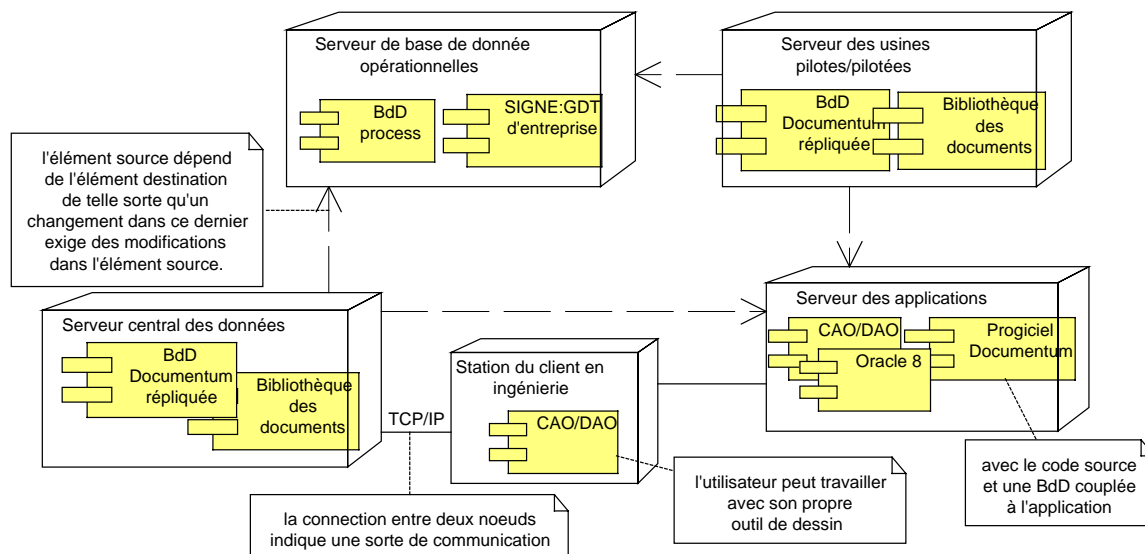


Figure 5.29 Infrastructure de l'application de GED. Diagramme de déploiement

III Fonctionnalités acquises

Les fonctionnalités obtenues du livrable en GED sont nombreuses. Nous les décrivons sommairement en s'appuyant sur des copies d'écran du livrable. Nous montrons en premier la structure d'accueil du progiciel élaborée suivant les métiers d'entreprise (III.1). La procédure de création d'un document est illustrée par la suite (III.2). La traçabilité des documents est assurée en mettant en place le processus de versionnement (III.3). Beaucoup de procédures métier sont automatisées grâce au système de workflow intégré au progiciel (III.4). Nous terminons l'illustration des fonctionnalités obtenues avec les requêtes pour l'édition des rapports (reporting), fonction également indispensable pour la gestion de projet (III.5).

III.1 Structure d'accueil conviviale : vision multi-vue

L'accès à l'application est sécurisé grâce aux profils des utilisateurs prédéfinis -par l'administrateur de l'application par exemple- comme le montre la [Figure 5.10](#).

Par ailleurs, la structure d'accueil telle qu'elle est spécifiée, facilite le travail de gestion documentaire aux différents acteurs des métiers d'ingénierie véhicule. Ces derniers accèdent aux documents en parcourant l'arborescence traduisant leur vue familière ([Figure 5.11](#)) :

- Vue purement produit : qui représente le découpage fonctionnel du véhicule.
- Vue produit/process : en choisissant la vision Nissan ou celle de Renault, toutes deux approuvées.
- Vue purement process : représente le découpage du système industriel.

Le gestionnaire de cabinets est maintenu automatiquement en fonction de nouvelles données. Le rangement des documents une fois créés se fait également d'une manière automatique selon les valeurs saisies des champs des documents que nous décrivons dans ce qui suit.

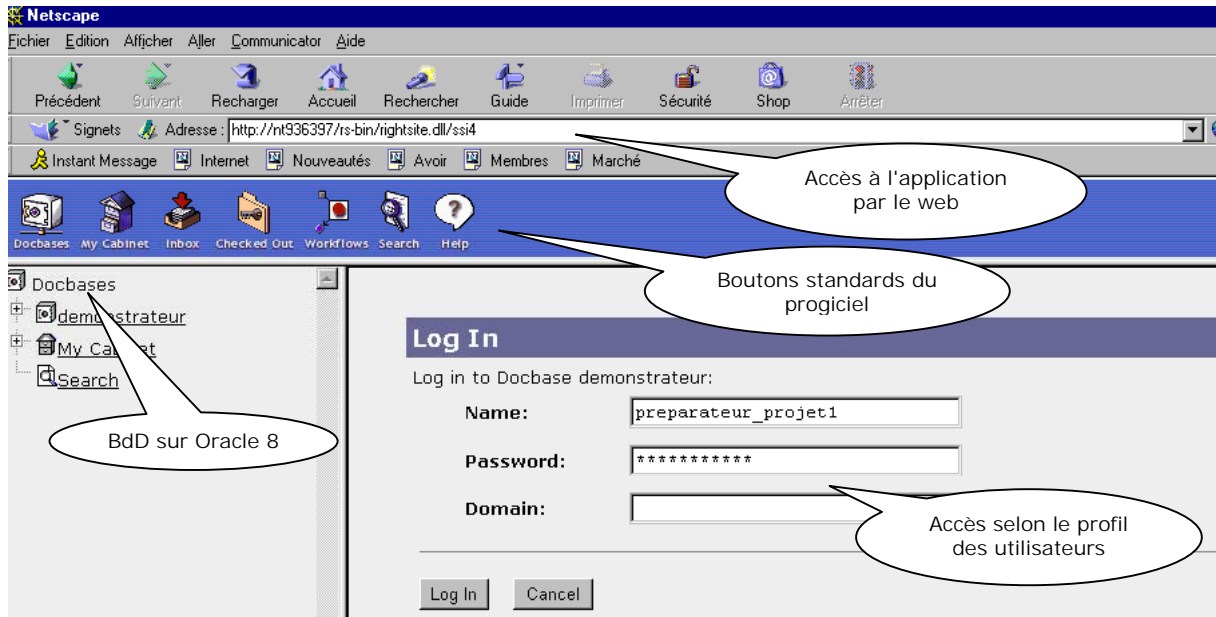


Figure 5.1040 Environnement utilisateur du progiciel Documentum

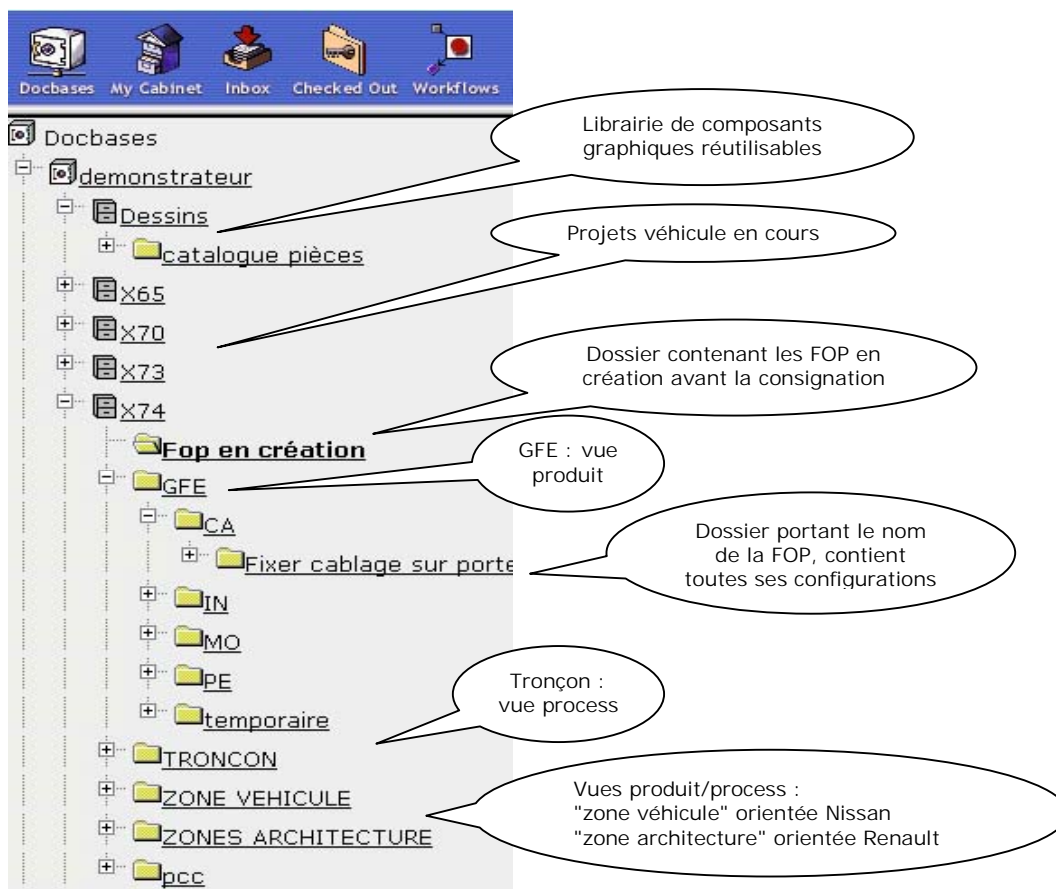


Figure 5.1144 Structure d'accueil de la GED : multiples vues

III.2 Gestion conjointe des documents et des données

La création de documents se fait d'une manière dynamique. Le formulaire de saisie des champs du document FOP A est présenté sur la [Figure 5.12](#) ~~Figure 5.12~~.

Fixer cablage sur porte ar
 Type: Fop A
 Document Lifecycle: None
 Version: 1.0,CURRENT,_NEW_
 Format: HTML Document

Properties

Nom Feuille Opération Process: Fixer cablage sur porte ar

Dessin: F_Cablage_Porte Choose from...

N° de l'opération process:
 N°Séq.Proc/N°Supp/N°Clas.Proc: 0100

Procédure:
 1) Enfiler cablage (N) du module (A) dans la boutonnière (P) du caisson.
 2) Clipser 2 locatings à l'intérieur du caisson

Notes:
 Même montage pour le cablage de porte ar droite

Tronçon: SE8 JUSQU AU REMONTAGE DE PORTES Choose from...

Zone Vehicule: F Portes arrières Choose from...

Fonction Famille:
 (Please click on the choose from button to modify this attribute)
 F16238--- CABLAGE PORTE AR G,F10191--- COM Choose from...

Diversité: TT Tampons dynamiques

Total pages: 0

Nombre tampon serrage: 0 Choose from...

Nombre tampon consommable: 0 Choose from...

Nombre tampon contrôle des lots: 0 Choose from...

Nombre tampon outil spécifique: 0 Choose from...

Save Changes Reset

Vehicule: X74
Usine: TCR Champs saisis automatiquement

Owner: preparateur_projet1
Permit: Delete

Create Document
 Create Document in Folder "Fop en création"
Nom Feuille Opération Process: Fixer cablage sur porte ar
Type: Fop A
Continue Create Cancel

Figure 5.12 ~~12~~ Formulaire de saisie des champs du document

Notons que la structure de donnée implémentée permet cette création dynamique, privilégiant la réutilisation de quelques données existantes et les liens entre les données. La fonction d'import d'objets graphiques, depuis l'outil avec lequel ils sont créés vers la BdD couplée au progiciel, se fait en amont de la création du document. Plusieurs formulaires peuvent être nécessaires pour créer les tampons dynamiques : objets graphiques contenant eux-mêmes des informations.

Après consignation du document, celui-ci est rangé automatiquement en fonction de son contenu dans les dossiers appropriés comme il apparaît sur la [Figure 5.13](#)~~Figure-5.13~~. Les dossiers lorsqu'ils n'existent pas sont créés automatiquement, maintenant le gestionnaire de cabinets.

Le document est en effet stocké une seule fois et un lien virtuel permet de l'afficher dans chaque arborescence traduisant une vue du métier. Dans la partie droite du gestionnaire de cabinets, montrant le contenu du dossier activé, nous pouvons ouvrir le document obtenu avec la possibilité de le mettre à jour, de le faire valider par l'approbateur (responsable hiérarchique par exemple)...

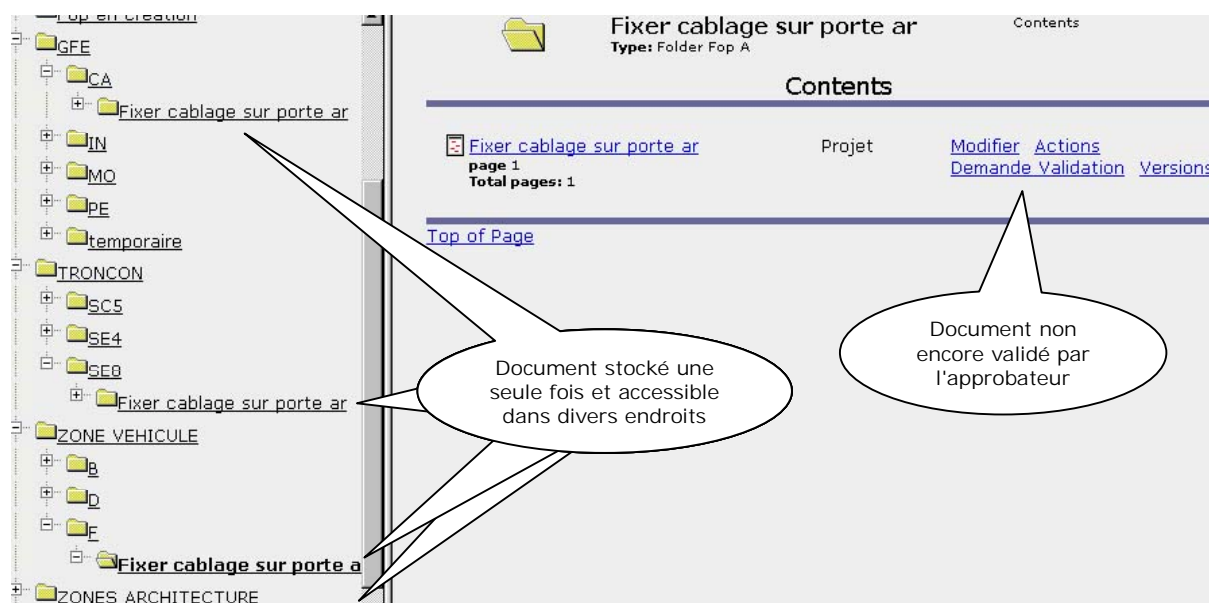


Figure 5.13 ~~13~~ Stockage unique du document et multiplicité d'accès

La [Figure 5.14](#)~~Figure-5.14~~ montre justement le document FOP A créé à l'aide du formulaire précédent. Certains champs ne sont pas remplis, comme les cases liées au nom de l'approbateur et la date de son approbation ; ces éléments sont saisis automatiquement dès le déclenchement du processus de validation du document. D'autres champs sont traités comme l'identifiant de l'opération. Celui-ci devient la concaténation de trois champs : zone véhicule (F), suivi du numéro saisi dans le formulaire (0100) et enfin le nom du projet véhicule (X74).

Constatons également l'absence de tampons car nous avons omis de remplir leurs formulaires spécifiques. Le cartouche du document (bas à droite) montre que le document est à sa version initiale. Les cases réservées vont recevoir les versions ultérieures du document, tâche décrite ci-après.

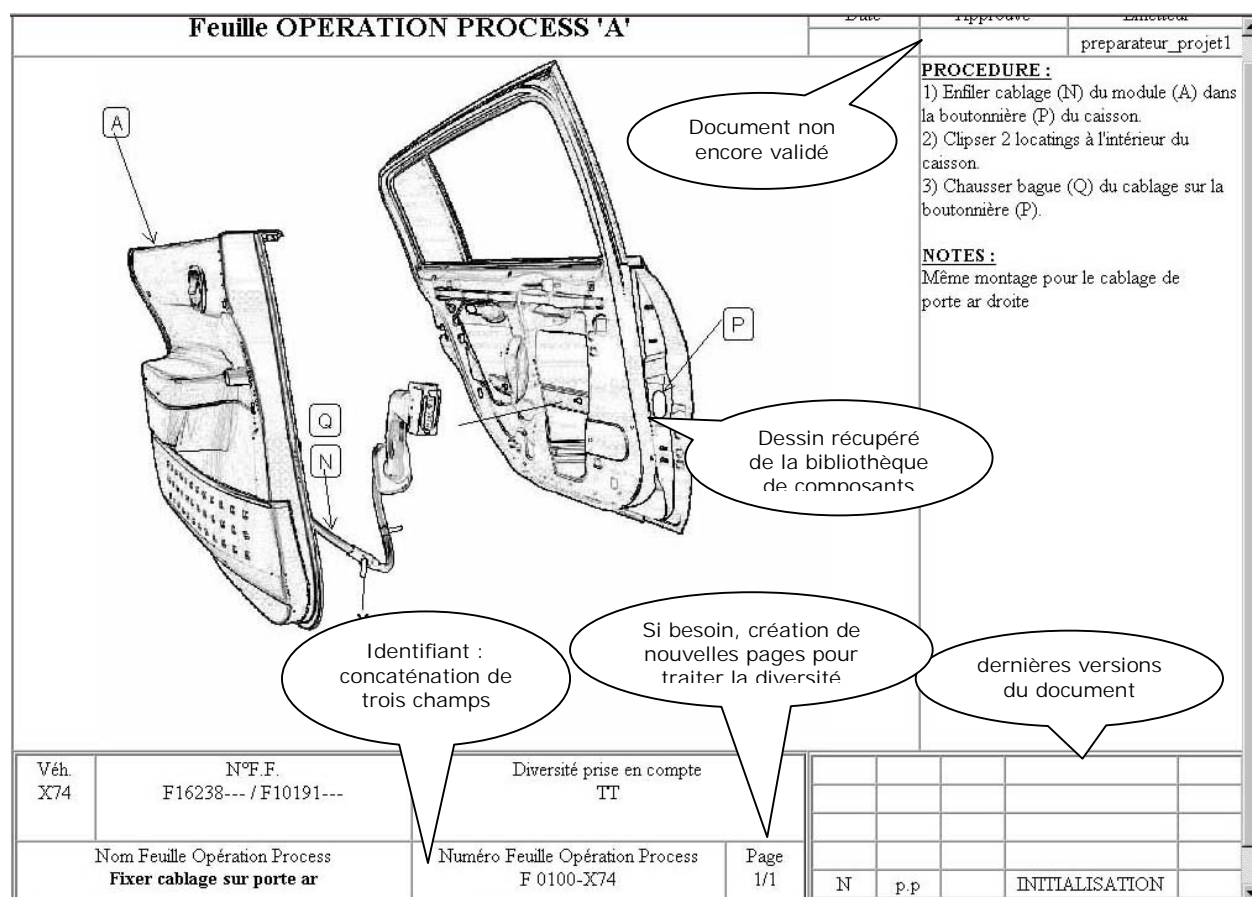


Figure 5.144 Document créé depuis le formulaire de saisie

III.3 Traçabilité et gestion des configurations du document

Le versionnement est aujourd'hui une fonction populaire aussi bien dans le monde de la GDT que de la GED. Dans ce dernier cas, nous nous focalisons sur le traitement des configurations des documents.

Comme tous les documents accompagnant le projet, les documents FOPA évoluent et il est indispensable par conséquent de gérer leur maturité. La [Figure 5.15](#) montre, comme nous l'avons spécifié dans le CdC, deux types de versionnement :

- Version majeure : assigne au document l'état Projet ou Série. Le premier indique que le document circule dans les bureaux d'ingénierie. Le second exprime que le document est basculé dans les sites de production pour son exploitation. Pour chaque jalon (projet, série), nous pouvons retenir plusieurs versions mineures.
- Version mineure : durant le cycle de vie du document (qu'il soit en projet ou en série), chaque évolution de celui-ci peut donner lieu à une incrémentation de sa version. Nous laissons l'acteur décider d'écraser ou non chaque modification de son document. Afin de faciliter la traçabilité des documents, il est préférable de commenter le motif de l'incrémentation de la version du document.

Le versionnement est une fonction généralement spécifiée suivant la culture de l'entreprise ; la notre traduit vraisemblablement la logique des métiers d'ingénierie véhicule, pouvant être différente de celle des autres constructeurs.

Par ailleurs, le processus de versionnement peut être automatisé, en déclenchant des routines de contrôle et de qualité par exemple. Cette automatisation s'effectue grâce au système de workflow, intégré au progiciel. Examinons quelques procédures métier déjà automatisées.

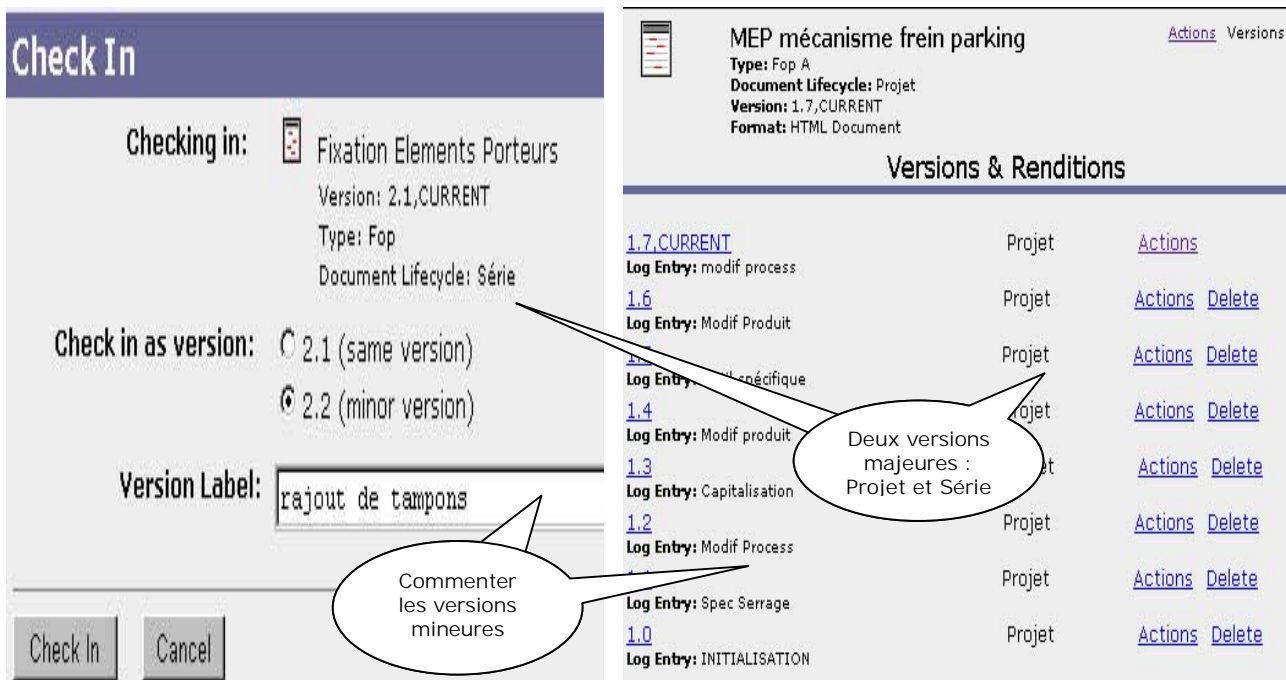


Figure 5.1545 Gestion des configurations du document : versionnement

III.4 Le workflow et automatisation des procédures métier

Nous avons auparavant défini ce qu'est le workflow et son intérêt dans la rationalisation et l'automatisation des procédures métier.

Durant la rédaction du CdC de cette application, nous nous sommes aperçus qu'il y avait plusieurs procédures récurrentes, telles le principe pilote/piloté ou encore le processus de validation des documents (voir par exemple la [Figure 5.5](#) ~~Figure 5-5~~).

Le résultat d'application du premier principe est illustré par la [Figure 5.16](#) ~~Figure 5-16~~. Le document, créé précédemment, est basculé en série selon ce principe. Le document est ainsi envoyé seulement à l'usine pilote qui est en charge des évolutions du document et de sa diffusion aux usines pilotées. S'il est nécessaire, ces dernières peuvent se démarquer de ce processus en créant des document spécifiques. Toutes les configurations du document demeurent visibles à l'ensemble des acteurs, mais avec des droits de modifications restreints. Ainsi, la dernière version du document signée pour être basculée en série se présente comme une préconisation pour les préparateurs en usine. Ces derniers, ayant configuré le document suivant les caractéristiques de leurs sites, fournissent aux acteurs d'ingénierie un moyen de capitalisation des expériences.

La [Figure 5.17](#) ~~Figure 5-17~~ illustre le second exemple qu'est le processus de validation du document. Celui-ci est construit suivant une règle métier. Elle stipule qu'un document doit être impérativement validé par le supérieur hiérarchique de l'émetteur avant même de lui attribuer un état et de le promouvoir dans son cycle de vie. La technologie Web et la messagerie électronique intégrées au progiciel Documentum, aident considérablement ce

travail collaboratif. Ainsi la demande de validation peut se faire par le préparateur en s'adressant au groupe de travail (partie gauche). Celui-ci (groupe ou personne) reçoit instantanément le document, l'examine et envoie une réponse favorable ou non concernant sa validation (partie droite).

Tout comme les profils des utilisateurs, la gestion des groupes de travail peut être assurée par l'administrateur de l'application de GED. Ce dernier est également en charge de la plupart des personnalisations de l'application comme la construction des requêtes de recherche que nous décrivons par la suite.



Figure 5.1646 Diffusion du document suivant les procédures métier

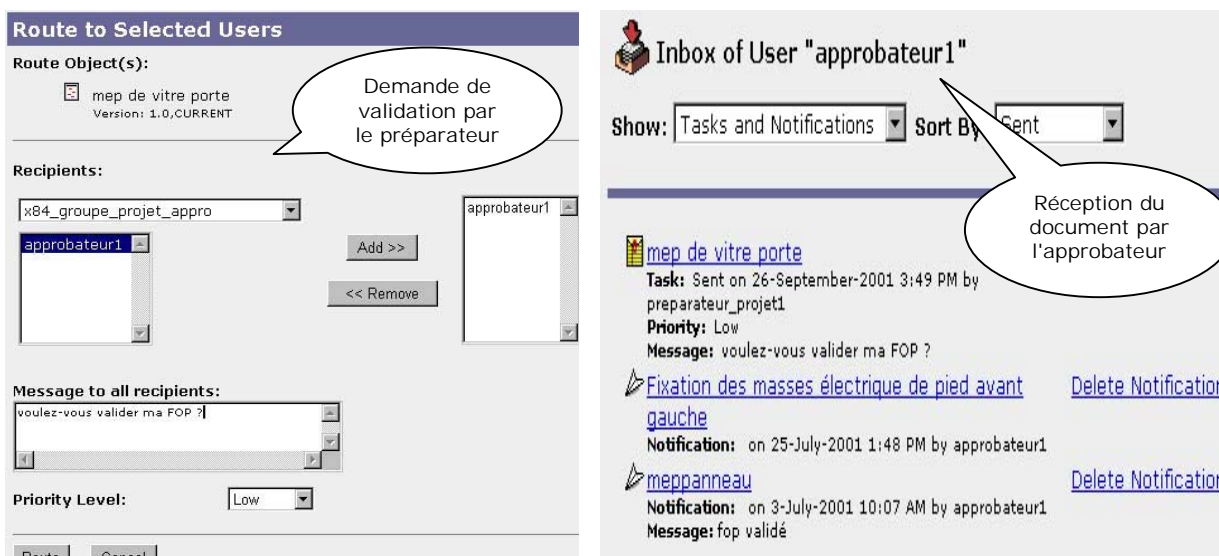


Figure 5.1747 Procédure d'approbation d'un document

III.5 Requêtes de recherche et édition de rapports

Dans les applications de GED, ne gérant pas les données des documents, la recherche de ces derniers se fait uniquement à l'aide des propriétés intrinsèques aux documents, telles que la désignation du document, la date de sa création ou de sa modification, son propriétaire etc.

De telles propriétés sont illustrées sur la [Figure 5.18](#)~~Figure 5-18~~, qui est le format standard de la requête de recherche de documents dans le progiciel.

Dans notre cas, où le contenu informationnel des documents est également géré, il est possible d'affiner l'outil de recherche suivant les besoins des acteurs. La [Figure 5.19](#)~~Figure 5-19~~ montre un exemple de requête personnalisée. En s'appuyant sur les données clés du document et sur les demandes récurrentes des clients utilisateurs de l'application, nous

pouvons construire autant de requêtes qu'il est nécessaire. L'édition des rapports est le résultat d'application de ces requêtes. Nous pensons que cette tâche est utile dans la gestion documentaire au même titre que les précédentes relatées dans cette partie.

Avec cette fonction, nous terminons la description des fonctionnalités de l'application obtenue. Ce projet de développement du nouveau SIT -la GED- s'associe aux nombreux projets d'entreprise dans lesquels nous avons contribué avec le référentiel et le cadre méthodologique proposés durant ce travail de thèse.

Bien entendu, toutes ces contributions étaient des opportunités pour confronter notre recherche aux réalités industrielles ; nous y avons tiré beaucoup de leçons. Mais avant, témoignons tout d'abord dans la conclusion suivante, des leçons retenues de ce cas industriel décrit tout au long de ce chapitre.

Search [Criteria](#) [Results](#)
[SmartLists](#)
[Custom Searches](#)

Criteria for Search

Search for in [DocBase\(s\)](#) *demonstrateur* where:

of the following:

Name	<input type="text" value="contains"/>	<input type="text"/>
Subject	<input type="text" value="contains"/>	<input type="text"/>
Modified	<input type="text" value="on or after"/>	<input type="text"/> <input type="text"/> <input type="text"/>
	<input type="text" value="on or before"/>	<input type="text"/> <input type="text"/> <input type="text"/>
Owner	<input type="text" value="contains"/>	<input type="text"/>
<input type="text" value="Created"/>	<input type="text" value="on or after"/>	<input type="text"/> <input type="text"/> <input type="text"/>
	<input type="text" value="on or before"/>	<input type="text"/> <input type="text"/> <input type="text"/>

And:

Possibilité de créer des requêtes personnalisées

Figure 5.1848 Requête de recherche standard du logiciel

of the following:

Vehicule	is	<input type="text" value="X74"/>
Usine	is	<input type="text" value="Sandouville"/>
Numero Fop	is	<input type="text" value="F0100"/>
Nom operation	<input type="text" value="contains"/>	<input type="text" value="fixer"/>
Modifié	<input type="text" value="on or after"/>	<input type="text"/> <input type="text"/> <input type="text"/>
	<input type="text" value="on or before"/>	<input type="text" value="Aug"/> <input type="text" value="30"/> <input type="text" value="2001"/>
Zone Vehicule	is	<input type="text" value="F Portes arrières"/>
Fonction Famille	is	<input type="text"/>
Tronçon	is	<input type="text" value="SE4 AVANT SOUS-CAISSE"/>
<input type="text" value="Diversité"/>	<input type="text" value="starts with"/>	<input type="text" value="B74"/>
<input type="text" value="Nombre tampon outil spécifique"/>	<input type="text" value="equals to"/>	<input type="text" value="2"/>
<input type="text" value="Dessin"/>	is	<input type="text" value="D_Cable_ABS"/>

Champs significatifs du document

Figure 5.1949 Exemple de requête de recherche personnalisée

IV Conclusion et ouvertures de l'application

Le développement d'un nouveau SIT (GED) pour l'ingénierie de production, nous a permis de montrer les capacités d'adaptation aussi bien du référentiel du domaine que du cadre méthodologique préconisés.

Le processus d'adaptation du référentiel s'effectue comme suit :

- Sélection des paquetages appartenant au champ d'application de la GED.
- Etude de nouveaux cas d'utilisation métier issus du domaine.
- Réexamen des cas d'utilisation métier existants.
- Génération de nouveaux objets du domaine.
- Structuration et raffinement de nouveaux objets en lien avec les composants existants.
- Enrichissement du référentiel du domaine.

Pour ce projet, l'adaptation du référentiel a généré des effets directs et positifs :

- Accélération du délai de spécification en réutilisant les paquetages existants dans le référentiel. Cette forme de réutilisation présente un gain de temps considérable pour le projet.
- Rationalisation de la communication entre la maîtrise d'ouvrage (que nous étions) et la maîtrise d'œuvre (le sous-traitant) et cela par l'emploi des diagrammes standards d'expression de besoins et de spécification fonctionnelle.

Par ailleurs, les décisions que nous avons été amenés à prendre durant le travail d'analyse et de conception sont liées au triptyque : intention du projet, produit final attendu et contraintes environnementales. Nous avons défini ce triptyque comme socle fondateur de toute démarche d'ingénierie des SIT en entreprise, pouvant se substituer ainsi aux méthodes formelles de la littérature moins appropriées aux projets d'ingénierie. Le choix de la technologie GED, la prise en compte des BdD opérationnelles... sont des exemples de telles décisions.

Nous avons toutefois adapté des approches de mieux en mieux acceptées en entreprise :

- La sous-traitance du développement même de l'application : le plus souvent, les technologies d'information et les progiciels (SGDT, GED...) sont difficiles à mettre en œuvre sans l'intervention de l'éditeur de ces produits. L'expérience a été réussie, grâce notamment à :
- La démarche participative et itérative où l'utilisateur (le client final dans la DIV et dans les sites de production) est mis au centre de celle-ci. La présence du client final dans les revues de projet nous a permis d'évaluer continuellement la satisfaction de ses attentes.

Plus globalement, ce travail est d'un apport significatif pour l'entreprise pour les raisons suivantes :

- L'intégration des SIT existants, notamment la GDT d'entreprise SIGNE.
- L'étude de la nomenclature de produit Nissan, auparavant inconnue, en prenant en compte ses atouts dans la structure de donnée de la GED.
- La vision globale de l'application de GED (par opposition à une GED locale) : la structure de donnée implémentée dans la GED transcende les cloisons entre directions pour supporter toute la gestion documentaire des métiers process.

Malgré le succès qu'a reçu ce travail, à nos yeux, nous n'avons pas étudié plusieurs aspects importants comme :

- Elargir le concept de la gestion des configurations : elle peut être étendue par exemple pour traiter la diversité causée par les différentes langues caractérisant les usines.
- Eviter la réplication des BdD : celle-ci apparaissait comme solution incontournable face à la rigidité du progiciel et au choix d'avoir intégré les BdD opérationnelles existantes.
- Montrer le lien avec l'ERP de l'entreprise : c'était en effet l'occasion d'étudier les éventuels liens entre la GDT supportée par le progiciel Documentum et les données du progiciel SAP homologué en entreprise, d'autant plus que les éditeurs de ces deux progiciels travaillent en partenariat.
- Manipuler les objets graphiques intelligents : ces derniers sont le produit de la dernière génération d'outils de CAO/DAO. Il aurait été intéressant de penser à une structure de données supportant ce type d'objets...

Enfin, l'ensemble des nos contributions au sein de l'entreprise ont enrichi notre travail de thèse et nous ont permis de dégager les perspectives possibles à celui-ci.

Nous récapitulons le tout dans la conclusion générale de ce manuscrit.

CONCLUSION GENERALE

Bilan et perspectives

Life is the art of drawing sufficient conclusions from insufficient premises.

Samuel BUTLER (1835-1902). Notebooks.

Conclusion générale

L'ingénierie des SIT en entreprise souffre davantage de **problèmes méthodologiques** que techniques ou organisationnels... C'est cette dimension que nous avons traitée dans ce travail, en ayant pour champ d'application l'**ingénierie de production**.

Les problèmes méthodologiques abordés se résument en ces points clés :

- Formalisation des expressions de besoins en GDT pour les principaux acteurs : la maîtrise d'ouvrage et la maîtrise d'œuvre. En phase amont d'ingénierie du domaine, le CdC de ces acteurs est incomplet et informel.
- Rationalisation de la démarche d'ingénierie des SIT. Les projets de mise en œuvre des SIT en entreprise sont généralement lents et présentent des ruptures à travers leurs jalons. Par ailleurs, les expériences industrielles ont montré la difficulté d'adaptation des méthodes académiques.

Nous avons alors développé :

- Un **référentiel du domaine** de l'ingénierie de production, constitué de deux composants que nous avons convenu d'appeler : business model et data model.
- Un **cadre méthodologique** pour l'élaboration du référentiel et plus globalement pour la conduite des projets de développement de SIT en entreprise.

Relativement à ces deux aspects, les **apports de notre contribution** peuvent se résumer ainsi :

Référentiel du domaine : ses propriétés et son expérimentation démontrent :

- L'accélération du délai de spécification par la réutilisation des fragments du référentiel.
- La rationalisation du domaine de l'ingénierie de production : le référentiel regroupe la taxonomie des objets process, les terminologies, les modèles fonctionnels et contextuels... Le domaine de l'ingénierie de production, nous l'avons vu, a longtemps été confondu avec la production stricte. Les travaux de recherche et les actions normatives commencent à peine à s'y intéresser, et ce, avec l'apparition des outils d'IPAO.
- L'indépendance du référentiel vis-à-vis des outils d'IPAO et de SGDT qui sont en évolution permanente. Les travaux de structuration de données techniques s'appuient souvent sur les spécificités de ces outils alors qu'ils ne sont pas suffisamment stables.
- L'évolutivité de celui-ci, car chaque adaptation du référentiel dans les projets d'entreprise lui donne l'occasion d'être mis à jour.
- La formalisation des échanges entre acteurs : cela est possible grâce notamment à l'emploi de diagrammes de spécification suffisamment compréhensibles et standards. Ces diagrammes représentent au mieux les besoins de la maîtrise d'ouvrage et les problèmes fréquemment posés par la maîtrise d'œuvre.
- L'intégration des SIT existants (legacy systems), particulièrement le SGDT d'entreprise SIGNE. Le référentiel capitalise ainsi les objets pertinents présents dans ces systèmes, tout en fédérant leur sémantique.

Cadre méthodologique : nous le caractérisons par les approches, les techniques et l'AGL :

- L'approche combine les stratégies ascendante et descendante –communément employées séparément. La première stratégie nous a permis d'intégrer les SIT existants par rétroingénierie. Grâce aux techniques traditionnelles d'analyse du domaine de la seconde

stratégie, nous avons pu prendre en compte des business requirements de l'entreprise. La complémentarité de ces stratégies a été donc démontrée.

- Les décisions prises durant l'élaboration du référentiel sont contextuelles, guidées par le triptyque (intention du projet, produit final attendu et contraintes environnementales). A ce titre, ce triptyque peut se substituer aux méthodes académiques, dotées de règles généralement rigides.
- L'interaction entre le business model et le data model. Celle-ci se concrétise par la formalisation du processus d'extraction des objets métiers depuis les processus opérationnels. Ce processus est souvent exécuté d'une manière implicite lors des développements de SIT. Cette formalisation peut vraisemblablement apporter des réponses à la question d'intégration des composants du SI, connus sous les noms de SI organisationnel et de SI informatisé.
- Le processus d'élaboration du référentiel est itératif et participatif : cela est possible grâce en grande partie aux propriétés de l'AGL utilisé. Il est doté d'une encyclopédie, favorisant la réutilisation. Il dispose en outre des techniques requises à l'élaboration du référentiel, à savoir : les techniques de modélisation des données, des processus et la notation UML.
- Le cadre de Zachman nous a permis d'organiser les connaissances d'entreprise durant la phase cruciale d'analyse du domaine. De plus, il nous a été possible de gérer aisément la diversité des techniques employées grâce à celui-ci. Dès lors, nous nous interrogeons des raisons pour lesquelles ce cadre est si peu connu par la communauté francophone.

Par ailleurs, les difficultés rencontrées durant ce travail nous ont ouvert des **perspectives**. En intégrant la DUAF (Direction Urbanisme et Architecture Fonctionnelle) de la direction informatique du groupe Renault, un nouveau projet "Référentiel Process" a été lancé. Il est conjointement piloté par la DUAF et la DIV dont la phase avant-projet est fixée pour janvier 2002. L'enjeu du projet, comme son nom l'indique, est de développer un référentiel, adoptant un périmètre plus large que celui délimité dans notre travail de thèse. Il a été décidé de se baser sur nos résultats de thèse, de tester le référentiel proposé à grande échelle et d'emprunter les aspects méthodologiques qui nous ont permis d'élaborer celui-ci.

Face à ces ambitions, nous nous voyons confrontés à des questions pouvant être étudiées dans le cadre d'un travail de recherche futur. Nous y inscrivons des problématiques, ayant pour enjeu une meilleure :

- **Formalisation du cadre méthodologique** : si le référentiel du domaine est formalisé par l'emploi des diagrammes de spécification standards, il n'en est pas de même pour le processus de son élaboration. La façon souhaitée pour formaliser ce cadre peut ressembler à la **démarche d'ingénierie de patrons** proposée dans la thèse de Gzara [GZA00]. Nous devons veiller toutefois à ne pas prescrire les étapes et à respecter le caractère créatif du métier d'ingénierie, sous peine de retomber dans les inconvénients des méthodes formelles.
- **Rationalisation de l'ingénierie de production** : durant notre travail de thèse, nous avons emprunté très peu de concepts abordés dans les références normatives **PSL et MANDATE** [GRU00;CUT00]. Celles-ci étaient encore à leurs débuts. Cependant, nous croyons au rôle important qu'elles peuvent jouer dans la standardisation des concepts de l'ingénierie de production.
- **Intégration des données en entreprise** : nous avons constaté que les frontières jusqu'ici imposées entre les données techniques et celles de gestion, ou encore entre le monde de l'ingénierie et de la production n'ont plus lieu d'être. Nous savons par ailleurs qu'il serait difficile d'intégrer toutes les données d'entreprise dans un seul système. De ce

fait, nous estimons qu'il est plus judicieux de mettre en place conjointement les systèmes PDM et ERP, sans toutefois négliger les liens entre eux. Dans ce contexte, nous souhaitons tester particulièrement si les nomenclatures proposées dans notre référentiel peuvent supporter les spécialisations et les détails de la production. La thèse de Ndiaye [NDI00] par exemple nous donne une idée de la grande aptitude de la **métamodélisation** à gérer les changements et les évolutions des modèles...

Une fois les problèmes méthodologiques traités, le respect des missions de chacun devrait éviter la plupart des problèmes humains, techniques ou organisationnels, notamment en :

- Faisant coexister les deux systèmes en entreprise : les PDM et les ERP.
- Intégrant les deux mondes : l'ingénierie et la production.
- Conciliant les deux rôles : la maîtrise d'ouvrage et la maîtrise d'œuvre.

REFERENCES BIBLIOGRAPHIQUES

... My library was duckedon large enough.

William SHAKESPEARE (1564-1616). The Tempest, 1, 2, Prospero

Références bibliographiques

Note : les références suivantes sont toutes citées dans le manuscrit.

- [AIT98] AIT (1998) AIT for European Manufacturing Industry - Integration Platform. Specification 1.0, Consortium distribution, www.ait.org.ukl/projects.
- [ALS99] Alsene, E. (1999) The Computer Integration of The Enterprise. IEEE Transactions on Engineering Management, 46.
- [AMB99] Ambler, S. W. (1999) Mapping Objects To Relational Databases. An AmbySoft Inc. White Paper, www.ambySoft.com/mappingObjects.pdf.
- [AVI00] Avison, D. E. and G. Fitzgerald (2000) Information Systems Development: Methodologies, Techniques and Tools. 2nd Edition, McGraw-Hill.
- [BAI98] Bailey, J. E. and R. H. Rucker (1998) Automated Management of Design Data : Product, Process and Resource structures. International Journal of Industrial Engineering, 5, 1, 7-16.
- [BAR96] Bartoli, J. A. et J. L. Le Moigne (1996) Organisation Intelligente et Système d'Information Stratégique. Economica.
- [BOU00] Bouzeghoub, M. et A. Rochfeld (2000) La conception objet des SI. Hermès-Science.
- [BRA85] Bravoco, R. R. and S. B. Yadav (1985) Requirement Definition architecture - An overview. Computers in Industry, 6, 237-251.
- [BRI93] Brinkkemper, S., Engmann, R., Harmsen, F. and R. van de Weg (1993) Complexity reduction and modeling transparency in CASE Tools. IN: Avison, D. E. et al. (1993) Human, Organisational, and Social Dimensions of IS Development. North Holland, Amsterdam.
- [BUC87] Buckingham, R. A., Hirschheim, R. A., Land, F. F. and C. J. Tully (1987) Information Systems Education: Recommendations and Implementation. CUP, Cambridge.
- [BUG98] Bugtai, N. and R. I. M. Young (1998) Information Models in an integrated fixture decision support tool. International Conference on Computer Aided Production Engineering, Warsaw Poland, 87-94.
- [CAS92] Castano, S., De Antonellis, V. and B. Zonta (1992) Classifying and reusing conceptual schemas. In Proceedings of the ER approach.
- [CAT87] Catchpole, C. P. (1987) Information systems design for the community health services. PhD thesis, Aston University, Birmingham.
- [CAU 01] Cauvet, C. & C. R. Sabroux (2001) Ingénierie des Systèmes d'Information. Hermès.
- [CHE76] Chen, P. (1976) The Entity-Relationship Model – Toward a unified view of data. Communications of the ACM Transactions on Database systems, 1, 1.
- [CHE90] Checkland, P. (1990) Systems Thinking, Systems Practice. Wiley, Chichester.
- [CHE97] Chen, Y. M. & Y. T. Hsiao (1997) A collaborative data management framework for concurrent product and process development. Int. J. Computer Integrated Manufacturing, 10, 6, 446-469.
- [CHE98] Chen, Y.M. & T.H. Tsao (1998) A Structured Methodology for Implementing Engineering Data Management Robotics and Computer Integrated Manufacturing, 14, 275–296.

- [CHE00] Cheung, W. M., J. Zhao, J. M. Dorador and R. I. M. Young (2000) Linking product design and manufacturing capability through a manufacturing strategy representation. International Conference on Computer-Aided Production Engineering, UK.
- [CIM01] CIMdata (2001) Strategies and solutions for collaborative engineering through the supply chain. 2001 conference. Palm Springs, California, www.cimdata.com.
- [COA92] Coad, P. (1992) Object-Oriented Patterns. Communications of the ACM, 35, 9.
- [COA99] Coad, P., Lefebvre, E. and J. D. Luca (1999) Java Modeling In Color With UML: Enterprise Components and Process, Prentice Hall.
- [COD70] Codd, E. F. (1970) A Relational model of data for large shared data banks. Communications of the ACM, 13, 6.
- [COL93] Colquhoun, G. J., Baines, R. W. and R. Crossley (1993) The state-of-the-art review of IDEF0. International Journal Of Computer Integrated Manufacturing, 6, 4.
- [COO96] Cook, M. A. (1996) Building Enterprise Information Architecture. Reengineering Information Systems. Hewlett-Packard Professional Books.
- [CSC00] CSC (2000) Catalyst Overview Version 1.0, © by Computer Sciences Corporation, USA.
- [CUT00] Cutting-Decelle, A. F. and J. J. Michel (2000) A standardised Data Model For Manufacturing Management. ISO 15531 TC184/SC4/WG8 N220 Mandate.
- [DAT95] Date, C. J. (1995) An introduction to Database Systems. 6th ed, Addison Wesley.
- [DHA97] Dhamija, D., D. A. Koonce and R. P. Judd (1997) Development of an unified data meta-model for CAD-CAPP-MRP-NC verification integration. Computers Industry Engineering, 33, 1-2,19-22.
- [DOU93] Doumeingts, G., Chen, D., Vallespir, B. and P. Fenie (1993) GIM (GRAI Integrated Methodology) and its evolutions - a methodology to design and specify advanced manufacturing systems. In: Yoshikawa, H. and J. Goossenaerts (Eds). Proceedings of the JSPE/IFIP TC5/WG5.3 Workshop on the design of information infrastructure systems for manufacturing, DIISM'93 Tokyo, Japan.
- [FOW96] Fowler, M. (1996) Analysis Patterns. Reusable Object Models. Addison Wesley.
- [FOW00] Fowler, M. (2000) Analysis Patterns, Work in Progress. www.martinfowler.com/ap2/index.html.
- [GAM95] Gamma, E., Helm, R., Johnson, R. and J. Vlissides (1995) Design Patterns, Elements of reusable Object-Oriented software. Addison-Wesley Publishing Company.
- [GAR83] Gardarin, G. (1983) BD : les systèmes et leurs langages. Eyrolles.
- [GRU00] Gruninger, M. and A. F. Cutting-Decelle (2000) Process Specification Language: General Overview ISO TC184/SC4/WG8 N220 – ISO/WD 18629-1.
- [GZA00] Gzara, L. (2000) Les patterns pour l'ingénierie des systèmes d'information Produit. Thèse de doctorat, ENSGI, INP Grenoble.
- [HAM93] Hammer, M. and J. Champy (1993) Reengineering the corporation: A manifest for business revolution. Harper Collins Publishers, Inc. Ed., New-York.
- [HAM98] Hameri, A. P. and J. Nihtilä (1998) Product data management - Exploratory study on state-of-the-art in one-of-a-kind indutstry. Computers in Industry, 35, 195-206, Ed. Elsevier.

- [HAM99] Hameri A. P. and J. Nikkola (1999) How engineering data management and system support the main process-oriented functions of a large-scale project. *Production Planning & Control*, 10, 5, 404-413.
- [HAR98] Harmon, P. and M. Watson (1998) *Understanding UML: The developers guide with a web-based application in Java*. Morgan kaufmann Publishers.
- [HAR99] Harding, J. A., Yu, B. and K. Popplewell (1999) Information Modeling: An Integration of Views of a Manufacturing Enterprise. *International Journal of Production Research*. 37, 2777-2792.
- [HEL00] Helms, R. W., Pels, H. J., Elzen, V. D. and E. E. Jans (2000) Simulating Product Development Processes from a PDM Perspective. *TMCE 2000*, Ed. I. Horvath et al. Delft University, 787-798.
- [HEN00] Hensinger C., Reichert, M., Bauer Th., Strzeletz, Th. and P. Dadam (2000) ADEPT workflow– Advanced Workflow Technology for the Efficient Support of Adaptive, Enterprise-wide Processes. 7th conference on Extending Database Technology, Germany.
- [HIR89] Hirschheim, R. and H. K. Klein (1989) Four paradigms of information systems development. *Communications of the ACM*, 32, 10.
- [INM97] Inmon, W. H., Zachman, J. A. and J. G. Geiger (1997) *Data Stores, Data Warehousing and the Zachman Framework*. Managing Enterprise Knowledge. McGraw-Hill.
- [JAC95] Jacobson, I., Ericsson, E. and A. Jacobson (1995) *The Object Advantage: Business Process Reengineering with Object Technology*. Addison Wesley Publishing Company.
- [JAY94] Jayratna, N. (1994) *Understanding and Evaluating Methodologies, NIMSAD: A systemic Framework*. McGraw-Hill, Maidenhead.
- [KAL97] Kalpic, B. and A. Polajnar (1997) Model of the holistic Information Integration of an enterprise. *Strojarsstvo*, 39, 6, 275–280.
- [KEL98] Kellert, P. and S. Ruch (1998) Méthodologie de modélisation orientée-objet des systèmes de production. *APII- JESA* , 32, 1.
- [KEN83] Kent, W. A. (1983) Simple guide to five normal forms in relational theory. *Communications of the ACM*, 26, 2.
- [KET98] Kettani, N., Mignet, D., Paré, P. et C. Rosenthal-Sabroux (1998) *De Merise à UML*. Ed Eyrolles, Paris.
- [KIM93] Kim, C., Kim, K. and I. Choi (1993) An object oriented information modeling methodology for manufacturing information systems. *Computers industrial engineering*, 24, 3, 337-353.
- [KOS96] Kosanke, K. (1996) Comparison of enterprise modeling methodologies – Proceedings of the IFIP TC5/WG5.3/WG5.7 International conference on the Design of Information Infrastructure Systems for Manufacturing (DIISM'96), Chapman & Hall.
- [KOV98] Kovacs, Z., Le Goff, J. M. & R. McClatchey (1998) Support for product data from design to production. *Computer Integrated Manufacturing Systems*, 11, 4, 285-290.
- [KUR93] Kurbel, K. and T. Schneider (1993) Integration issues of information engineering based I-Case tools. IN: Avison, D. E. et al. (1993) *Human, Organisational, and Social Dimensions of IS Development*. North Holland, Amsterdam.
- [MAI94] Maiden, N., Sutcliffe, A., Taylor, C. and D. Till (1994) A set of formal problem abstractions for reuse during requirement engineering. *Ingénierie des systèmes d'information*, 2, 6, 679-698.

- [MAD83] Maddison, R. N. (1983) Information systems Methodologies. Wiley Heyden Chichester.
- [MAR99] Martin R. and E. Robertson (1999) Formalization of Multilevel Zachman Frameworks. Technical Report n° 522, Computer Science Department, Indiana University.
- [MAR00] Martin, R. C. (2000) Design principles and Design Patterns. www.objectmentor.com.
- [MAR'00] Marshall, C. (2000) Enterprise Modeling with UML: Designing Successful Software through Business Analysis. The Addison-Wesley Object Technology Series.
- [MCI95] McIntosh, K. G. (1995) Engineering Data Management - A Guide to Successful Implementation. McGraw-Hill Book Company.
- [MIC00] Michel, J. J. and A. F. Cutting-Decelle (2000) Relationship between manufacturing related standards. ISO TC184/SC4/WG8 N216.
- [MIL98] Million, O. (1998) De l'intégration des métiers par les données techniques vers la maîtrise de la modélisation conceptuelle : la méthode VIM. Thèse de doctorat, université H. Pointcarré, Nancy.
- [MOR94] Morand, B. (1994) Trois arguments et trois propositions pour concevoir des systèmes d'information organisationnels. IN Ingénierie systémique : de la conception orientée objet à la conception orientée projet, GRASCE, URA CNRS 935.
- [MOR00] Morley, C., Hugues, J. et B. Leblanc (2000) UML pour l'analyse d'un SI : le cahier des charges du maître d'ouvrage. Dunod.
- [MUL99] Muller, R. J. (1999) Database Design for smarties. Using UML for Data Modeling. Morgan Kaufmann Publishers.
- [NDI00] Ndiaye, D. (2000) Apport des technologies orientées objet dans l'étude et la mise en place d'un référentiel de conception pour les systèmes automatisés de production. Thèse de doctorat, université de Lille 1.
- [OLL91] Olle, T. W.,Hagelstein, J., Macdonald, I. G., Rolland, C., Sol, H. G., Van Assche, F. J. M. and A. A. Verrijn-Stuart (1991) Information systems methodologies – A framework for understanding. 2nd Edition, Addison-Wesley, Wokingham.
- [OMG99] OMG (1999) Unified Modeling Language Specification. Version 1.3.
- [OPA96] OPAL (1996) Architecture Definition. OPAL EP20377, deliverable 3.1.1.
- [OUS99] Oussalah, C. (1999) Génie Objet – analyse et conception de l'évolution d'objets. Hermès.
- [PEL00] Pels, H. J., Helms, R. W. and T. H. Mandemaker (2000) RapidPDM: Faster implementation of PDM. Ed. B. Mesley, 245-254, www.rapidpdm.org/lhints.html.
- [POP00] Popkin software (2000) Building an Enterprise Architecture, Draft version 1.00 www.popkin.com.
- [RAN95] Randoing, J. M. (1995) Les SGDT, Hermès.
- [RIE99] Rieu, D. (1999) Ingénierie des SI. BdD, Bases de Connaissances et Méthodes de Conception. Mémoire d'Habilitation à Diriger les Recherches. INPGrenoble.
- [ROU00] Rouibah, K. (2000) Specification of an Engineering Workflow Methodology – Deliverable 2A SIMNET.BE.032.DFC- EP26780.
- [ROL96] Roland, C. (1996) L'ingénierie des processus de développement de systèmes : un cadre de référence. Ingénierie des Systèmes d'information, 4, 6, 705-744.
- [ROL00] Roland, C. and N. Prakash (2000) From Conceptual modeling to Requirement engineering. Special issue of annals of software engineering on comparative studies of engineering approaches for software engineering.

- [SAN93] Sandoval, V. (1993) CALS - Introduction et mise en œuvre, Ed. Hermès.
- [SOW92] Sowa, J. F. & J. A. Zachman (1992) Extending and Formalizing the Framework for Information Systems Architecture. IBM Systems Journal, 31, 3.
- [SID96] Siddiqi, J and M. C. Shekaran (1996) Requirement engineering: the emerging wisdom. IEEE Software, 03.
- [TAB93] Tabourier, Y. (1993) Propositions for changing the methodological perspective of organisational information systems design In Autour et alentour de Merise. Les méthodes de conception en perspectives. CERAM.
- [TAR91] Tardieu, H. et B. Guthmann (1991) Le triangle stratégique : stratégie, structure et technologie de l'information. Édition d'Organisation, Paris.
- [TES95] Tessier, C. (1995) La pratique des méthodes en informatique de gestion, les éditions des organisations, Paris.
- [TOH99] Toh, K. T. K, Harding J. A. and D. Thomson (1999) An holonic approach for the modelling of enterprise functionality and behaviour. Int. J. computer Integrated Manufacturing, 12, 6, 541-558.
- [TOL99] Tollenaere, M. (1999) Internet, Intranet, SGDT : couvertures fonctionnelles et complémentarités, Journée PRIMECA, IFMA, Clermont-Ferrand.
- [VLI97] Van Vliet, P. J. A. (1997) Information Systems Architecture & Organization Syllabus - Information Systems & Quantitative Analysis Department, College Of Information Science and Technology, University of Nebraska, Omaha.
- [VER96] Vernadat, F. (1996) Enterprise Modeling and Integration: Principles and Applications, Chapman & Hall.
- [VER99] Vernadat, F. (1999) Techniques de modélisation en entreprise - Application aux processus opérationnels. Economica.
- [WES99] Weske, M. (1999) Adaptive Workflows based on Flexible Assignment of Workflow Schemas and Workflow Instances, Proc. Workshop Informatik'99, enterprise-wide and cross-enterprise Workflow Management: Concepts, Systems, Applications, 6.
- [WES98] Westfechtel, B. and R. Conradi (1998) Software Configuration Management and engineering Data Management: differences and similarities. In Boris Magnusson, editor, Software Configuration Management - ECOOP'98 SCM8 Symposium, 95-106, Brussels, Belgium.
- [WFM96] WfMC (1996) WfMC-TC-1011. Terminology and Glossary. Technical Report, Workflow Management Coalition, Brussels, Belgium.
- [WHI01] Whitten, J. L., Bentley L. D. and K. C. Dittman (2001) Systems analysis and design methods. McGraw-Hill 5th Edition.
- [WIL93] Williams, T. (1993) The Purdue Enterprise Reference Architecture. In Yoshikawa, H. and Goossenaerts, J. (Eds), Information Infrastructure Systems for Manufacturing (B-14), Elsevier Science B. V. (North-Holland).
- [YOU92] Yourdon, E. (1992) Decline and Fall of the American Programmer. Yourdon Press, Englewood Cliffs, New Jersey.
- [YU00] Yu, B., Harding J. A. and K. Popplewell (2000) A reusable Enterprise Model. International Journal of Operations & Production Management. 20, 50-69.
- [ZAC87] Zachman, J. A. (1987) A framework for information systems architecture. IBM Systems Journal, 26, 3.

- [ZAC97] Zachman, J. A. (1997) Enterprise Architecture: the issue of the century. Database programming and design, www.dbpd.com.
- [ZHA99] Zhao, J., Cheung, W.M. and R.I.M. Young (1999) A Consistent Manufacturing Data Model to Support Virtual Enterprises. International Journal of Agile Manufacturing Systems, 1, 3, 150-158.
- [ZHA'99] Zhang, W. J. and Q. Li (1999) Information Modelling for Made-to-Order Virtual Enterprise Manufacturing Systems. Computer Aided Design, 31, 99, 611-619.
- [ZHA00] Zhao, J., Cheung, W. M. and R.I.M. Young (2000) The influence of manufacturing information models on product development systems. Engineering Design Conference, Brunel university, UK.

ANNEXES

Annexe A – Glossaire

Annexe B – Problématiques en ingénierie de production

Annexe C – Etude comparative des ateliers logiciels

On fait, défait, refait ce beau dictionnaire qui, toujours très bien fait, sera toujours à faire.

Ponce Denis ECOUCHARD-LEBRUN (1729-1807). Epigrammes.

Annexe A – Glossaire

Note : les définitions qui sont déjà regroupées dans des tableaux ou encarts ne sont pas reprises ici.

Bill of Materials (nomenclature) *An ordered list of the parts, sub-assemblies, and assemblies and raw materials that define a product. Normally created and maintained with the project structure management function, it defines the type, number, quantity, and relationships of parts and assemblies [CIM01]. A listing of the parts used in a manufacturing process along with the relation of one product to another insofar as assembly of the final product is concerned. The bill of material is a classical recursive structure [INM97].*

Business Process Re-engineering (reconfiguration des processus) *The fundamental rethinking and radical redesign of business processes to achieve dramatic improvements in critical, contemporary measures of performance, such as cost, quality, service and speed [HAM93].*

Computer-Aided Software Engineering (atelier de génie logiciel) *Tools that are software programs that automate or support the drawing and analysis of system models and provide for the translation of system models into application programs [WHI01]. Any computer and/or system which is specifically designed to support any sub-stage, stage or stages of the information systems development process of a computer-based information system or any aspect of the management of these tasks and processes [AVI00].*

Configuration Management (gestion des configurations) *The process of defining and controlling a product, its structure, associated documentation and other data (facilities, processes...). CM includes maintaining revision control and history information about all changes to a document, parts, assemblies and all other related data [CIM01].*

Cross-functional IS *System that supports relevant business processes from several business functions without regard to traditional organizations boundaries such as divisions, departments, centers, and offices [WHI01].*

Digital mock-up (maquette numérique) *Extension to 3D modeling capabilities that enable complete products to be built in electronic form. The mock-ups can be used to check for problems such as interference and clashes between components. Using digital mock-ups reduces the cost of development, as physical material models do not need to be built [CIM01].*

Enterprise Architecture (architecture d'entreprise) *The set of descriptive representations (ie, models) that are relevant for describing an Enterprise such that it can be produced to management's requirements (quality) and maintained over the period of its useful life (changed) [ZAC97].*

Enterprise Architecture Framework *The framework as it applies to Enterprise is simply a logical structure for classifying and organizing the descriptive representations of an enterprise that are significant to the management of the enterprise as well as to the development of the Enterprise's systems. It was derived from analogous structures that are found in the older disciplines of Architecture/Construction and Engineering/Manufacturing that classify and organize the design artefacts created over the process of designing and producing complex physical products (e.g. building or airplanes) [POP00].*

Enterprise Modeling (techniques de modélisation en entreprise) *Is the art of externalising enterprise knowledge which can be reused or add value to the enterprise... EM describes various aspects of enterprise structure, behaviour and organisation [VER96].*

Information Engineering (ingénierie de l'information) *A model driven and data-centered, but process-sensitive technique to plan, analyze, and design information systems [WHI01].*

Information System (Système d'information) *A system which assembles, stores, processes, delivers information relevant to an organisation, in such a way that information is accessible and useful to those who wish to use it, including managers, staff, clients and citizens. An information system is a human activity (social) system which may or may not involve the use of computer systems [BUC87].*

Interview (en ingénierie de domaine) *A fact-finding technique whereby the systems analysts collect information from individuals through face-to-face interaction [WHI01].*

Legacy Systems *Older Information System applications that are crucial day-to-day operation of a business, but which may use technologies considered old or outdated by current standards [WHI01].*

Method (méthodologie de développement des SI) *A recommended collection of philosophies, phases, procedures, rules, techniques, tools, documentation, management, and training for developers of information systems [MAD83].*

Outsourcing (acte d'externalisation) *The act of contracting with an outside vendor to assume responsibility for one or more IT functions or services. In many cases, ownership of IT assets (including technology and employees) is transferred to the outsourcer [WHI01].*

Package (paquetage) *A general purpose mechanism for organizing elements into groups. Packages may be nested within other packages [OMG99].*

Pattern (Patron objet) *An abstraction of a doublet, triplet, or other small grouping of classes that is likely to be helpful again and again in object-oriented development [COA92].*

cPDM (collaborative Product Definition management) *is a strategic business approach; applying a consistent set of business solutions that collaboratively manage the product definition lifecycle across the extended enterprise. These business solutions focus on the complete product definition, including all mechanical, electronic, software, and documentation components, and the processes that are used during the lifecycle [CIM01].*

Problem analysis (Analyse de problème) *The activity of identifying the problem, understanding the problem (including causes and effects), and understanding any constraints that may limit the solution [WHI01].*

Repository (encyclopédie ou dictionnaire de donnée) *A data store(s) of accumulated system knowledge-system models, detailed specifications, and any other documentation that has been accumulated during the system's development. This knowledge is reusable and critical to the production system's ongoing support. The repository is implemented with various automated tools, and it is often centralized as an enterprise business and IT resource [WHI01].*

Reverse Engineering (rétroingénierie) *Technology that reads the program code for an existing database, application program, and/or user interface and automatically generates the equivalent system model [WHI01].*

Scope (domaine) *The boundaries of a project-what part of the business is to be studied, analyzed, designed, constructed, implemented, and ultimately improved ? [WHI01].*

Timeboxing (développement incrémental) *A technique that delivers information systems functionality and requirements through versioning. The development team selects the smallest subset of the system that, if fully implemented, will return immediate value to the system owners and users. That subset is developed, ideally with a time frame of six to nine months or less. Subsequently, value-added versions of the system are developed in similar time frame [WHI01].*

View (vue d'un système) *A selective perception of the enterprise, which accentuate very much the special aspects and neglect others [AIT98].*

Workflow *The automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules [WFM96].*

Zachman Framework *Is a true framework (that is, based on rules of classification) which hypothesizes a comprehensive set of descriptive representations of an enterprise. It is based on analogous descriptive representations of the physical products found in older disciplines of classical architecture and/or manufacturing. The logic construct is a matrix comprised of perspectives (planner, owner, designer, builder, subcontractor and user) versus interrogatives (what, how, where, who, when, and why) [SOW92].*

Annexe B – Problématiques de l'ingénierie de production

Note : les tableaux suivants sont extraits de plaquettes commerciales de la société Matra Datavision, aujourd'hui rachetée par le groupe DELMIA / Dassault Systèmes.

Ils décrivent sous forme de questions clés, les problématiques en ingénierie de production et leur résolution à l'aide des outils IPAO (ingénierie de Process/Production Assistée par Ordinateur). Ces outils, servent à l'évaluation et à l'optimisation des activités process. Ils concernent particulièrement la conception et la simulation :

- Des flux et des lignes d'assemblages
- Robotique et la programmation hors ligne
- Des procédés manuels et automatisés
- De l'implantation des lignes et des ressources etc.

Analyse des flux

Problématiques	Avantages de la simulation	Bénéfices industriels
Comment chiffrer le gain d'une plus grande polyvalence des opérateurs ?	Possibilité de simuler différentes catégories d'opérateurs.	Améliorer le lissage de la charge de travail.
Quels sont les taux d'utilisation des ressources ?	Edition des rapports d'analyse et des statistiques (courbes)	Quantifier rigoureusement l'utilisation des ressources autorisant des affectations rationnelles
Quelles sont les quantités produites ?	Visualisation du nombre d'articles produits et des encours en temps réel.	Valider la capacité de production d'un système industriel.
Comment optimiser les stocks tampons ?	Lancement de plusieurs scénarios intégrant des situations de crise (taux de retouche élevé, pannes longues...)	Déterminer les stocks au juste nécessaire. Traitement automatique des résultats.
Quel est l'impact du mix produits et des aléas de production (pannes) ?	Prise en compte des phénomènes aléatoires par l'intermédiaire de distributions paramétrables.	Prendre en compte les variations de la demande client dans la définition du système de production. Formation des opérateurs aux conséquences des pannes.
Quel est le meilleur scénario de production ou comment optimiser la "fonction objectif" ?	De manière traditionnelle, par itération. A l'aide d'un optimiseur permettant dans certains cas de converger vers une solution "acceptable".	Détecter les cas aux limites, les configurations défavorables du système industriel résultant de la conjonction de certains paramètres.
Comment modifier un système existant ? (exemple : montée en régime d'un nouveau produit)	Description des données du système nécessaires à la modélisation et indispensables pour l'équipe projet	Synthétiser l'ensemble des paramètres dimensionnants du système.
Comment concevoir une nouvelle ligne de production ?	Conception non fondée sur l'extrapolation d'un système existant auquel on applique une marge de sécurité	Diminuer les coûts des systèmes qui ne sont plus inutilement surdimensionnés.

Assemblage

Problématiques	Avantages de la simulation	Bénéfices industriels
Comment réduire les coûts des prototypes et des tests ?	Complément de la maquette numérique BE par des données de production	Réduire voire supprimer les prototypes et les tests.
Comment faire pour que les différents acteurs (BE, atelier, SAV) parlent le même langage ?	Simulation sur le même outil, sur les mêmes données et dans le même temps, permettant d'assurer le partage d'informations cohérentes.	Impliquer les différents acteurs dès la genèse du produit apporte une véritable interactivité au sein de l'entreprise.
L'opération de montage est-elle réalisable en tenant compte des contraintes spécifiques ?	Garantie de la faisabilité du montage par les trajectoires et les zones enveloppes clairement matérialisées et par les fonctions d'évitement automatique combinées à la détection dynamique des collisions.	Minimiser les temps de mise au point. Améliorer la qualité.
Comment optimiser la séquence de montage d'un matériel en fonction des ressources employées ?	Association aisée de tout type de ressources aux produits par le biais d'un environnement IPAO intégré.	Adéquation optimale des ressources pour une opération donnée.
Comment prévoir les zones de débatement des câbles et autres flexibles	Modélisation fidèle des flexibles permettant d'anticiper leur comportement de façon réaliste.	Supprimer les modifications tardives. Assurer la sécurité des personnels.
Comment diffuser les documents d'instruction de montage ou les fiches de maintenance ?	Mise à disposition d'outils de communication puissants (vidéos de montage, films de maintenance)	Assurer la diffusion de l'information de façon fiable et cohérente.

Simulation des moyens robotisés

Problématiques	Avantages de la simulation	Bénéfices industriels
Comment valider le process envisagé ?	Simulations des ressources utiles à l'enchaînement des opérations (robots, opérateurs...). Simulation de la logistique interne (approvisionnement de pièces, ...)	Faire bien du premier coup
Comment réaliser une implantation optimale dans l'espace imparti ?	Placement automatique des robots en fonction des contraintes environnementales. Exploitation directe des données CAO natives.	Optimisation de l'allocation des espaces utiles. Suppression des échanges de données.
Comment vérifier les temps de cycle avec certitude ?	Mise à disposition de la signature robot.	Réduction des phases test.
Comment choisir l'outil adapté ?	Recherche automatique intelligente multicritères.	Gain de temps dans la conception de la cellule.
Comment programmer les robots sans arrêter la production ?	Programmation Hors-Ligne des robots (PHL)	Réduction des temps d'intervention sur ligne.
Comment exploiter et capitaliser les informations ?	Simulation sur le même outil, les mêmes données et dans le même temps pour assurer le partage d'informations cohérentes.	Enrichissement de la base de connaissances : produit, process et ressources.

Simulation des procédés manuels

Problématiques	Avantages de la simulation	Bénéfices industriels
Comment implanter le poste de travail afin d'optimiser les tâches ?	Prise en compte des données physiques (morphologie, champ de vision, étude des efforts, des fatigues, des postures, de la dépense énergétique).	Valider un process avant investissement.
Respecte-t-on la législation sur les conditions de travail ?		Maîtriser les temps opératoires. Valider l'implantation du poste.
Les outillages utilisés sont-ils efficaces ?	Intégration des gestes dès la conception (temps de cycle, accessibilité, allocation d'espaces, coordination tâches manuelles, tâches mécanisées, vidéos éducatives)	Eviter la création de maquette onéreuses.
Le poste de travail est-il adapté ?		Disposer d'un poste de travail plus confortable pour les opérateurs, ce qui réduira la fatigue, les erreurs, l'absentéisme.
L'objet est-il accessible ?		Augmenter la productivité
Quels sont les temps opératoires ?		Assurer la maintenance des matériels.
Les actions de maintenance sont-elles possibles ?		

Annexe C – Eléments sur l'étude comparative des AGL

Note : cette étude a été effectuée tout au début de notre thèse (Avril 1999). Il est inutile de présenter les résultats des évaluations ; les produits et le marché évoluent en permanence et de ce fait ces résultats peuvent paraître obsolètes aujourd'hui. Néanmoins, nous voulons récapituler ici les aspects plus ou moins pérennes. Nous décrivons, en premier, une classification arbitraire des AGL suivant les fonctionnalités qu'ils offrent (1). La présentation de chaque outil repose sur plusieurs paramètres que nous évoquons sommairement (2). Nous traduisons par la suite les critères d'évaluation de ces outils (3). Ces derniers peuvent bien entendu évoluer en fonction des besoins du projet. Nous donnons les adresses des sociétés spécialisées dans l'évaluation de ces progiciels dont certaines ont été sollicitées pour effectuer notre étude (4). Enfin, nous listons les adresses de quelques AGL des plus populaires (5).

1. Classification des AGL

▪ Outils de dessin

Ils sont limités aux fonctions graphiques de dessin ; ils ne disposent pas d'encyclopédie ou de notations standards. Beaucoup d'AGL ont débuté avec ces fonctions rudimentaires pour évoluer plus tard vers les outils suivants.

▪ Outils de modélisation et de simulation

D'origine manufacturière, beaucoup d'entre eux sont pauvres en aspects de modélisation. Une collecte fastidieuse de données est nécessaire avant toute modélisation et simulation de modèles.

▪ Outils de workflow et de BPR

Une partie de ces outils est plutôt destinée à la modélisation et la simulation des processus administratifs ; elle demeure inefficace face aux processus complexes des activités d'ingénierie. L'autre partie supporte les nouvelles techniques d'entreprise modeling, plus appropriées pour ce type d'activités.

▪ Outils de conception des SI

Ces outils sont spécialement dédiés pour accompagner les projets de développement des SI. Chaque AGL peut supporter une ou plusieurs notations des modèles : OO (notamment la dernière notation UML), de données (E-R...), d'analyse structurée... Cependant, beaucoup d'AGL de cette catégorie n'offrent pas la totalité des fonctions requises pour mener à bien ces projets. Quelques-unes de ces fonctions sont décrites ci-après.

2. Description des AGL

▪ Présentation

Nous devons avoir des informations concernant l'image et la position de la société conceptrice de l'AGL, les distributeurs de celui-ci, son coût, les prestations du SAV...

▪ Description des modules principaux du produit

Il faut décrire le produit et ses modules quand il en dispose, les points forts et les points faibles des fonctions qu'il propose etc.

- **Environnements supportés et architecture informatique**

Nous devons étudier la plate-forme du produit ainsi que la procédure de sa mise en œuvre en entreprise. Il faut penser également à connaître ses interfaces avec les environnements informatiques, son interface utilisateur etc.

- **Notations et modèles supportés**

Les diagrammes de spécifications dont dispose un AGL doivent être évalués par rapport aux normes de qualité et aux standards des méthodes et des techniques d'ingénierie des SI qui les représentent.

- **Autres fonctions de conduite de projet**

Citons la génération de code, la rétroingénierie (reverse engineering), l'édition des rapports et la documentation... Nous pouvons également évaluer la robustesse de l'encyclopédie quand elle est présente dans l'AGL, la nature des métriques disponibles etc.

Toutes ces caractéristiques peuvent être groupées, priorisées afin de former les critères d'évaluation des AGL que nous présentons par la suite.

3. Critères d'évaluation et indicateurs de mesure

Les critères ci-après sont plus ou moins universels. Les choix de progiciels se distinguent le plus souvent par la priorisation de ces critères.

- **Qualité des modèles**

Elle se mesure de diverses façons : les métriques, les fonctions de simulation et le respect des standards des notations. Ces indicateurs de mesure garantissent la cohérence et l'intégrité des modèles. En effet, les problèmes d'intégrité peuvent empêcher ou altérer la génération de code et conduire à des contradictions.

- **Facilité d'adoption**

L'AGL doit être facile à utiliser. Son environnement de conception doit être convivial et facile à personnaliser. La productivité des concepteurs peut être ainsi améliorée.

La mise en œuvre de l'AGL doit être simplifiée ; les règles de cohérence et d'intégrité devraient être personnalisées suivant le caractère et l'exigence du projet.

- **Gestion du travail en équipe**

Pour mener à bien le travail en équipe, il faut contrôler l'accès à l'AGL en fonction des responsabilités et des droits tout en éliminant les conflits multi-utilisateurs.

De plus, l'architecture informatique doit favoriser le travail collaboratif en partageant des éléments de modélisation. Cela ne peut se faire sans la présence d'une encyclopédie.

- **Conservation de l'information entre les étapes**

Grâce aussi à la présence d'encyclopédie, une démarche itérative est possible, garantissant la traçabilité des informations. La documentation quand elle est automatique facilite considérablement la conservation des informations entre les étapes de développement.

- **Complétude de modélisation**

Les aspects de modélisation ne sont pris en compte qu'en intégrant les notations et les diagrammes de spécification les plus significatifs : l'orienté-objet, la modélisation E-R, les techniques de modélisation en entreprise (enterprise modeling), les techniques d'analyse structurée...

Il faut également intégrer les nombreuses fonctions pouvant intervenir en projet : la rétroingénierie (reverse engineering, lecture de code) afin de prendre en compte les

composants existants, la génération de code, de schéma de BdD pour pouvoir effectuer des tests et les prototypes intermédiaires...

4. Sociétés de conseil et organismes spécialisés

AT KEARNEY : <http://www.atkearney.fr/>

BUSINESSWARE : <http://www.bizware.com/>

CXP : <http://www.cxp.fr/>

GARTNER GROUP : <http://www4.gartner.com/Init>

ORGAconseil : <http://www.orgaconseil.com/>

Process Product Watch : <http://www.enix.co.uk/ppw.htm>

SODAN : <http://www.sodan.co.uk/bpmodel.htm>

Yphise : <http://www.yphise.com/>

Université de Toronto – laboratoire d'intégration d'entreprise : <http://www.ie.utoronto.ca/>

Université de Warwick – Business Process Ressource Center : bprc.warwick.ac.uk/bp-tool.html

Fraunhofer Institute for Computer Graphics : www-a5.igd.fhg.de/reuse_m/bpr_description.html

Quelques logiciels testés par des consultants : www.sellers-associates.com/products.htm

5. Quelques outils de modélisation BPR et de conception de SI

ArchétypeSeer : <http://www.seer.com/>

ARIS Toolset - IDS : <http://www.ids-scheer.de/>

Bonapart UBIS : <http://www.ubis.de/>

Cimtool : <http://www.rgcp.com/pub/download/cimtool.zip>

CoolJEX - Sterling Software : <http://www.cool.sterling.com/> ou <http://www.sterling.com/>

Design CPN : <http://www.daimi.aau.dk/>

Designer 2000 – Oracle : <http://www.oracle.com/>

Extend+BPR - Imagine That : <http://www.imaginetthatinc.com/>

FirstStep – Interfacing Tech : <http://www.interfacing.com/>

GD Pro : <http://www.gdpro.com/>

Grade Modeler DATI – Infologistik : <http://www.latnet.lv/lu/mii/grade/vecais.htm>

GroundWorks – Cayenne : <http://www.cayennesoft.com/>

Imagine That : <http://www.imaginetthatinc.com/>

IDEF (suite) - Kbsi : <http://www.kbsi.com/>

ISOA : <http://www.mega.com/>

Live Model Intelicorp : <http://www.intellicorp.com/>

Modeler – CaseWise : <http://www.casewise.com/>

Objecteering : <http://www.softeam.com/> ou <http://www.objecteering.com/>

Paradigm+ : <http://www.cai.com/>

Ptech : <http://www.ptech.com/>

Primeobjects : bruno@polito.it

Process Engineer - LBMS : <http://www.lbms.com/>

Provision Workbench - Proforma : <http://www.proformacorp.com/>

Rational Rose : <http://www.rational.fr/>

Rethink - Gensym : <http://www.gensym.com/productoverview/rethink.html>

Select Enterprise : <http://www.princetonsofttech.com/>

SpecWare Kestrel : <http://www.kestrel.edu/>

System Architect - Popkin Software : <http://www.popkin.com/>

Workflow Modeler : <http://www.metasoftware.com/>

De la gestion des données techniques pour l'ingénierie de production

Référentiel du domaine et cadre méthodologique pour l'ingénierie des systèmes d'information techniques en entreprise

Résumé : ce travail traite des problèmes méthodologiques dans les projets de mise en œuvre des Systèmes d'Information Techniques (SIT) en entreprise. Il a pour champ d'application le domaine de l'ingénierie de production. L'objectif est l'amélioration de deux aspects de ces problèmes : le cahier des charges de la maîtrise d'ouvrage et de la maîtrise d'œuvre, et la démarche globale de conduite de ces projets. Pour ce faire, nous proposons un référentiel du domaine, élaboré à l'aide de diagrammes de spécification standards. Orienté expression des besoins en GDT, le référentiel est réutilisable dans les développements futurs de SIT en phase d'analyse du domaine. La démarche d'élaboration du référentiel est rationalisée ; elle se veut un guide méthodologique pour l'ingénierie même des SIT. Elle implique les principaux acteurs suivant un processus itératif. Cette démarche est de plus contextuelle pour mieux appréhender les singularités des projets et suffisamment flexible pour respecter le caractère créatif des métiers d'ingénierie.

Mots clés : ingénierie des Systèmes d'Information Techniques (SIT), ingénierie de production, Gestion des Données Techniques (GDT).

Technical Data Management for Process Engineering

Reference and Methodological Framework for Technical Information Systems Engineering

Abstract This research deals with the methodological problems involved in technical Information Systems (IS) engineering within manufacturing company. The application area is the Process Engineering. The goal is the improvement of two aspects of these problems: the technical requirements of industrial agents, and the global guideline of such projects. With this intention, a reference frame is elaborated that uses standard specification diagrams. The reference frame formalises PDM needs. Thus, it is reusable in future developments of IS, especially in the domain analysis stage. Its development process is rationalised. It may be even used as a methodological guideline for generic IS engineering. The reference frame implementation involves the main concerned agents following an iterative process. Moreover, this approach is contextual for better apprehending project singularities and it is flexible enough to respect the creative character of engineering works.

Keywords Information System (IS) Engineering, Product/Process Data Management (PDM), Manufacturing Engineering.