



HAL
open science

PROPOSITION D'UN CADRE DE MODÉLISATION MULTI-ÉCHELLES D'UN SYSTÈME D'INFORMATION EN ENTREPRISE CENTRÉ SUR LE PRODUIT

Jean-Philippe Auzelle

► **To cite this version:**

Jean-Philippe Auzelle. PROPOSITION D'UN CADRE DE MODÉLISATION MULTI-ÉCHELLES D'UN SYSTÈME D'INFORMATION EN ENTREPRISE CENTRÉ SUR LE PRODUIT. Modélisation et simulation. Université Henri Poincaré - Nancy I, 2009. Français. NNT: . tel-00371290v1

HAL Id: tel-00371290

<https://theses.hal.science/tel-00371290v1>

Submitted on 27 Mar 2009 (v1), last revised 27 Mar 2009 (v2)

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.

U.F.R. Sciences et Techniques Mathématiques, Informatique et Automatique

École Doctorale IAEM Lorraine

Département de Formation Doctorale Automatique

Thèse

Présentée pour l'obtention du titre de

Docteur de l'Université Henri Poincaré, Nancy I

en Automatique, Traitement du Signal, Génie Informatique

par **Jean-Philippe AUZELLE**

PROPOSITION D'UN CADRE DE MODÉLISATION MULTI-ÉCHELLES D'UN SYSTÈME D'INFORMATION EN ENTREPRISE CENTRÉ SUR LE PRODUIT.

Soutenue publiquement le 11 mars 2009 devant la commission d'examen :

Membres du jury :

Rapporteurs :	M. David Chen	Professeur, Université Bordeaux 1, Président du jury
	M. Angel Ortiz Bas	Professeur, Université Polytechnique de Valencia
Examineurs :	M. Dominique Luzeaux	Directeur adjoint de l'UMESIO à la DGA (Délégation Générale pour l'Armement), HdR
	M. Gérard Morel	Professeur, UHP Nancy I (Directeur de thèse)
	M. Hervé Panetto	Professeur, UHP Nancy I (Codirecteur de thèse)
	Mme. Frédérique Mayer	Maître de conférences, INPL (Codirectrice de thèse)
Invité :	M. François Vernadat	Professeur, Université Paul Verlaine de Metz

Je tiens à remercier très sincèrement mon directeur de thèse, Gérard Morel, pour la confiance qu'il a placée en moi, en m'accompagnant dans mon apprentissage de 'jeune' chercheur. Avoir été formé sous sa direction, a été une expérience extrêmement riche, qui a permis l'aboutissement de mes travaux après de nombreux échanges passionnés et passionnants, et qui m'a ouvert de nouveaux horizons intellectuels ... et professionnels. Bien entendu, je remercie Hervé Panetto, mon co-directeur, qui m'a soutenu jusqu'au bout en m'apportant de précieux conseils, et qui a toujours été vigilant sur la qualité des modèles que je proposais... et aussi sur la « qualité » de mon Anglais encore largement perfectible, isn't it ? Un très grand merci à Frédérique Mayer, qui m'a encadré en DEA, puis encouragé à continuer en thèse en me co-encadrant. J'ai pu ainsi, grâce à elle, enrichir mes connaissances en modélisation d'entreprise durant ces dernières années ... sachant qu'il reste encore beaucoup de chemin à parcourir et qu'il reste aussi encore beaucoup de concepts à faire « émerger » dans ce domaine.

Merci aussi à Angel Ortiz Bas et David Chen qui ont accepté de rapporter sur mon travail de thèse, et qui ont ainsi contribué chacun à apporter une valeur ajoutée à mes travaux. Je remercie aussi Dominique Luzeaux et François Vernadat qui ont examiné mon travail avec leur regard d'expert avisé.

D'autre part, rien n'aurait été possible sans l'effervescence intellectuelle et amicale qui anime le laboratoire du CRAN, qui a fait émerger certaines idées dans mes travaux de recherche, notamment à la BSR (Brain Storming Room). C'est ainsi que tout naturellement, je remercie non seulement l'ensemble des chercheurs seniors, des doctorants, mais aussi l'ensemble des personnels du laboratoire qui ont apporté, directement ou indirectement, aussi leur part de contribution à ce mémoire, par leurs compétences, leur disponibilité et leur gentillesse.

Je tiens aussi à remercier l'ensemble du personnel de l'AIPL, au travers de son directeur, Jean-Yves Bron, qui a mis à ma disposition un environnement d'étude parfaitement adéquat et opérationnel pour mes recherches. Je remercie sincèrement Olivier Nartz, pour sa collaboration et sa forte implication dans le cas d'étude de mon mémoire, et qui a ainsi grandement participé à la mise au point de ma proposition méthodologique.

Un merci particulier aussi aux membres du groupe de travail de l'AFIS sur les Systèmes de Systèmes et les Systèmes Complexes, avec qui j'ai eu de nombreux échanges fructueux et riches, et qui m'ont permis de mieux cerner certaines problématiques industrielles.

Un remerciement particulier à Sophie, qui m'a toujours soutenu et qui m'a permis de tenir le cap jusqu'au bout, même dans les moments difficiles. Merci à Flora et Jules qui ont su patienter pour avoir un papa plus disponible.

Merci aussi à mes parents, qui par leur éducation et leur amour, m'ont transmis les valeurs fondamentales qui me structurent au quotidien.

SOMMAIRE

Sommaire	5
Introduction générale & contexte de reflexion de la thèse	7
Introduction générale et démarche de réflexion	8
Références bibliographique de l'introduction générale	20
Chapitre 1. D'un Système d'Information à un Système de Systèmes d'Information	24
Introduction du chapitre 1	25
1 Le système d'information d'entreprise vu comme un Système de Systèmes d'Information	27
1.1 Caractérisation d'un SdSI comme un SDS	30
1.2 Caractérisation de la relation d'interopérabilité dans un SdSI	34
2 Proposition de Formalisation d'un SdS	37
2.1 Formalisation de la démarche de construction d'un SdSI.....	38
2.2 Formalisation d'un SdSI.....	41
Conclusion du chapitre 1.....	49
Chapitre 2. ... Au Système de systèmes d'ingénierie pour faire un système de systèmes d'information	52
Introduction du chapitre 2	53
1 De l'ingénierie d'un SI à l'Ingénierie d'un SdSI	53
1.1 Ingénierie Dirigée par les processus – les modèles.....	56
1.2 Ingénierie Dirigée par les Modèles (IDM)	57
1.3 Ingénierie Système Basée sur les Modèles (ISBM).....	57
1.4 Les cadres d'architectures.....	58
1.5 Cadre de modélisation pour une Ingénierie basée sur les modèles d'un SdSI	66
2 Proposition d'une Ingénierie Système Basée sur les Modèles.....	66
2.1 Système pour Faire de l'Ingénierie de notre ISBM.....	67
2.2 ISBM d'un Système à Faire : instanciation d'un cadre de modélisation de Zachman.....	85
3 Proposition d'un prototype d'outillage pour notre ISBM	89
3.1 Atelier logiciel pour la capitalisation et la cohérence de la modélisation.....	89
3.2 Prototype d'Outillage de modélisation	89
Conclusions du chapitre 2	93
Chapitre 3. Scénario de mise en oeuvre	96
Introduction et contexte de l'étude globale	97
1 Environnement du système à faire	97

2	Ingénierie Système Basée sur les Modèles du projet « E-Production ».....	101
2.1	Préparation du projet (MOA)	102
2.2	Définition du SAF « système de traçabilité du produit à l’AIPL» (MOE de rang 1)	108
2.3	Spécification du SAF « système de traçabilité du produit à l’AIPL» : processus de la réception matières (MOE de rang 2).....	116
	Conclusion du chapitre 3.....	126
	Conclusion générale & perspectives	128
1	Conclusion générale	129
2	Perspectives.....	131
2.1	Méthodes et outils de l’interopérabilité de SdS.....	131
2.2	Le produit-actif pivot de l’interopération des ingénieries au sein d’un SdS	133
2.3	Perspectives de mise en application de l’ISBM : L’Université Henri Poincaré de Nancy	137
	Acronymes.....	140
	Index des illustrations	143
	Index des tableaux.....	145
	Références.....	146

**INTRODUCTION GÉNÉRALE
&
CONTEXTE DE REFLEXION DE LA THÈSE**

Introduction générale et démarche de réflexion

Mes travaux portent sur la **modélisation d'Entreprise**, et plus précisément, sur une vision **Ingénierie Système** de celle-ci afin de faciliter son application en Entreprise.

Modélisation d'Entreprise : (Verdanat 1996)

"Enterprise modelling is a generic term which covers the set of activities, methods, and tool related to developing models for various aspects of an enterprise."

Systems Engineering: (INCOSE 2006)

"Systems Engineering is an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, and then proceeding with design synthesis and system validation while considering the complete problem. Systems Engineering considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs."

Ingénierie Système : (AFIS 2009)

« L'Ingénierie Système est une démarche méthodologique coopérative et interdisciplinaire qui englobe l'ensemble des activités adéquates pour concevoir, développer, faire évoluer et vérifier un ensemble de produits, processus et compétences humaines apportant une solution économique et performante aux besoins des parties prenantes et acceptable par tous (inspirée de IEEE 1220). Cet ensemble est intégré en un système, dans un contexte de recherche d'équilibre et d'optimisation sur tout son cycle de vie. »

Alors que la modélisation d'entreprise arrive à maturité scientifique (Vallespir 2003; Chen 2005; Panetto 2006) et technologique (Ferrarini et al. 2006; Berre and al. 2007), force est de constater que sa pratique est encore trop peu courante, comme nous l'avons remarqué auprès de divers groupes industriels impliqués dans les travaux de recherche du laboratoire ainsi que dans le projet d'unification du système d'information de notre Université dans lequel nous sommes impliqués.

Constat (Baptiste et al. 2007)

Nous arrivons au moment où la modélisation d'entreprise, au sens large de l'ensemble des modèles nécessaires au pilotage des organisations du court au long terme, pénètre enfin l'entreprise et bientôt les administrations, à la satisfaction de la communauté internationale des chercheurs qui, depuis deux décennies au moins, en assurent la promotion."

Mais aussi :

Constat (Chapurlat 2007)

« Il n'existe pas à proprement parler de tâche de modélisation, de maintenance et de gestion de modèle dans l'entreprise. Un modèle est bâti à la demande et considéré à priori comme inutilisable en dehors du périmètre du projet pour lequel il a été construit. De plus, le travail de modélisation est généralement fait par un acteur extérieur, consultant en organisation ou chargé de l'installation d'un ERP par exemple. Le résultat reste donc difficilement exploitable par les acteurs de l'entreprise. »

Du point de vue de la **maîtrise d'ouvrage** (MOA) d'un projet à laquelle nous nous intéressons, cette vision 'Ingénierie Système de la modélisation en Entreprise' (Figure 1) devrait englober, dans un Système-Entreprise, les différentes vues relatives au Système à Faire évoluer et au Système-Projet qui lui est associé pour répondre (Faire) à un besoin exprimé de façon interne (les exploitants) ou externe (les clients).

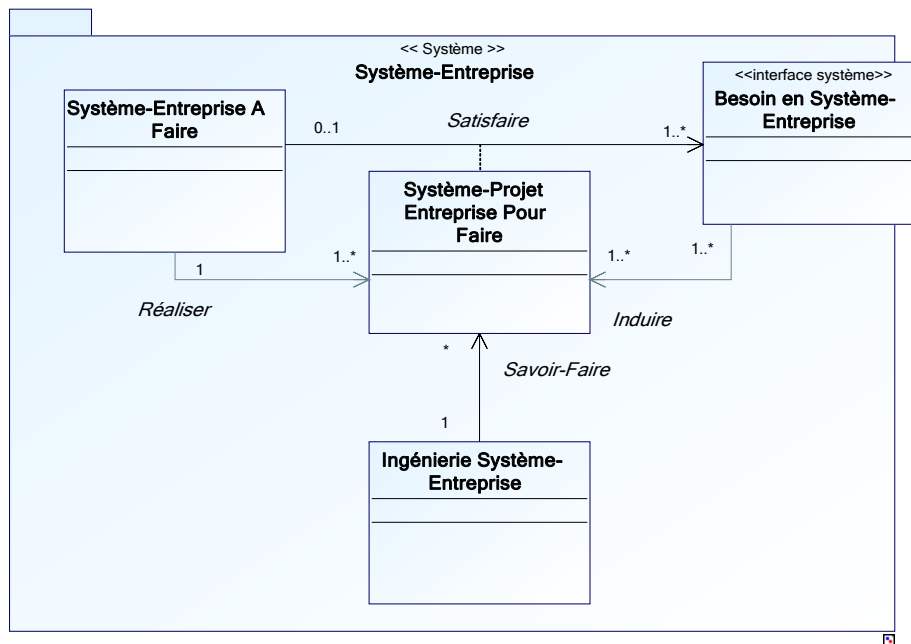


Figure 1. Vision Ingénierie Système en Maîtrise d'Ouvrage de la modélisation en Entreprise adapté de [AFIS 2008]

Cependant, lors de ses évolutions, ce Système-Entreprise est le plus souvent composé de façon ad-hoc de sous-systèmes (composants), entre autres de **COTS**, qui ont pour particularité d'encapsuler leur propre Savoir-Faire (ingénierie), rendant dès lors difficile, pour le Système-Entreprise, sa maîtrise à travers le temps.

Définition d'un COTS : (Source Wikipedia)

"COTS (Commercial off-the-shelf) is a term for software or hardware, generally technology or computer products, that are ready-made and available for sale, lease, or license to the general public ...The motivation for using COTS components is that they will reduce overall system development costs and involve less development time because the components can be bought or licensed instead of being developed from scratch ... However, since COTS software specifications are written by external sources, government agencies are sometimes wary of these products because they fear that future changes to the product will not be under their control."

Mais, si couramment le **Système-COTS** réalisé devient une partie tangible du Système-Entreprise, il n'en est pas de même pour l'**Ingénierie-COTS** dont l'interface avec une Ingénierie unifiée du Système-Entreprise (Figure 2: relation de composition) n'est pas explicitement requise (Figure 2 : relation en pointillé) de façon systématique.

Un premier objectif de nos travaux est donc de supporter cette Ingénierie Système d'Entreprise par un cadre de modélisation suffisamment complet mais sans être compliqué pour limiter 'à minima' l'effort requis de modélisation en entreprise. Notre choix s'est porté sur le **cadre de modélisation de Zachman** (Sowa and Zachman 1992), qui outre son usage bien établi en BPE (Business Process Engineering), s'appuie, d'une part sur un questionnement d'un problème (en colonne) proche de celui mis en œuvre en démarche qualité et, d'autre part sur des niveaux d'abstraction (en ligne) proche de ceux mis en œuvre dans une démarche systémique (de type MERISE). Une analyse plus fine, relative notamment au positionnement entre l'ingénierie de processus métiers avec BPE et l'ingénierie basée sur des modèles avec UML (ISO/IEC-19501:2005 2005), SysML¹, ou tout autre langage de modélisation², nous permet de définir ce 'minima' aux niveaux contextuel et conceptuel (planificateur/propriétaire) du cadre de modélisation de Zachman.

Le cadre de modélisation de Zachman : (MEGA Modelling Suite³)

« Le cadre pour l'architecture des entreprises introduit en 1987 par John Zachman propose une structure logique pour classer et organiser les représentations de l'entreprise qui sont d'intérêt aussi bien pour sa direction que pour le développement de ses systèmes. Il trouve son origine dans des structures analogues que l'on trouve dans des disciplines plus anciennes telles que l'architecture, la construction, l'ingénierie ou l'industrie et qui classent et organisent les éléments créés durant le processus de conception et de fabrication de produits techniques complexes (ex : bâtiment, aéronautique) ».

¹ SysML 1.1 specification, OMG, 2008, <http://www.omg.org/spec/SysML/1.1/>

² Dans nos travaux, nous avons choisi UML comme langage pour la structuration de notre raisonnement.

³ MEGA International : <http://www.mega.com>

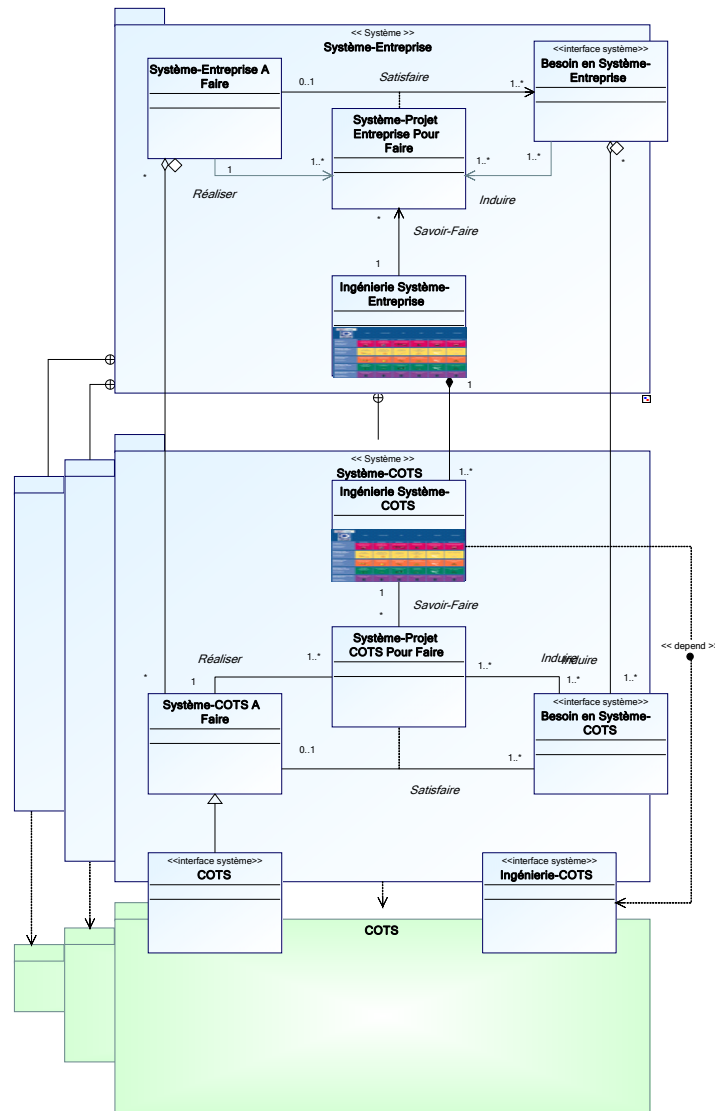


Figure 2. Vision en Maîtrise d’Ouvrage de l’Ingénierie d’un Système-Entreprise

La question s’est alors posée de la relation d’interopération entre ces systèmes (A Faire⁴, Pour Faire⁵, COTS, ...) et leurs Ingénieries respectives, en incluant le Système-Produit (son besoin, son projet, ...) à la source de chaque **recomposition ‘à la demande’ du Système-Entreprise**.

⁴ Le Système À Faire : « Le système est un concept, une abstraction, le résultat de sa réalisation est un produit (Produit-Système). Un système peut être réalisé en un ou plusieurs exemplaires. Ces exemplaires peuvent éventuellement différer, selon des variantes prévues par la définition du système. » (AFIS 2005)

⁵ Le Système Pour Faire : « Pour organiser, exécuter et coordonner toutes les activités qui conduisent à la réalisation et à la mise à disposition du système à faire, il est nécessaire de mettre en place un système doté de ressources humaines et techniques : le système pour faire ou projet. » (AFIS 2005)

Constat (Baptiste et al. 2007)

« L'évolution actuelle la plus marquante – du point de vue organisationnel – est liée au déploiement des partenariats d'entreprises pour le développement de produits à forte valeur ajoutée et/ou pour leur production (« supply chains »). Dans cette évolution, le paradigme d'intégration qui prévalait jusqu'ici à l'intérieur de l'entreprise (organisation cohérente de moyens visant à l'optimalité globale du système, concept CIM) montre aujourd'hui ses limites et cède le pas à l'interopérabilité des organisations, des processus et des outils informatiques de gestion qui se trouvent interconnectés, en dépit de leur caractère souvent hétérogène, en vue d'une activité collaborative – durable ou éphémère – de création de valeur. Le saut du système intégré au système de systèmes interopérables s'illustre ainsi clairement dans l'émergence des réseaux d'entreprises. »

Cette évolution organisationnelle du Système-Entreprise est aussi à la base du projet '**Systèmes Contrôlés par le Produit**' (SCP) du CRAN (rapport d'activité 2004-2007) dans lequel se sont développés nos travaux.

(Rapport d'activité 2004-2007 du Groupe Thématique 'Systèmes de Production Ambiants') :

« Ce projet s'appuie sur une interprétation du paradigme 'Holon Manufacturing Systems' (HSM) de l'initiative 'Intelligent Manufacturing Systems' (IMS) en phase avec les travaux du projet Européen 'FP6/PABADIS/PROMISE' (PABADIS/PROMISE 2006) pour expérimenter l'intérêt de rendre le produit et, au-delà l'ensemble du procédé, interactif afin d'organiser de façon plus collaborative l'interopérationalité des différents systèmes hétérogènes de pilotage et de gestion (APS, ERP, MES, SCEM) composant la chaîne de production et de logistique des entreprises. Cette interprétation tire parti des progrès et de la miniaturisation croissante des technologies infotroniques (RFID, communications sans-fils, etc.) et mécatroniques (composants logiciels embarqués) pour faire de ce produit actif un objet composite assurant une relation récursive logiciel-matériel entre les services et les biens associés aux produits en reliant tout objet logique de contrôle ou de gestion à au moins un objet physique du procédé. Ce paradigme SCP (Figure 3) est une alternative pragmatique pour permettre aux entreprises de migrer d'une organisation intégrée vers une organisation adaptable puisque l'originalité de notre approche est de combiner des décisions centralisées prises a priori sur des horizons à moyen, voire à long terme, avec des décisions décentralisées prises 'en exécution' sur des horizons à court terme, voire sur événements. SCP a ainsi un trouvé un écho important auprès d'industriels pour résoudre un verrou technologique majeur au moment où le paradigme d'intégration visant à l'optimalité globale du système entreprise montre ses limites pour re-synchroniser les flux physiques et d'informations (de décisions) en prenant en compte la gestion événementielle des problèmes de l'ensemble de la chaîne de production et de logistique, ou pour reconfigurer les systèmes existants afin de déployer de nouvelles applications telles que la traçabilité de produits tout au long de leur cycle de vie. »

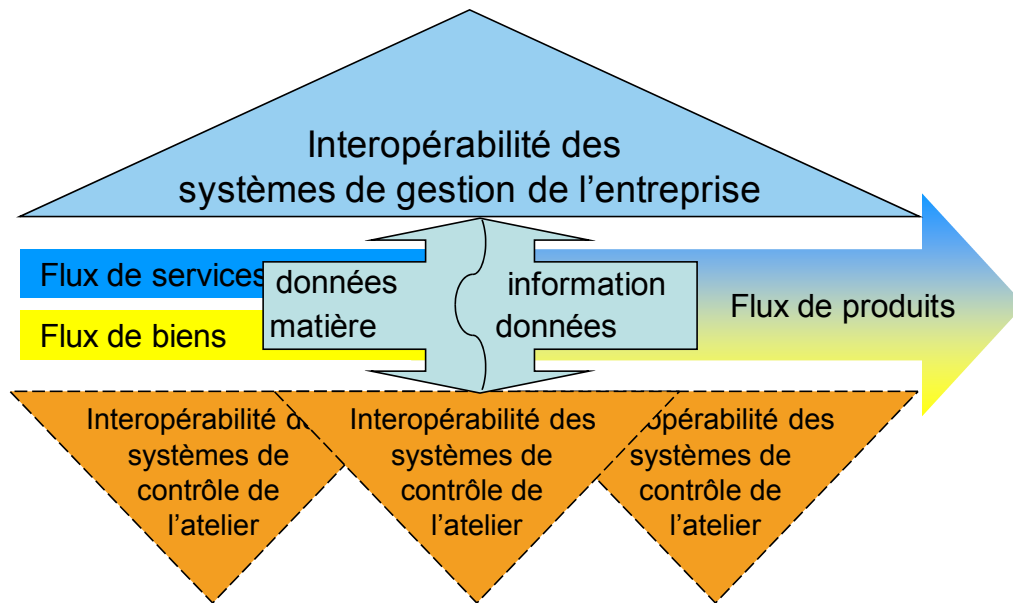


Figure 3. Product-driven manufacturing enterprise-wide control (Morel et al. 2007b)

Ces travaux complètent ainsi le cadre courant de l'**Interopérabilité** d'Entreprise (Man-Sze. Li et al. 2006) en centrant autour du produit (finalité retenue du Système-entreprise) l'implémentation par des systèmes de la **relation d'interopération**.

Constat (Baptiste et al. 2007)

« L'interopérabilité, concept informatique caractérisant la compatibilité et l'interconnectivité d'architectures logicielles et matérielles hétérogènes, est en passe de devenir une philosophie plus générale de conception de systèmes de systèmes. Dans le domaine Productique, sa portée a permis d'établir un pont entre l'interconnectivité des solutions informatiques hétérogènes d'entreprise (systèmes d'informations, bases de données, ERP), et l'interconnectivité des processus fonctionnels de création de valeur ajoutée (business models) mis en œuvre au sein d'un réseau hétérogène d'acteurs économiques (donneurs d'ordres, sous traitants, Fournisseurs). »

Relation d'interopération : (Carney et al. 2005)

"The essence of interoperation is that it is a relationship between systems, where systems are the entities in the above definition. While our focus will be on computer-based systems, the definition extends to beyond the world of mechanical systems to organizational and other contexts. To interoperate one system must provide a service¹ that is used by another. This cannot be achieved without, at a minimum, communication from the provider to the consumer of the service. Our focus is the relationship and not the manner of communication."

Le deuxième objectif de nos travaux est donc de faciliter l'interopération entre les ingénieries de ces systèmes interopérants (autour du produit A Faire) et l'Ingénierie du Système-Projet d'Entreprise (Pour Faire), par exemple pour des applications de type PLM (Terzi 2005).

Il nous est alors apparu que la définition couramment admise d'un système ne suffisait pas à refléter les aspects **récurifs** et **multi-échelles** de ce contexte organisationnel évolutif, dont nous avons réduit la complexité aux seuls aspects informationnels centrés autour du produit en terme '**d'une sorte de Système de Systèmes d'Information (SdSI)**'.

Définition d'un système : (INCOSE 2006)

"A system is a construct or collection of different elements that together produce results not obtainable by the elements alone. The elements, or parts, can include people, hardware, software, facilities, policies, and documents; that is, all things required to produce systems-level results. The results include system level qualities, properties, characteristics, functions, behavior and performance. The value added by the system as a whole, beyond that contributed independently by the parts, is primarily created by the relationship among the parts; that is, how they are interconnected (Rechtin, 2000)."

Définition d'un système (Morin 1977; Penalva 1997), travaux en lien avec le courant systémique (De Rosnay 1975 : Le Moigne 1977) :

« Un système est une unité globale d'interrelations entre éléments, activités et individus, organisée pour une finalité dans un environnement. »

Définition d'un Système de Systèmes : (INCOSE 2006)

"System of systems applies to a system-of-interest whose system elements are themselves systems; typically these entail large scale inter-disciplinary problems with multiple, heterogeneous, distributed systems."

Définition d'un Système de Systèmes : (Luzeaux and Ruault 2008a)

« Un système de Systèmes est un assemblage de systèmes pouvant potentiellement être acquis et/ou utilisés indépendamment, pour lequel le concepteur, l'acquéreur et/ou l'utilisateur cherche à maximiser la performance de la chaîne de valeur globale, à un instant donné et pour un ensemble d'assemblages envisageables. »

Nous nous sommes alors inspirés de la proposition de Kuras (Figure 4) parce qu'elle nous fournissait ainsi un cadre de définition plus précis pour formaliser notre Système-Entreprise (en fait son Ingénierie) dans ce contexte de SdSI. Ce cadre utilise notamment le concept d'**Holon** (Koestler 1967) pour extraire le **patron de conceptualisation (design pattern)** le plus adéquat pour modéliser notre système-cible, comme Baïna (Baïna 2006) l'a intuitivement fait dans ses travaux en Interopérabilité Dirigée par les Modèles (IDM) des Systèmes d'Entreprise.

(Kuras 2006)

"The currently accepted definition of a system is no longer adequate. It is no longer sufficiently general, and too much is left implicit in its articulation. While the current definition is adequate for "traditional" systems, it is lacking when it is used as the foundation for the engineering of systems associated with the more recent and ambitious efforts."

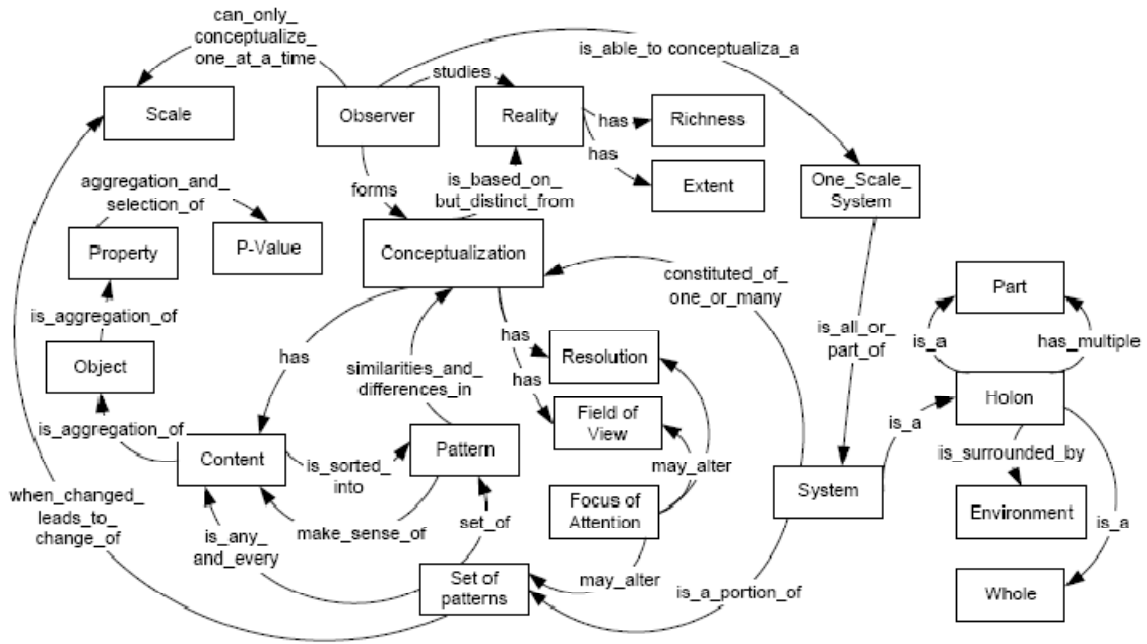


Figure 4. Relationship between concepts in a Kuras' multi-scale definition of a system by (Bjelkemyr and Lindberg 2007a)

En nous appuyant sur la théorie des patterns (Lea 1994; Alexander 1999), nous avons identifié puis dérivé le 'pattern composite' (Gamma et al. 1995) pour mieux décrire la nature 'Tout et Partie' d'un Holon Système-Entreprise (Figure 5) tel que modélisé en Figure 1 et de façon récursive en Figure 2. Parmi tous les points de vue d'un système, cette dérivation nous permet de représenter ceux du Système A Faire et ceux du Système Pour Faire avec son contenu d'Ingénierie associée guidée par le cadre de modélisation de Zachman.

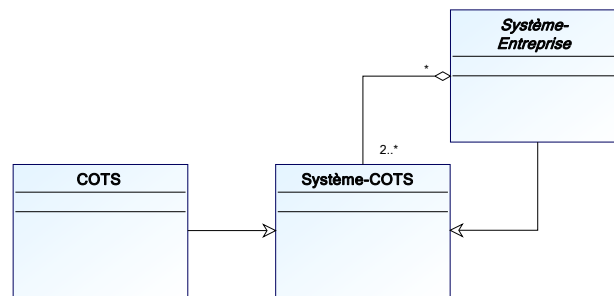


Figure 5. Modèle sémantique : Pattern 'Composite' appliqué au Système-Entreprise

Pour préserver le savoir-faire relatif à son Système, le procédé d'Ingénierie Système Basée sur des Modèles (ISBM) (Wymore 1993) que nous proposons (Figure 6) considère que l'Entreprise doit faire office de maîtrise d'ouvrage et de maîtrise d'œuvre de rang 1 'à minima' aux niveaux contextuels et conceptuels des cadres de modélisation de Zachman en assurant une relation syntaxique par héritage d'invariants de modélisation. Cela ne préjuge pas qu'un métier particulier, par exemple pour un

COTS, n'utilise pas une **Ingénierie Dirigée par des Modèles (IDM)** (Favre et al. 2006) pour assurer la transformation des modèles (par exemple, en CFAO du Système-Produit) menant au déploiement de l'application à partir de l'invariant hérité (par exemple, un modèle sémantique de données en cellule Conceptual-What du cadre de modélisation de Zachman).

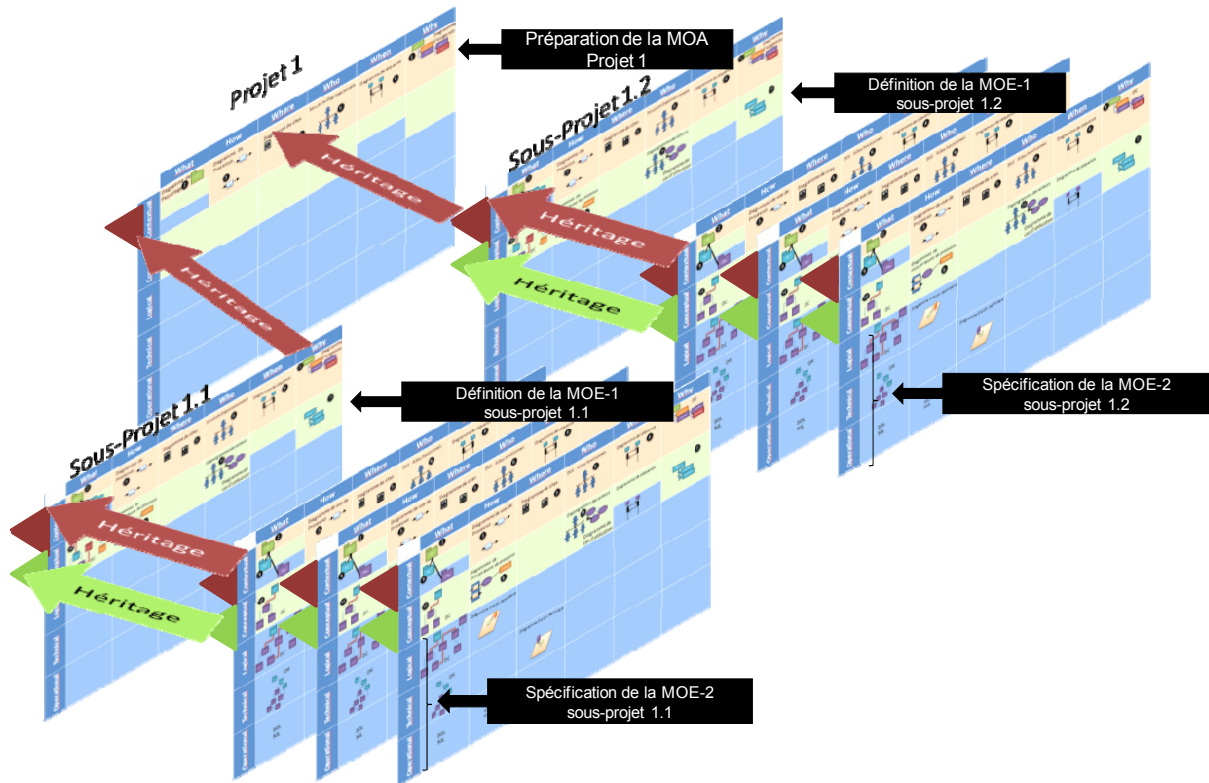


Figure 6. Ingénierie Basée sur les Modèles guidée par le cadre de modélisation de Zachman

Nous avons outillé ce cadre d'ISBM en étendant le meta-modèle de l'outil 'MEGA Modelling Suite' ainsi qu'en développant les interfaces utilisateurs nécessaires et nous l'avons appliqué à l'Ingénierie d'un Système de Traçabilité d'un Produit dans le contexte d'un scénario PLM plausible entre l'AIPL⁶ et DIMEG⁷ (Figure 7).

⁶ AIPL : Atelier Inter-établissements de Productique Lorrain

⁷ DIMEG : Dipartimento di Ingegneria Meccanica e Gestionale – Politecnico di Bari, Italie

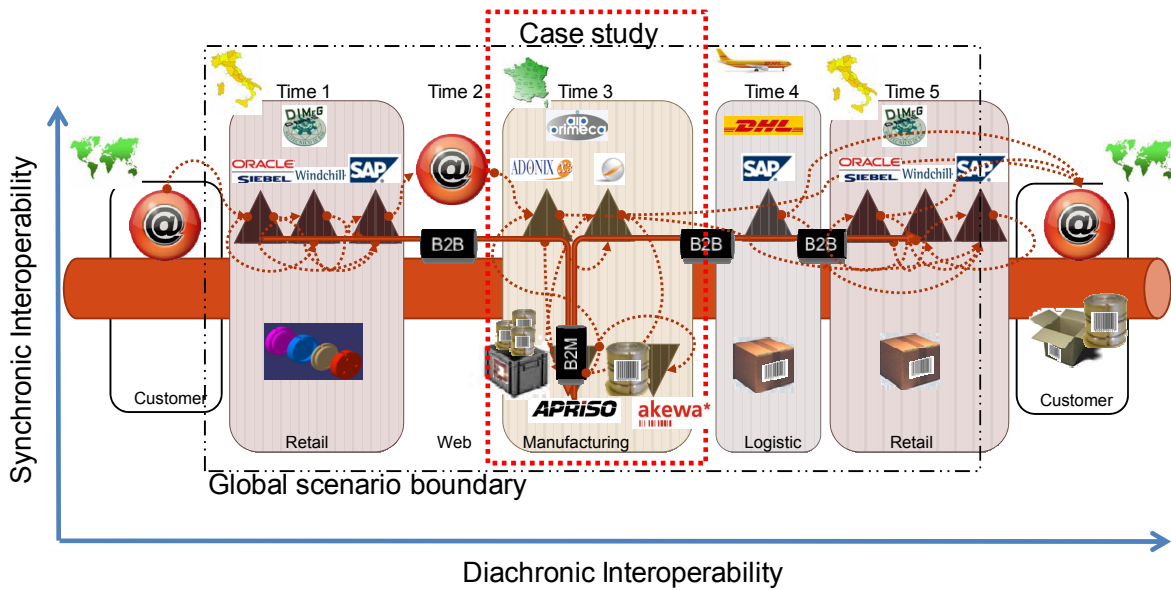


Figure 7. Scénario général du cas d'étude

L'organisation de ce mémoire reflète la démarche de l'IS (Figure 8) appliquée à nos travaux :

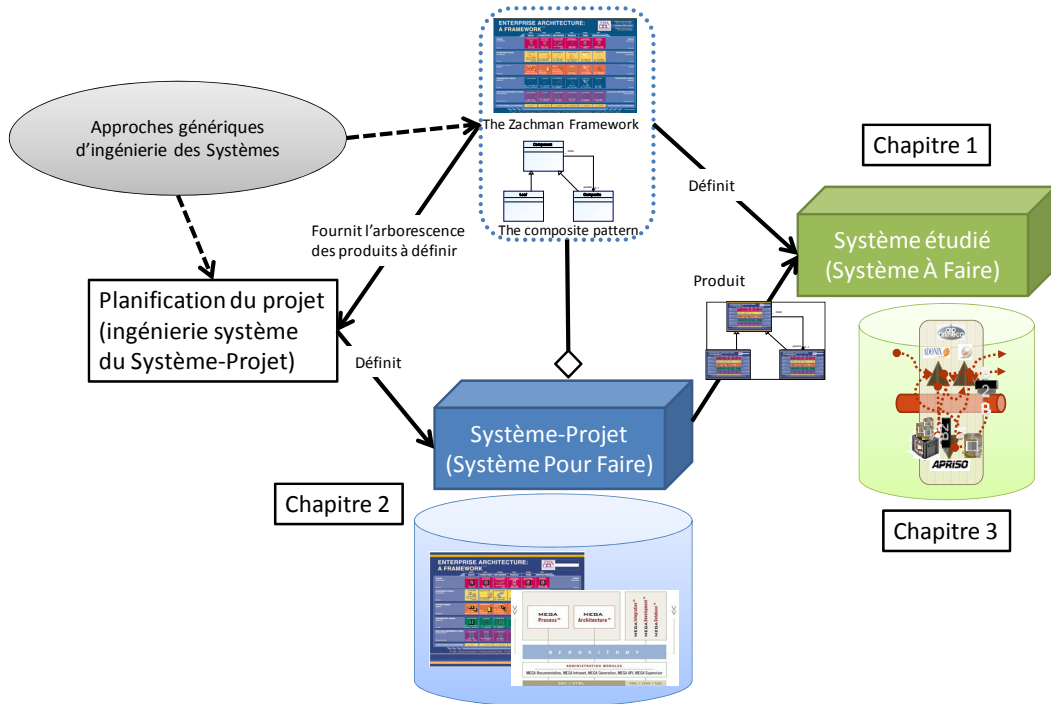


Figure 8. Approche didactique de l'IS : Ingénierie Système Basée sur les Modèles

En partant d'une exigence d'interopération de SI distribués pour répondre à un besoin de traçabilité du produit sur un cas d'étude opérationnel (Chapitre 3), nous proposons d'appliquer une Ingénierie Système Basée sur les Modèles, centrée sur le produit et guidée par le cadre de modélisation de Zachman (Chapitre 2). Ce chapitre développera aussi une proposition d'outillage de modélisation supportant notre méthodologie. Notre contribution méthodologique à la modélisation d'entreprise est

scientifiquement fondée sur la théorie des patterns, en dérivant un pattern de la littérature pour conceptualiser, au sens de Kuras, un Système-Entreprise.

Chapitre 1 : *D'un système d'information à faire ...* Ce chapitre a pour objectif, sur la base de la théorie des patterns, de proposer le pattern d'un « Système A faire » par une ISBM tenant compte des particularités d'un SdSI et de ses interopérations. Nous basons ainsi notre raisonnement sur une dérivation du « pattern composite » afin de modéliser les caractéristiques structurelles et fonctionnelles d'un système composite ainsi que l'agrégation de systèmes à l'image d'un Système de Systèmes (SdS). Notre proposition s'appuie sur le paradigme SCP qui centre ces interopérations sur le produit afin d'assurer la cohérence de l'ensemble des SI interopérables au sein d'un cadre de modélisation adapté.

Chapitre 2 : *... à une ingénierie de SI pour faire.* Ce chapitre a pour objectif de proposer le processus d'ingénierie pour modéliser un SdSI interopérables, en la rendant récursive du fait de la conception en tout point gigogne d'un SdS. Nous postulons que les aspects multi-échelles des métiers et des points de vue d'un SdSI interopérables, peuvent être rendus cohérents à l'aide d'un référentiel commun de modélisation tel que Zachman. Ce cadre de modélisation nous permet de structurer et d'organiser les différents modèles d'ingénierie concernés au sein d'un modèle cohérent de « Système Pour faire ». Le langage que nous utilisons pour spécifier notre ISBM est UML. Nous utilisons également des méthodes de modélisation telles que la modélisation de processus afin d'exprimer le point de vue métier propre à chaque entreprise.

Chapitre 3 : *cas d'étude d'une ISBM appliquée pour faire un système de production de biens.* Ce chapitre a pour objectif d'illustrer les chapitres précédents à travers l'étude d'un cas d'ingénierie de l'interopérabilité entre des SI (ERP, CRM, MES,...) hétérogènes, autonomes, et distribués géographiquement. Nous avons restreint notre cas d'étude à la problématique de mise en œuvre de la traçabilité du produit dans un environnement de réseau d'entreprises. Ce scénario diachronique, de dimension suffisante assurant la crédibilité de notre proposition, sera supporté par un ensemble de modèles et un prototype d'outillage de modélisation, nécessaires à l'ingénierie de l'interopérabilité synchronique de SdSI centré sur le produit. (Encadré rouge « case study » - Figure 7). L'acceptation d'un tel cadre de modélisation reste à confirmer dans un contexte réel d'ingénierie.

En conclusion, en nous basant sur le principe de récursivité des systèmes et de leur ingénierie, nous proposons dans cette thèse de guider la MOA et la MOE de rang 1 dans son acte de modélisation de Système de Systèmes d'Information. Notre proposition d'ISBM participe ainsi à la capitalisation des modèles à travers le temps, au sein d'un cadre de modélisation fournissant les règles de modélisation et les invariants de modélisation, créant ainsi la cohérence et la complétude des modèles à des facteurs d'échelles et des niveaux d'observation différents. Parallèlement, notre nouvelle fonction au sein de

l'Université Henri Poincaré nous ouvre des perspectives d'application de notre proposition d'ISBM sur le Système d'Information de l'Université que l'on pourrait centrer sur l'entité pour laquelle l'Université est missionnée à former : l'étudiant.

Le temps imparti à nos travaux de recherche dans le cadre d'un doctorat ne nous a pas permis d'étudier plus formellement certains aspects, ni de concrétiser de façon tangible la relation d'interopération centrée autour d'un produit-actif. Nous l'avons cependant explorée et nous présentons en perspectives quelques pistes de réflexions en ce sens. Une extension possible de nos travaux serait de proposer une formalisation des relations sémantiques entre les modèles des différents niveaux d'observation afin de vérifier et de valider la cohérence des modèles de l'ISBM proposée.

Références bibliographique de l'introduction générale

- AFIS (2006). "Découvrir et Comprendre l'Ingénierie Système." (Association Française d'Ingénierie Système, groupe de travail en IS)p 126.
- Alexander, C. (1999). "The Origins of Pattern Theory: The Future of the Theory, and the Generation of a Living World." IEEE software (v16, n°5), ISSN 0740-7459, pp 71-82
- Baïna, S. (2006). "Interopérabilité dirigée par les modèles : une approche orientée produit pour l'interopérabilité des systèmes d'entreprise." Thèse de Doctorat de l'Université Henri Poincaré. 7 décembre 2006.
- Baptiste, P., Bernard, A., Bourrières, J.-P., Lopez, P., Morel, G., Pierreval, H. and Portmann, M.-C. (2007). "Comité d'Experts Productique : perspectives de recherche." 15 janvier 2007.
- Berre, A. and al. (2007). "ATHENA deliverable A4.2, Specification of Interoperability Framework and Profiles, Guidelines and Best Practices." (Advanced Technologies for Interoperability of Heterogeneous Enterprise Networks and their Application), FP6-IST IP n°507849, March.
- Bjelkemyr, M. and Lindberg, B. (2007). "The Effects of Limits to Human Abilities on System of Systems Properties." in The 7th International Conference on Complex Systems October 28-November 2, A. Minai, D. Braha and Y. Bar-Yam, eds. (Boston, MA).
- Carney, D., Fisher, D. and Patrick, P. (2005). "Topics in Interoperability: System-of-Systems Evolution." in Carnegie-Mellon Univ Pittsburgh Pa Software Engineering, Inst (Defense Technical Information Center), Technical Note, CMU/SEI-2005-TN-002,
- Chapurlat, V. (2007). "Vérification et validation de modèles de systèmes complexe : application à la modélisation d'entreprise." Habilitation à Diriger les Recherches de l'Université de Montpellier 2. 1er mars 2007.
- De Rosnay, J. (1975). "Le microscope". Collection Points Ed. Le Seuil, 305 pages, 1975.
- Favre, J.-M., Estublier, J. and Blay-Fornarino, M. (2006). L'ingénierie dirigée par les modèles : au delà du MDA, Traité IC2, série informatique et Systèmes d'Information. Lavoisier, 226 Pages.
- Ferrarini, L., Kalogeras, A., Lüder, A. and al. (2006). "Deliverable 6.1, PABADIS' PROMISE, Next Generation Control Devices." (PABADIS based Product Oriented Manufacturing Systems for Reconfigurable Enterprises), FP6-IST-016649, September 22, p 77.
- Gamma, E., Helm, R., Johnson, R. and Vlissides, J. (1995). Design Patterns Elements of Reusable Object-Oriented Software. ISBN 0-201-63361-2, 396 Pages.
- INCOSE (2006). "Systems Engineering Handbook, a guide for system life cycle processes and activities." C. Haskins, ed. (INCOSE-TP-2003-002-03).
- Koestler, A. (1967). The Ghost in the Machine. London, Arkana, ISBN 978-0140191929, 400 Pages.

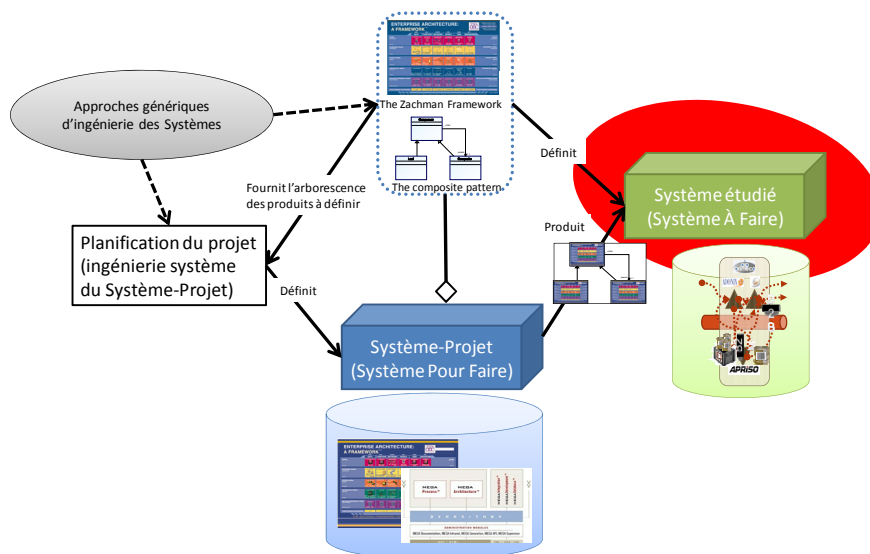
- Kuras, M.L. (2006). "A Multi-Scale Definition of a System." M. T. Report, ed. (Bedford, MA: The MITRE Corp.), Report MTR 06B0000060, August, www.mitre.org.
- Lea, Doug (1994). "Christopher Alexander: an introduction for object-oriented designers." ACM SIGSOFT Software Engineering Notes (v19, n°1), pp 39-46.
- Le Moigne, J.L. (1994). La théorie du système général : théorie de la modélisation. mexapc.org, ISBN 2-13-043323-5, 352 Pages.
- Luzeaux, D. and Ruault, J.R. (2008). Ingénierie des systèmes de systèmes - concepts et illustrations pratiques. Hermes Science, ISBN 978-2-7462-1875-8, 225 Pages.
- Man-Sze. Li, Ricardo. Cabral, Guy. Doumeingts and Keith. Popplewell (Editors) (2006). "Enterprise Interoperability: Research Roadmap (Version 4.0)." (European Commission, Objective 1.3 ICT for Enterprise Networking).
- Morel, G., Valckenaers, P., Faure, J.-M., Pereira, C.E. and Diedrich, C. (2007). "Manufacturing plant control challenges and issues."
- Morin, E. (1977). La methode, Tome 1: la nature de la nature. Le Seuil (coll. Points). Paris, France, ISBN 2-02-004634-2, 399 Pages
- NoE-INTEROP (2003-2006). "IST 508011-Interoperability research for Networked Enterprises Applications and Software."
- PABADIS'PROMISE (2006). "Deliverable 6.1, PABADIS'PROMISE, Next Generation Control Devices." (Project funded by the European Community under the "Sixth Framework Programme" (2002-2006) within the "Information Society Technologies" (IST) priority), FP6-IST-016649, September 22, p 77,
- Panetto, H. (2006). "Meta-modèles et modèles pour l'intégration et l'interopérabilité des applications d'entreprises de production." Habilitation à Diriger des Recherches de l'Université Henri Poincaré. 4 décembre 2006.
- Terzi, S. (2005). "Elements of Product Lifecycle Management: Definitions, Open Issues and Reference Models." Thèse de Doctorat de l'Université Henri Poincaré & du Politecnico di Milano. 27 Mai 2005.
- Sowa, J.F. and Zachman, J.A. (1992). "Extending and formalizing the framework for information systems architecture." IBM Systems Journal (v31, n°3), ISSN 0018-8670, pp 590-616 (519 ref.).
- Vallespir, B. (2003). "Modélisation d'entreprise et architectures de conduite des systèmes de production." Habilitation à Diriger des Recherches de l'Université Bordeaux 1. 19 Décembre 2003.
- Vernadat, F. (1996). Enterprise Modeling and Integration: Principles and Applications. Springer, ISBN 0412605503, 9780412605505, 513 Pages.
- Wymore, A.W. (1993). Model-Based Systems Engineering. CRC Press, Inc. Boca Raton, FL, USA, ISBN 0-8493-8012-x, 700 Pages.

CHAPITRE 1.

D'UN SYSTÈME D'INFORMATION À UN SYSTÈME DE SYSTÈMES D'INFORMATION

...

Formalisation d'un système de systèmes d'information centré sur le produit



Introduction du chapitre 1

Les entreprises sont vues comme des **systèmes sociotechniques complexes** (Verdanat 2006) composés de ressources interdépendantes avec leurs **finalités** propres, organisées dans un processus (**fonctionnel**) lui-même lié à un comportement global (**organisationnel**) avec une finalité ou un objectif global (**stratégique**). Beaucoup de caractéristiques rapprochent ces systèmes sociotechniques complexes aux systèmes de systèmes. En effet, lorsque l'on aborde la gouvernance de tels systèmes avec en particulier les indicateurs et le tableau de bord proposés par (Boardman and Sauser 2006), les interactions multiples entre les ressources, les activités, les processus, et les sous-systèmes déterminent la capacité de réactivité et de flexibilité de l'entreprise (Chapurlat 2006).

Définies comme telles, ces entreprises, peuvent être assimilées à un réseau réticulaire de système d'entreprises ouvert (Oberndorf 1998), et par extension le Système d'Information (SI) de l'entreprise peut être lui-même considéré comme un réseau réticulaire de SI.

La spécification d'un tel réseau de SI implique d'évoluer du seul paradigme d'intégration vers un paradigme d'interopération (Fisher 2006) où l'agrégation est le mécanisme « de construction » des relations entre des SI distribués, et participant au pilotage du Système Entreprise. Ce mécanisme se différencie du mécanisme de composition⁸ qui est assimilé à une vision monolithique d'un SI décomposé en SI structurel tel que préconiser par MERISE⁹ (Tardieu et al. 1983).

Ce changement implique donc de prendre en compte ces aspects d'autonomie et d'hétérogénéité des SI constitutifs tant dans leur ingénierie propre pour les faire que dans l'ingénierie intervenant au niveau global du SI. Dans ce chapitre, nous montrerons que cette problématique peut trouver des réponses en se référant au concept de Système de Systèmes (SdS), ainsi que par des mécanismes particuliers tels que l'émergence et la récursivité, pour appréhender les aspects multi-échelles de ces nouvelles organisations distribuées.

Par analogie, nous proposons de percevoir l'ensemble de SI constitutifs d'un SE comme un SdS d'Information, dans lequel les interopérations entre les éléments doivent être synchronique (**B2M**¹⁰) et diachronique (**B2B**¹¹) afin de former un tout, système d'entreprise, cohérent.

⁸ Afin d'éviter toute confusion, la notion de composition exprime ici la notion de « décomposition » structurelle d'un système au sens classique de l'Ingénierie Système : un assemblage de systèmes. Nous verrons par la suite, que cette notion aura un tout autre sens lors de la formalisation d'un Système de Systèmes en UML avec les notions de composition vs agrégation de systèmes.

⁹ MERISE : méthode d'analyse, de conception et de réalisation de système d'information informatisé.

¹⁰ **B2M** : Business to Manufacturing.

¹¹ **B2B** : Business to Business.

Pour ce faire, dans un premier temps, et afin de s'appropriier le concept de Système de Systèmes, nous proposons de formaliser partiellement ce concept en nous appuyant sur un mécanisme mathématique qu'est le diagramme commutatif, permettant de décrire la démarche de construction d'un SdSI au fur et à mesure de sa caractérisation par les propriétés d'un SdS que sont l'autonomie, l'appartenance, la diversité, la connectivité, et l'évolution. L'émergence, caractéristique nécessaire de définition d'un SdS, étant difficilement formalisable par cette technique mathématique, nous proposons de la prendre en compte progressivement à travers notre processus de formalisation s'appuyant sur des techniques complémentaires.

La conception d'un tel système étant en tout point gigogne (un SdS est un système d'éléments agrégés, chacun d'eux étant lui-même un système), il convient de prendre en compte d'une part, un premier niveau de complexité lié à l'observation multi-échelle et multipoint de vue d'un SdSI et d'autre part, un deuxième niveau de complexité induit par la récursivité de leur conception.

Pour résoudre ce premier niveau de complexité, nous proposons d'utiliser les travaux de (Kuras 2006) permettant de croiser, à travers un cadre de modélisation, les différents niveaux d'échelle de conception d'un système avec les différents points de vue d'observation d'un système que sont sa conceptualisation, sa résolution et son contenu ou ses informations. Ceci nous permet également de renforcer l'aspect évolutif caractérisant un SdS. Le résultat de notre formalisation est une « équation » faisant apparaître simultanément la structure d'un SdS et son cadre d'ingénierie à travers duquel il est étudié.

Ces aspects formalisés, nous nous sommes attachés à la description de l'aspect récursif d'un SdSI en nous appuyant sur la théorie des patterns (Lea 1994; Alexander 1999) et en dérivant le « pattern composite » communément appliqué dans le domaine du génie logiciel (Gamma et al. 1995). Ce pattern nous permet de compléter l'aspect structurel d'un SdS, tel que proposé précédemment, et de décrire le mécanisme particulier d'assemblage par agrégation d'un SdS.

Le résultat de notre formalisation est un pattern de SdSI prenant en compte les éléments constitutifs et leurs interopérations ainsi que la caractéristique d'émergence du SdSI décrite à l'aide du mécanisme particulier de « classe d'association » d'UML.

Ce pattern que nous considérons comme un méta-modèle de SdSI est au cœur du cadre de modélisation d'un SI que nous proposons dans cette thèse. Le cadre d'ingénierie qu'il convient de lui associer systématiquement est basé sur le cadre de modélisation proposé par Zachman (Sowa and Zachman 1992). Nous décrivons cette particularité dans le chapitre 2 de ce mémoire.

Nous notons que ce travail de formalisation participe à la résolution d'une de nos problématiques initiales, à savoir le manque de capitalisation et le besoin d'internalisation des savoirs et savoir-faire de l'entreprise.

1 Le système d'information d'entreprise vu comme un Système de Systèmes d'Information

Dans ce paragraphe, nous proposons de définir l'ensemble des systèmes d'information composant le système d'information d'une entreprise, vu comme un système de systèmes d'information afin de passer d'une vue intégrative autour d'un référentiel commun pour partager l'information entre les différents domaines fonctionnels de l'entreprise ... à une vue réticulaire du système d'information pour distribuer l'information entre les différents SI participant à la gestion de l'information de l'entreprise et leur ingénierie associée.

La difficulté majeure rencontrée dans l'ingénierie de SI distribués, hétérogènes et autonomes, concerne la complexité de leur assemblage (Figure 9) pour former un SI d'entreprise cohérent au regard d'une performance globale à atteindre (Mayer and Auzelle 2007; Auzelle et al. 2008).

En effet, ces SI étant pour la plupart considérés comme des composants sur étagères (COTS), leur configuration doit prendre en compte les propriétés et les fonctionnalités d'un assemblage plus global auquel ils vont participer tout en conservant les conditions opérationnelles propres à chacun.

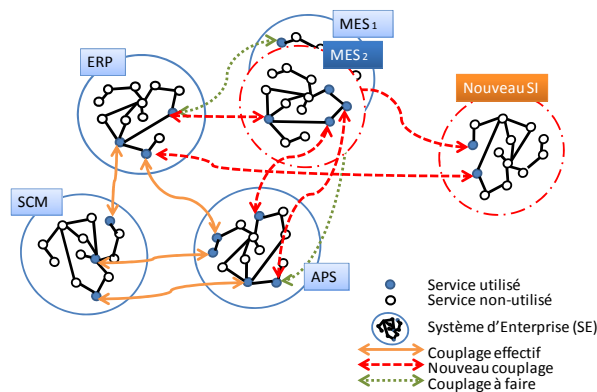


Figure 9. Complexité de l'interopérabilité des Systèmes d'entreprise – des COTS (Auzelle et al. 2008).

L'exigence de l'optimisation de l'assemblage de ces SI distribués, hétérogènes et autonomes dans des contextes potentiellement changeant de coopérations intra et inter-entreprises, nous amène à étudier ce que doit être le mécanisme de construction d'un SdSI par **agrégation**¹² de SI constitutifs ainsi que ce que doivent être les interactions entre ces systèmes.

En effet, ces assemblages constituent des enjeux majeurs pour lesquels les grands groupes industriels se regroupent en réseaux d'entreprises (entreprise étendue, entreprise virtuelle), de manière souvent

¹² **Agrégation** : forme contrainte d'une association dans laquelle l'une des classes décrit un tout alors que la classe associée décrit des parties. On appelle le tout un composé, une partie un composant. Une agrégation est une association qui représente une relation d'inclusion structurelle ou comportementale d'un élément dans un ensemble.

éphémère, pour échanger des informations afin d'augmenter leur potentiel de marché et ou accéder à de nouveaux marchés (SESSI 2007).

Focalisé sur l'agrégation de systèmes en un système plus global, le concept de Système de Systèmes (SdS) nous apparaît comme pertinent afin de décrire la complexité de l'assemblage de SI distribués, hétérogènes et autonomes devant interopérer entre eux pour une mission satisfaisant le besoin d'un donneur d'ordres. Nous proposons dans ce chapitre de formaliser un SI d'entreprise comme un Système de Systèmes d'Information (SdSI) en prenant en compte les définitions originelles d'un SdS mais aussi la relation particulière d'interopération permettant aux différents SI constitutifs de participer à un SdSI d'entreprise bénéficiaire.

Aujourd'hui une entreprise ne se pilote plus seulement par les flux informationnels concentrés dans un **Système d'Information** (SI) monolithique, mais plutôt par l'objet à forte **Valeur Ajoutée** dont l'entreprise est créatrice : le **produit** ou le **service**. Dans le cadre de cette thèse nous nous focaliserons sur le produit en particulier. L'« explosion » des SI monolithiques du concept CIM¹³ au profit de SI composés de COTS de types **progiciels** comme des **CRM**¹⁴, des **ERP**¹⁵, des **MES**¹⁶, ..., rend plus complexe encore le traitement des flux d'information toujours plus croissants. Le premier constat (Baptiste et al. 2007) de l'introduction de cette thèse, montre aussi une évolution rapide du concept CIM vers l'**interopérabilité** des organisations, des processus et des outils informatiques de gestion. Cette relation d'interopérabilité, souvent éphémère, amène au développement de SI **distribués, hétérogènes et autonomes** avec comme point commun la partie du SI relative au produit.

Le **Centre de Recherche en Automatique de Nancy** (CRAN) étudie ces questions d'interopérabilité depuis plusieurs années en particulier avec son équipe de recherche « **Système Contrôlé par le Produit** » (SCP). Ce paradigme défend l'idée d'une **interopérabilité centrée sur le Produit** (Morel et al. 2003 ; Panetto 2006). En particulier, dans cette thèse, nous avons étudié ces questions et la formalisation de l'interopérabilité où un SE traite de l'information dans un contexte de fabrication. Ainsi, Le paradigme SCP postule que le « Produit », de part ses propriétés qui le rendent « Actif », collabore avec les applications constituant le SE aux moments opportuns et à la volée pour que ces dernières puissent permettre son évolution durant son cycle de vie (Figure 10) et en particulier dans sa phase de production. Dans ce contexte, un produit embarquant des informations sur lui-même et pouvant communiquer avec son environnement peut être qualifié de « **Produit-Actif** ». D'un point de vue SCP, la collaboration entre l'ensemble de ces applications de SE autonomes, hétérogènes et distribués devient effective au travers de la **connectivité** dont « Produit-Actif » est l'initiateur. Cette mise en relation de courte durée des applications est alors centrée sur le « Produit-Actif », et va

¹³ **CIM** : Computer Integrated Manufacturing

¹⁴ **CRM** : Customer Relationship Management

¹⁵ **ERP** : Enterprise Resource Planning

¹⁶ **MES** : Manufacturing Enterprise System

déclencher en chacun d'eux un comportement individuel au service de ce même « Produit-Actif » (Ferrarini et al. 2006; Morel et al. 2007a).

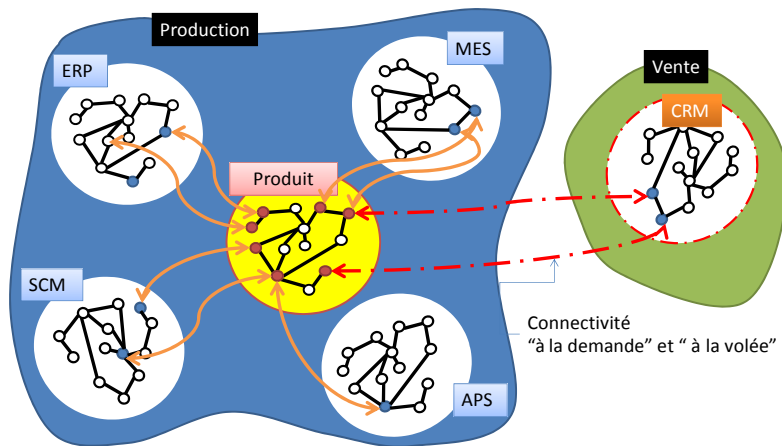


Figure 10. Le produit vecteur de l'interopérabilité technique et sémantique entre le SI

1.1 Caractérisation d'un SdSI comme un SDS

Un système de systèmes est un assemblage de systèmes pouvant potentiellement être acquis et/ou utilisé indépendamment, pour lequel le concepteur, l'acquéreur et/ou l'utilisateur cherche à maximiser la performance de la chaîne de valeur globale, à un instant donné et pour un ensemble d'assemblages envisageables (Luzeaux and Ruault 2008a).

Le concept de SdS (Berry 1964; Ackoff 1971) se distingue de celui d'un système monolithique par 5 caractéristiques de base définies par (Maier 1998) :

- l'**indépendance opérationnelle** des éléments : si l'on dissocie d'un système de systèmes les systèmes qui le composent, ces systèmes peuvent fonctionner indépendamment.
- l'**indépendance managériale** des éléments : les systèmes constitutifs sont acquis et intégrés séparément, et leur exploitation se poursuit indépendamment du système de systèmes.
- le **développement évolutionniste** : un SdS n'est jamais vraiment achevé, il évolue au gré des ajouts, des disparitions, des modifications de ses constituants.
- le **comportement émergent** : les systèmes constituants, évoluant dans le temps, ont la capacité de s'auto-organiser afin d'assurer la continuité de la mission globale du SdS.
- la **distribution géographique** : les systèmes constitutifs sont physiquement, géographiquement dispersés.

Ces critères sont pour l'essentiel relatifs à la description de ce qu'est un système en général, mais font apparaître la possibilité de comportements émergents souvent occultés dans la définition de systèmes du fait de la complexité induite. Afin d'introduire une notion d'organisation des ressources techniques et humaines de ces systèmes au sein d'un SdS, DeLaurentis (2005) complète ces 5 premiers critères par 3 autres critères permettant de décrire les aspects collaboratifs entre les systèmes constitutifs :

- l'**interdisciplinarité** : les systèmes constitutifs sont relatifs à des domaines différents ;
- l'**hétérogénéité** : les systèmes constitutifs sont de natures différentes avec une dynamique différente et sur des échelles de temps différentes ;
- les **réseaux de systèmes** : les systèmes sont en réseau, régis par des règles d'interaction qui définissent la connectivité entre chacun d'entre eux.

De leur côté, (Boardman & Sauser, 2006) ont proposé de requalifier certains critères initiaux de Mark Maier, mais aussi de compléter ces critères afin de mieux répondre au besoin de composition de systèmes en réseaux réticulaires de systèmes d'entreprises :

- l'**autonomie** : L'autonomie des systèmes constituants permet d'atteindre la finalité du SdS ;

- l'**émergence** : Le comportement d'un SdS n'est pas systématiquement prévu, a priori, à sa conception mais résulte, de manière non prescriptive, d'un ensemble de conditions de fonctionnement, dans un environnement, lui assurant une certaine flexibilité pour répondre à une mission ;
- la **connectivité** : Pour renforcer l'efficacité d'un SdS, la connectivité/l'interopérabilité entre ses constituants se construit dynamiquement par l'intermédiaire d'une architecture centrée sur la communication et les services (**NetCentric**)¹⁷;
- la **diversité** : Une condition nécessaire à l'existence d'un SdS concerne l'hétérogénéité de ses constituants ;
- l'**appartenance** : les systèmes constituants choisissent d'appartenir à un SdS sur une base de coût/avantage, afin d'en tirer profit pour leur(s) propre(s) but(s), et aussi parce qu'ils adhèrent (croyance) à la mission de ce SdS.

En particulier, le critère d'**appartenance**, autour de la valeur ajoutée, nous semble pertinent afin de mettre l'accent sur une collaboration « centrée sur le produit » entre des SI distribués, hétérogènes et autonomes. Nous notons que cette notion d'appartenance a fait l'objet d'une enquête sur les nouveaux modes d'organisation d'entreprises de toutes tailles par le Ministère de l'Économie, des Finances et de l'Emploi (SESSI 2007). D'autre part, nous pensons qu'un SdS ne peut exister en tant que tel si les éléments qui le constituent n'ont pas adhéré (notion d'appartenance) à la nouvelle mission qui leur est proposée, afin d'en tirer individuellement profit tout en partageant des informations pour une finalité commune au travers d'un réseau.

Le concept de SdS, et plus particulièrement les critères fonctionnels qui le définissent, nous permettent de distinguer les aspects fonctionnels des aspects techniques d'un assemblage d'un SI distribué, hétérogène et autonome, afin d'optimiser plus facilement la configuration de cet assemblage et de favoriser leur agilité pour faire face à un environnement économique changeant. D'autre part, ces critères fonctionnels nous permettent de s'absoudre des obsolescences (Luzeaux and Ruault 2008a) des technologies de l'information, en remontant d'un COST pouvant devenir obsolète, à sa fonction, voire à l'exigence concernée, pour mieux le cerner individuellement tout en permettant d'exercer une véritable gestion de sa configuration dans le but de favoriser ses interopérations au sein d'un **SdS d'Information**.

Ces différentes définitions nous conduisent à postuler que comprendre la complexité de l'assemblage de SI distribués, autonomes et hétérogènes au sein d'un SI entreprise peut se faire à travers le concept de système de systèmes qu'il convient cependant de particulariser à notre problématique de système de

¹⁷ **NetCentric** : http://nzuwah.com/NCEA_Overview.html

systèmes d'information en entreprise. Pour ce faire, il convient de vérifier que les différentes propriétés d'un SdS, retenues dans ce travail de thèse, soient des conditions nécessaires et suffisantes pour décrire notre SdSI.

Ainsi, les travaux de (Gorod et al. 2007) proposent un cadre de pilotage de l'ingénierie d'un SdS, utilisant des matrices pour le management opérationnel d'un SdS. Un des objectifs de ces travaux est de fournir une méthode permettant de préserver la caractérisation d'un SdS par l'application des meilleures pratiques extraites du pilotage de réseaux de systèmes en général et d'entreprises en particulier. Cette méthode s'appuie, entre autres, sur les 5 domaines de pilotage de réseaux tels que définis par l'ISO/IEC 7498 (ISO/IEC-7498 1994) et sur les 5 critères d'un SdS tels que définis par (Boardman and Sauser 2006).

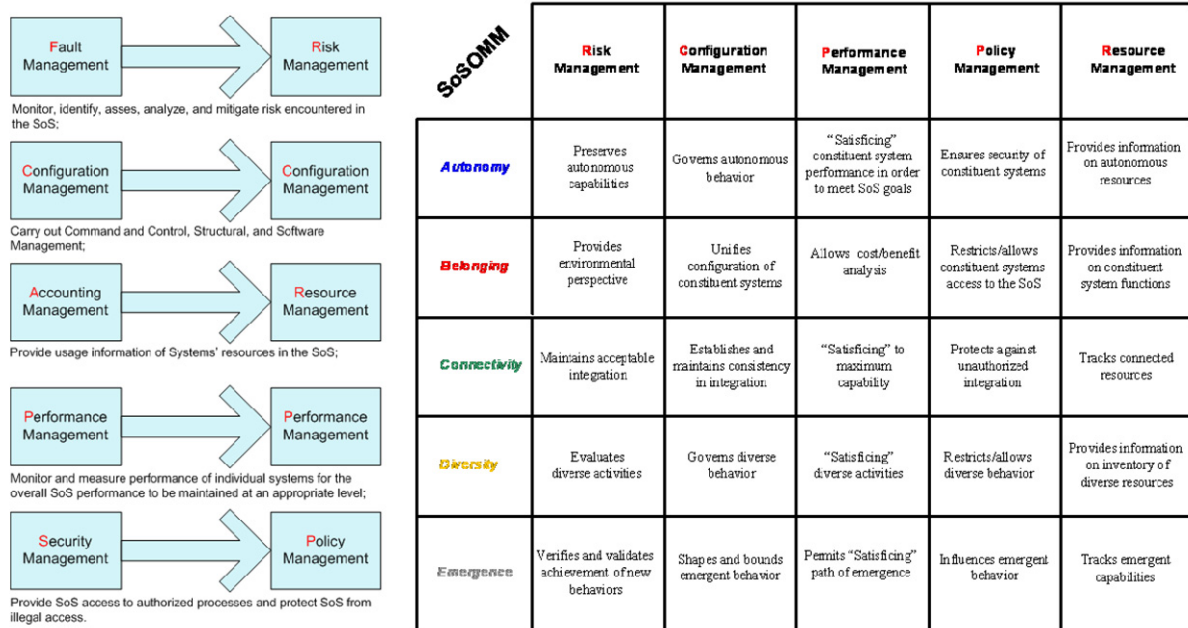


Figure 11. SoS Operational Management Matrix (Boardman and Sauser 2006).

Utilisée comme un « tableau de bord » de pilotage d'un SdS, cette matrice assimile les critères d'un SdS proposés par Boardman et Sauser (2006) comme des indicateurs dont les valeurs varient suivant le niveau d'exigence désiré pour les 5 domaines de pilotage : le pilotage des risques, le pilotage de configuration, le pilotage de la performance, le pilotage des règles et le pilotage des ressources.

Parce que nous faisons une analogie entre la problématique ayant amené à définir ce cadre de pilotage d'un SdS et notre propre problématique de définition d'un cadre de modélisation d'un système d'information d'entreprise centré sur le produit, nous retiendrons dans la suite de nos travaux les 5 critères de Boardman et Sauser (2006) pour caractériser un SI d'entreprise comme un SdSI.

D'autre part, nous retiendrons également le critère de « développement évolutionniste » ou « évolution » tel que défini par Maier (1998) afin d'illustrer la capacité des systèmes constitutifs d'un SdS à se configurer ou à se reconfigurer pour permettre *un ensemble d'assemblages envisageables*

(Luzeaux and Ruault 2008a) dans le but d'évoluer en fonction des exigences, des performances, des missions ou des finalités à atteindre par le SdS.

En conclusion, dans nos travaux, nous postulons qu'un système de systèmes d'information se distingue d'un système d'information monolithique par 6 critères (Figure 12) issus de la combinaison des critères précédents et rendant à notre sens, le mieux compte de l'agrégation de SI constitutifs en un SdSI.

Figure 12. Proposition de critères de SdS sur la base des travaux de (Maier 1998; DeLaurentis 2005; Boardman and Sauser 2006)

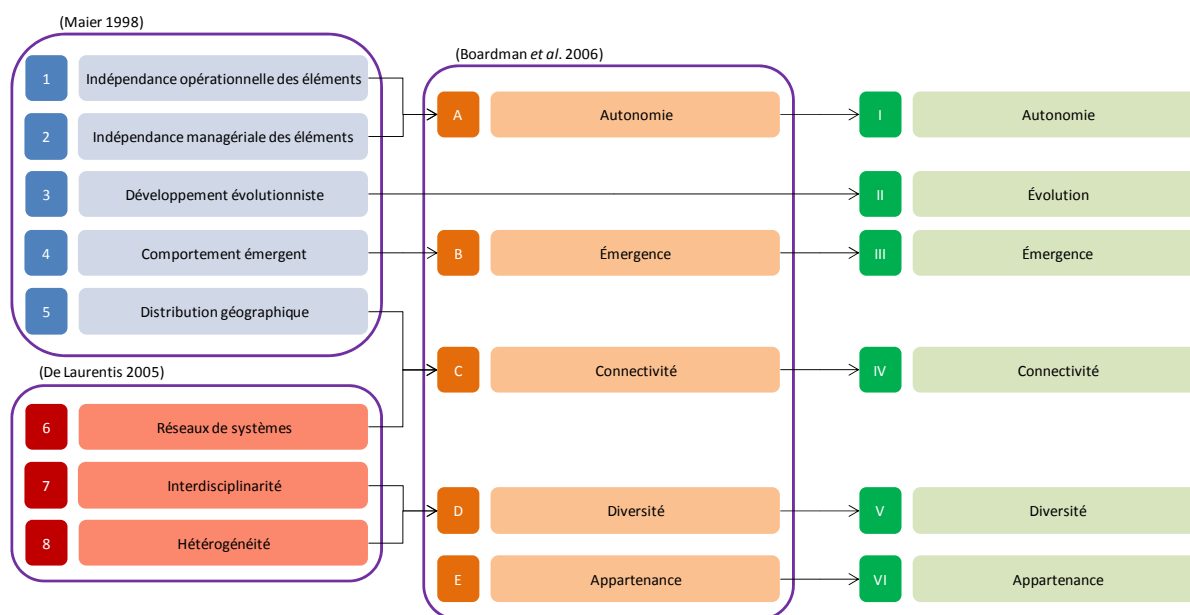


Tableau 1. Récapitulatif des critères retenus de SdS sur la base des travaux de (Maier 1998; DeLaurentis 2005; Boardman and Sauser 2006)

Élément	SdS	Mots clés
Autonomie	L'autonomie des systèmes constituants permet d'atteindre la finalité du SdS	Dirigé ; Planifié ; Embarqué ; Autonomie
Évolution	Un SdS s'adapte au gré des coopérations qui évoluent au cours du temps au travers d'auto-adaptation de ses constituants.	Évolution ; Dynamique ; Auto-adaptation
Émergence	Le comportement d'un SdS n'est pas systématiquement prévu, a priori, à sa conception mais résulte, de manière non prescriptive, d'un ensemble de conditions de fonctionnement, dans un environnement, lui assurant une certaine flexibilité pour répondre à une mission.	Intelligence ; Le tout est plus que la somme des parties ; Comportements ; Émergence ; Dynamique ; Adaptatif
Connectivité	Pour renforcer l'efficacité d'un SdS, la connectivité/l'interopérabilité entre ses constituants se construit dynamiquement par l'intermédiaire d'une architecture centrée sur la communication et les services (<i>NetCentric</i>).	Interdépendance ; Distribué ; Travail en réseau ; Solutions multiples ; Loose Coupling ; Intégration ; Interopérabilité ; Synergisme
Diversité	Une condition nécessaire à l'existence d'un SdS concerne l'hétérogénéité de ses constituants.	Indépendance ; Diversité ; Hétérogénéité
Appartenance	Les systèmes constituants choisissent d'appartenir à un SdS sur une base de coût/avantage, afin d'en tirer profit pour leurs propres buts, et aussi parce qu'ils adhèrent (croyance) à la mission de ce SdS.	Entreprise ; Partagé ; Mission

1.2 Caractérisation de la relation d'interopérabilité dans un SdSI

Connectivity – Dynamically supplied by constituent systems with every possibility of myriad connections between constituent systems, possibly via a NetCentric architecture, to enhance SoS capability (Gorod et al. 2007)

La particularité du concept de SdS réside dans sa conception puisqu'il convient de considérer simultanément ses éléments constitutifs et les relations d'interopération qui s'exercent entre ces éléments. C'est cette particularité qui distinguera les interactions au sein d'un système d'information monolithique des interactions qui s'exercent dans un SdSI. En effet, si les premières sont relatives, comme préconisé en Ingénierie Système, à une structure hiérarchisée contrôlé par un SI censé posséder la visibilité globale des échanges entre les éléments, les secondes sont quant à elles issues de l'agrégation des éléments constitutifs pour réaliser la fonction globale du SdS mise en œuvre par le maître d'œuvre pour satisfaire les exigences d'un maître d'ouvrage et pour répondre au(x) besoin(s) d'un donneur d'ordres.

Selon (Boardman and Sauser 2006; Gorod et al. 2007), les interactions entre les éléments constitutifs d'un SdSI sont caractérisées par le critère de « connectivité » assimilables aux organisations de type « **Network-Centric**¹⁸ ». Ainsi, la qualité des interactions entre les éléments d'un SdSI dépend de la qualité des relations d'interopération permettant de réaliser dynamiquement la connectivité au sein d'un SdSI. L'interopération entre ces éléments, également appelée relation d'interopérabilité (Fisher 2006), consiste en l'échange et l'utilisation coopérative d'information entre constituants autonomes. Cependant, la complexité de mise en œuvre de ces relations d'interopérabilité implique, dans une démarche d'ingénierie d'un SdSI, d'englober les métriques, les modèles, les méthodes et les outils du domaine de l'ingénierie de l'interopérabilité des systèmes.

Bien que notre travail de recherche ne participe pas directement aux travaux relatifs au domaine de l'interopérabilité¹⁹, nous nous intéressons cependant à ce mécanisme particulier d'ingénierie d'un SdSI afin d'une part, de caractériser les types d'échanges d'information (**B2B**²⁰ & **B2M**²¹) pouvant être modélisés dans notre cadre de modélisation d'un SdSI et d'autre part, de contribuer au paradigme de système contrôlé par le produit. En effet, les perspectives de nos travaux (voir le chapitre

¹⁸ **Network-Centric** : axé sur les réseaux et la connectivité. Un SI centré réseau sera constitué de plusieurs machines, connectées entre elles; la charge du système se répartissant sur l'ensemble des processeurs du réseau ce qui est l'inverse d'un système centralisé.

¹⁹ Nous entendons par là que nos travaux ne proposent pas de résultats sur l'interopérabilité purement technique ou syntaxique qui nous semble déjà mature, mais proposent plutôt de considérer l'interopérabilité sémantique entre les objets invariants de modélisation que nous manipulons dans notre ingénierie.

²⁰ **B2B** : Business to Business.

²¹ **B2M** : Business to Manufacturing.

« conclusion et perspectives » en fin de document) seront de définir le produit comme un système actif & autonome, participant à travers des relations d'interopérabilité dont il est l'initiateur, à la mission dont le « système entreprise de systèmes » sera bénéficiaire.

Interoperability (IEEE): Ability of a system or a product to work with other systems or products without special effort on the part of the customer. Interoperability is made possible by the implementation of standards (IEEE 2000).

Il existe de très nombreuses définitions de l'interopérabilité dans la littérature (Tableau 3). De manière générale, on peut cependant constater que l'interopérabilité a de larges implications, dans de nombreux de domaines, à la fois sur des objets techniques, mais aussi sur des organisations sociotechniques. Ainsi, l'interopérabilité couvre les aspects **techniques**, **syntactiques**, **sémantiques**, **pragmatiques** et **organisationnels**, mais aussi d'autres d'aspects cités de manière non exhaustive dans le tableau ci-dessous inspiré des travaux de (Panetto 2006) (Tableau 3).

Tableau 2. Types d'interopérabilité (Panetto 2006).

Types d'interopérabilité→										
	Technique	Syntaxique	Sémantique	Pragmatique	Organisationnelle	Fonctionnelle	Politique	Sociale	Légale	Inter standard
(Tsagkani 2005)	x		x	x						
(EPAN 2004)	x		x		x					
(Whitman et al. 2006)	x	x	x	x						
(Lilleng 2005)	x	x	x							
(IDABC/EIF 2004)	x		x		x					
(CEN/ISSS 2005)		x	x			x				
(NAHIT 2005)	x									
(TMAB 2005)	x				x		x	x		
(ETSI 2005)										x

L'interopérabilité a beaucoup inspiré la communauté académique et militaire dans le but de déterminer une typologie et de proposer des métriques (Tableau 3) afin de mieux décrire ses aspects qualitatifs. Nous noterons que les aspects quantitatifs de l'interopérabilité font l'objet de travaux en cours (Ford et al. 2007).

Tableau 3 Récapitulatif des principaux modèles d'interopérabilité avec leur référence bibliographique

Modèles	Références	Organisations
SoIM	(LaVean, 1980)	Defense Information systems Agency (DISA)
QoIM	(Mensch, Kite, & Darby, 1989)	MITRE corporation
LISI	(C4ISR, 1998)	
MCISI	(Amanowick & Gajewski, 1996)	Military University of Technology, Warsaw, Poland
IAM	(Leite, 1998 (revised Aug 2003))	Joint Theater Air and Missile Defense Organization (JTAMDO) contractor SIM, Inc
OIM	(Clark & Jones, 1999)	Australian Defence Science and Technology Organization (DSTO)
OIAM	(Kingston, Fewell, & Richer, 2004)	
Stoplight	(Hamilton, Rosen, & Summers, 2002)	Joint Forces Command (JFCOM) Joint Forces Program Office (JFPO)
LCI	(Tolk & Muguira, 2003)	Old Dominion University Virginia Modeling Analysis And Simulation Center. (Tolk, Beyond Technical Interoperability : Introducing a Reference Model for Measures of Merit for Coalition Interoperability, 2003)
LCIM		
NMI / NIE	(NATO, 2003) (NATO, 2003) (revised 2005)(Moxley, Simon, & Wells, 2001)	North Atlantic Treaty Organization (NATO) NATO Interoperability Environment
SoSI	(Morris, Place, & Smith, 2006)	Carnegie Mellon Software Engineering Institute (CMU-SEI)
NTI	(Stewart, Clarke, Goillau, & Varrall, 2004)	Defence Science and Technology Laboratory (DSTL) and Contractor, QinetiQ, plc
EIF	(Chen et al. 2006) (CEN/ISO-11354-1 2008)	Interop Enterprise Interoperability Framework
ATHENA	(ATHENA-IP 2007)	Interoperability Framework and Services for Networked Enterprises
i-Score	(Ford, Colombi, Graham, & Jacques, 2007)	Air Force Institut of Technologie (AFIT)

Si ces aspects permettent de typer les relations d'interopération en fonction de la nature des échanges réalisés, il convient désormais de cerner comment s'établissent ces relations entre les SI constitutifs d'un SdSI. En effet, l'organisation résultante de la combinaison de SI est conditionnée par la façon dont les relations d'interopération connectent les SI entres-eux.

De manière générale, combiner des SI entre eux consiste à appliquer un mécanisme de médiation afin d'améliorer les interactions entre ces SI, mais aussi entre les différentes parties prenantes de ceux-ci, entre les services, voire les entreprises auxquels ils appartiennent. Ainsi, différents scénarii d'intégration sont possibles (Vernadat 2007; Chen et al. 2008) :

- Une **intégration « complète »** signifiant que les éléments constitutifs des systèmes ne se distinguent plus du système bénéficiaire auquel ils appartiennent puisqu'ils forment un tout indissociable.
- Une **intégration « forte »** signifiant que les éléments constitutifs des systèmes se distinguent des uns des autres, mais cependant toute modification sur un des éléments impacte directement les autres,
- Une **intégration « faible »** signifiant que les éléments constitutifs des systèmes continuent d'exister par eux-mêmes et peuvent participer comme un élément d'un système intégré.

Notre problématique d'assemblage de SI hétérogènes et autonomes en un réseau réticulaire évoluant dans un contexte potentiellement changeant, nous amène à considérer le mécanisme d'intégration faible sur lequel sont bâties les relations d'interopération. Ainsi ces interopérations sont soumises à des interactions coopératives entres des SI autonomes faiblement couplés afin d'atteindre, de manière adaptative, les objectifs du système à grande échelle ainsi constitué (Fisher 2006).

(Vernadat 2007) compare ce mécanisme d'intégration faible à un mécanisme d'interopérabilité permettant à deux ou plusieurs SI d'échanger ou de partager de l'information, mais aussi leur permettant d'utiliser les fonctionnalités des uns et des autres, tout en préservant à chacun leur autonomie.

Résoudre nos problématiques initiales nous amène pour la suite de ce travail à considérer les relations d'interopération entre des SI comme étant mis en œuvre par un couplage faible. Nous notons que ce couplage faible correspond aux différents critères caractérisant un SdS, en particulier le critère d'émergence qui, lors d'assemblage de systèmes constitutifs, met en évidence une certaine forme d'agilité puisqu'un SdS continue à remplir sa mission malgré les évolutions de ses éléments constitutifs.

Dans notre proposition de formalisation de SdS présentée dans le paragraphe 2 suivant, nous retiendrons donc que les relations d'interopération supportant la structure d'un SdS seront mises en œuvre par un mécanisme d'interopérabilité et que les aspects synchronique et diachronique de cette dernière nous permettront d'identifier les types d'échanges B2B et B2M réalisés au sein d'un SdSI. Ce dernier point sera développé dans le chapitre 2, afin de proposer que toutes ingénieries associées systématiquement au SAF doivent participer également à ces relations d'interopération.

Aussi les collaborations au sein d'un réseau réticulaire de SI nous amène à prendre en compte ce besoin d'interopération à la fois d'un point de vue des SI, éléments constitutifs de type COTS propre à un SAF, et à la fois d'un point de vue des modèles d'ingénieries qui sont nécessaires pour la mise en œuvre des interopérations synchroniques et diachroniques.

Aussi, nous prenons en compte ce besoin d'un tout composé d'interopérations entre un SAF et son SPF dédié au travers d'une ingénierie guidée par un cadre de modélisation, objet de nos travaux, et proposé dans le chapitre 2 de ce mémoire.

2 Proposition de Formalisation d'un SdS

Dans le paragraphe précédent nous avons caractérisé de manière informelle ce que nous entendions par SdS et plus particulièrement par SdSI. Cependant, afin de proposer un cadre de modélisation d'un SI centré sur le produit, cette définition informelle doit être formalisée pour d'une part, capitaliser le concept de SdS appliqué à nos travaux, et d'autre part, pour contribuer à une Ingénierie Système Basée sur des Modèles (ISBM) pour concevoir un SdSI.

La formalisation d'un SdS, que nous proposons dans ce paragraphe 2, doit être prise comme semi formelle, puisque nous utilisons pour modéliser les particularités d'un SdS le diagramme commutatif ainsi que la notion de pattern composite décrite en UML.


2.1 Formalisation de la démarche de construction d'un SdSI

Nous avons formalisé la dynamique de construction d'un SdS au travers de diagrammes commutatifs. Le principe de construction d'un diagramme commutatif consiste à définir des objets apparaissant comme autonomes et des morphismes²² entre des paires d'objets, tels que le chemin entre deux objets soit équivalent à une composition de morphismes entre les objets concernés. Ces éléments d'un diagramme commutatif peuvent alors participer à la définition d'un ensemble d'objets. Dans nos travaux, nous interprétons cet ensemble par un SdSI composé d'objets (les SI constitutifs) et de morphismes (les relations d'interopérabilité).

Formaliser la démarche de construction d'un SdSI nous amène à formaliser de manière « diachronique » les différents assemblages d'objets tout en utilisant les critères comme éléments de progression dans les différentes étapes de formalisation.

Ainsi, du critère d'autonomie des SI d'un SdSI, nous définissons un objet S1 représentant un SI autonome possédant, de ce fait, sa finalité intrinsèque (Tableau 4). A cet objet, nous lui adjoignons un deuxième objet S2 afin de former, par anticipation, une paire d'objets dont les éléments seront des morphismes.

Tableau 4. Étape 1 du diagramme commutatif : définition de S1 et S2

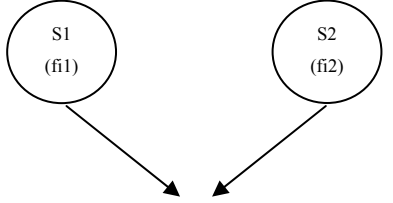
Critère d'autonomie	Étape 1		
	Définition de S1 Finalité intrinsèque de S1 : f1 Définition de S2 Finalité intrinsèque de S2 : f2		

Le critère d'appartenance nous conduit à définir un morphisme particulier consistant à inclure au sein d'un même ensemble les deux précédents objets tout en conservant leurs propriétés d'autonomie (Tableau 5). Nous considérons ce morphisme de « plongement²³ » comme permettant de représenter la potentialité de deux SI hétérogènes à participer à une mission commune, sans perdre leur intégrité ni leur finalité intrinsèque originelle.

²² Morphisme : un morphisme est une application entre deux ensembles munis d'une même espèce de structure algébrique, qui respecte cette structure.

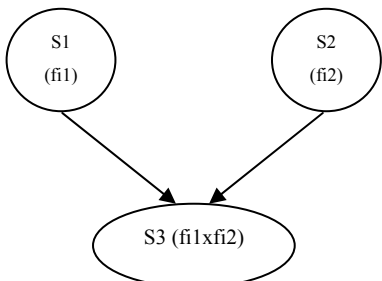
²³ Plongement : un plongement est une application f injective, qui préserve (au sein de l'ensemble d'arrivée : le $S1 \times S2$) les structures de départ des systèmes S1 et S2.

Tableau 5. Étape 2 du diagramme commutatif : définition du morphisme de plongement

Critère d'appartenance	Étape 2	
	Définition du morphisme de plongement	

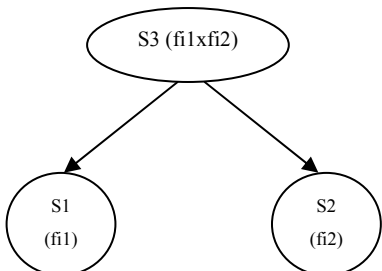
A ce niveau de construction du diagramme commutatif, le critère de connectivité est représenté par une agrégation (X) entre les deux objets S1 et S2 afin de définir un nouvel objet S3 (Tableau 6). Les propriétés de ce nouvel objet sont issues du produit cartésien entre les propriétés de deux objets. Nous notons que cette phase de « définition/caractérisation en compréhension » doit impliquer la maîtrise d'œuvre de rang 1 afin de prescrire les solutions envisagées pour établir les relations d'interopérabilité entre SI hétérogènes (critère de diversité) dans le but de former un SdSI.

Tableau 6. Étape 3 du diagramme commutatif : définition de la composition S1xS2

Critère de connectivité Critère de diversité	Étape 3	
	Définition de la composition S1xS2 Définition de S3 Finalité intrinsèque de S3 : f3	

L'étape 4 consiste à affirmer que les objets S1 et S2 désormais inter-opérants, conservent malgré tout leur autonomie d'un point de vue de leur fonctionnement et de leur finalité (Tableau 7).

Tableau 7. Étape 4 du diagramme commutatif : définition du morphisme de projection

Critère d'autonomie Critère d'appartenance Critère de connectivité Critère de diversité	Étape 4	
	Définition du morphisme de projection	

L'étape 5 participe partiellement au critère d'évolution en permettant, à travers une combinaison de morphismes de plongement (Étape 5 - Tableau 8) et de projection (Étape 7 - Tableau 10), de représenter tant l'évolution d'un SdSI (objet S3 (S1xS2)) que l'évolution de chacun des deux objets S1

& S2 constitutifs du SdSI (Tableau 8). Cette étape se doit de conserver les critères d'autonomie, d'appartenance, de connectivité et de diversité de l'ensemble des objets.

Tableau 8. Étape 5 du diagramme commutatif : définition du morphisme de plongement

	Étape 5 & Étape 7	
Critère d'évolution Critère d'autonomie Critère d'appartenance Critère de connectivité Critère de diversité	Définition du morphisme de plongement.	

L'étape 6 (Tableau 9) est une étape qui formalise l'ensemble des propriétés des solutions envisagées pour concevoir tant les systèmes d'information constitutifs que le système de SI auquel ils participent. Cette étape de caractérisation en extensionnalité s'apparente à la production du cahier des charges produit par la maîtrise d'ouvrage en Ingénierie Système qui décrit des finalités particulières, converties en exigences fonctionnelles ou organisationnelles pour chaque système constitutif (ici S1 et S2). Ces exigences seront à la base du cahier des charges que le maître d'œuvre de rang 1 devra respecter.

Tableau 9. Étape 6 du diagramme commutatif : définition des propriétés des solutions envisagées

	Étape 6	
Critère d'évolution Critère d'autonomie Critère d'appartenance Critère de connectivité Critère de diversité	Définition des propriétés des solutions envisagées.	

Tableau 10. Étape 7 du diagramme commutatif : définition du morphisme de projection

	Étape 7	
Critère d'évolution Critère d'autonomie Critère d'appartenance Critère de connectivité Critère de diversité	Définition du morphisme de projection.	

A ces différentes étapes de construction du diagramme commutatif de la constitution d'un SdSI, il est nécessaire d'y appliquer entre les étapes 4 & 6 une application d'identité afin de s'assurer, lors de la

démarche de formalisation de la conservation des propriétés et des critères définissant un SdSI, le résultat de cette démarche de construction.

Nous notons que dans cette phase de formalisation de la démarche de construction d'un SdS, nous n'avons pas tenu compte du critère d'émergence car nous n'avons pu, avec les principes mis en œuvre par la méthode des diagrammes commutatifs, l'intégrer dans la démarche de construction.

2.2 Formalisation d'un SdSI

Si le paragraphe précédent nous a amenés à proposer une semi-formalisation de la démarche de construction d'un SdSI pour appréhender les différents critères, nous proposons désormais de semi-formaliser le résultat de cette démarche, à savoir un SdSI. En effet, il s'agit de proposer un modèle, voire un méta-modèle représentant un SdSI, afin de capitaliser les différents critères et propriétés le caractérisant.

La difficulté de représenter le critère d'émergence par la méthode précédente nous a amenés à chercher un autre langage de semi-formalisation afin de prendre en compte ce mécanisme très particulier de construction de systèmes.

De manière générale, formaliser l'émergence est un exercice de modélisation très difficile puisqu'il s'agit de représenter que les propriétés du comportement émergent dans un SdS ne peuvent pas se réduire à la somme des propriétés des comportements de ses systèmes constitutifs et que de nouvelles propriétés émergent de la combinaison de ces systèmes constitutifs (Simon 1996). Ce type de problématique fait l'objet de nombreux travaux (Johnson C.W.; Cariani P 1991 ; Bedau M 1997; Ueda K 2001; Bar-Yam 2004) dans la communauté scientifique internationale afin de proposer des solutions à cette modélisation de l'émergence.

Cependant, nous notons que le mécanisme d'émergence dans un SdS, et plus généralement la complexité qui en découle, sont fortement liés, d'une part, au facteur d'échelle ou de granularité de points de vue sous lesquels sont observés tant un SdS que ses systèmes constitutifs (Kuras 2006) et d'autre part, à la structure d'un SdS, en tout point gigogne, dont la conception, dans un contexte donné et pour un but précis, est basée sur le principe de récursivité (Carney et al. 2005).

Dans un premier temps, nous proposons dans ces travaux de considérer chaque système constitutif d'un SdS selon l'approche de (Kuras 2006) afin de tenir compte du facteur d'échelle d'un SdS. Cette approche consiste à modéliser un système et ses différents niveaux d'observation (par exemple, le niveau de la maîtrise d'ouvrage, le niveau de la maîtrise d'œuvre, ...) selon six modalités de conceptualisation (*le focus ; la correspondance entre le point de vue et l'étendue ; la correspondance entre la richesse et la résolution ; l'identification ; le cadre de référence ; le changement d'échelle*) guidées par trois attributs : le point de vue de la conceptualisation → (Field of

View, FOV), la résolution de la conceptualisation → (Résolution, RES) correspondant à un facteur d'échelle v , et le contenu (de l'information) → (CONTENT). (Figure 13)

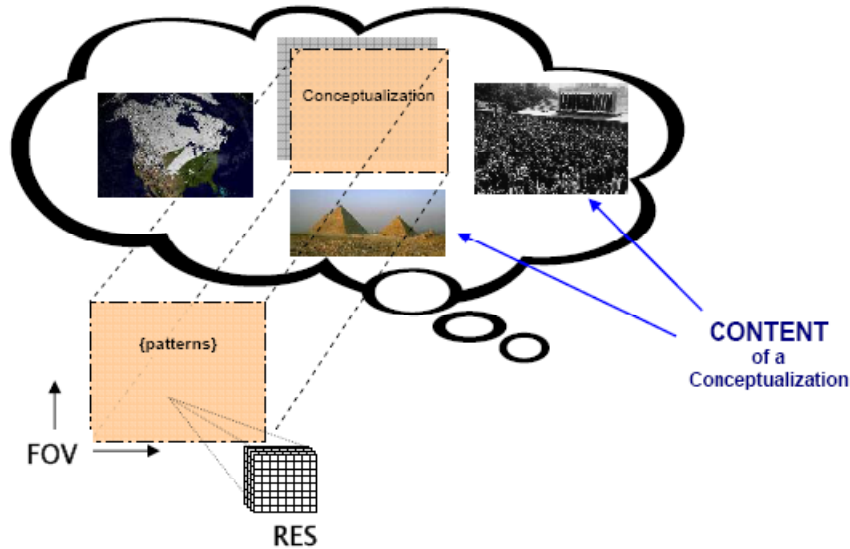


Figure 13. Conceptualisation d'un système suivant des points de vue et des différentes résolutions (Kuras 2006).

Ainsi, suivant cette approche, un système observé à un certain niveau d'échelle peut être formalisé par l'équation suivante :

$$s^v \in \{S^v\} \mid S^v \equiv \{E_i; REL_j\}_{I^v} H^v \quad \text{Équation 1 (Kuras 2006)}$$

où $\{E_i; REL_j\}$ est un « pattern » $\{pattern\}$ formalisant l'ensemble des conceptualisations humainement possible d'un système S observé à un **facteur d'échelle** v ; E **élément** de S et REL les **relations** entre ses éléments ; H^v (H étant un **Holon**) est un cadre de référence filtrant parmi l'ensemble des conceptualisations possibles ceux qui représentent le système à l'échelle v .

Ce cadre utilise notamment le concept d'**Holon** (Koestler 1967) pour extraire le **patron de conceptualisation (design pattern)** le plus adéquat pour modéliser notre système-cible, comme Baïna (Baïna 2006) l'a intuitivement fait dans ses travaux en Interopérabilité Dirigée par les Modèles (IDM) des Systèmes d'Entreprise. Notons aussi que ce concept est la base de diverses réflexions en 'system thinking' dans la communauté d'IS (Sausser and Boardman 2007).

Afin d'exprimer qu'un système est un agrégat d'objets, qui peuvent être eux-mêmes des systèmes, Kuras (2006) propose de conceptualiser un système à l'image d'un ensemble récursif de "pyramides" où chaque objet est une propriété de la "pyramide" à laquelle il appartient et forme avec un ou d'autres objets une agrégation de valeurs de ces propriétés (Figure 14).

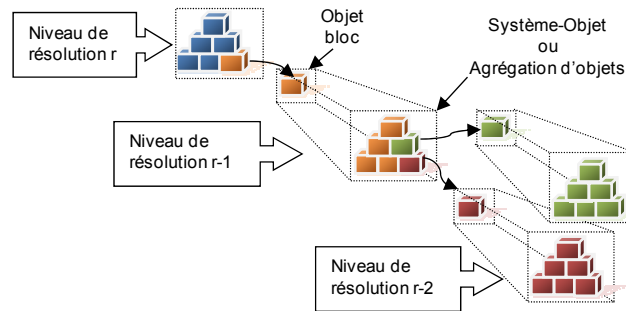


Figure 14. Notion d'agrégation et de résolution d'un système.

Chacun de ces objets est observé à des niveaux de point de vue ou de résolution différents. Ainsi, cet ensemble d'objet, conceptualisé par un pattern, composant un système peut s'exprimer par $S^v \equiv \{pattern_k\} \subseteq \{pattern_{k'}\}$ avec $k \in K$ où K est un ensemble d'index qui énumère tous les « patterns » dans des conceptualisations humainement possibles de la réalité avec un facteur d'échelle v ; $k' \in K'$ où K' est un ensemble d'index qui énumère tous les « patterns » dans des conceptualisations humainement possibles de la réalité avec des points de vue différents et des niveaux de résolutions plus importants.

Parce que notre objectif est de proposer un cadre de modélisation permettant aux parties prenantes d'un SdSI de partager des invariants de modélisation afin d'assurer la cohérence de l'ingénierie qui lui est appliquée, nous retenons dans notre travail la conceptualisation d'un système de systèmes d'information multi-échelles telle que proposée par (Kuras 2006). Ainsi, ces invariants de modélisation sont constitués par :

- les **différents niveaux de points de vue** ou de résolutions d'un SdSI. Dans notre travail, nous retiendrons essentiellement :
 - o le niveau de la **maîtrise d'ouvrage** (MOA) en tant qu'architecte du système à faire pour satisfaire les objectifs et les exigences du donneur d'ordre bénéficiaire du système final. Il s'agit pour la MOA de piloter un projet de conception d'un système de systèmes d'information et de garantir le respect des exigences quant à la coopération mise en œuvre entre les SI constitutifs pour réaliser la mission attendue.
 - o Le niveau de la **maîtrise d'œuvre** (MOE) en tant que spécialiste métier du SdSI devant être déployé et répondant aux exigences de la MOA.
- les **différents objets** qui constituent les différents SI constitutifs d'un SdSI.

Afin d'appliquer la conceptualisation d'un système telle que préconisée par (Kuras 2006), nous avons choisi de la filtrer à travers d'un patron de modélisation permettant de rendre compte de l'assemblage de systèmes pour former un SdS.

Pour ce faire, nous avons retenu la notion de '**pattern**' pour formaliser les éléments et leurs interrelations dans un SdSI et dans ses éléments constitutifs. En particulier, nous nous sommes

intéressés au principe de patron de conception (design pattern en Anglais) issu de la communauté du génie logiciel (Gamma et al. 1995). Dans ce domaine, un patron de conception est utilisé comme élément de capitalisation des procédés de conception d'applications logicielles et permet de décrire un problème et sa solution de telle façon que l'on puisse réutiliser ce patron chaque fois que l'on rencontre ce problème. Pour la conception d'un SdSI étant en tout point « gigogne », nous proposons donc d'utiliser ce mécanisme de récursivité des patrons de conception.

En particulier, nous nous sommes intéressés à l'un des 23 patrons de conception proposés par (Gamma et al. 1995), à savoir le pattern composite (Figure 15) permettant de décrire la structure récursive d'objets.

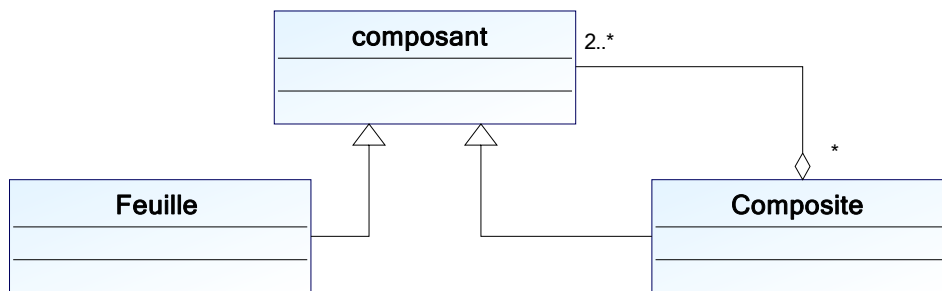


Figure 15. Modèle sémantique inspiré du pattern composite de (Gamma et al. 1995).

Dans un tel patron, les éléments d'un système de systèmes sont représentés à deux niveaux de composition et de manière récursive de type arborescente. Ainsi, appliqué à nos travaux, nous considérons les SI constitutifs d'un SdSI comme des composants formant par assemblage un composite c'est-à-dire un SdSI. L'arborescence entre « composite » et « composant » est semi-formalisée par une relation de généralisation, telle que définie en UML, afin d'exprimer que chaque « composite » peut devenir lui-même un « composant » d'un nouveau « composite ». Cette relation permet d'exprimer la récursivité de la structure en tout point équivalente dans un SdSI. La fin du mécanisme de récursivité pour modéliser un SdSI est exprimée par la notion de « feuille », représentant la description ultime d'un composant de système. Enfin, une association d'agrégation de multiplicité (2..* ; *) nous permet d'exprimer les relations d'interopération supportant l'assemblage des SI constitutifs d'un SdSI.

Ainsi, en dérivant et adaptant l'équation de Kuras (2006) (équation 1), nous proposons de conceptualiser un SdSI, à un niveau de résolution \mathcal{V} , selon :

$$SdSI^{\mathcal{V}} \equiv \{pattern_k\}_{K^{\mathcal{V}}}; pattern\ composite^{\mathcal{V}} \quad \text{Équation 2}$$

où $\{pattern_k\}$ formalise à un facteur d'échelle \mathcal{V} l'ensemble des SI constitutifs et leurs relations d'interopération ; $k \in K$, K étant l'ensemble des index des patterns ; $Pattern\ composite^{\mathcal{V}}$ est un

cadre de référence filtrant, à une échelle v , l'ensemble des conceptualisations possibles pour en isoler celles définissant le système $SdSI^v$ concerné. Ainsi, un SdSI observé à un niveau d'échelle v , est un ensemble de patterns auxquels on associe systématiquement un cadre de modélisation pour l'observer à ce niveau d'échelle. Après avoir proposé une démarche tenant compte de la complexité due au facteur d'échelle d'un SdS, il convient désormais d'étudier la structure de celui-ci, dont la particularité est d'être construite de manière récursive (Carney et al. 2005).

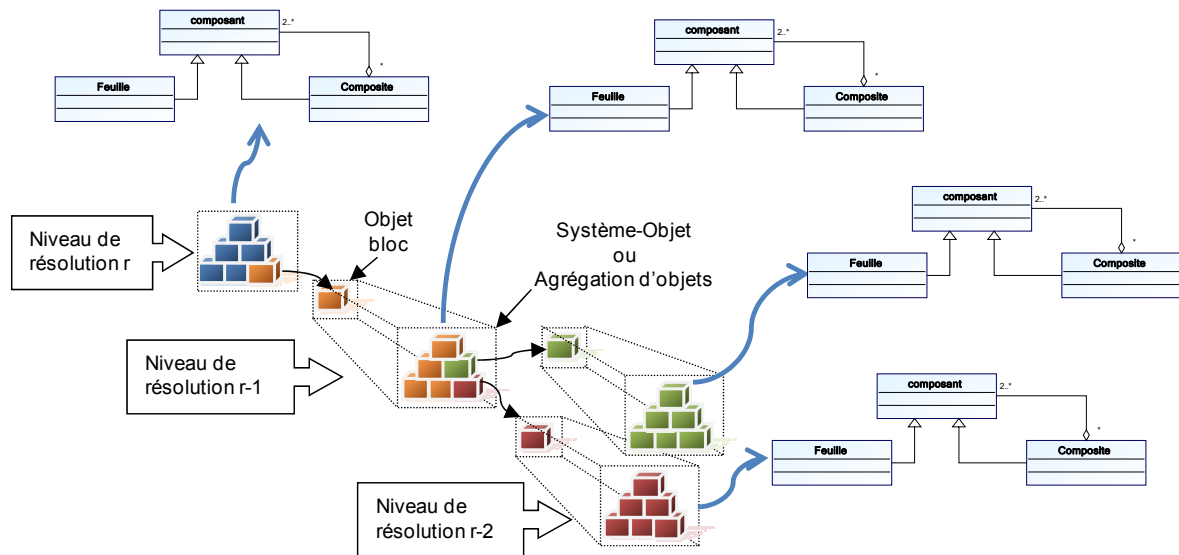


Figure 16. Conceptualisation des SI constitutifs à Faire "filtrés" par le pattern composite.

Selon (Vernadat 2007) il existe trois types de relations d'interopération entre des composants pour former un composite :

- des relations où les composants sont complètement couplés (ou intégrés) au point où l'on n'est plus capable de les distinguer des uns des autres. Le composite résultant est un système « **Fully integrated** ».
- des relations où les composants sont fortement couplés (ou intégrés) se distinguant des uns des autres mais dont les moindres évolutions impactent sur les autres. Le composite résultant est un système « **Tightly coupled** ».
- des relations où les composants sont faiblement couplés se distinguant des uns des autres mais dont les moindres évolutions n'impactent pas sur les autres. Le système résultant est un système « **Loosely coupled** ».

À un niveau de récursivité supérieur, les systèmes composites présentés précédemment peuvent eux-mêmes participer à un nouvel assemblage pour former un nouveau système composite :

- Un « **fully integrated system** » est une sorte de boîte noire pour laquelle on considère qu'elle ne peut pas par conséquent être le composite de systèmes faiblement, étroitement ou/et complètement intégrés.
- Un « **tightly coupled system** » peut être **composé** uniquement de « fully integrated system », de « tightly coupled system, de « loosely coupled system » ou d'un mixte des trois types de systèmes.
- Un « **loosely coupled system** » peut être **agrégé** uniquement de « fully integrated system », de « tightly coupled system, de « loosely coupled system » ou d'un mixte des trois types de systèmes.

Afin de rendre compte des différents niveaux de récursivité de construction de ces systèmes qu'ils soient composite ou qu'ils soient composant, nous introduisons les notions « abstraites » de « **tight-component-system** » afin de symboliser les étapes intermédiaires de composition d'un système et « **loose-component-system** », afin de symboliser une étape intermédiaire d'agrégation d'un système. Nous entendons par là, qu'il n'est pas possible de définir la nature du système composite ou composant *à priori*, tant que l'on n'a pas défini l'ensemble des types d'assemblages.

À partir de ces différents postulats appliqués à notre problématique de SdS, nous proposons un méta-modèle (Figure 17) rendant compte des relations d'interopérations particulières que sont la composition et l'agrégation entre des composants pour former des composites. Les « ports » matérialisant le point de départ et le point d'arrivée d'interopérations sont typés respectivement par une interface « input » et une interface « output ».

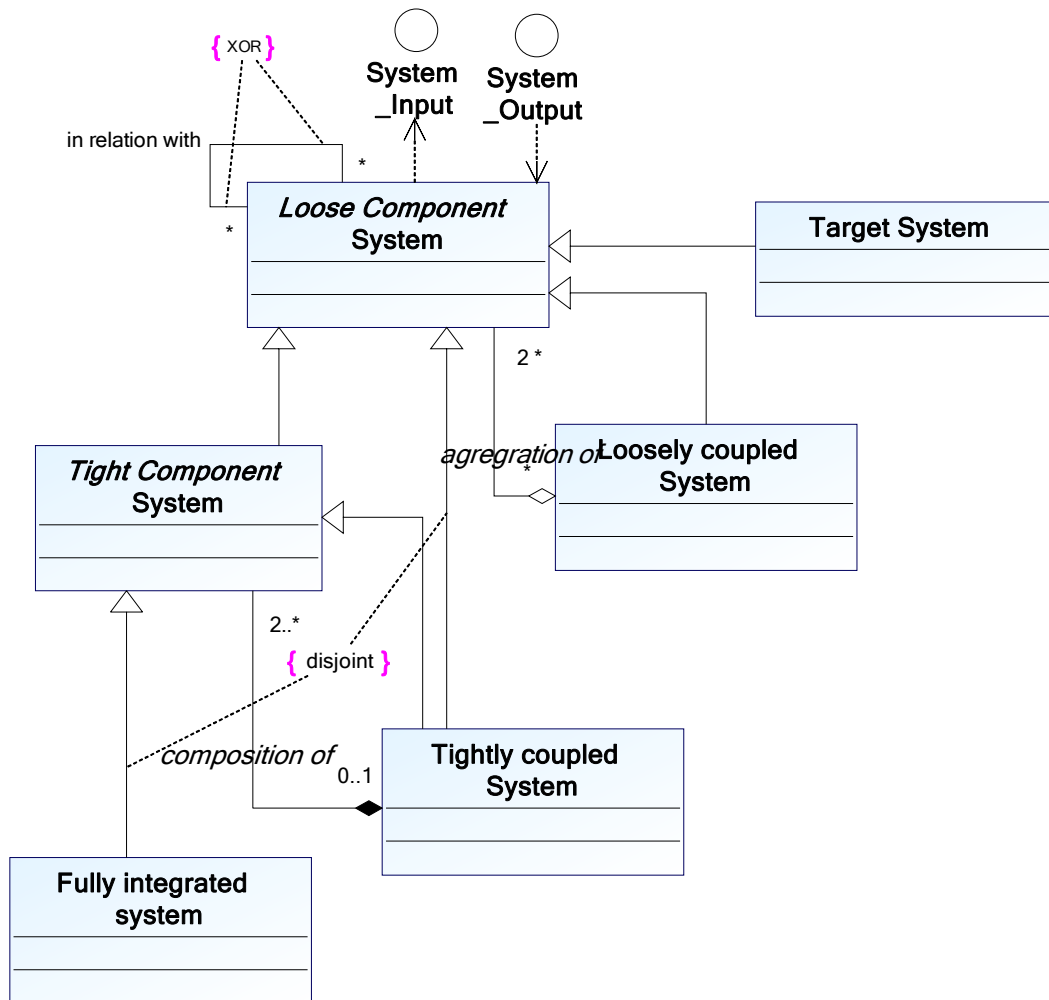


Figure 17. Modèle sémantique : formalisation UML de systèmes « loosely coupled », « tightly coupled », « fully integrated »

Dans ce méta-modèle, les systèmes, qu'ils soient composants ou composites, sont :

- **autonomes** et **divers** (hétérogènes puisqu'ils sont représentés par des classes indépendantes)
- des éléments **appartenant** par agrégation ou par composition à un ou plusieurs composites
- **connectés** entre eux par des associations,
- **évolutifs** parce qu'ils peuvent soit être composants, soit composites et qu'ils peuvent évoluer en fonction des types de relations.

À partir des définitions précédentes à propos d'un SdS et des différents types de relations entre des systèmes, nous postulons qu'un SdS est le résultat d'une agrégation de systèmes, construit à travers la notion abstraite de « *tight-component-system* » et de « *loose-component-system* », qui peuvent être de classe « *loosely coupled system* », « *tightly coupled system* », ou « *fully integrated system* ».

Le résultat de cette agrégation est lui-même un « *loosely coupled system* », condition nécessaire et non suffisante pour satisfaire aux critères de définition d'un SdS. De plus, le changement d'échelle observé entre les différents niveaux de récursivité, sous tend le mécanisme d'émergence,

caractéristique d'un SdS. Pour représenter cette émergence (Mayer and Lavigne 2001), nous proposons d'utiliser un stéréotype UML, nommé « weak emergence », basé sur le concept UML de classe d'association, dont nous avons étendu la sémantique par la notion de « graphe connexe²⁴ » (un unique SdS peut émerger d'un ensemble de 2 ou plus composants en interrelation, formant un graphe connexe) et la contrainte des 6 critères de définition d'un SdS. Ainsi, à un facteur d'échelle ν , un SdS émerge de l'agrégation d'au moins deux systèmes quels qu'ils soient de classes quelconques et formant un « graphe connexe » afin de satisfaire au critère d'appartenance d'un SdS.

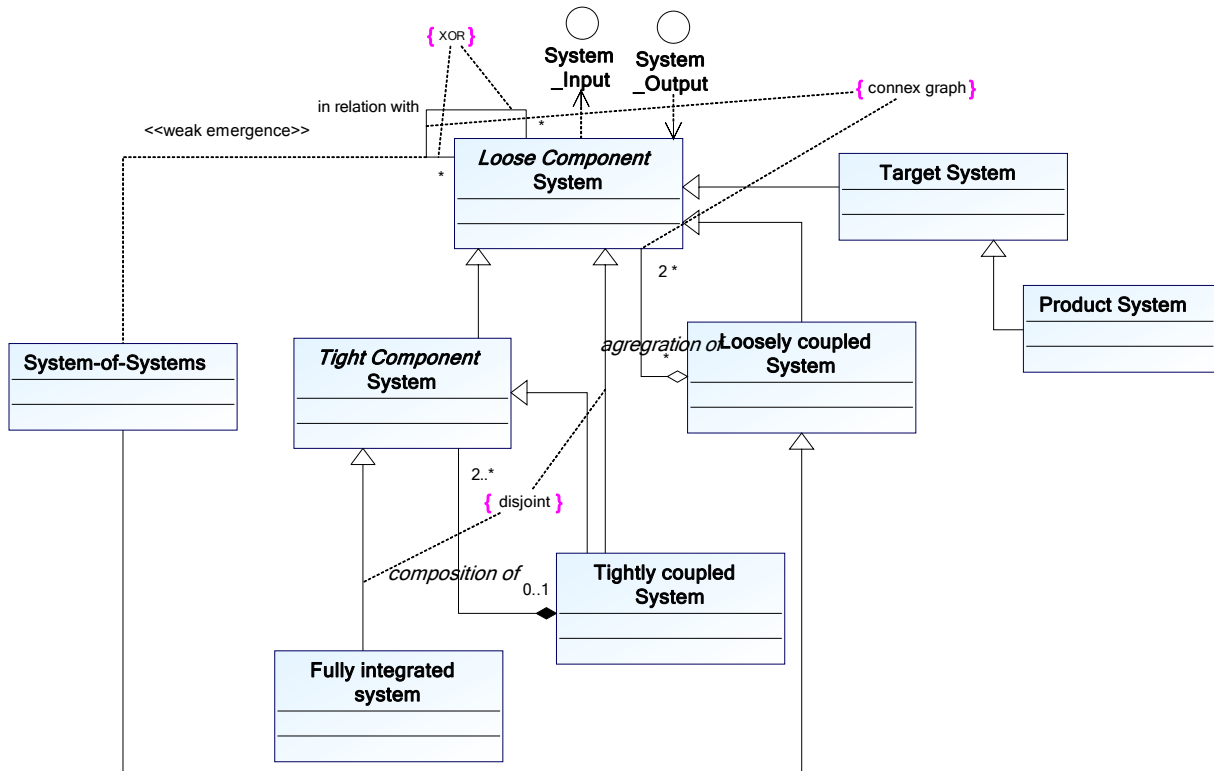


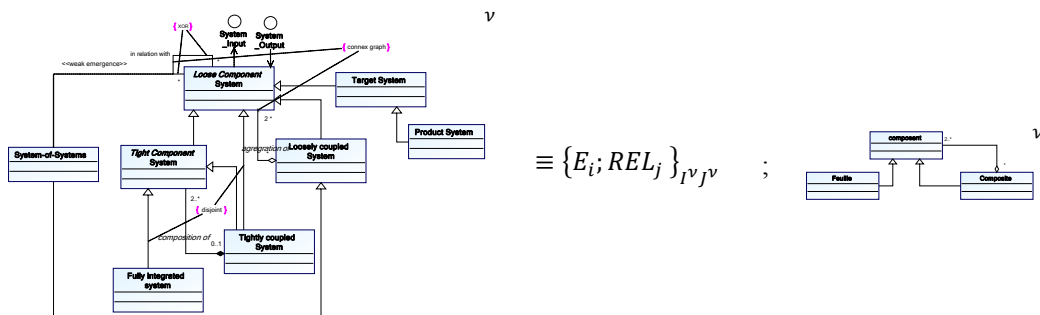
Figure 18. Proposition d'un modèle sémantique de méta-modèle de Système de Systèmes centré sur le produit

Nous avons vu précédemment que Kuras (2006) propose de conceptualiser un système comme étant un ensemble d'éléments et de relations, mais, nous avons vu également qu'il ne propose pas de règle qui définissent les relations entre ces éléments et donc la structuration du système à conceptualiser. Ainsi, c'est le système lui-même qui pose les règles définissant les relations entre ces éléments constitutifs.

Or notre proposition de méta-modèle (Figure 18) fournit des règles qui définissent les relations entre les systèmes constitutifs (ou composants ou composites), mais ce méta-modèle ne formalise pas les notions de points de vue ou de résolution des systèmes composites ou composants entre eux. Pour

²⁴ Un graphe est connexe lorsqu'il est possible de trouver au moins un parcours permettant de relier les nœuds deux à deux.

que la MOA et la MOE de rang 1 & 2 puissent conceptualiser un SdSI suivant les points de vue de chacun avec un niveau de résolution qu'il leur est propre, nous proposons que l'« Holon » soit un « système pour faire » comme le cadre de modélisation de Zachman qui établit les règles de changement de résolution (d'échelle) et de changement de point de vue (MOA, MOE de rang 1 & 2). Ainsi, un SdSI peut être conceptualisé par un pattern instancié *{instanciation de notre méta – modèle de SdSk}* pour une résolution ou facteur d'échelle donné v , et dont la conceptualisation des points de vue du SdSI est réalisée par le pattern composite (Équation 3). Aussi, nous proposons dans ce mémoire de définir un système de systèmes d'information multi-échelles à partir de l'équation 2 (page 44 de ce mémoire) selon :



Équation 3

Conclusion du chapitre 1

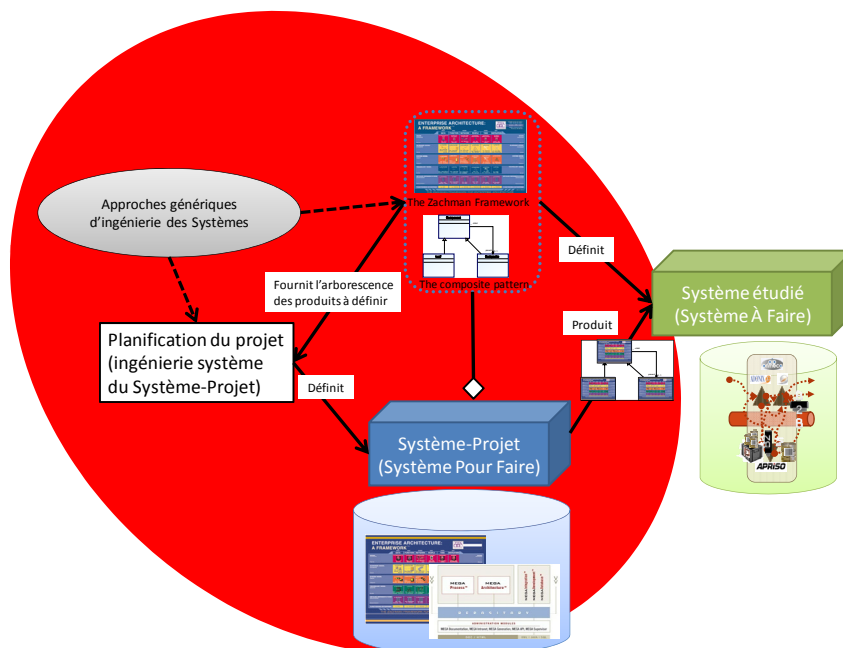
Nous avons vu dans ce chapitre que l'interopérabilité entre des SI appartenant à un **réseau réticulaire d'entreprises** devient de plus en plus difficile à mettre en œuvre, de par les **interopérations éphémères** entre les SI d'entreprises qui désirent contribuer ensemble à la réalisation d'un bien à forte valeur ajoutée : dans notre cas un produit. Nous avons montré aussi que les causes qui engendrent ces difficultés sont dues aux propriétés particulières de chacun de ces SI comme l'autonomie, l'évolution, l'émergence, la connectivité, la diversité ou bien encore l'appartenance, dans un contexte de composition réticulaire de SdS. Aussi, nous avons proposé une semi-formalisation d'un **SdS** pour distinguer son mécanisme d'**agrégation**, du mécanisme de **décomposition** d'un système plus « classique » dit monolithique. Pour semi-formaliser le mécanisme de construction d'un SdS, nous avons utilisé un diagramme commutatif qui décrit diachroniquement cette construction. Or, nous avons noté que ce mécanisme ne nous permet pas de représenter le critère d'**émergence** de comportements. D'autre part, nous nous sommes aussi intéressés à la particularité de composition en tout point gigogne d'un SdS et à sa représentation récursive en langage UML utilisée en génie logiciel : le **pattern composite**. Ainsi, un premier résultat de nos travaux a été de proposer un méta-modèle d'un SdS, basé sur le pattern composite.

Ce méta-modèle participe de la sorte à la **capitalisation** des modèles dès lors où il est instancié par des SI particuliers contribuant à un SAF pour une mission particulière. Ainsi, nous proposons que chaque « entrée-sortie » de chaque système (Figure 18 ; Chapitre 1.2.2) participe, en tant qu'interface, à la résolution des problématiques d'interopération de leur SI respectif. Aussi, nous proposons dans le chapitre 2 de ce mémoire que l'ingénierie de ces interfaces et de leurs relations d'interopération se basent sur les mécanismes de composition et d'agrégation de SI définis à partir de notre méta-modèle instancié à un Système À Faire (SAF) particulier que nous proposons mais aussi à son Système Pour Faire (SPF) qui lui est associé. Ainsi, nous postulons que le modèle de l'ingénierie des interopérations d'un SdSI, doit suivre le même mécanisme de construction, ce que nous proposerons dans le chapitre suivant.

CHAPITRE 2.

... AU SYSTÈME DE SYSTÈMES D'INGÉNIERIE POUR FAIRE UN SYSTÈME DE SYSTÈMES D'INFORMATION

Formalisation d'un cadre de modélisation d'un SI vu comme un SdSI centré sur le produit



Introduction du chapitre 2

Ce chapitre a pour objectif de proposer un processus d'ingénierie basée sur les modèles pour modéliser un SdSI interopérable, de manière récursive du fait de la conception en tout point gigogne d'un SdS.

Intuitivement, notre démarche de réflexion s'est calquée notamment sur GERAM (Generalised Enterprise Reference Architecture and Methodology) développé par un groupe de réflexion sur les architectures (IFAC-IFAC Task Force on Architecture for Enterprise Integration) (GERAM 1999). Comme pour GERAM, nous nous sommes intéressés à choisir un cadre d'architecture de référence ainsi qu'un langage de modélisation, à proposer des concepts génériques, une méthodologie de modélisation, un outil de modélisation, des modèles, des modèles réutilisables et un système opérationnel. D'autre part, comme pour GERAM, nous partageons l'idée que l'ensemble des ingénieries (projet et sous-projets associés) doit être séquencé dans le temps au travers d'étapes bien identifiées. Pour notre part, nous nous sommes inspirés des phases clés proposées en IS : Préparation, définition et spécification.

Nous postulons que les aspects multi-échelles des métiers et des points de vue d'un SdSI interopérable peuvent être rendus cohérents à l'aide d'un référentiel commun de modélisation tel que Zachman. Ce cadre de modélisation nous permet de structurer et d'organiser les différents modèles d'ingénierie concernés au sein d'un modèle cohérent de « Système Pour faire ». Le langage que nous utilisons pour spécifier notre Ingénierie Système Basée sur les Modèles (ISBM) est UML. Nous utilisons également des méthodes de modélisation telles que la modélisation de processus afin d'exprimer le point de vue métier propre à chaque entreprise.

Comme nous l'avons vu dans l'introduction générale, le constat de Chapurlat (Chapurlat 2007) s'appuie sur le fait qu'il n'existe pas de tâche de modélisation ou même de gestion de modèles dans les entreprises, malgré les efforts de Recherche et de Développement de la communauté scientifique dans le domaine. Ce constat nous a ainsi amené à étudier tant les méthodes qui vont contribuer à modéliser le Système-Entreprise, que les outils qui vont participer à la capitalisation et au maintien de la cohérence des modèles de ce Système-Entreprise.

1 De l'ingénierie d'un SI à l'Ingénierie d'un SdSI

Les travaux de cette thèse se veulent être une contribution à l'ingénierie d'un système d'information (SI), venant en appui aux outils et méthodes d'Ingénierie Systèmes (IS) déjà existants, afin de tenir compte non seulement de l'évolution du paradigme d'intégration vers un paradigme de l'interopération (Fisher 2006; Panetto 2006) mais surtout des problèmes de capitalisation de la connaissance métier pour internaliser les savoir-faire d'ingénierie associés au SI d'entreprise.

En particulier, ces problèmes de capitalisation sont dus essentiellement d'une part, à l'externalisation des tâches de modélisation des systèmes d'information d'entreprise et d'autre part, au caractère particularisé des modèles d'entreprise construits pour une demande (un projet) et donc non réutilisables.

Résoudre ces problèmes, outre les démarches de management et de capitalisation de la connaissance, passe par la définition d'une démarche capable de fournir des invariants ou des objets de modélisation, voire des patterns, pour bâtir chaque modèle d'un SI de manière la plus générique possible. Ce type de démarche a fait l'objet de nombreux travaux notamment dans la proposition de méta-modèles ou de référentiel commun (Bruneau et al. 1992; Dieng et al. 1999; Zacklad and Grundstein 2001; Grundstein 2004; Lievre 2007).

Plus récemment, ce type de démarche a été proposé sous la forme d'une ingénierie d'un système intégrant *l'ensemble des produits, processus et compétences humaines pour apporter une solution économique et performante à ses donneurs d'ordre* (selon la définition de l'AFIS²⁵) (AFIS 2005).

En particulier, nous notons que l'IS est une démarche méthodologique basée sur des **éléments invariants** (en rapport avec les secteurs d'activité, les types de problème, les types de système-solution, les types d'organisation industrielle) issus de :

- L'expérience : les bonnes pratiques pouvant devenir transposables d'un projet à un autre deviennent des invariants pour réaliser l'ingénierie d'un système,
- La science : le concept de base de l'IS étant la notion de système, tout ce qui, scientifiquement, traite de ce concept fournit des invariants à la base de toute démarche de modélisation et de simulation des systèmes réalisés par l'IS.

Dans ce cadre de capitalisation, nos travaux contribuent à l'IS en proposant des éléments invariants ou des **invariants de modélisation** sur lesquels est bâti le cadre de modélisation d'un système d'information centré sur le produit afin de fournir à la MOA et aux MOE de rangs 1 & 2 une démarche d'ingénierie générique basée sur les modèles avec pour objectifs d'associer systématiquement tout SI à son ingénierie particulière et de capitaliser l'ensemble des modèles ainsi acquis. Cette distribution des modèles sur les systèmes d'information qu'ils représentent contraste avec d'autres approches de modélisation, telle que dans l'**Ingénierie Dirigée par les Processus**, qui consiste en une cascade de modèles de définition de SI, de conception de SI, ... (Estefan 2007). Notre approche d'**Ingénierie Basée sur les Modèles (IbM)** (Figure 19) se différencie également des approches d'**Ingénierie Dirigées par les Modèles (IDM)** puisque nous n'avons pas l'objectif de transformer des modèles mais simplement de les créer, de les capitaliser tout en assurant leur traçabilité et leur inter-relations.

²⁵ AFIS (<http://www.afis.fr>)

Dans cette partie du mémoire nous proposerons un état de l'art sur ces différentes ingénieries. Bien que ne les utilisant pas intégralement dans notre démarche, nous utilisons en partie leurs éléments invariants afin de supporter certains aspects métiers de l'ingénierie que nous proposons et ainsi compléter certaines ingénieries particulières associées aux SI constitutifs d'un SdSI. Cette façon de procéder permet aux acteurs de la MOA et de la MOE de rang 1 & 2 des SI constitutifs de garder un certain degré de liberté quant au choix des méthodes et outils dans les phases de spécification et de développement.

Afin de respecter les différentes problématiques évoquées dès l'introduction de ce mémoire, nous proposons donc un cadre de modélisation pour une Ingénierie Système Basée sur les Modèles (système pour faire) afin de développer un système de systèmes d'information centré sur le produit (système à faire).

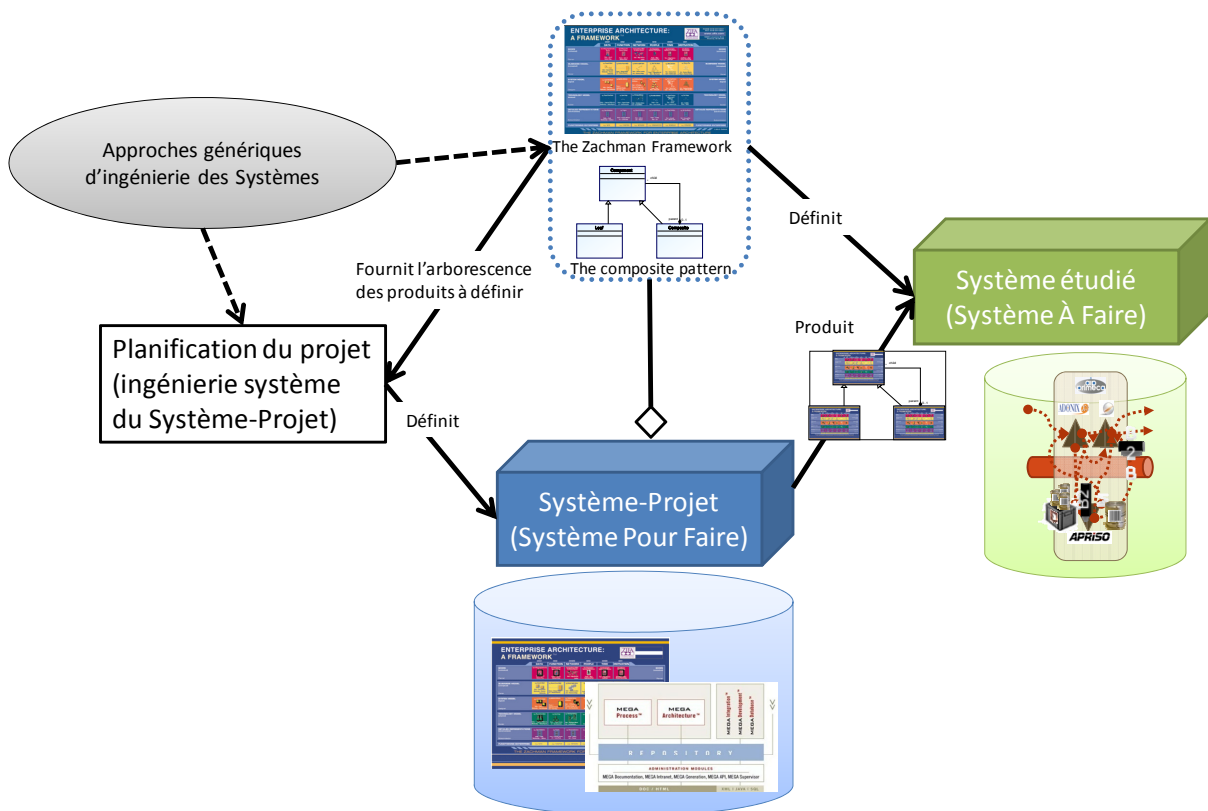


Figure 19. Ingénierie Basée sur les Modèles

*Le **Système Pour Faire** : « Pour organiser, exécuter et coordonner toutes les **activités** qui conduisent à la réalisation et à la mise à disposition du système à faire ., il est nécessaire de mettre en place un système doté de ressources humaines et techniques : le système pour faire ou projet. » (AFIS 2005)*

*Le **Système À Faire** : « Le système est un concept, une abstraction, le **résultat** de sa réalisation est un produit (Produit-Système). Un système peut être réalisé en un ou plusieurs exemplaires. Ces exemplaires peuvent éventuellement différer, selon des variantes prévues par la définition du système. » (AFIS 2005)*

1.1 Ingénierie Dirigée par les processus – les modèles

Généralement, l'ingénierie s'inspire de normes et standards issus de consensus entre des experts dans des domaines variés et particuliers, pour concevoir des processus d'ingénierie eux-mêmes particuliers, par de bonnes pratiques et par des méthodes propres aux entreprises qui cherchent à réaliser un SAF. L'IS va ainsi favoriser le développement de méthodes et d'outils en se basant sur des normes et standards permettant de comprendre et de gérer la complexité des SAF (Rochet 2007). Parmi les normes et standards utilisés en IS, la norme ISO/IEC 15228 (ISO/IEC-15288 2002) identifie 25 processus, eux-mêmes regroupés dans 4 familles de processus : les processus d'entreprise, les processus contractuels, les processus du projet et les processus techniques. La norme ISO/IEC 15288 est affinée par 2 autres normes. Ainsi l'on retrouve notamment la norme IEEE 1220 (IEEE-Standard-1220 2005) qui se focalise sur les processus techniques d'Ingénierie Système allant de l'analyse des exigences jusqu'à la définition physique du système et la norme (ANSI/EIA-632 1999) qui complète les processus techniques de définition du système en couvrant la réalisation des produits jusqu'à leur mise en service (transfert du concepteur à l'utilisateur). De plus, elle incorpore les processus contractuels d'acquisition et de fourniture.

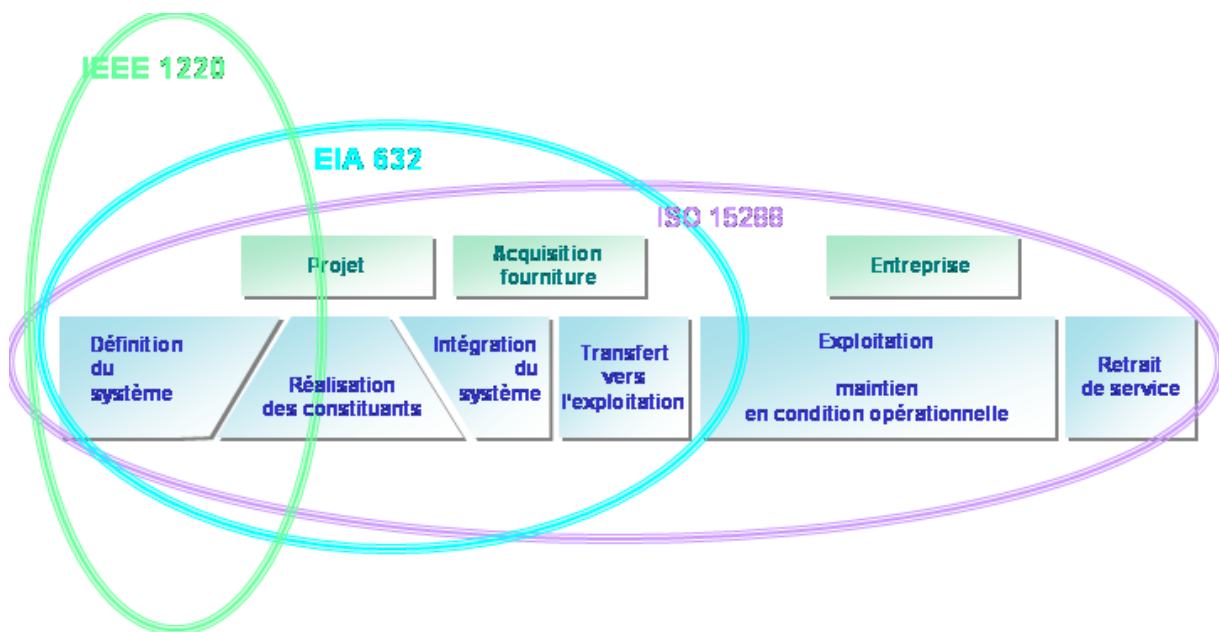


Figure 20. Couverture des normes relatives à la mise en œuvre d'une IS durant le cycle de vie d'un système. (AFIS : <http://www.afis.fr/doc/normes/normes.html>)

Notre périmètre de recherche méthodologique ne se borne pas seulement à deux des phases de la norme ISO/IEC 15288 qui sont : la phase de « **définition du système** » et la phase « **d'intégration du système** ». En effet, nos travaux vont aussi considérer la phase amont de l'ingénierie d'un « Système à Faire » : la **préparation du projet** (encadré « périmètre de la thèse » - Figure 21).

- La préparation du projet par la maîtrise d'ouvrage (MOA) : J0 à J1
- La définition du SAF par la maîtrise d'œuvre (MOE) de rang 1 : J1 à J2
- La spécification du SAF par la maîtrise d'œuvre (MOE) de rang 2 : J2 à J3

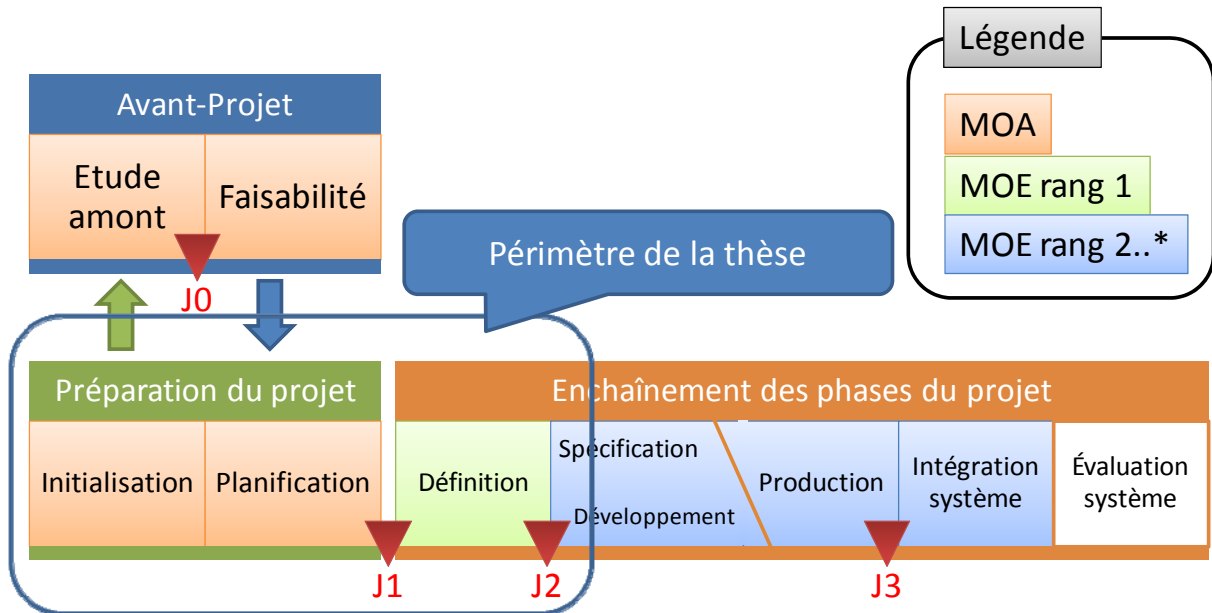


Figure 21. Cycle de vie générique d'un projet, inspiré de (Tucoulou 2008)

1.2 Ingénierie Dirigée par les Modèles (IDM)

L'IDM, issue du domaine de l'informatique, propose une « mécanisation » du processus d'élaboration de logiciels par modélisation. Cette « mécanisation » est bâtie sur la base de connaissances d'ingénieurs expérimentés et de leurs bonnes pratiques en développement de logiciels. L'ordonnancement de l'ensemble des transformations de modèles est alors vu comme un processus de conception. Ce processus doit inclure l'ensemble des étapes de modélisation des plus conceptuelles pour aboutir à la réalisation d'une application exécutable.

Nos travaux s'inspirent de l'IDM sans en reprendre le cœur car, bien qu'elle réponde bien au besoin d'un ingénieur système de vouloir transformer des modèles et de les mettre en cohérence. Nos travaux de recherche ne se focalisent pas sur les problématiques de transformation de modèles mais plutôt sur les problématiques de capitalisation de modèles que nous qualifierons ainsi **d'Ingénierie Système Basée sur les Modèles (ISBM)** pour éviter tout amalgame et ambiguïté.

1.3 Ingénierie Système Basée sur les Modèles (ISBM)

L'objectif majeur de cette thèse est de proposer une méthodologie favorisant les interopérations entre des SI et la capitalisation de leurs ingénieries à travers le temps et l'espace. Cependant, comme nous venons de le voir, l'IDM n'est pas adaptée à notre problématique car elle se focalise principalement

sur la transformation de modèles. Or, comme nous désirons capitaliser des modèles issus d'ingénieries particulières (dans le temps et l'espace) pour des SI distribués et hétérogènes, nous nous sommes intuitivement orientés vers la recherche de cadres de modélisation.

1.4 Les cadres d'architectures

(Software Productivity Consortium²⁶)

« Un cadre fournit un espace générique de problèmes et un vocabulaire commun dans lesquels les individus peuvent coopérer pour résoudre un problème spécifique. Les cadres ne sont pas nécessairement complets, mais ils peuvent être accrus pour fournir au moins un ensemble de questions de bases et des problématiques qui doivent être adressées dans le développement d'architecture. Pour beaucoup d'organismes et professionnels techniques, l'architecture était traditionnellement assimilée à un ou deux diagrammes incompréhensibles qui ont été à la base d'un nouveau projet et qui ne peuvent pas être changés parce que « trop de choses dépendent d'eux », ni ne peuvent être interprétés parce que leur signification n'est pas vraiment claire. Qu'est ce qui dépend d'eux, et comment cela a été décidé ? Les réponses à ces questions et à bien d'autres encore sont trop souvent perdues à cause des pressions dues aux exigences calendaires ou aux demandes du marché. Les cadres de modélisation peuvent fournir un guide aussi bien pour obtenir une vision plus large d'architecture que pour obtenir des schémas fonctionnels. »

Initialement, les cadres d'architecture ou cadre de modélisation d'architecture ont été développés à des fins militaires dans le but de définir des exigences par rapport aux fournisseurs de systèmes d'armes et de technologies de l'information. Aujourd'hui on les retrouve aussi bien dans l'administration territoriale que dans l'industrie. Les plus connus définissent des catégories de vues par rapport au domaine qu'ils traitent - par exemple "affaires", "technique", "services", etc. dans un contexte particulier, pour un objectif ou mission particulière (Tableau 11). Fondamentalement, les cadres de modélisation d'architecture répondent au problème de communication que rencontrent tous les acteurs humains qui souhaitent concourir à un objectif commun : un « Système A Faire ». Ainsi, il est avantageux de baser une modélisation sur un cadre d'architecture afin de circonscrire la complexité du système à modéliser. En effet, suivant les parties prenantes, le « découpage » en de multiples vues réduit cette complexité puisque chaque vue (Figure 22) correspond à un modèle particulier d'un système.

²⁶ <http://www.software.org>

Tableau 11. Synthèse des cadres de modélisation et langages de modélisation

	Références	Année	Cadre de modélisation
Ministères de la défense & organisations de défense	(TOGAF 1995)	1995	TOGAF, promu par l'Open Group Architecture Framework
	(Architecture_Working_Group 1997)	1997	C4ISR, (Computerized Command, Control, Communications Intelligence Surveillance Reconnaissance) promu par le département de la défense Américaine
	(Office Of Force Transformation Washington 2005)	2001	Net-Centric Operations and Warfare, directive et implémentation de l'architecture réseau-centré au département de la défense Américaine
	(AGATE 2002)	2002	AGATE, Atelier de Gestion de l'Architecture Technique des Systèmes d'Information de Commandement, promu par la Délégation Générale pour l'Armement en France
	(MODAF 2007)	2007	MoDAF, promu par le ministère de la défense Britannique
	(DoD 2001 (As Amended Through 16 October 2006))	200x	DoDAF, promu par le département de la défense Américaine
	(Doumeings 1984)	1984	Méthode GRAI
Entreprise	(Zachman 1987; Sowa and Zachman 1992)	1987	Cadre de modélisation de Zachman
	(CEN/ENV-40003 1990)	1990	CEN ENV 40 003 : computer integrated manufacturing - CIM systems architecture framework for modelling
	(Spewak and Hill 1992)	1992	EAP, Enterprise Architecture Planning
	(Williams 1992)	1992	PERA (Purdue Enterprise Reference Architecture)
	(Scheer 1992)	1992	ARIS - Business Process Modelling
	(AMICE 1993)	1993	CIMOSA (Computer Integrated Manufacturing Open System Architecture).
	(Zanettin 1994)	1994	GIM (GRAI Integrated Modelling)
	(Doumeings et al. 1996)	1996	GIM: GRAI Integrated Methodology For Product Development
	(CEN/ENV-12204 1995)	1996	CEN ENV 12 204 : advanced manufacturing technology - systems architecture - constructs for enterprise modelling
	(GERAM 1999)	1997	GERAM, Generalised Enterprise Reference Architecture and Methodology (Version 1.6.3)
	ANSI/IEEE 1471-2000 (ISO IEC DIS 25961)	2000	Recommandation pour la description de l'architecture de systèmes Logiciels complexes
	(ISO-15704 2000)	2000	Industrial automation systems — Requirements for enterprise-reference architectures and methodologies
	(ISO/CEN-19439 2002)	2002	ISO/CEN 19439 : Enterprise Integration – Framework for enterprise Modelling
	(ISO/CEN-19440 2002)	2002	ISO/CEN 19440 : Language Constructs for Enterprise Modelling
	(Panetto et al. 2004)	2004	UEML : Unified Enterprise Modelling Language
	Administration Territoriale	(TISAF 1997)	1997
(FEAF 2006)		1999 2006	FEAF, Federal Architecture Framework, Version 1.1, promu par le gouvernement fédéral, avec laquelle DoDAF a des dépendances
(TEAF 2000)		2000	TEAF, The Treasury Enterprise Architecture Framework

La Figure 23 ci-dessous montre la diversité des cadres d'architecture et des langages de modélisation d'entreprise développés depuis plusieurs décennies par influences successives (Bernus et al. 2003). Cette diversité tient non seulement de la multiplicité des domaines d'application de ces cadres, mais aussi de la sensibilité et des besoins de leurs auteurs. Notre problématique s'inscrit dans le domaine de la production de biens et de services avec une sensibilité orientée vers les systèmes d'information et un besoin d'ingénierie de ces systèmes. Nous nous sommes intéressés à 3 cadres répondant au moins partiellement à notre problématique (C4ISR (C4ISR 1998), TOGAF (TOGAF 2000) et Zachman (Zachman 1987))

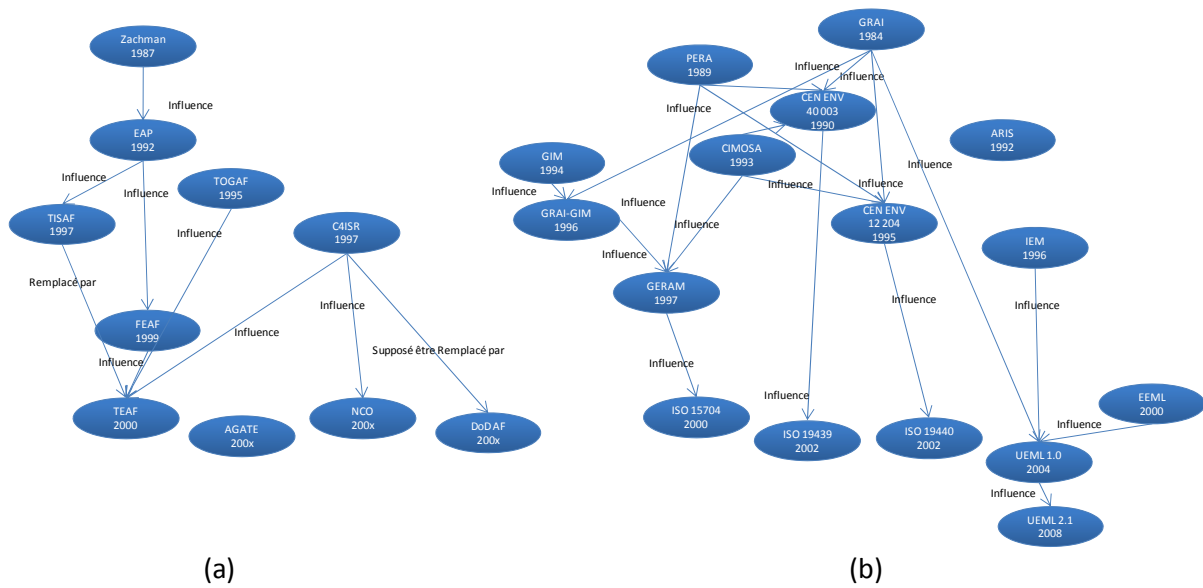


Figure 23. Influences et historiques des principaux cadres de modélisation (a) et langages de modélisation d'entreprise (b)

1.1.1 C4ISR

Le **C4ISR**²⁷ (Architecture_Working_Group 1997) a été proposé par un groupe de travail en architecture relatif à la commande, au contrôle, à la communication, aux ordinateurs, à la veille, à la surveillance et à la reconnaissance. Ce groupe de travail s'est intéressé notamment à l'interopérabilité des systèmes d'information au travers de la proposition d'un modèle de maturité : **LISI**²⁸. Aux 5 niveaux d'interopérabilité du LISI, sont opposés 4 types de caractéristiques : **Procédures**, **Applications**, **Infrastructure**, **Données**. On retrouve ainsi les 5 niveaux suivants :

- Niveau 0 - Systèmes isolés : pas de connexion existante. Les systèmes doivent échanger des données ou des services, mais ne peuvent pas interagir directement.
- Niveau 1 - Systèmes connectés : échanges homogènes possibles. Les systèmes sont connectés électroniquement par des liens de type peer-to-peer. Cela peut concerner simplement les échanges de fichiers entre des systèmes. Les types d'échanges sont typiquement hétérogènes suivant leur contexte (ex. texte, bitmap, gif, tiff, etc.)
- Niveau 2 - Systèmes distribués : échanges hétérogènes possibles de fichiers. Cela concerne des annotations d'images, de liens hypertexte, de multimédia,...Les systèmes sont connectés à de multiples systèmes dans un réseau local. L'intérêt de ce niveau est d'accéder à des systèmes ou des applications afin d'échanger des données par le Web.
- Niveau 3 - Systèmes intégrés : partages d'applications et de données. Ce niveau est caractérisé par de multiples interactions entre applications. Les systèmes et les applications sont interconnectés, mais opèrent sur une fonction unique et généralement sur un seul ensemble de données fonctionnelles. Généralement, les implémentations à ce niveau ont uniquement une vue « localisée » de l'espace de l'information distribuée et du domaine opérationnel ou fonctionnel traversé.
- Niveau 4 - Systèmes universels : Systèmes partagés au niveau de l'entreprise. Le niveau est l'objectif ultime de recherche d'interopérabilité des systèmes d'information à travers des activités fonctionnelles et des domaines d'information (intelligence, logistique, etc.). À ce niveau de l'entreprise, les informations sont partagées globalement via une architecture distribuée de l'information. Les applications et les systèmes fonctionnent, le cas échéant à travers des domaines fonctionnels de données. L'espace de travail « virtuel » utilise les applications d'exploitation par rapport à un espace intégré d'informations partagées. Ce niveau représente les fonctions nécessaires pour atteindre les concepts proposés dans les documents « vision joint 2010 » du DoD.

Ce cadre est particulièrement intéressant lorsqu'il s'agit de qualifier le niveau de maturité d'un SI, mais il ne fournit pas de guide ou de méthode permettant de le spécifier à des niveaux d'échelles

²⁷ **C4ISR** : (Command, Control, Communications, computers, Intelligence, Surveillance and Reconnaissance)

²⁸ **LISI** : Levels of Information Systems Interoperability

différents et sur des niveaux d'observations différents (MOA, MOE de rang 1 & 2) à travers le temps afin d'en capitaliser ses modèles.

1.1.2 TOGAF

Le principal objectif de **TOGAF**²⁹ (TOGAF 2000) est de fournir un cadre pour à la fois la conception, l'évaluation et le développement d'architectures d'entreprise. Un élément clé de TOGAF est **ADM**³⁰ qui spécifie le processus de développement d'une architecture d'entreprise. ADM est une méthode générique itérative en 7 phases :

- Phase A : l'« architecture vision » inclue les hauts niveaux de description de l'architecture du système en cours (« as-is ») et du système à faire (« to-be ») à partir des perspectives du « business » et techniques.
- Phase B : la « Business Architecture » décrit les fondements de l'architecture « business » et de l'analyse des écarts entre ces fondements et l'objectif visé.
- Phase C : l'« Information System Architecture » décrit l'objectif relatif aux données et aux applications par une analyse des exigences des données et des applications.
- Phase D : la « Technologie Architecture » est utilisée pour développer l'architecture qui doit être réalisée. Cette phase se décompose en 8 sous-étapes qui sont la création des modèles d'architecture, la sélection des services, la confirmation des objectifs du « Business », la détermination de critères, la définition de l'architecture et le suivi d'analyse des écarts.
- Phase E : l'« Opportunities and Solutions » réalise l'évaluation et le choix des options d'implémentation.
- Phase F : la « Migration Planning » est relative à la planification des projets d'implémentation.
- Phase G : l'« Implementation Governance » est relative au pilotage global de l'implémentation et de déploiement.
- Phase H : l'« Architecture Change Management » est relative au suivi continu et à la veille technologique en fonction des contraintes environnementales du « Business ».

TOGAF recommande d'utiliser des vues comme la vue d'architecture business, la vue architecture des données, la vue ingénierie système, la vue sécurité d'entreprise, la vue management d'entreprise, la vue qualité de service et la vue mobilité d'entreprise.

Ce cadre de modélisation fournit un guide à la conduite du changement d'un SI (du « as-is » au « to-be ») très intéressant, notamment en termes de description de processus par lesquels transitent et sont traitées des informations. On peut noter aussi, que ce cadre est relativement prescriptif au sujet des processus qui devraient être employés pour recueillir et maintenir l'information d'organisation. D'autre

²⁹ **TOGAF** : The Open Group Architecture Framework

³⁰ **ADM** : (Architecture Development Method)

part on peut noter que TOGAF ne donne pas de méthode ou de guide pour décrire le « comment faire », le « pourquoi le faire » et « sur quoi agir pour faire », l'architecture d'un SI.

1.1.3 Le cadre de Modélisation de Zachman

Zachman (Sowa and Zachman 1992) propose un cadre de modélisation structuré dont le but est de rendre le plus exhaustif possible l'étude des points de vue des modélisateurs sur le système à étudier.

« Le cadre pour l'architecture des entreprises introduit en 1987 par John Zachman propose une structure logique pour classer et organiser les représentations de l'entreprise qui sont d'intérêt aussi bien pour sa direction que pour le développement de ses systèmes. Il trouve son origine dans des structures analogues que l'on trouve dans des disciplines plus anciennes telles que l'architecture, la construction, l'ingénierie ou l'industrie et qui classent et organisent les éléments créés durant le processus de conception et de fabrication de produits techniques complexes (ex : bâtiment, aéronautique) » (MEGA software).

Le cadre de modélisation de Zachman est une matrice à deux dimensions avec 6 colonnes et 6 lignes. Chaque cellule représente une vue partielle d'un système complexe ou non. Les lignes définissent le niveau de modélisation et donc le **point de vue du modélisateur** : contextuel (vue du planificateur), conceptuel (vue du propriétaire), logique (vue du concepteur), physique (vue du réalisateur), formel (vue des sous-traitants), d'exécution. Les colonnes représentent le **point de vue de l'observation du modélisateur** : quoi, comment, où, qui, quand, pourquoi. L'union des cellules doit couvrir l'ensemble des points de vue d'un système. Les grands principes qui guident l'application du cadre Zachman comprennent :

- Un système « complet » pouvant être modélisé en répondant à des questions simples (perspectives) comme : pourquoi, qui, quoi, comment, où et quand ?
- Les six perspectives (pourquoi, qui, quoi, comment, où, quand) capturent tous les modèles critiques exigés pour l'analyse, la spécification et le développement d'un système.
- Les contraintes pour chaque perspective sont cumulatives. Celles d'une rangée inférieure sont ajoutées à celles des lignes supérieures pour fournir un nombre croissant de restrictions.
- Les colonnes représentent différentes abstractions dans un effort de réduire la complexité de n'importe quel modèle simple qui est établi.
- Les colonnes n'ont pas d'ordre.
- Les modèles dans chaque colonne sont uniques.
- Chaque ligne représente une perspective unique.
- Chaque cellule est unique en son genre.
- La logique est **récursive**.

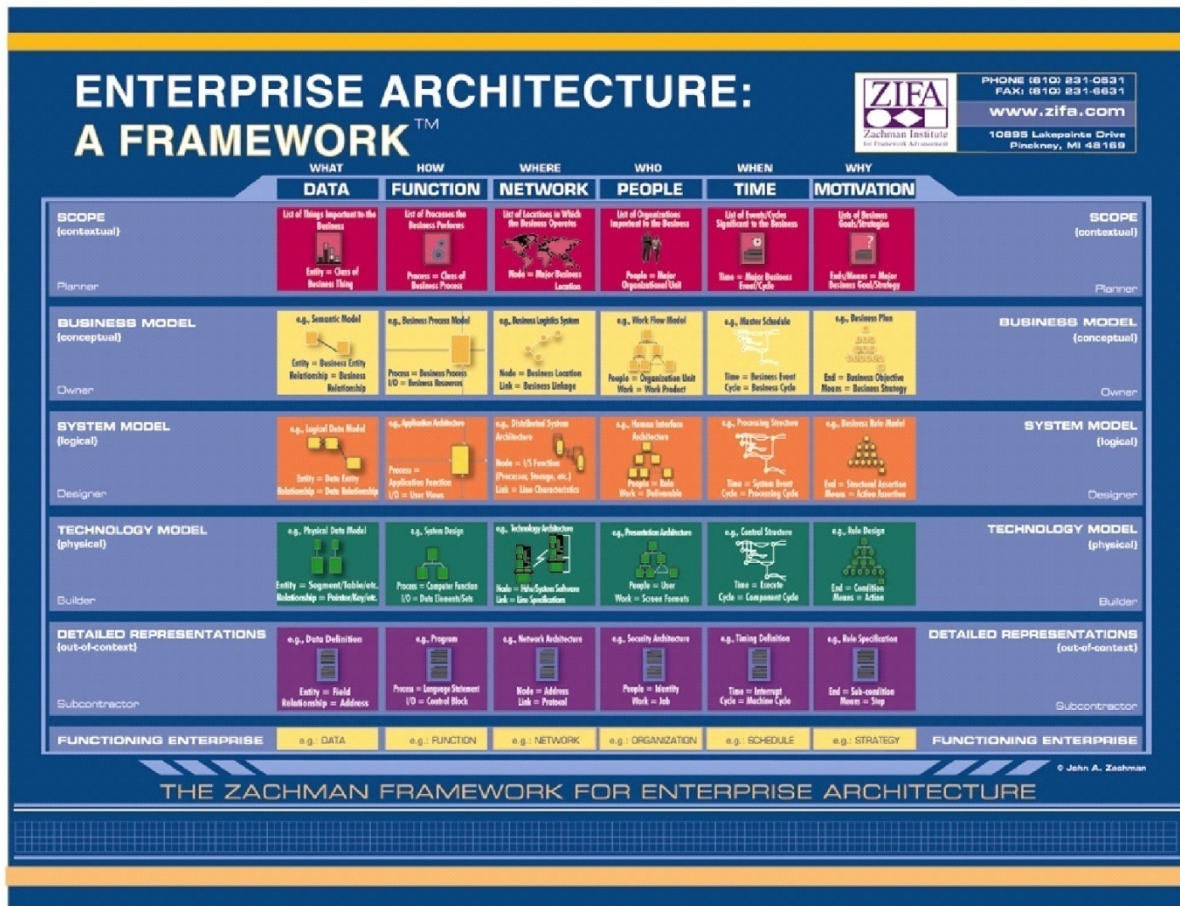


Figure 24. Cadre de Zachman (Sowa and Zachman 1992).

Un des avantages majeur du cadre de modélisation de Zachman est d'offrir au modélisateur une vision complète du système complexe qu'il doit modéliser. Ainsi, ce cadre fournit un guide (ou filtre) avec des croisements de points de vue simples à interpréter (la matrice 6X6) et permet d'avoir une vision complète rapidement accessible des objets de modélisation d'un SAF.

Nous avons noté que TOGAF apporte une dimension importante en termes de gestion temporelle dans la conduite au changement d'un SI, mais qu'il ne fournit pas tous les points de vue nécessaires à l'ingénierie d'un SAF (comment, pourquoi, quoi). De son côté C4ISR n'aborde pas ces aspects temporels de gestion d'un projet et donc ne participe pas à la capitalisation d'un système à faire.

Nous retiendrons pour ce mémoire le **cadre de modélisation de Zachman** parce qu'il :

- est adapté aux Systèmes d'Information,
- propose un guide simple et relativement générique,
- laisse une grande liberté d'instanciation à son utilisateur dans son action de modélisation.
- permet la spécification d'un système à des niveaux d'observations différents,
- permet la spécification d'un système de manière récursive.

1.5 Cadre de modélisation pour une Ingénierie basée sur les modèles d'un SdSI

Une des particularités d'un SdSI est sa constitution en tout point « gigogne ». Cette assemblage concerne des SI distribués et hétérogènes dont l'ingénierie est réalisée par des personnes différentes sur des lieux éventuellement différents. Ainsi, nous avons une ingénierie propre au SdSI avec un point de vue macroscopique du SAF et un ensemble d'ingénieries propres à chaque SI constitutif. Compte tenu des aspects récursifs des systèmes constitutifs, il convient conjointement de rendre récursives leurs ingénieries au travers d'une instance particulière du cadre de modélisation de Zachman. Ainsi, cette approche récursive d'ingénierie peut contribuer à réduire la complexité de la modélisation de grands systèmes complexes sur des échelles différentes et selon des niveaux d'observation différents avec des points de vue différents.

Nous postulons que le cadre de modélisation de Zachman s'apparente à une sorte de « pattern » (Chapitre 1.2.2) où chaque objet de modélisation est en lien avec un autre au travers de relations et de règles de modélisation. Ce sont ces relations et ces règles de modélisation qui garantissent la cohérence entre les modèles distribués et capitalisés au sein de chaque cadre de modélisation de Zachman mais aussi entre chacun des cadres de modélisation de Zachman propres aux ingénieries particulières et distribuées des SI constitutifs.

Avec ce mécanisme de récursivité de l'ingénierie basée sur des modèles cohérents et capitalisés au sein de multiples Zachman associés à chaque SAF et à chaque SI constitutif, la MOA et la MOE vont pouvoir partager et capitaliser leurs modèles à travers le temps. Aussi, nous proposons que les modèles soient représentés dans un langage unifié comme UML afin de conserver leur sémantique. Ainsi, ces cadres de modélisation de Zachman viennent en appui à notre ISBM afin de capitaliser l'ensemble des modèles relatifs à l'ingénierie des interopérations de ces systèmes.

2 Proposition d'une Ingénierie Système Basée sur les Modèles

Nous nous basons sur l'approche de l'Ingénierie Système proposant les relations et les rôles entre le Système À Faire et le Système Pour Faire qui lui est associé (Figure 25).

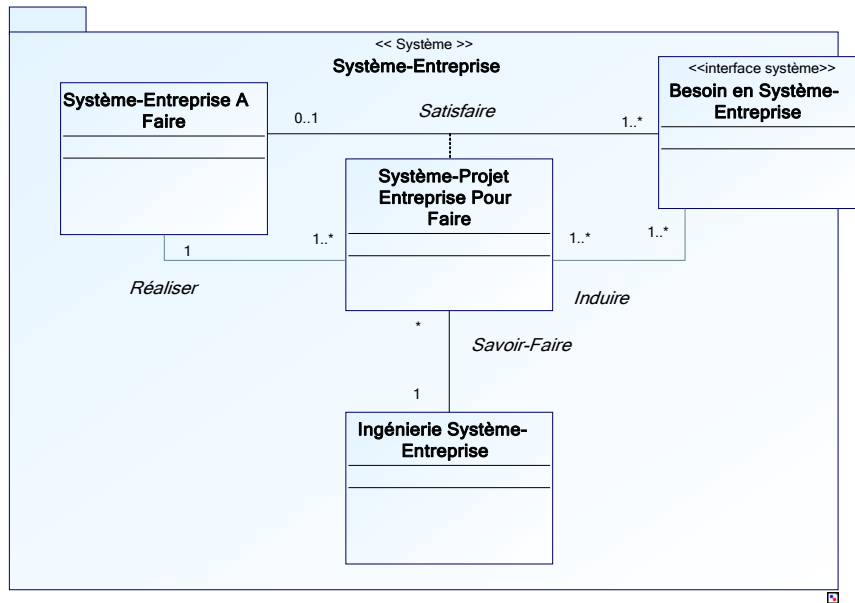


Figure 25. Vision Ingénierie Système en MOA de la modélisation en Entreprise, adapté de [AFIS 2008]

Du fait de la récursivité en tout point « gigogne » du Système Pour Faire (SPF) qui est lui-même un système (au même titre qu'un Système À Faire (SAF)), il a les mêmes propriétés qu'un SAF : la récursivité des ingénieries, et la conception en tout point « gigogne ».

Afin de déterminer les invariants de modélisation du SPF, nous utilisons le cadre de modélisation de Zachman comme un pattern de modélisation, en ne retenant pour notre SPF que les objets de ce cadre de modélisation qui sont relatifs à l'ingénierie (Tableau 12).

Tableau 12. Correspondance entre un système au sens de Kuras, un SAF et un SPF

(Kuras 2006)	SAF	SPF
s	Système à faire	Système pour faire
$\{pattern_k\} \subseteq \{pattern_{k'}\}$	$\{Système_k\} \subseteq \{Système_{k'}\}$	$\{Zachman_k\} \subseteq \{Zachman_{k'}\}$
échelles (de μ à ν) représentées par : s^u invariants dans le temps	SdS, site, atelier, cellule de production, etc.	Zachman du SdS, du site, de l'atelier, de la cellule de production,...
à chaque échelle un ensemble d'objets $\{E_i\}$ et un ensemble de relations $\{REL_j\}$	SdS {sites} ; Site {ateliers} ; Atelier {cellules de production} ; etc.	Objets de modélisation, relations entre les objets de modélisation
H^v	A un SAF correspond un Zachman	A un Zachman correspond un SAF
P_m	Agrégation de propriétés d'un SdS, d'un site, d'un atelier, d'une cellule de production, etc.	Agrégation de propriété des objets de modélisations
REL_n	Relations entre les éléments constitutifs SdS, site, atelier, cellule de production,...	Relation entre les objets de modélisations

2.1 Système pour Faire de l'Ingénierie de notre ISBM

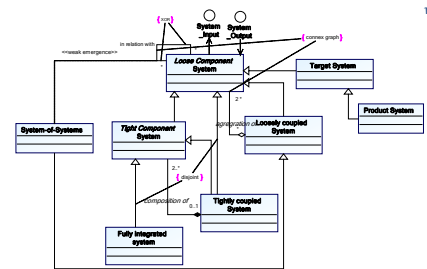
Notre méthodologie a comme objectifs d'assurer la cohérence de la modélisation d'un assemblage de systèmes d'information appartenant à un réseau réticulaire de Systèmes-Entreprises, et de faciliter les interopérations entre leurs applications au gré des collaborations conjoncturelles.

Cette cohérence est assurée par les relations qui lient les éléments de modélisation créés par la MOA dont l'objectif est de réaliser un Système à Faire que l'on qualifiera de SAF-Système-entreprise. De part l'assemblage récursif des éléments constitutifs de ce SAF-global, les différentes MOE participant à la réalisation de ce SAF- Système-entreprise vont hériter des éléments de modélisation et des modèles de la MOA pour la réalisation de leur propre SAF-Sn (Figure 26). Ces héritages et cette cohérence des modèles au travers du cadre de modélisation de Zachman est ce que nous appelons l'Ingénierie Système de notre Système pour Faire : Ingénierie Système Basée sur les Modèles.

Ainsi, en nous basant sur la proposition de Kuras (Kuras 2006), l'ingénierie d'un SdSI peut être conceptualisé par un pattern instancié $\{instanciation\ de\ notre\ méta - modèle\ d'ingénierie\ de\ SdS_k\}$ pour une résolution ou un facteur d'échelle donné v , et dont la conceptualisation des points de vue de l'ingénierie d'un SdSI est réalisée par un pattern composite (Équation 4).

Aussi, nous proposons dans ce mémoire de définir l'instanciation d'un système (le Système à Faire, le Système pour Faire, l'Ingénierie Système-Entreprise, le besoin en Système-Entreprise) en nous basant sur l'équation proposée dans le chapitre 1 selon (Équation 4):

$$SdS^v \equiv \{(SàF; SpF; Ing.; Besoin); REL_{IS_j}\}_{I^v, J^v};$$



Équation 4

Ainsi le pattern de SdS^v de que nous proposons (Figure 26) tient compte des relations entre le 'besoin en un Système-Entreprise' et le 'Système-Entreprise A Faire'. De l'association de ces deux systèmes va émerger un 'Système-Projet Pour Faire' dont le rôle est de faciliter la réalisation du 'Système-Entreprise A Faire'. Les savoir-faire seront alors directement associés à une ou plusieurs 'Ingénierie Système-Entreprise'. Cette ingénierie va ainsi être le 'pilote' des ingénieries particulières de chacun des COTS (voir le Chapitre 1.1) au travers de leur 'Ingénierie-Système COTS'. Pour ingénieriser ces COTS nous avons choisi d'être guidé par le cadre de modélisation de Zachman, ainsi représenté dans notre proposition de méta-modèle d'ISBM (Figure 26) par les concepts 'Ingénierie Système-Entreprise' et 'Ingénierie Système-COTS'. Cette ingénierie va, récursivement, avoir son propre 'Système-Projet COTS Pour Faire'. Ce système sera issu de l'émergence d'un 'Besoin en Système-COTS' (pour contribuer à la réalisation du 'Besoin en Système-Entreprise') et du 'Système-COTS A Faire'. De plus, cette 'Ingénierie-Système COTS' ne correspond pas à 'l'ingénierie-COTS' déployée

par son éditeur qui a été nécessaire à la conception du COTS. Par contre, pour que les COTS puissent communiquer entres-eux, il est nécessaire de relier (d'interfacier) chacune des 'ingénieries-COTS' à leur 'Ingénierie Système-COTS' associée pour appliquer les méthodes particulières de l'éditeur.

En nous référant au Meta-Modèle que nous proposons dans le chapitre 1, décrivant un Système de Systèmes (Figure 18), et en nous référant à l'Ingénierie Basée sur les Modèles du point de vue l'Ingénierie Système (Figure 19), l'application de l'équation 3 du chapitre 1 nous permet de proposer une instantiation particulière d'un modèle sémantique d'une ISBM d'un réseau réticulaire de Système-Entreprise (Figure 26).

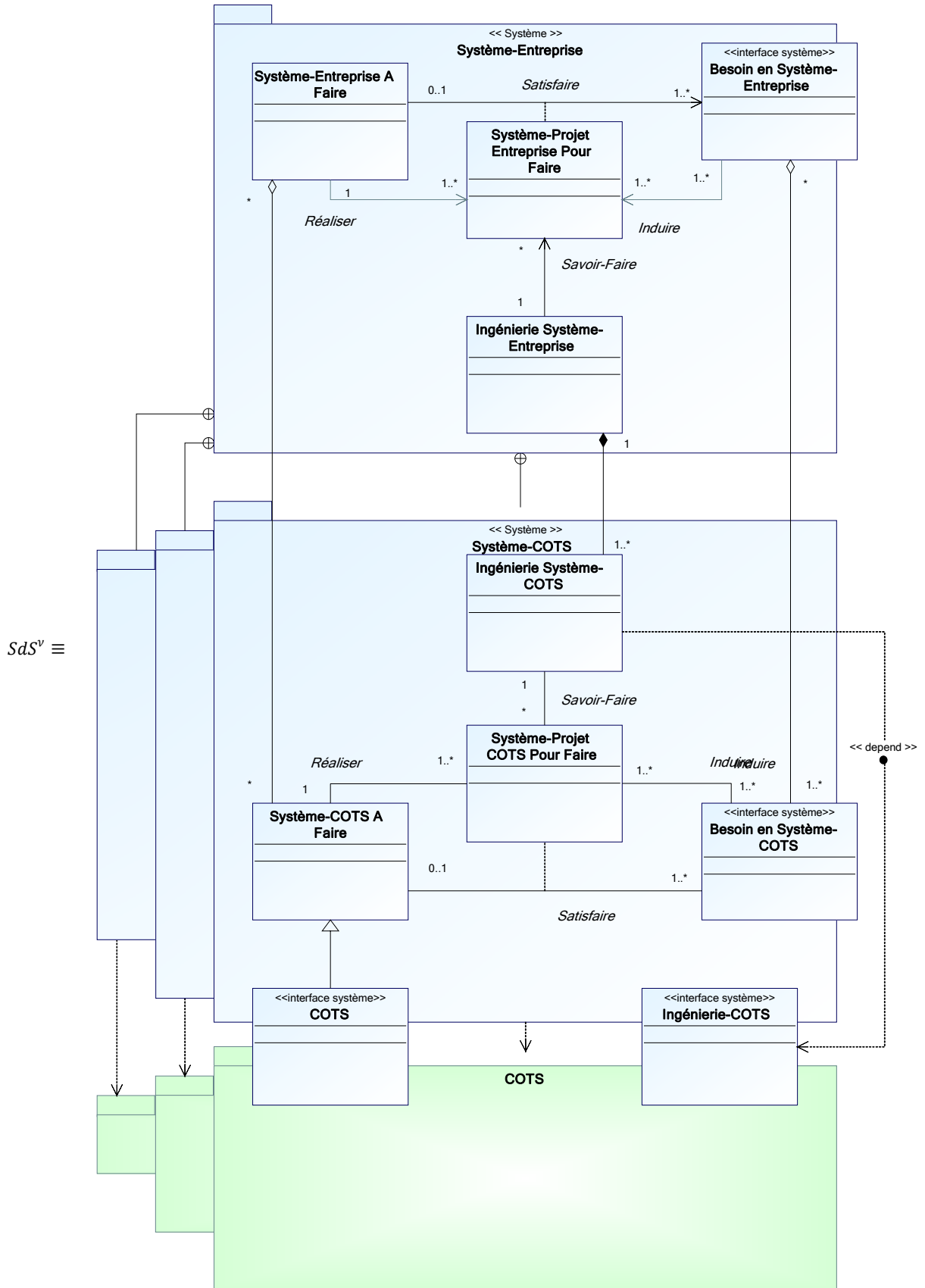
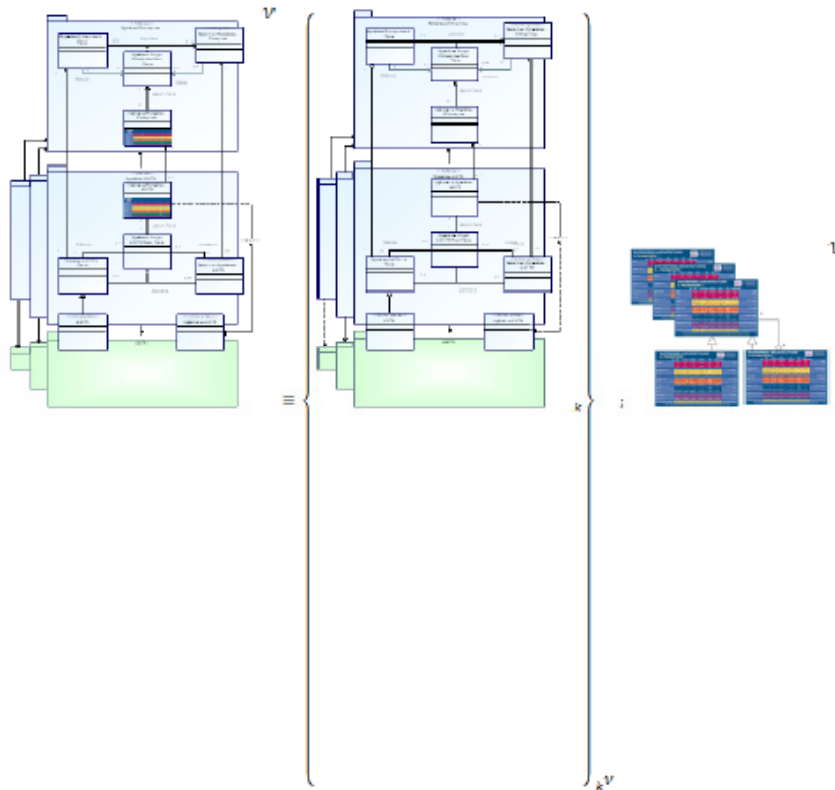


Figure 26. Modèle sémantique de l'ingénierie Système Basée sur Les Modèles d'un réseau réticulaire de Système-entreprise.

Parce que notre ISBM se base sur la récursivité du cadre de modélisation de Zachman, nous proposons dans ce mémoire de définir le modèle sémantique d'un système de systèmes d'information (SAF & SPF) multi-échelles à partir de l'équation 5 selon (Kuras 2006) :



Équation 5

Mais avant de mettre en œuvre d'une telle ISBM, la MOA doit se poser la question sur la réelle utilité d'une telle démarche pour son SAF. Notre ISBM est-elle adaptée à la problématique que doit résoudre la MOA ? Ainsi la MOA doit se poser la question suivante : pourquoi modélise-t-on ?

Si le Système à Faire justifie la mise en œuvre d'une ISBM, la MOA doit définir le contexte dans lequel doit se dérouler la modélisation afin d'assurer son pilotage, à savoir :

- Qui modélise ? → Attribution d'un rôle pour chacun des acteurs du système projet (notre Système pour Faire) : MOA, MOE 1 & 2, etc.
- Où modéliser ? → Attribution à chacun des acteurs d'un niveau d'observation dans le cadre de modélisation de Zachman pour sa propre modélisation : niveaux « contextual – conceptual – logical – physical - technical ».
- Quoi modéliser ? → Définition de quels types de modélisations chaque acteur peut employer à chaque perspective d'observation du cadre de modélisation de Zachman (« what – how – where – who – when – why ») : UML, BPMN, etc.

- Quand modéliser ? → Organisation temporelle du système projet (notre Système pour Faire) avec des jalons pour ponctuer les moments clés du projet.
- Comment modéliser ? → Description détaillée des enchaînements de modélisation pour chaque acteur du système projet.

2.1.1 Pourquoi modéliser ?

Comme nous désirons conceptualiser des systèmes multi-échelles (Bjelkemyr and Lindberg 2007b), nous avons choisi le cadre de Zachman pour sa relative complétude et simplicité d'usage et la notion de récursivité pour sa contribution à la réduction de la complexité de conceptualisation d'un grand système. En effet, comme nous l'avons vu précédemment, une seule matrice de Zachman n'est pas suffisante pour modéliser un système complexe de type SdS. Plus un système est une succession d'assemblages, plus la récursivité du cadre de modélisation devient une nécessité. On retrouve de la sorte une certaine forme de parallélisme entre la décomposition³¹ récursive d'un système (le Système à Faire) et celle de sa modélisation, de son ingénierie (le Système pour Faire). Ainsi, sur la base du pattern composite, les cadres de modélisation récursifs assurent la cohérence effective entre les modèles des systèmes dans le contexte de SdS. En nous appuyant sur ces principes de bases, un des objectifs majeurs de cette thèse est de réaliser un prototypage d'outillage méthodologique de modélisation récursif et multi-échelles (Figure 27).

D'un point de vue général, les entreprises n'utilisent pas de cadres de modélisation et en particulier pas de cadre de modélisation de Zachman. Bien souvent la modélisation s'effectue au coup par coup par des personnes en interne ou des consultants extérieurs avec pour chacun des formalismes différents et hétérogènes. Comme il n'existe pas de base commune de modélisation sur un formalisme consensuel, ni même de base commune des objets de modélisation, quand une modélisation existe, l'interopérabilité entre les modèles de l'entreprise et ceux du « Systèmes A Faire » est inexistante ou inefficace.

Voilà pourquoi nous proposons de définir une méthode de modélisation des niveaux « Contextual » et « Enterprise Model » du cadre de modélisation de Zachman afin d'aider l'ingénieur système dans son acte de modélisation pour qu'il puisse faire interopérer son SI constitutif au sein d'un SdSI bénéficiaire respectant les invariants de modélisation imposés par la MOA et la MOE de rang 1.

De la sorte, sur la base des invariants de modélisation, à chaque COTS dont on a la maîtrise des intrants, des extrants ainsi que ses fonctions internes, on associe un cadre de modélisation de Zachman pour capitaliser son ingénierie.

³¹ Nous voulons dire par là un « désassemblage » ou « désagrégation » de systèmes.



Figure 27. Des modèles et cadres génériques vers une hiérarchisation de modélisation réursive. (Morel et al. 2007a)

Ainsi, la finalité de l'Ingénierie Système (le système projet) que nous proposons est d'obtenir la structure composite et d'instancier des modèles génériques (y compris les cadres génériques), pour un projet spécifique. Les modèles particuliers à chaque nouveau système sont, une fois instanciés, récursivement inclus dans un nouveau cadre de Zachman. Nos travaux se sont alors focalisés sur l'ingénierie réursive d'un SAF basée sur un ensemble de cadres de modélisation de Zachman récursivement imbriqués, de sorte que cette imbrication ne doive jamais perdre de vue la cohérence des modèles instanciés dans chacun des cadres récursifs particuliers de modélisation.

Le point fort de l'approche réursive de la modélisation basée sur un cadre comme Zachman est de proposer un changement d'échelle à chaque succession de points de vue qui vont de la vue globale du SAF bénéficiaire jusqu'à l'ensemble des vues particulières propres à chaque système constitutif. Chaque ingénierie particulière se voit alors guidée par une ingénierie de SdS (Figure 28) qui est garante de la cohérence des modèles du SAF et des modèles des systèmes constitutifs. Cette ingénierie de SdS portée par un cadre de modélisation de Zachman particulier sera le « portail » d'entrée des ingénieries particulières, c'est-à-dire des cadre de modélisation de Zachman particuliers de l'ensemble des systèmes constitutifs.

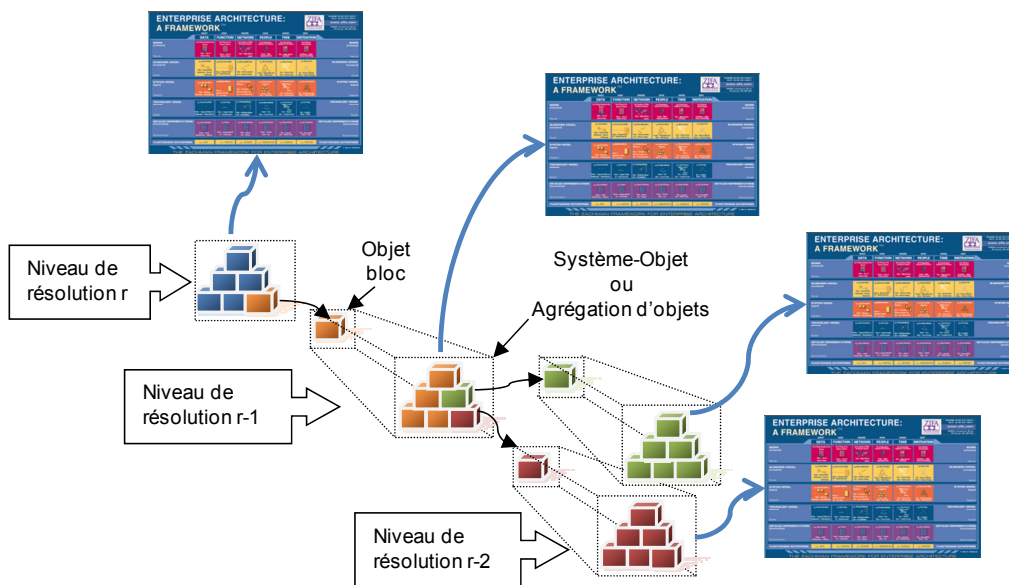


Figure 28. Modélisation récursive multi-échelle de systèmes.

2.1.2 Qui modélise ?

L'Ingénierie Système de ce SAF met en activité 4 types d'acteurs principaux :

- Le **donneur d'ordres** : il est l'initiateur du projet du SAF. Il va fixer les objectifs de hauts niveaux du SAF et va définir l'Appel d'Offre. C'est lui qui choisit le Maître d'Ouvrage suite à l'Appel d'Offre parmi ceux qui y ont répondu. C'est enfin lui qui réceptionne le SAF une fois que celui-ci est terminé en temps et en heures.
- La **maîtrise d'ouvrage - MOA** : elle est l'architecte principal du SAF. Elle est en lien direct avec le donneur d'ordres en tant que candidat à un appel d'offre dans un premier temps puis, en second temps il devient le fournisseur direct du SAF suite à une RAO³². C'est elle qui détermine les objectifs et qui pilote le projet, en lien direct avec la maîtrise d'œuvre de rang 1. Elle est ainsi l'élément charnière entre le donneur d'ordres et la maîtrise d'œuvre de rang 1. Elle est le garant de l'interopérabilité sémantique entre les concepts des différents systèmes constitutifs qu'elle doit faire coopérer.
- La **maîtrise d'œuvre de rang 1 - MOE de rang 1** : La MOE de rang 1 déploie une ingénierie propre au paramétrage et à la configuration des interfaces entre les SI du site dont elle a la charge. Elle doit respecter les contraintes et les exigences définies par la MOA dont hérite son système. Elle peut ajouter des besoins et des exigences propres à son système constitutif mais ils ne devront jamais être en opposition ou avoir une influence négative sur les besoins et les exigences de la MOA. Elle est le spécialiste métier du système qui doit être déployé et interopérer avec d'autres systèmes. Elle participe à la définition des correspondances (mapping) sémantiques des concepts de son SI et ceux du SAF que pilote la MOA. Elle fournit les solutions techniques répondant aux exigences du SAF et des SI du site. Elle est le garant de l'interopérabilité technique entre son système et tous ceux avec qui il doit coopérer.
- La **maîtrise d'œuvre de rang 2 - MOE de rang 2** : Elle est le spécialiste métier du système qui doit être déployé et interopérer avec d'autres systèmes. Elle devra fournir les solutions techniques répondant aux exigences du SAF et de son propre système. Elle est le garant de l'interopérabilité technique entre son système et tous ceux avec qui il doit coopérer.

³² **RAO** : Réponse à l'Appel d'Offre

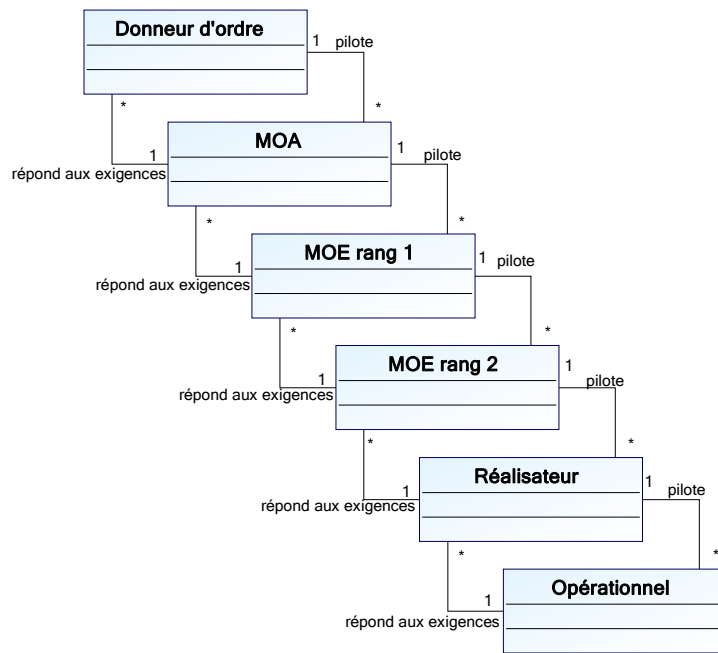


Figure 29. Relations entre les acteurs du système projet (SPF).

2.1.3 Où modéliser ?

	What	How	Where	Who	When	Why
Contextual (MOA)						
Conceptual (MOE-1)						
Logical (MOE-2)						
Technical (réalisateur)						
Operational (Opérationnel)						

Figure 30. Zachman + MOA + MOE rang 1 + MOE rang 2

Chaque modélisateur instancie une zone particulière du cadre de modélisation. On a ainsi pour chaque niveau du Zachman :

- La **MOA** qui traduit le SAF au niveau contextuel (niveau « Contextual ») suivant les besoins du donneur d'ordre,
- la **MOE de rang 1** qui décrit le SAF au niveau conceptuel (niveau « Conceptual») en respectant les objectifs et les exigences définis par la MOA,
- la **MOE de rang 2** qui formalise le SAF aux niveaux logiques (niveaux « Logical »),
- Le **réalisateur** qui produit le SAF au niveau physique (niveau « Technical ») en suivant les directives du MOE de rang 2,
- l'**opérationnel** qui fournit les éléments minimaux nécessaires au réalisateur en fournissant le code par exemple (niveau « Operational »).

Dans le cadre de cette thèse nous nous focalisons sur les invariants de modélisation qui sont portés par les deux premiers niveaux du cadre de modélisation de Zachman, c'est-à-dire les niveaux de la MOA et de la MOE de rang 1.

2.1.4 Quoi modéliser ?

L'IS, de par sa nature systémique, se limite à des correspondances sémantiques entre modèles, difficiles à faire. Ce qui justifie l'usage du cadre de modélisation de Zachman qui ne fait pas de correspondances formelles entre modèles (cellules). Comme nous désirons modéliser un SAF dans un même formalisme afin de garantir une cohérence sémantique de modélisation et d'interprétation avec l'ensemble des acteurs travaillant sur un même projet, nous nous sommes orientés vers le langage unifié de modélisation **UML**³³ adaptée à la modélisation de système. UML permet la spécification, l'analyse, la conception, la vérification et la validation de systèmes. Nous avons complété notre modélisation par un **modèle des exigences** qui nous semble être le point de départ de toutes modélisations d'un système.

Les modèles que nous proposons d'utiliser pour les invariants de modélisation dans notre ingénierie sont (Figure 32) :

- Au niveau « Contextual » : le modèle des exigences, le modèle de séquence, le modèle de paquetages, le modèle de processus, le structure organisationnelle (sites), l'organigramme fonctionnel.

³³ C'est en 1997 que l'OMG³³ adopta l'Unified Modeling Language (UML)[®] et le Meta-Object Facility (MOF[™]). La base de MOF a été rendue officielle en Août 2004. Le MOF définit un modèle comme une instance d'un méta modèle. Un méta modèle permet de décrire des propriétés d'une plate-forme particulière Mellor, S.J., Scott, K., Uhl, A. and Weise, D. (2002). "Model-Driven Architecture." in *Proceedings of the 8th Workshops on Advances in Object-Oriented Information Systems*, 2 September 2002, (Montpellier , France), ISBN 3-540-44088-7, pp 290-297..

- Au niveau « Concept » : le modèle de séquence, le modèle d'activité, le modèle de cas d'utilisation et le modèle de classe, l'organigramme des acteurs.

Le modèle des exigences représente le diagramme fondateur de tous les modèles qui vont suivre. En effet, l'ingénierie des exigences (McConnell 1996; Wiegers 2003) est le point de départ de toute IS, puisqu'elle spécifie un SAF qu'un donneur d'ordres a commandé. Un système, non limité par des **objectifs** ou **finalités** est indéfinissable (il est indescriptible, avec un nombre de caractéristiques infini). Ce postulat impose le caractère obligatoire de l'association d'un système à une **finalité**, et/ou son **appartenance** à un groupe de systèmes pour partager une finalité commune (Le Gallou and Bouchon-Meunier 1992). Aussi le SAF doit répondre à un ensemble d'objectifs (et donc d'exigences). Le recueil des exigences est un point essentiel avant toutes démarches de modélisation. Nous avons choisi d'aborder les exigences d'un point de vue de l'« analyse orientée objet » puisque nous formalisons les spécifications sous la forme de modèles avec des objets de modélisation uniques (Figure 31).

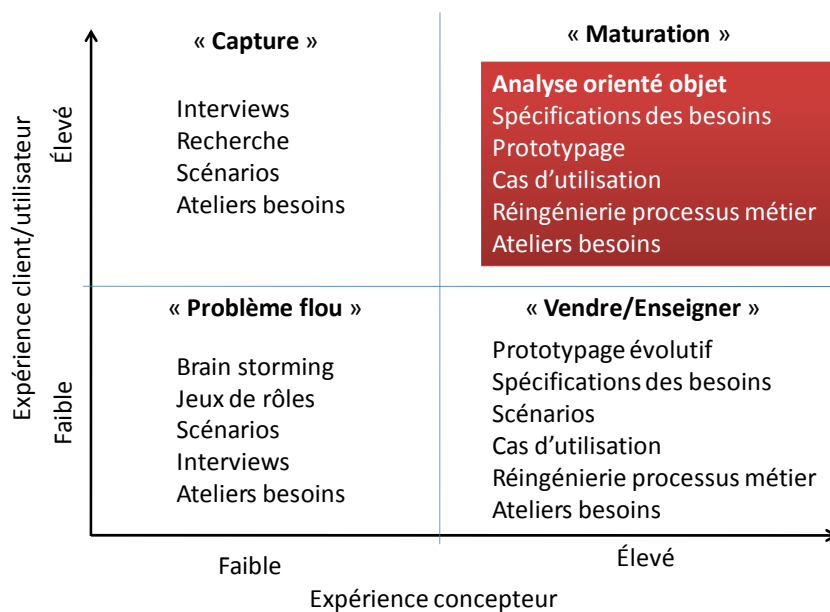


Figure 31. Types de recueil des exigences inspiré de ADN - R3 Management (Risk-Requirement-Regulatory Management)

Ce sont donc bien les exigences d'un SAF bénéficiaire qui conduisent à faire émerger le ou les besoins de systèmes qui devront contribuer à sa réalisation. La découverte des systèmes constitutifs est une tâche qui incombe à la MOE de rang 1 dans sa phase de spécification au niveau « Conceptual » (ou « Concept »). Ainsi, concrètement, chaque nouveau système utile à cette contribution est représenté par un paquetage dans un diagramme de paquetage du SAF. En s'appuyant sur ces deux diagrammes, la MOA et la MOE de rang 1 modélisent les autres diagrammes qui participent aux invariants de modélisation.

	What	How	Where	Who	When	Why
Contextual (MOA)	Diagramme de paquetages	Diagramme de processus	Diagramme de sites	Organigramme fonctionnel	Diagramme de séquence	Diagramme des exigences
Conceptual (MOE-1)	Diagramme de classes	Diagramme d'activités		Organigramme des acteurs	Diagramme de séquence	Diagramme des exigences
				+ cas d'utilisations		
Logical (MOE-2)	Diagramme de classes	Diagramme d'archi. applicative				
Technical (réalisateur)	Diagramme relationnel		Diagramme d'archi. technique			
Operational (Opérationnel)		Fichiers plats, XML, XSLT,...				

Figure 32. Types de formalisme par cellule du cadre de modélisation de Zachman

2.1.5 Quand modéliser ?

Nous proposons que notre SPF soit le support d'une **modélisation diachronique** garantissant le respect des exigences du point de vue global, au niveau d'un SdSI, jusqu'au niveau local, dans chaque SI ou COTS constitutifs. Les exigences imposées par le donneur d'ordres « traduites » par la MOA sont ainsi respectées au travers de **l'invariant de la modélisation**³⁴. Comme notre ingénierie est basée sur le cadre de modélisation de Zachman, cet invariant correspond aux spécifications de la maîtrise d'ouvrage (**niveau CONTEXTUAL du cadre de modélisation de Zachman**) et de la maîtrise d'œuvre de rang 1 (**niveau Enterprise Model du cadre de modélisation de Zachman**).

Notre ingénierie doit s'insérer dans une logique de gestion de projet (aspect diachronique) que l'on pourrait appeler la gestion du SPF. Comme nous l'avons présenté (encadré bleu -Figure 21 page 57), la frontière de notre ingénierie se situe entre la phase de définition et la phase d'intégration du système. On retrouve alors 3 étapes clés :

³⁴ Nous entendons par « invariants de modélisation » tout objet de modélisation définit « a minima » par la MOA et qui sera hérité par les MOE pour qu'ils puissent formaliser le même système mais suivant leur propre niveaux d'observation et d'abstraction.

- La préparation du projet par la MOA : J0 à J1
- La définition du SAF par la MOE de rang 1 : J1 à J2
- La spécification du SAF par la MOE de rang 2 : J2 à J3

Un projet commence généralement par un Appel d'Offre. Les MOA potentielles répondent à l'Appel d'Offre suite à des échanges itératifs avec le donneur d'ordres. Le but de ces échanges est de s'assurer de la bonne compréhension des objectifs et des exigences du donneur d'ordres. Cette phase est essentielle et incontournable. La fin de l'itération indique que la MOA estime être en accord avec les objectifs et les exigences du donneur d'ordres. Une fois cette phase passée, la MOA fournit une définition du SAF que le donneur d'ordres à commandé. La RAO représente la définition du SAF que la MOA s'engage à fournir si elle est retenue par le donneur d'ordres. Nous proposons que l'étape de définition soit bâtie autour d'une ISBM de premier niveau. Si la MOA est retenue pour le projet, les définitions qu'elle a présentées deviennent contractuelles (temps J1- Figure 21 page 57). Elles deviennent par conséquent notre **invariant de modélisation**. La MOA peut alors engager une étude plus approfondie du SAF pour fournir une spécification détaillée, en engageant autant de MOE de rang 1 nécessaires qui apportent leurs compétences métiers (temps J2- Figure 21 page 57). Ainsi la MOA devient le pilote des multiples projets entre les temps J2 et J3 (Figure 21 page 57) que doit réaliser chacun des MOE de rang 1. La phase d'intégration sera pilotée par la MOA et organisée autour de l'ensemble des systèmes constitutifs fournis par les MOE de rang 1. La phase de vérification peut alors être mise en œuvre afin de contrôler l'adéquation entre les exigences de départ du donneur d'ordres traduites par la MOA et le SAF développé par l'ensemble des MOE de rang 1. Une fois le système du projet réalisé par la MOE de rang 2, la phase de validation vient ensuite pour clore le projet. Nous la nommerons la réception du SAF. Notre ISBM n'inclut pas ces deux dernières phases.

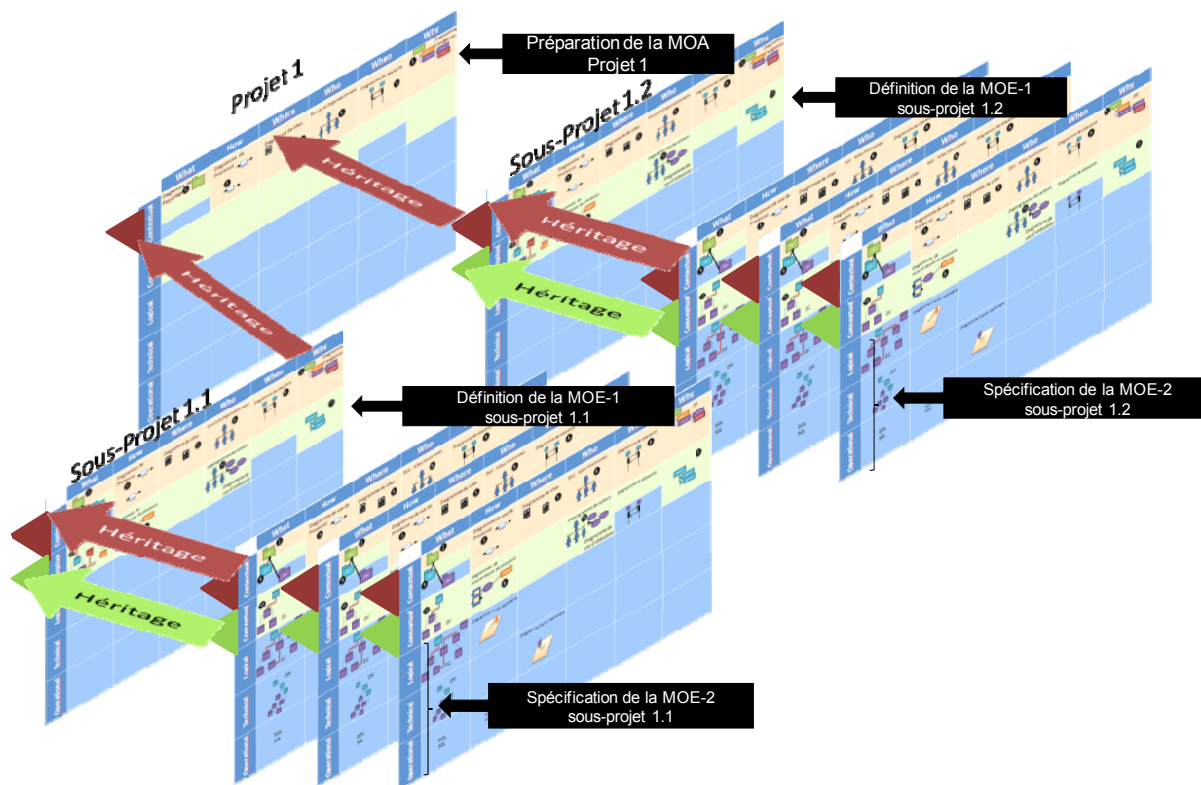


Figure 33 Phases de l'ingénierie récursive basée par les modèles en relations avec la gestion du projet à faire.

2.1.6 Comment modéliser ?

Comme les travaux de recherche de cette thèse se basent sur le paradigme SCP (Morel et al. 2003 ; Panetto 2006), le produit est l'élément central des interopérations **synchroniques** et **diachroniques** entre les SI. Par conséquent, le produit devient le vecteur des objectifs et des exigences de la MOA et de la MOE de rang 1, au travers des invariants de modélisation rattachés à des cellules particulières des niveaux « Contextual » et « Enterprise Model » du cadre de Zachman.

Ainsi, on retrouve 3 parties de cadre de modélisation de Zachman (§ 2.1.3 - Figure 30) : La partie « **Contextual** » de la MOA, la partie « **Conceptual** » de la MOE de rang 1 et la partie « **Logical** » + « **Technical** » + « **Operational** » des MOE de rang 2. Chacune des parties est instanciée à des instants différents (Figure 34) :

- Entre J1 et J2 : la MOA et la MOE de rang 1 avec leurs modèles invariants,
- Entre J2 et J3 : l'ensemble des MOE de rang 2 sur chaque site avec leurs modèles particuliers.

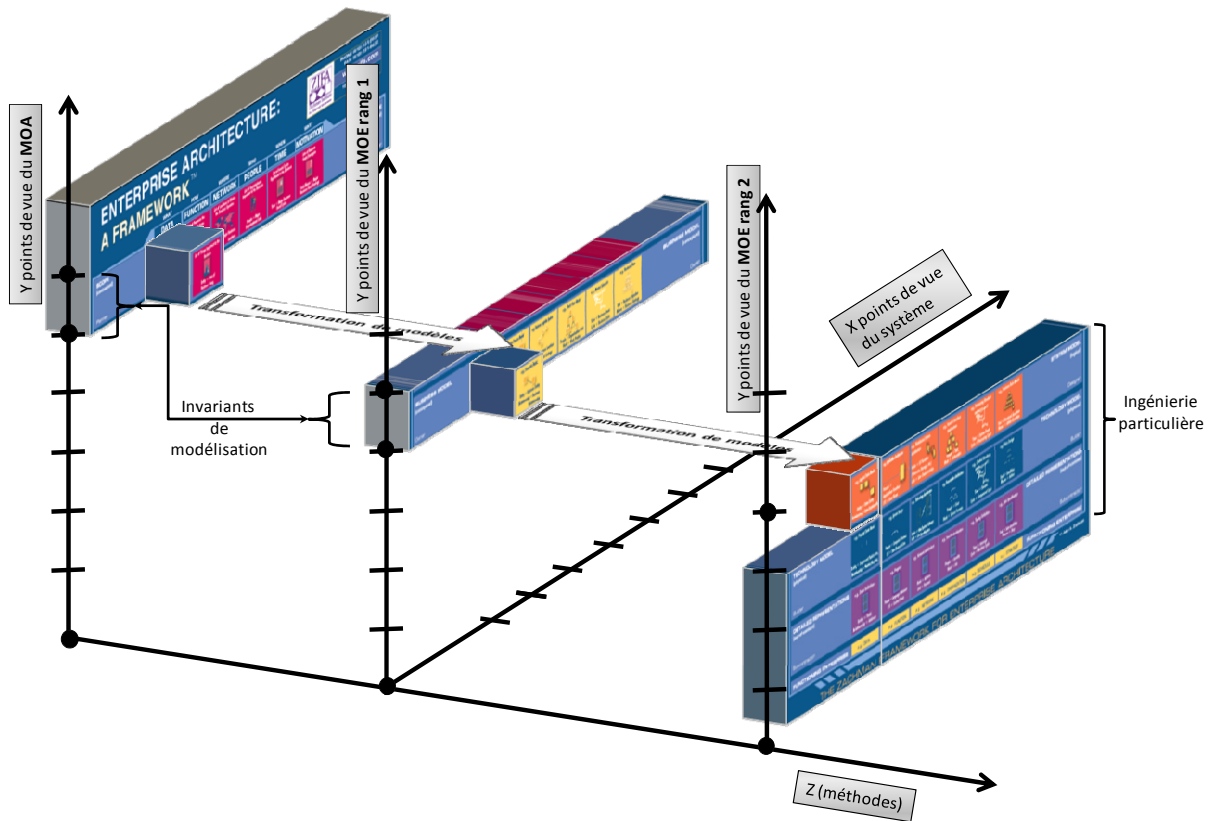


Figure 34. Processus d'ISBM dans le Zachman

L'objectif de notre méthodologie est de mettre à disposition des modélisateurs (MOE de rang 2) l'ensemble des objets de modélisations et modèles invariants sur lesquels ils vont se baser pour modéliser à leur tour leur SI. Par exemple, si dans la cellule « Contextual-What » la MOA propose un diagramme de paquetage avec le Paquetage du SAF (Figure 35), la MOE de rang 1 va récupérer ce diagramme puis ajouter l'ensemble des paquetages des systèmes constitutifs liés (Figure 36). Les invariants de cette cellule sont ainsi créés. La MOE de rang 2 va récupérer à son tour ces invariants, mais elle va garder seulement les paquetages qui la concernent (Figure 37).

	What	
Contextual	Diagramme de Paquetage 2	Diagramme de Processus

Figure 35. Spécification de la MOA dans le cadre de modélisation de Zachman

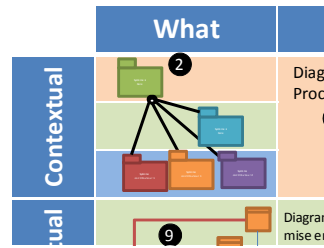


Figure 36. Spécification de la MOE de rang 1 dans le cadre de modélisation de Zachman

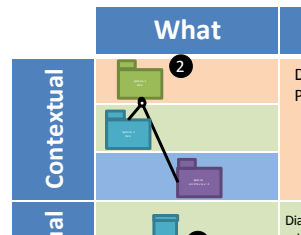


Figure 37. Spécification de la MOE de rang 2 dans le cadre de modélisation de Zachman

Parce que nous nous appuyons sur le paradigme SCP, nous considérons que le **produit** (au sens du Produit-Holon (Bajic and Chaxel 1997; Morel et al. 2003) doit être le **fil conducteur de la modélisation des invariants**. En effet, nous nous focalisons sur les informations intrinsèques au produit dont les applications de gestion du SI ont besoin pour qu'il puisse être réalisé. Ainsi, et au même titre qu'un système constitutif, le produit possède son propre paquetage rattaché au paquetage du SAF. Ce paquetage est le porteur des informations intrinsèques au produit (Tursi et al. 2007), il est donc extrêmement important car il pose la base sémantique des concepts qui le concernent. Les travaux de thèse d'Angela Tursi (Tursi et al. 2009). sont principalement orientés sur la définition de l'ontologie du produit. Cette ontologie pose les briques de base des concepts fondamentaux d'un produit manufacturier. Comme ces travaux sont en cours de réalisation, nous nous baserons dans nos travaux sur une ontologie incomplète formalisée dans un diagramme de classe. Aussi, à chaque interaction avec un SI, la MOE de rang 2 n'utilisera que les classes de l'ontologie du produit qui le concerne. Il pourra ainsi établir une relation sémantique entre ces concepts et les concepts formalisés dans le modèle de son propre SI (Figure 38).

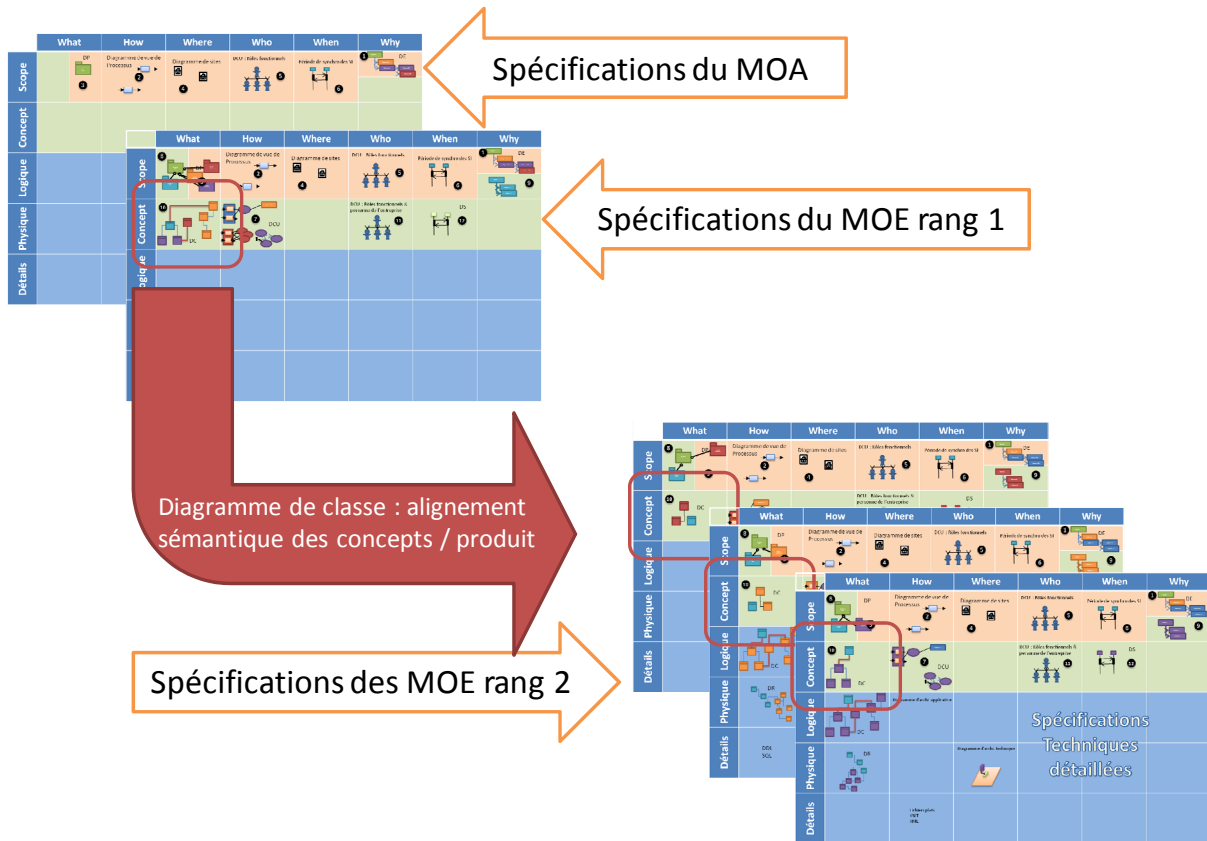


Figure 38. Alignement sémantique des concepts des SI par rapport aux concepts du produit

2.2 ISBM d'un Système à Faire : instanciation d'un cadre de modélisation de Zachman

Il est admis que des démarches méthodologiques cohérentes doivent être développées pour rendre opérationnels des langages de modélisation de type UML, qui visent à proposer des primitives couvrant de manière générale les divers aspects d'une modélisation systémique. Notre proposition contribue en ce sens à outiller méthodologiquement une ingénierie d'un SI d'Entreprise (qui peut être de type SdSI) en définissant, pour chaque cellule, du cadre de modélisation de Zachman, les objets de modélisation qui nous semble les plus pertinents du point de vue de l'IS (Tableau 13). L'implémentation de cette méthodologie consiste en un enchaînement de requêtes sur la base (Meta-Modèle) de l'outil 'MEGA Modelling Suite'. Par exemple, pour la cellule SCOPE-WHY la requête qui définit les objets de modélisation que nous recommandons est de la forme suivante : pour le 'SAF_SdSI_Tracer_le_produit' dans la cellule SCOPE-WHY du cadre de modélisation de Zachman on préconise la création d'un 'diagramme des Exigences' → `Select [Diagramme] Where [Cellule Zachman de rattachement] ="Contextual-Why" And [DiagramType].[Nature] ="RQMNT" And [Paquetage décrit].[Nom court] = "SAF_SdSI_Tracer_le_produit".`

Tableau 13. Proposition d'objets de modélisation pertinents du point de vue de l'IS et rattachés au cadre de modélisation de Zachman

Quoi	Comment	Où	Qui	Quand	Pourquoi
SAF et SI	Diagrammes et colonne du Zachman	Niveau du Zachman	Modélisateur	Jalon	Objectifs
Le SAF :	<ul style="list-style-type: none"> • (Why), diagramme des exigences • (What), diagramme de paquetage • (How), diagramme de processus • (When), diagramme de séquence • (Where), diagramme de site • (Who), organigramme fonctionnel 	« Contextual »	La MOA : direction de l'AIPL	Entre J1 et J2	Spécifications des objectifs et exigences globales du SAF
Les SI constitutifs + produit :	<ul style="list-style-type: none"> • (Why), diagramme des exigences • (What), diagramme de classe • (How), diagramme de cas d'utilisation, diagramme de mise en œuvre de processus • (When), diagramme de séquence • (Where), diagramme de site • (Who), organigramme opérationnel 	« Enterprise Model »	La MOE de rang 1 : 1 ingénieur d'étude + 1 doctorant	Entre J1 et J2	Spécifications des objectifs et exigences des SI au sein du SdSI
Chaque SI constitutif	<ul style="list-style-type: none"> • (Why), diagramme des exigences • (What), diagramme de paquetage • (How), diagramme de processus • (When), diagramme de séquence • (Where), diagramme de site • (Who), organigramme fonctionnel 	« Contextual », par héritage	La MOE de rang 2 : 1 ingénieur de recherche + 3 ingénieurs d'étude	Entre J2 et J3	Spécifications techniques détaillées du SI
	<ul style="list-style-type: none"> • (When), diagramme de séquence • (How), diagramme de cas d'utilisation, diagramme de mise en œuvre d'un processus 	« Model » par héritage			
	<ul style="list-style-type: none"> • (What), diagramme de classe • (How), diagramme d'arch. applicative 	« Logical »			
	<ul style="list-style-type: none"> • (What), diagramme relationnel • (Where), diagramme d'arch. Technique 	« Technical »			
	<ul style="list-style-type: none"> • (What), DDL, SQL • (How), fichiers plats, XML, XSLT,... 	« Operational »			
	<ul style="list-style-type: none"> • (How), fichiers plats, XML, XSLT,... 	« Operational »			
	<ul style="list-style-type: none"> • (How), fichiers plats, XML, XSLT,... 				
	<ul style="list-style-type: none"> • (How), fichiers plats, XML, XSLT,... 				

2.2.1 Préparation du projet « Système A Faire » par la MOA

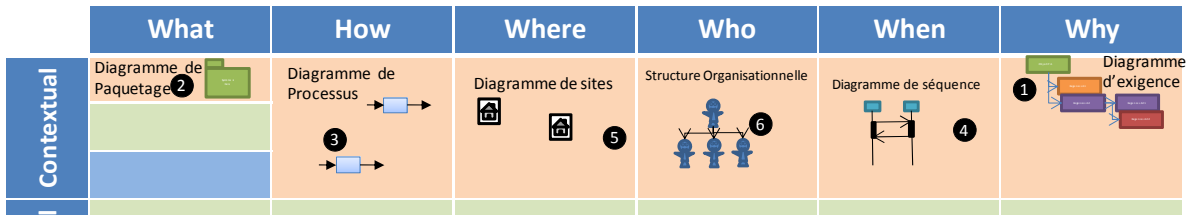


Figure 39 Zachman du SAF - spécifications de la MOA.

Tableau 14 Séquencement de la modélisation des spécifications du SAF par la MOA

Modélisation	Modèles utilisés
1 Définition des objectifs et exigences en fonction des besoins exprimés par la MOA	Diagramme des exigences
2 Définition du Système à Faire	Diagramme de paquetage
3 Définition des domaines fonctionnels et de leurs processus génériques	Diagramme de processus
4 Définition des exigences de synchronisation des SI (horizon long et moyen terme)	Diagramme de séquence
5 Description du contexte géographique des sites	Diagramme de sites
6 Définition des rôles fonctionnels et organisationnels du SdSI	Organigramme structurel

2.2.2 Définition du « Système A Faire » par la MOE de rang 1

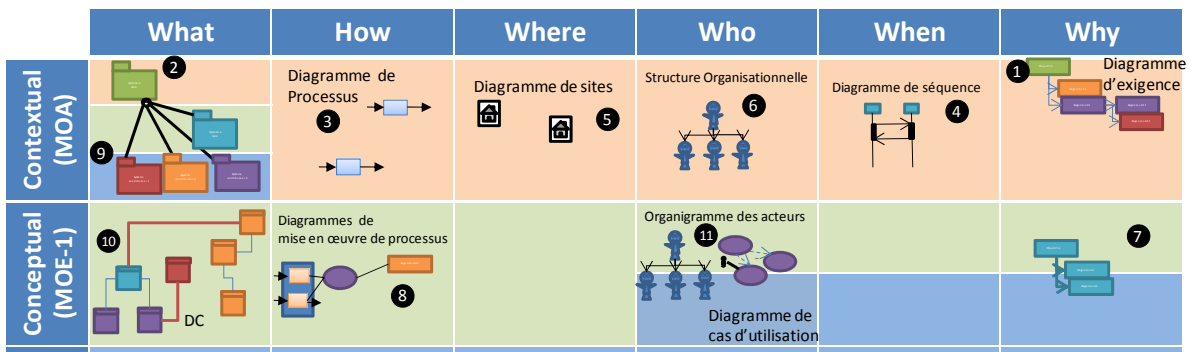


Figure 40 Zachman du SAF - spécifications de la MOE de rang 1.

Tableau 15 Séquencement de la modélisation des spécifications du SAF par la MOE de rang 1

Modélisation	Modèles utilisés
7 Définition des objectifs et des exigences en fonction des besoins exprimés par la MOE de rang 1	Diagramme des exigences
8 Modélisation des procédures et opérations métiers avec rattachement de cas d'utilisation et association aux exigences prédéfinies	Diagramme de procédures
9 Modélisation des SI constitutif au SAF	Diagramme de paquetage
10 Définition des rôles opérationnels propres au site	Organigramme + cas d'utilisation
11 Formalisation sémantique des concepts des SI	Diagramme de classes

2.2.3 Spécifications détaillées du SAF par la MOE de rang 2

Les spécifications détaillées de chaque SI sont propres aux ingénieurs responsables des SI et spécialistes métiers : les MOE de rang 2. Ils doivent prendre en compte les invariants de modélisation fournis par la MOA et la MOE de Rang 1. Ainsi la MOE de rang 1 devient en quelque sorte une MOA de rang 2 qui pilotera autant de sous-projets qu'il existe de Systèmes d'Entreprise.

	What	How	Where	Who	When	Why
Contextual (MOA)	2, 9	Diagramme de vue de Processus 3	Diagramme de sites 5	DCU : Rôles fonctionnels 6	Diagramme de séquence 4	DE 1
Conceptual (MOE-1)	10, DC	Diagrammes de mise en œuvre de processus 8		Organigramme des acteurs 11 Diagramme de cas d'utilisation	Diagramme de séquence	7
Logical (MOE-2)	DC	Diagramme d'archi. applicative				
Technical (réalisateur)	DR		Diagramme d'archi. technique			
Operational (Opérationnel)	DDL SQL	Fichiers plats XSLT XML				

Figure 41 Zachman du système constitutif au SAF – exemple de spécifications de la MOE de rang 2.

3 Proposition d'un prototype d'outillage pour notre ISBM

3.1 Atelier logiciel pour la capitalisation et la cohérence de la modélisation

Il existe un très grand nombre de progiciels de modélisation (Schekkerman 2007) qui pourraient répondre à nos besoins, mais notre choix s'est porté sur le progiciel MEGA Modelling Suite³⁵. L'expertise du laboratoire sur cet environnement a facilité l'accès à son méta-modèle et donc de le « modéliser » en fonction de nouvelles exigences de recherche. De plus, MEGA propose une réelle cohérence des modèles, à savoir que comme chaque objet de modélisation est unique, toute modification d'un objet dans un modèle se répercute dans l'ensemble des modèles qui l'utilisent.

3.2 Prototype d'Outillage de modélisation

Afin de supporter notre méthodologie de modélisation, nous avons prototypé un outillage, sur la base d'une adaptation de l'environnement de modélisation MEGA Modelling Suite. En effet, cet environnement fournit un ensemble d'outils de modélisation structuré autour d'un méta-modèle commun paramétrable et une base de modèles unifiée assurant une cohérence inter-modèles (Figure 42).

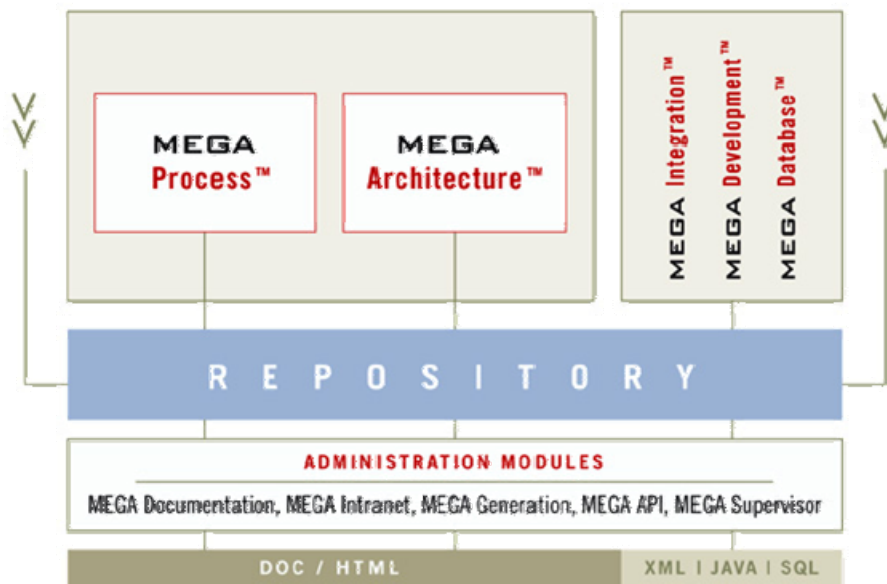


Figure 42. Architecture de MEGA Modelling Suite (Source : Mega International)

MEGA Modelling Suite offre aussi un portail de représentation de l'architecture des modèles d'un projet au travers du cadre de modélisation de Zachman. Notre problématique de modélisation récursive, basée sur le cadre de modélisation de Zachman, nous a amenés à étendre le méta-modèle

³⁵ <http://www.mega.com/>

unifié de l'environnement MEGA Modelling Suite afin de formaliser le concept de « point de vue » relatif au cadre de modélisation de Zachman (cf. Chapitre 4.1.1.3), ainsi que la spécification relative d'une typologie des systèmes (bénéficiaires et/ou constitutifs) au travers de leur syntaxe abstraite représentée sous la forme de paquets (Figure 43). Ainsi, un système constitutif participe à la réalisation de la mission d'un SdSI, système bénéficiaire.

Nous avons ainsi créé une méta-association « Système bénéficiaire – Système constitutif » qui formalise la sémantique liant des systèmes constitutifs au sein d'un SdS bénéficiaire. Pour un système donné, représenté par son paquetage, est associée une instance de cadre de modélisation de Zachman au travers d'un ensemble de cellules spécifiant un point de vue de ce système. Ce point de vue est caractérisé par son niveau d'abstraction (**Contextual, Conceptual, Logical, Technical, Operational**) et sa perspective d'observation (**What, How, Where, Who, When, Why**). La syntaxe abstraite de ce concept est formalisée par la méta-classe « Point DeVue » associée à un paquetage.

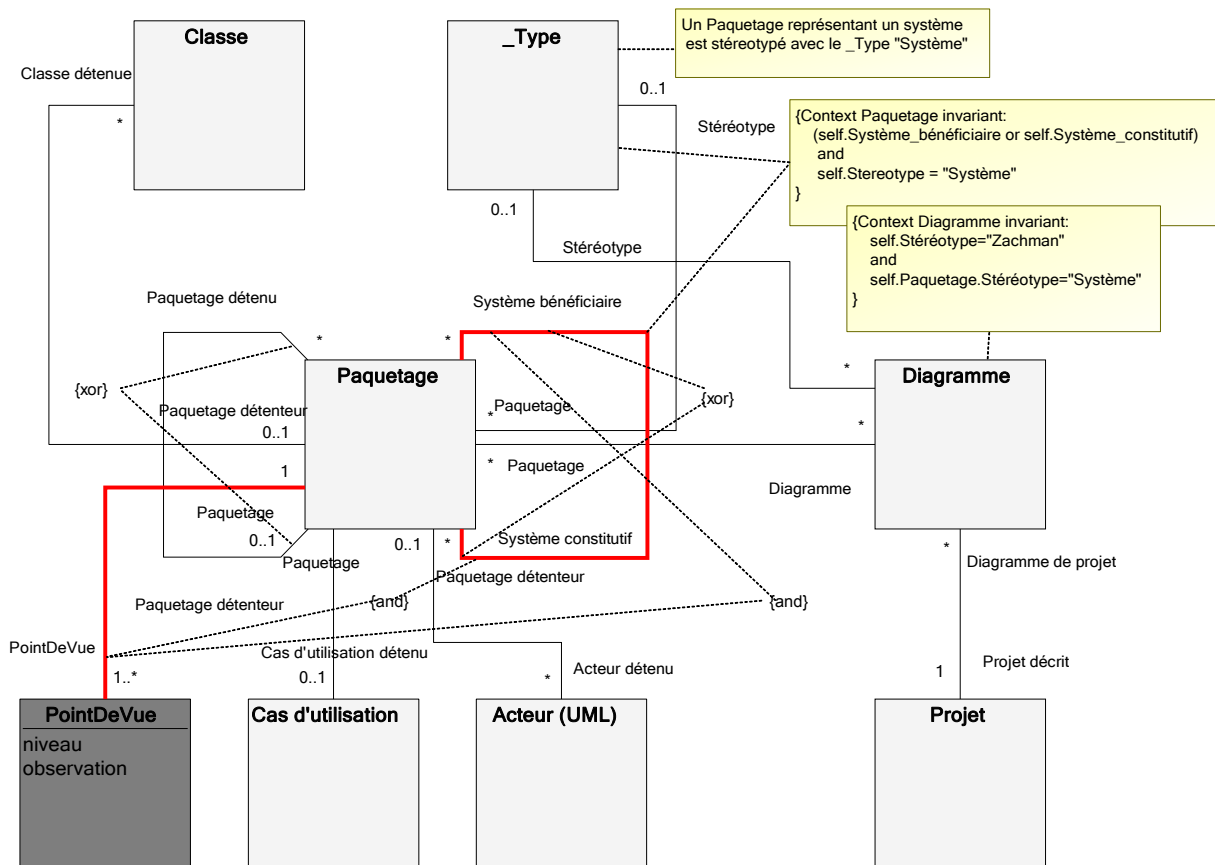
Les modifications du méta-modèle de MEGA (cf. annexe 7) concernent alors (Tableau 16) :

1. La définition du concept de PointDeVue : son niveau d'observation et son niveau d'abstraction.
2. La formalisation de 2 stéréotypes basés respectivement sur la méta-classe du concept de « paquetage » et celle du concept de « diagramme ».

Tableau 16. Modification du méta-modèle de MEGA Modelling Suite

Nom	Meta-Classe	Propriétés	Type	Multiplicité	Contraintes
« Système »	Paquetage	<i>PointDeVue</i>	PointDeVue	1..*	Paquetage.Système_bénéficiaire or Paquetage.Système_constitutif Paquetage.Système_bénéficiaire.typSystème = 'bénéficiaire'
		<i>typeSystème</i>	EnumTypeS	1	Paquetage.Système_constitutif.typSystème = 'constitutif'
« Zachman »	Diagramme				Diagramme.Paquetage.Stéréotype = « système »

EnumTypeS = {bénéficiaire, constitutif}



niveau ∈ {Contextual, Conceptual, Logical, Technical, Operational}

observation ∈ {What, How, Where, Who, When, Why}

Figure 43. Extrait du Méta-modèle unifié de MEGA et nos extensions

Sur la base de l'extension du méta-modèle de l'environnement MEGA Modelling Suite, nous avons prototypé un outillage de modélisation au sein de cet environnement, afin de faciliter la mise en œuvre de notre proposition méthodologique. Ce prototype est intégré comme un portail web attaché à l'environnement MEGA et accède ainsi à l'ensemble des modèles stockés dans le référentiel commun (MEGA Repository). La représentation graphique de l'arborescence des systèmes bénéficiaires et constitutifs est générée à l'aide de l'application issue du monde du libre Graphviz³⁶. La figure ci-dessous (Figure 44) définit l'architecture applicative de notre prototype.

³⁶ Graphviz : <http://www.graphviz.org/>

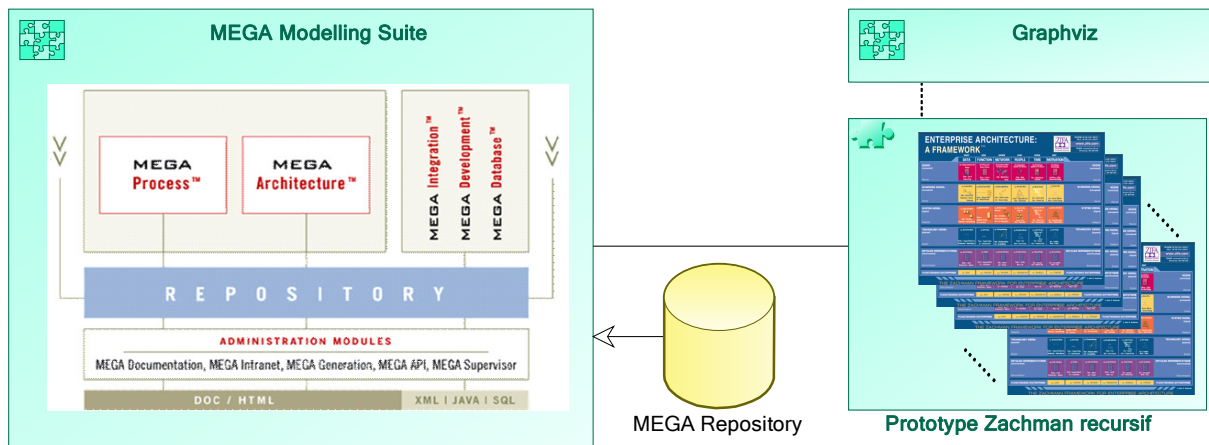


Figure 44. Architecture applicative du prototype d'outillage de modélisation

Le portail de MEGA Modelling Suite a aussi été modifié pour y ajouter un lien vers une cartographie de navigation du système en cours de modélisation. La construction de cette cartographie est basée sur l'application Graphviz³⁷. Elle est ainsi générée dynamiquement au gré de l'émergence de nouveaux systèmes donc de nouveaux paquetages. Cette cartographie permet d'avoir une vue synthétique du « système à faire » et surtout de naviguer par des liens hypertextes sur chacun des systèmes constitutifs ou bénéficiaires et donc sur leur Zachman respectifs. Les systèmes constitutifs et bénéficiaires sont distingués par des Zachman avec des couleurs de fond différentes : système bénéficiaire (couleur orange), système constitutif (couleur verte). Si le modélisateur rencontre un système de type « boîte noire » dont il a juste la connaissance des intrants et des extrants, la cartographie affiche le paquetage du système en noir.

Une fois entré dans un cadre de Zachman particulier à un système (constitutif ou bénéficiaire), le modélisateur peut afficher les objets de modélisations qu'il a créés au préalable en cliquant sur le lien hypertexte de chaque cellule. Pour ce faire, des requêtes SQL faisant appel aux objets de modélisation sont exécutées sur la base de modèle de l'environnement. Ainsi, le modélisateur peut bâtir d'autres objets de modélisation à partir des boîtes de dialogue associées à chaque cellule du cadre de Zachman et cela de manière récursive.

³⁷ <http://www.graphviz.org/>

Conclusions du chapitre 2

En nous basant à la fois sur notre proposition de Meta-Modèle d'un Système de Systèmes ('à faire' ou 'pour faire') et sur son aspect récursif au travers du cadre de Kuras, tout les deux étudiés dans le chapitre 1, nous avons cherché, dans ce chapitre 2, à instancier ce Meta-Modèle au 'Système-Projet' d'un SdSI multi-échelles.

Ainsi, nous avons étudié les cadres de modélisation majeurs d'ingénierie de systèmes et notamment le **cadre de modélisation de Zachman**. La relative généricité de ce cadre de modélisation lui confère une relative souplesse d'usage. En effet, le modélisateur garde une grande liberté d'action dans son acte de modélisation d'un système, tout en étant cadré par des points de vue à la fois sur le **niveau d'abstraction** d'un SI (contextuel, conceptuel, logique, technique et opérationnel) et sur le **niveau d'observation** d'un SI (quoi, comment, où, qui, quand, pourquoi). D'autre part, nous nous sommes intéressés à la **composition (gigogne) de l'ingénierie** de notre SdSI et conjointement à la **composition (gigogne) de son cadre de modélisation de Zachman** afin de représenter notre SdSI et ses éléments constitutifs d'un **point vue multi-échelles**. Ainsi, en réponse aux problématiques d'**interopération** de SI et de **capitalisation** de leurs modèles dans un contexte de SdSI, nous proposons de « distribuer » les « bons modèles », à la « bonne place », au « bon moment », au travers d'un cadre de modélisation de Zachman représentant l'ingénierie particulière de chaque SI constitutif d'un SdSI, suivant des points de vue différents relatifs à la **MOA** ou à la **MOE**. La récursivité des systèmes nous a amené à considérer notre ingénierie comme elle-même récursive afin de circonscrire la complexité des coopérations et donc des interopérations entre les SI constitutifs au sein d'un SdSI. Notre méthodologie d'ingénierie n'a pas pour objectif de remettre en cause les bonnes pratiques et les méthodologies déjà existantes en IS. Elle contribue cependant à guider le modélisateur et à cadrer ses bonnes pratiques afin de ne jamais perdre l'objectif originel du SAF, explicité par une exigence initiale d'un donneur d'ordres, et traduite par la MOA dans un modèle d'objectifs et d'exigences. De plus, elle participe à une certaine forme de capitalisation de la modélisation au fil de temps. Ainsi, les exigences sont le point de départ d'une conceptualisation cohérente d'un SAF complexe. Bien que ce mémoire ne traite pas de 'l'ingénierie des exigences', nous postulons que cette ingénierie doit d'être le processus d'IS pivot dans notre méthodologie.

Après avoir défini ce que nous entendions par une ISBM guidée par le cadre récursif de modélisation de Zachman, il nous fallait nous interroger sur 'avec quoi modélise-t-on ?' pour prototyper ('outiller') notre méthodologie.

Nous nous sommes appuyés sur l'environnement MEGA Modelling Suite dont le Meta-Modèle fournit les objets de base de modélisation d'un Système d'Information y compris certains relatifs à son ingénierie comme la gestion de projet et le cadre de Zachman. Nous avons étendu ce Meta-Modèle

des objets de modélisation nécessaires à mettre en œuvre notre méthodologie. Ce Meta-Modèle assure aussi la cohérence sémantique entre les différents objets de modélisation.

Nous ne nous sommes pas placés ainsi dans le contexte d'une entreprise qui ne dispose pas de cet environnement unifié de modélisation systémique mais d'un ensemble d'outils dont il faut assurer la cohérence sémantique entre leurs objets hétérogènes de modélisation. Cette problématique a particulièrement été explorée par les travaux de recherche autour d'UEML³⁸ (Verdanat 2002; Panetto et al. 2004; Anaya et al. 2008), qui visent « à assurer l'indépendance et l'interopérabilité entre chacun des formalismes initiaux, permettant ainsi de couvrir davantage les problématiques relevant de la modélisation d'entreprise » (Vallespir et al. 2003). Ces travaux ont montré la faisabilité d'un tel langage pivot, mais doivent encore être validés pour une éventuelle standardisation et une possible implémentation industrielle.

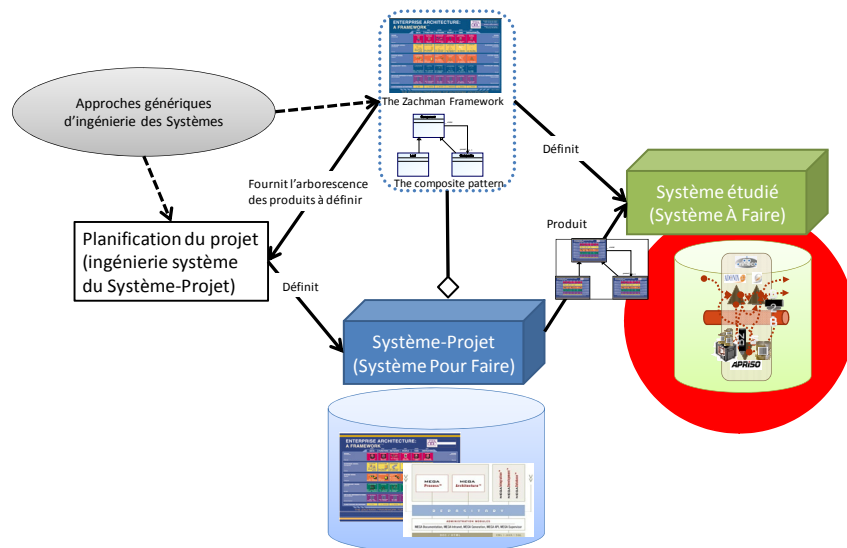
Nous pensons que SySML peut être candidat comme langage support à notre ISBM guidée par notre cadre récursif de modélisation. La taxonomie des diagrammes de SySML contient les diagrammes « d'exigences », de « paquetage », de « bloc », de « cas d'utilisation », de « séquence » qui nous semblent suffisants pour les niveaux de MOA et de MOE-1 que nous avons retenus dans notre méthodologie. Ces diagrammes sont en outre compréhensibles par les niveaux inférieurs de MOE, à charge d'interfacer les différents systèmes (COTS) composants le Système-Entreprise. Notons que notre démarche contribue à un usage méthodique de ce type de langage auquel il est souvent fait le reproche... de n'être qu'un langage.

³⁸ UEML : Unified Enterprise Modelling Language.

CHAPITRE 3.

SCÉNARIO DE MISE EN OEUVRE

Pratique d'un cadre de modélisation multi-échelles pour Faire un SI en entreprise centré sur le produit



Introduction et contexte de l'étude globale

Le contexte général de nos travaux de thèse concerne la production manufacturière dans un environnement multi-sites comprenant un site de conception, un site de production, et un transporteur et ses sites logistiques. Chaque site contribue au cycle de vie du produit et produit/consomme de l'information nécessaire à la réalisation de ses processus. A cette fin, chaque site échange des flux physiques et informationnels relatifs au produit à fournir au client final. Les sites sont distribués en Europe (France-Italie), sont autonomes opérationnellement et managérialement, ils évoluent au cours du temps et sont hétérogènes de part leurs propriétés intrinsèques. La mission commune à chacun des sites est de fournir le service juste nécessaire à la production d'un produit répondant aux exigences d'un client final. Nous cherchons ainsi à valider notre proposition méthodologique sur la base d'un scénario particulier. Nous nous intéressons ainsi à la **traçabilité du produit** sur la partie de son cycle de vie allant de sa conception jusqu'à sa livraison chez le client final. Ce système de traçabilité concerne alors notre « Système À Faire » (SAF), pour lequel nous proposerons une ingénierie système récursive dirigée par des modèles, en référence au cadre de modélisation résultat de nos travaux.

1 Environnement du système à faire

L'AIPL s'est engagé depuis plusieurs années dans un processus d'amélioration des services rendus à ses clients au travers d'une démarche qualité, ayant conduit à la certification ISO 9001 :2000. En complément de cette démarche, l'AIPL a mis en place un projet de mise en cohérence de ses SI et de mise en œuvre de nouvelles technologies : le projet « eProduction ». Ce projet a comme objectif principal de mettre à disposition des étudiants, des enseignants et des chercheurs une plateforme technologique émulant le fonctionnement d'une entreprise manufacturière.

Notre équipe de recherche contribue au projet « eProduction » de l'AIPL par l'étude des problématiques scientifiques d'interopérabilité entre des SI hétérogènes, distribués et autonomes, dans un contexte d'entreprise de production.

De manière plus particulière, le cas d'étude traité dans notre thèse concerne l'ingénierie d'un système de traçabilité des produits. Ce système de traçabilité est constitué d'un ensemble de SI autonomes de par leur implication dans le cycle de vie des produits (*propriété d'autonomie* (Tableau 3 ; page 33).

Chaque système constituant notre système de traçabilité produit de l'information au profit des autres systèmes constituants. De manière complémentaire, chaque système constituant utilise l'information mise à disposition par d'autres systèmes (*propriété d'appartenance* (Tableau 3 ; page 33).

Les SI constituant notre système de traçabilité fournissent des services de communication permettant leur interconnexion au réseau de l'entreprise et aux ressources productrices et consommatrices d'information (*propriété de connectivité* (Tableau 3 ; page 33).

Notre système de traçabilité est composé de SI divers et hétérogènes, le long du cycle de vie des produits (*propriété de diversité* (Tableau 3 ; page 33).

Chaque SI constituant notre système de traçabilité est spécifique à une étape du cycle de vie des produits. En conséquence, chacun a une vue particulière (métier) de l'information relative à ces produits. Il contribue, à sa manière, à l'émergence du système d'information global assurant sa mission de traçabilité des produits (*propriété d'émergence* (Tableau 3 ; page 33).

Notre SAF – le système de traçabilité des produits – est caractérisé par les 5 propriétés (A (autonomie) – B (émergence) – C (connectivité) – D (diversité) – E (appartenance)) qui définissent un SdS. Il est composé de sous-systèmes d'information. Nous pouvons donc le considérer comme un Système de Systèmes d'Information (SdSI).

Nous avons élaboré un scénario de traçabilité de produits, multi-sites (Italie-France), mettant en œuvre un ensemble hétérogène de technologies et de SI autonomes pour la gestion et la production de produits manufacturés (figure ci-dessous).

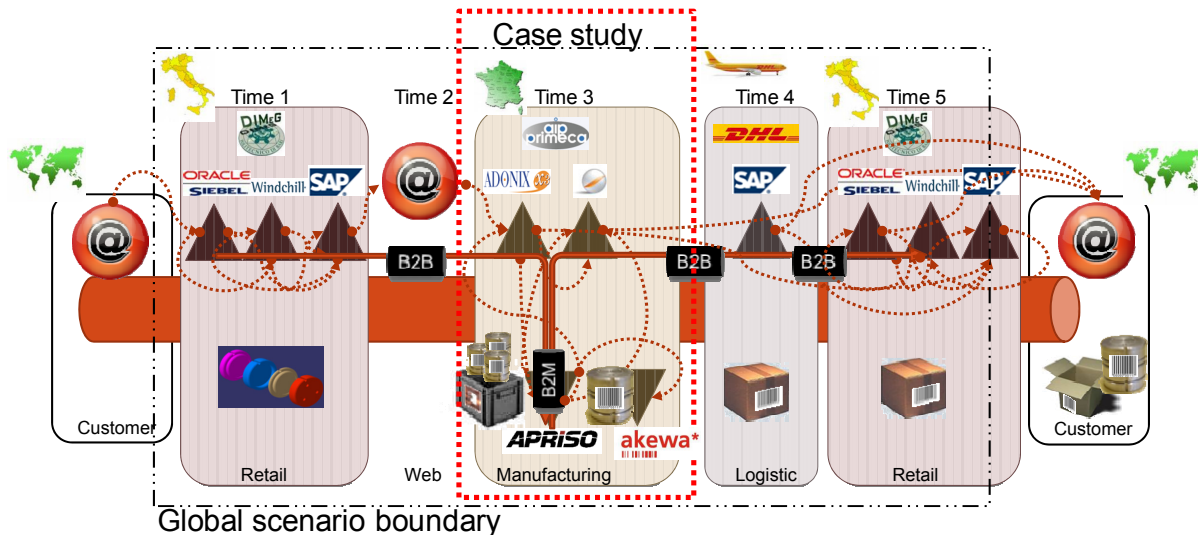


Figure 45. Scénario global

Le site français (AIPL³⁹) fournit une infrastructure de type industrielle comprenant des ingénieurs, des Machines Outils, une ligne de montage automatique, un Logiciel de conception (CAD), un ERP et un MES. Le site italien (DIMEG⁴⁰) fournit une infrastructure pour la conception et la commercialisation de pièces mécaniques, comprenant des ingénieurs, un CAD, un ERP, un PDM, et un CRM (Figure 46).

³⁹ AIPL : Atelier Inter-établissements de Productique Lorrain

⁴⁰ DIMEG : Dipartimento di Ingegneria Meccanica e Gestionale – Politecnico di Bari, Italie

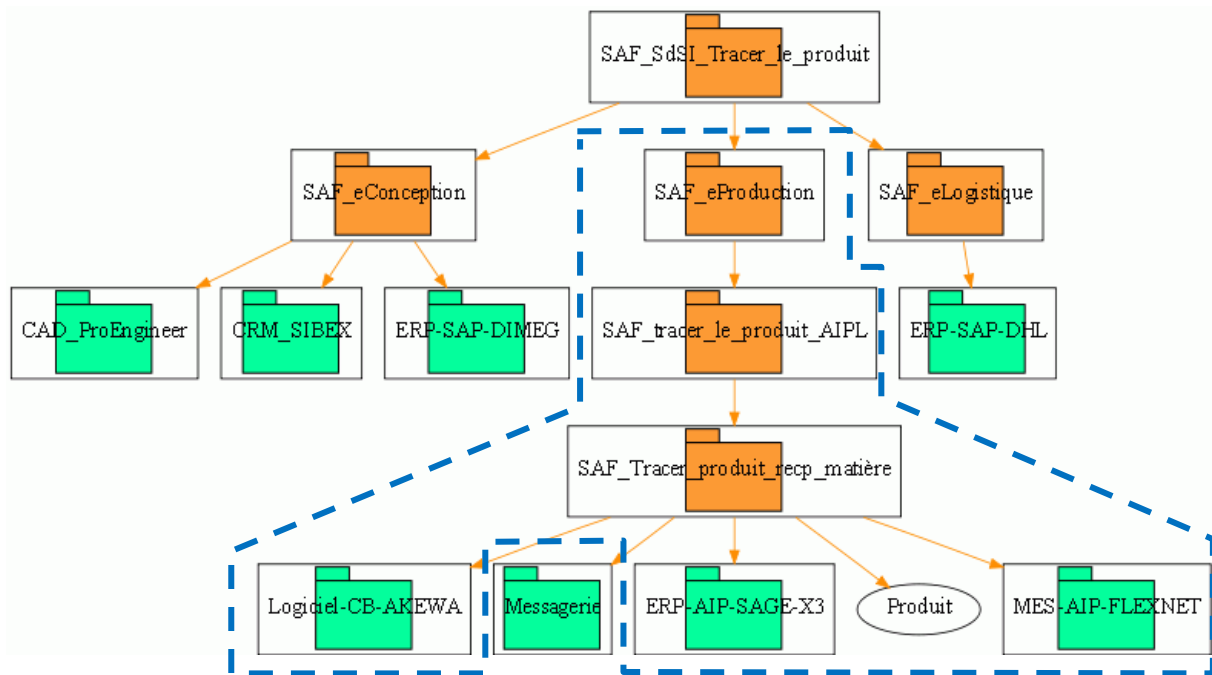


Figure 46. Arborescence des systèmes bénéficiaires et constitutifs du cas d'étude.

Afin de valider notre Ingénierie Système Réursive Basée sur les Modèles (ISBM) et le prototype d'outillage de modélisation associé, nous avons focalisé notre étude sur une partie de ce scénario : **l'interopérabilité synchronique de SdSI centré sur le produit** sur le site de production en France (partie encadrée en pointillé ; Figure 45). Nous avons étudié, en particulier, le début du cycle de vie du produit sur ce site de production : la **réception matières**. Ainsi, pour illustrer nos propos, nous mettrons en œuvre notre méthodologie de modélisation afin de spécifier rigoureusement le cahier des charges de développement de notre SAF, avec une focale particulière sur l'interopérabilité entre les SI concernés par le processus de **réception matières**.

Une interview des parties prenantes de ce scénario nous permet de dégager les principaux processus du cycle de vie du produit dans un diagramme de collaboration (Figure 47):

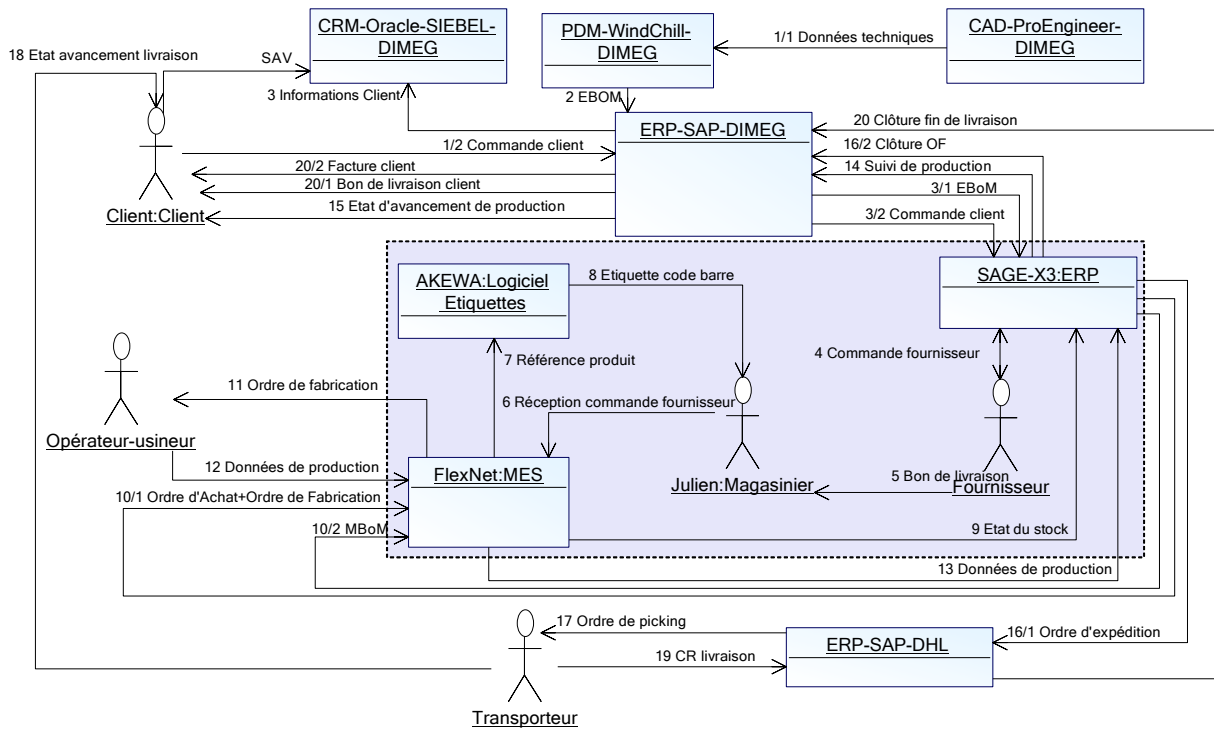


Figure 47. Diagramme de collaboration du scénario global

Afin de proposer au lecteur une meilleure compréhension de la méthode de modélisation, nous avons choisi de ne développer notre méthodologie que sur le système de traçabilité des produits du projet « e-Production » et de n'implémenter qu'un processus du scénario : la **réception matières sur le site de production en France à l'AIPL** (étape 6 du scénario global de la Figure 48 ; zone encadrée de la Figure 47).

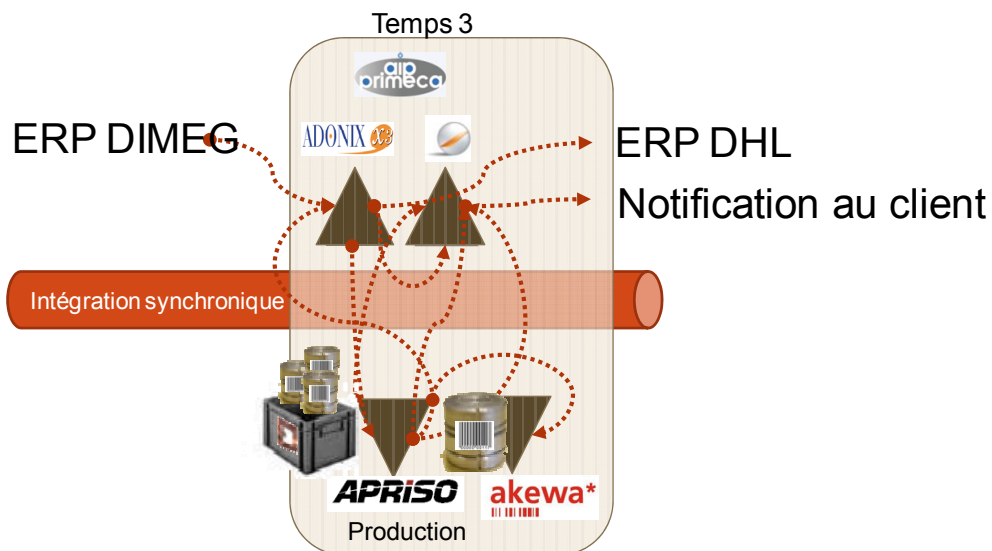


Figure 48. Vue synchrone du SAF de traçabilité à l'AIPL

Ce processus met en œuvre des SI tels qu'un **ERP** (SAGE X3), un **MES** (APRISO-FLEXNET93), et un **Logiciel d'étiquetage de codes barres** (AKEWA).

- L'interview du magasinier concernant ce processus a fait apparaître un ensemble d'activités : l'ERP du site Français calcule chaque semaine ses besoins nets et lance des ordres d'achats (OA) et/ou des ordres de fabrications (OF) en fonction de sa planification à long terme.
- Les OA et OF sont exportés de l'ERP vers le MES du site Français.
- L'ERP exporte l'OF de la commande client vers un serveur de messagerie.
- Le serveur de messagerie sur le site Français envoie un mail au client pour l'informer de la mise en production de sa commande.
- Le MES répercute les OA et OF vers les ressources.
- Les ressources (ressources humaines et équipements) renvoient des comptes rendus depuis l'IHM⁴¹ du MES pour stipuler la prise en compte de l'ordre pour son exécution, la fin de l'exécution et éventuellement des informations qualitatives.
- Le MES envoie chaque jour l'état des stocks, des en-cours et l'état des OA-OF à l'ERP du site Français.

2 Ingénierie Système Basée sur les Modèles du projet « E-Production »

Nous allons maintenant illustrer notre proposition méthodologique, ainsi que le prototype d'outillage de modélisation, à travers l'**Ingénierie Système Basée sur les Modèles** (ISBM) du système de traçabilité des produits à l'AIPL.

Après avoir présenté notre prototype d'outillage de modélisation, basé sur une adaptation de l'environnement MEGA Modelling Suite de la société MEGA International⁴², nous mettrons en application notre proposition méthodologique (chapitre 2) sur notre cas d'étude.

Notre démarche sera décomposée en 3 étapes :

- La préparation du projet par la MOA (§Chapitre 4.2.2.1)
- La définition du SAF par la MOE de rang 1 (§ Chapitre 4.2.2.2)
- La spécification du SAF par la MOE de rang 2 (§ Chapitre 4.2.2.3)

Pour faciliter la compréhension des modèles par le lecteur, l'ensemble des formalismes symboliques mis en œuvre par MEGA Modelling Suite est répertorié en Annexe.

⁴¹ **IHM** : Interface Homme-Machine













⁴² **MEGA International** : <http://www.mega.fr>

2.1 Préparation du projet (MOA)

2.1.1 Gestion du projet d'ingénierie de la de traçabilité des produits

Les frontières du projet « eProduction » de l'AIPL ont été formalisées textuellement dans un document Word (en annexe). Ce document a été rédigé par l'équipe de direction de l'AIPL. Il est le point de départ de notre étude de cas. Le projet « eProduction » de l'AIPL est lui-même décomposé en sous-projets dont le projet de mise en place d'un système de traçabilité des produits. La gestion du projet « eProduction » avec ses différentes phases est détaillée dans le tableau ci-dessous et son GANTT associé (en annexe).

Tableau 17. Gestion du projet « eProduction » – extrait du Gantt du projet.

N°		Nom de la tâche	Durée	Début	Fin
1		Processus de préparation du projet	8 jours	Mar 13/05/08	Jeu 22/05/08
2		Initialisation	4 jours	Mar 13/05/08	Ven 16/05/08
3		Planification	4 jours	Lun 19/05/08	Jeu 22/05/08
4		Processus de définition du projet	96 jours?	Lun 26/05/08	Lun 17/11/08
5		Processus de réception matière	245 hr	Lun 26/05/08	Ven 11/07/08
6		Processus d'assemblage automatique	182 hr?	Lun 29/09/08	Lun 17/11/08
7		Processus de collage	182 hr?	Lun 29/09/08	Lun 17/11/08
8		Processus de transitique	182 hr?	Lun 29/09/08	Lun 17/11/08
9		Processus d'usinage	182 hr?	Lun 29/09/08	Lun 17/11/08
10		Processus de colisage	182 hr?	Lun 29/09/08	Lun 17/11/08
11		Processus de spécification	75,71 jours?	Lun 14/07/08	Lun 08/12/08
12		Processus de réception matière	28 hr	Lun 14/07/08	Jeu 17/07/08
13		Processus d'assemblage automatique	40 hr?	Mar 18/11/08	Mar 25/11/08
14		Processus de collage	28 hr?	Mar 18/11/08	Ven 21/11/08
15		Processus de transitique	28 hr?	Mar 25/11/08	Lun 01/12/08
16		Processus d'usinage	28 hr?	Mar 18/11/08	Ven 21/11/08
17		Processus de colisage	35 hr?	Lun 01/12/08	Lun 08/12/08
18		Processus de développement	78,14 jours?	Ven 18/07/08	Mer 17/12/08
19		IHM pour réception matière	43 hr	Ven 18/07/08	Lun 25/08/08
20		IHM pour assemblage automatique	45 hr?	Mar 25/11/08	Jeu 04/12/08
21		IHM pour collage	45 hr?	Lun 24/11/08	Mar 02/12/08
22		IHM pour transitique	45 hr?	Lun 01/12/08	Mer 10/12/08
23		IHM pour usinage	45 hr?	Lun 24/11/08	Mar 02/12/08
24		IHM pour colisage	45 hr?	Lun 08/12/08	Mer 17/12/08
25		Intégration des systèmes	74 jours?	Lun 25/08/08	Ven 19/12/08
26		Processus de réception matière	2 jours?	Lun 25/08/08	Mer 27/08/08
27		Processus d'assemblage automatique	2 jours?	Jeu 04/12/08	Lun 08/12/08

La MOA ne s'attache qu'à définir le niveau contextuel du SAF (niveau « contextuel » du cadre de modélisation de Zachman). A partir de l'analyse du cahier des charges, la MOA va enchaîner (Figure 49) :

1. Un diagramme d'objectifs et d'exigences du SAF principal qui spécifie les grands objectifs du SAF,

2. Un diagramme de paquetage qui spécifie les SAF constitutifs répondant aux objectifs et exigences du SAF principal,
3. Un diagramme d’environnement de processus du SAF principal qui spécifie ses processus principaux,
4. Un diagramme de séquence qui spécifie les points de synchronisation entre les SAF constitutifs,
5. Une cartographie des sites contributeurs,
6. Un organigramme qui spécifie les noms des responsables des SAF constitutifs.

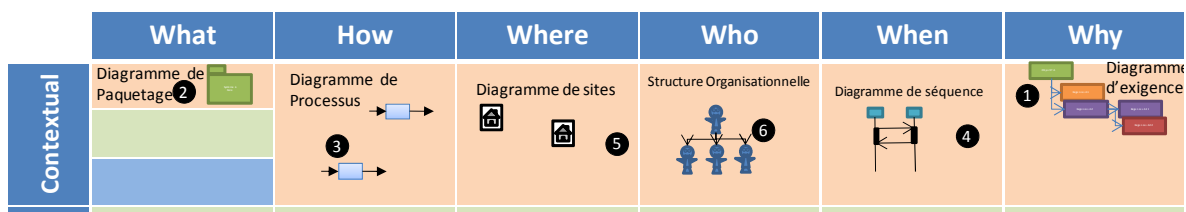


Figure 49. Zachman du SAF (niveau « contextual »).

2.1.2 (Contextual-why du SAF), Diagramme des exigences

Le premier travail de la MOA a été de comprendre les objectifs du donneur d’ordres à partir d’un document Word©. Un premier diagramme des exigences a été proposé puis réajusté en fonction des interviews avec le donneur d’ordres. Une fois le diagramme des exigences validé par le donneur d’ordres (Figure 50), il devient le document de référence pour l’ensemble des autres modèles. Il devient de la sorte le document contractuel entre le donneur d’ordres et la MOA. Ainsi, parmi les objectifs principaux du donneur d’ordres, la MOA s’est focalisée dans un premier temps sur un SAF particulier : le **système de traçabilité du produit à l’AIPL**.

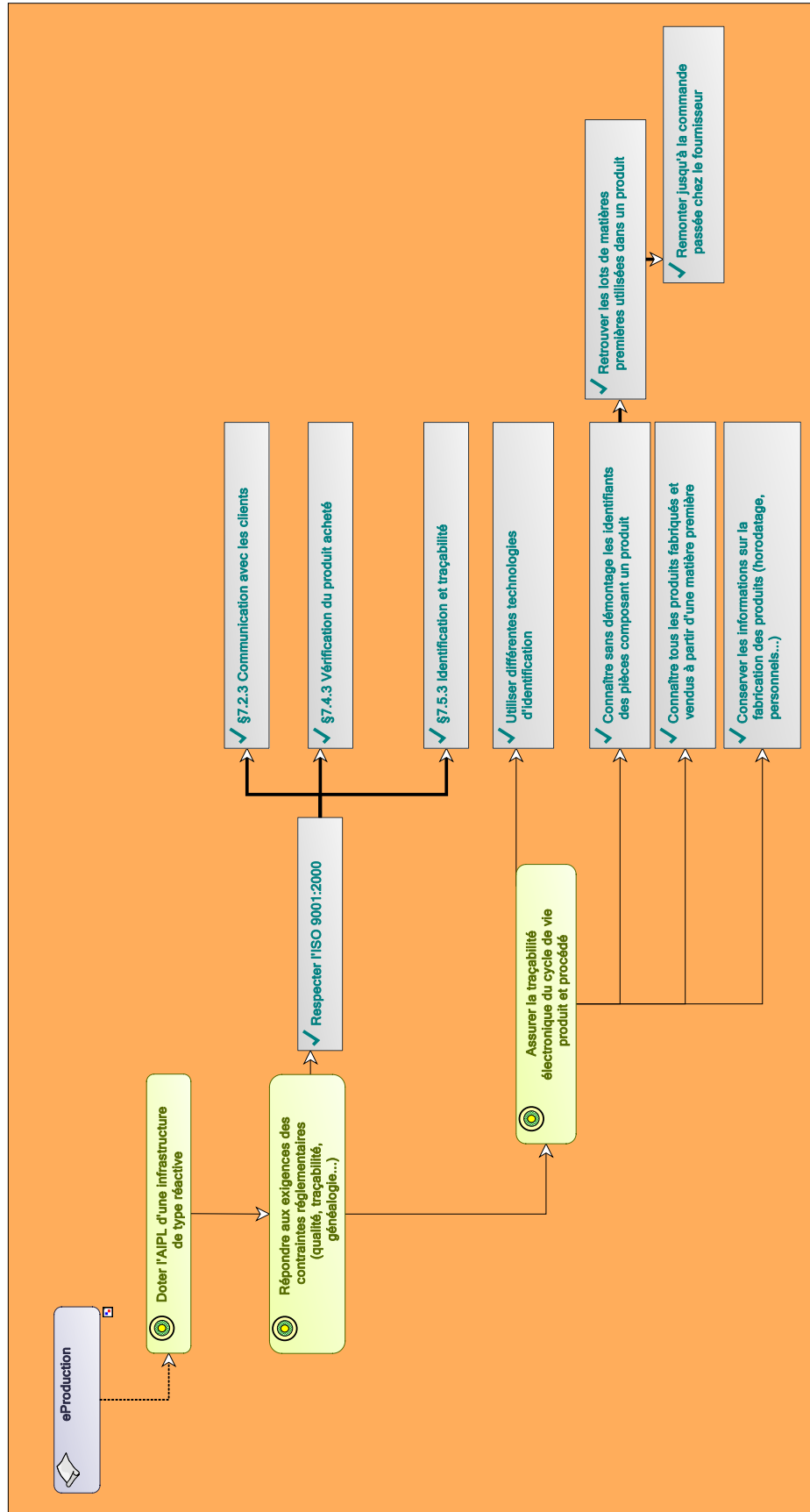


Figure 50. Diagramme des exigences du projet "eProduction"

2.1.3 (Contextual-What du SAF), diagramme de paquetage

La MOA du SAF principal « e-Production » représente ce système par un paquetage stéréotypé « Système ». Comme nous lui appliquons une ingénierie spécifique, la MOA lui associe son propre cadre de modélisation de Zachman. À partir du diagramme des exigences précédent, elle détermine les systèmes constitutifs supports de ces exigences (Tableau 18).

Tableau 18. Correspondances entre les exigences du SAF "e-Production" focalisé sur la réception matières et les applications informatiques existantes à l'AiPL

	Exigences du projet « e-Production »	Systèmes contraints
	<ul style="list-style-type: none"> § 7.2.3 Communication avec les clients 	
Réception matières	<ul style="list-style-type: none"> § 7.4.3 Vérification du produit acheté 	SAGE X3
	<ul style="list-style-type: none"> § 7.5.3 Identification et traçabilité 	FLEXNET 93
	<ul style="list-style-type: none"> Utiliser différentes technologies d'identification 	FLEXNET 93 ; AKEWA
	<ul style="list-style-type: none"> Connaître sans démontage les identifiants des pièces composants un produit 	FLEXNET 93
	<ul style="list-style-type: none"> Retrouver les lots de matières premières utilisées dans un produit 	FLEXNET 93
	<ul style="list-style-type: none"> Remonter jusqu'à la commande passée chez le fournisseur 	SAGE X3
	<ul style="list-style-type: none"> Connaître tous les produits fabriqués et vendus à partir d'une matière première 	
	<ul style="list-style-type: none"> Conserver les informations sur la fabrication des produits (horodatage, personnels,...) 	

Le diagramme de paquetage résultant (Figure 51) spécifie ainsi l'ensemble des systèmes constitutifs du système principal. Chaque système constitutif pourra ainsi donner lieu à un projet spécifique d'ingénierie.

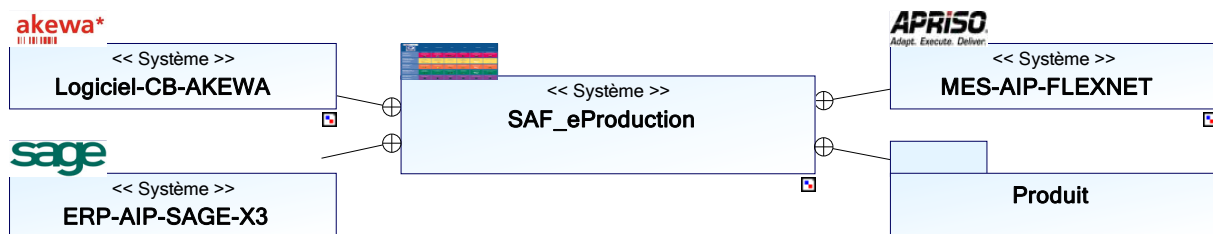


Figure 51. Diagramme de paquetage du SAF « e-Production »

2.1.4 (Contextual-How du SAF), diagramme de processus

Ce modèle spécifie le ou les domaines fonctionnels relatifs au SAF principal et les processus s’y rattachant. Notre SAF « e-Production » concernant le domaine fonctionnel de la production de biens et de services, nous avons ainsi choisi de dériver les processus relatifs au domaine fonctionnel de la production à partir de ceux proposés par le modèle MES de MESA International⁴³ (Annexe 9). La spécification des flux informationnels échangés entre ces processus (et leurs sous-processus) est dérivée du modèle fonctionnel proposé par le standard PRM⁴⁴ (Annexe 10). Nous spécifions ainsi une vue d’ensemble des processus de notre SAF et leurs interrelations informationnelles (Figure 52).

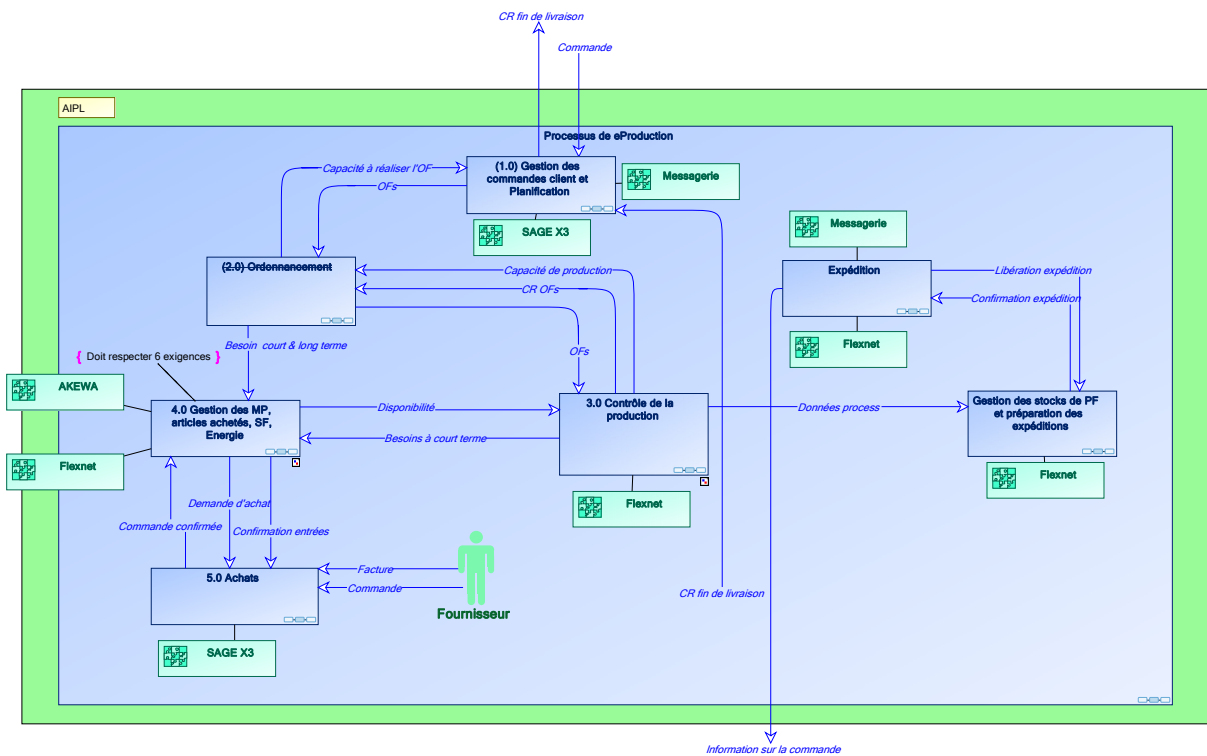


Figure 52. Diagramme de processus du SAF « e-Production », dérivé du modèle fonctionnel PRM

Notre étude de cas détaillée étant relative à la « réception matières », nous allons donc nous focaliser par la suite sur le processus 4.3 (Gestion des réceptions matières premières et énergies), sous processus du processus 4.0 (gestion des MP, articles achetés, SF, énergie) de notre modèle et ses interactions avec les processus connexes (Figure 53).

⁴³ MESA International : <http://www.mesa.org/>

⁴⁴ PRM : Purdue Reference Model, <http://www.pera.net>

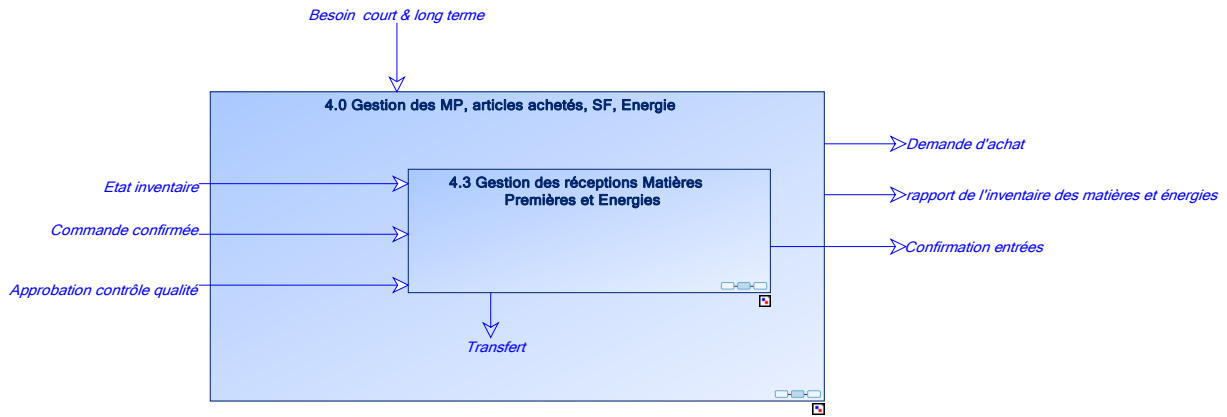


Figure 53. Processus de réception matières dérivé du PRM

2.1.5 (Contextual-When du SAF), diagramme de séquence

Ce diagramme de séquence (Figure 54) spécifie et caractérise les échanges informationnels entre les systèmes constitutifs intervenant dans la réalisation du processus de réception matières identifié à l'étape précédente (sous processus 4.3).

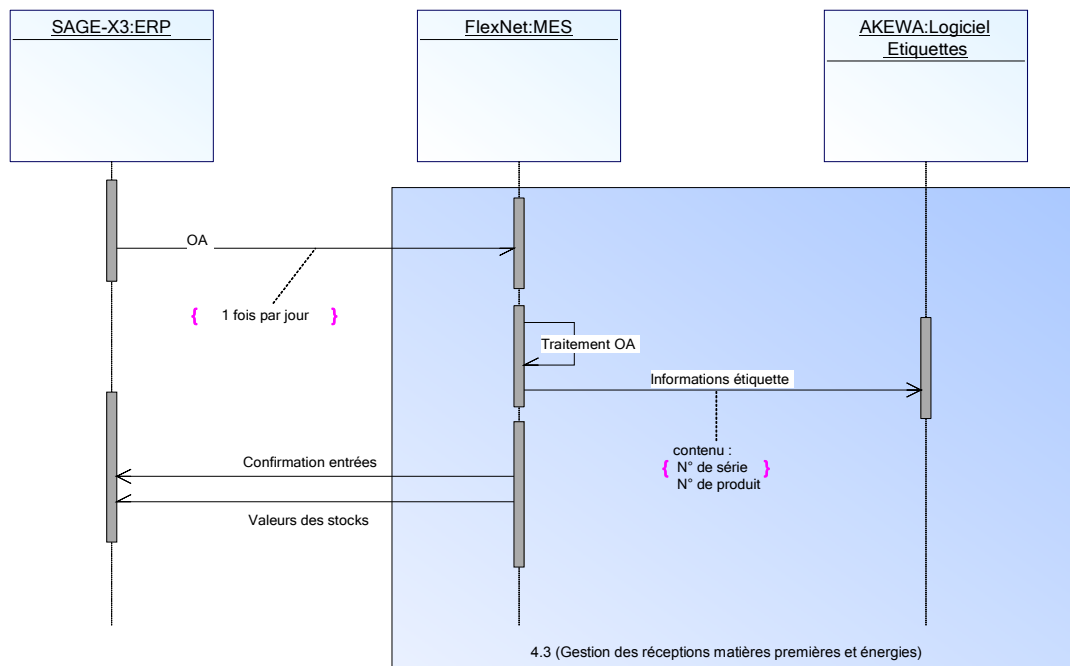


Figure 54. Diagramme de séquence relatif au sous processus 4.3 (Gestion des réceptions matières premières et énergies)

2.1.6 (Contextual-Where du SAF), diagramme de site

Ce diagramme spécifie les zones que l'AIPL utilise dans le cadre du « SAF_eProduction » focalisé sur la réception matière. À cet instant de la modélisation (étape : réception matière), ce diagramme ne recense que la zone de réception matières au sein de l'infrastructure de l'AIPL (Figure 55).

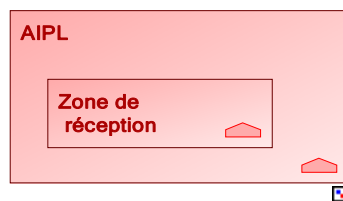


Figure 55. Diagramme de sites du SAF « e-Production » focalisé sur la réception matières

2.1.7 (Contextual-Who du SAF), structure organisationnelle

La MOA du « SAF_eProduction » spécifie aussi la structure organisationnelle nécessaire au fonctionnement nominal du SAF au niveau de la réception matière. Cet organigramme recense les rôles organisationnels mis en œuvre lors de la réception matières (Figure 56).

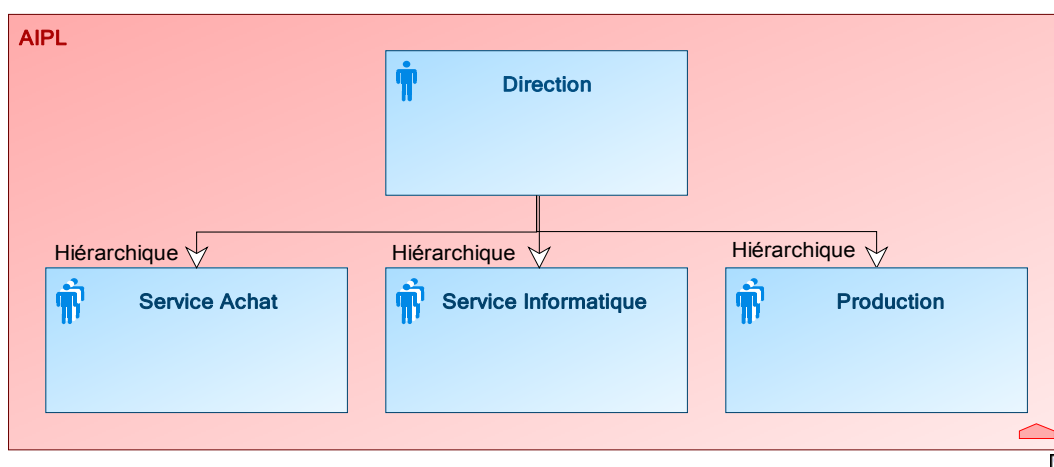


Figure 56. Structure organisationnelle du SAF « e-Production » focalisé sur la réception matières

2.2 Définition du SAF « système de traçabilité du produit à l'AIPL » (MOE de rang 1)

La MOE de rang 1 ne s'attache qu'à définir le niveau conceptuel du SAF « système de traçabilité du produit à l'AIPL » (niveau « Conceptual » du cadre de modélisation de Zachman). A partir de l'analyse du cahier des charges spécifié précédemment par la MOA (§ Chapitre 3.2.1), incluant des modèles concernant le SAF « système de traçabilité du produit à l'AIPL » mais focalisé sur la réception matière, la MOE de rang 1 va enrichir les modèles du SAF « système de traçabilité du produit à l'AIPL » (focalisés sur la réception matière), avec le point de vue métier de l'AIPL (Figure 57). La MOE de rang 1 apporte ces connaissances sur les processus, les procédures, les modes opératoires, les procédés de l'AIPL, ainsi que les connaissances sur l'ingénierie des SI. Les modèles vont s'enrichir progressivement, en suivant l'évolution du cycle de vie du produit. Dans notre cas d'étude, la MOE de rang 1 a spécifié différents modèles :

7. Diagramme des exigences particulières à chacun des SI internes à l'AIPL et contributeurs à la réception matières,

8. Diagramme de mise en œuvre des processus spécifiant la procédure de réception matières à l'AIPL,
9. Diagramme de paquetage du SAF « système de traçabilité du produit à l'AIPL » de l'AIPL complété par les paquetages des SI internes constitutifs à la réception matière,
10. Diagramme de classe sémantique relatif au produit,
11. Organigramme opérationnel des acteurs de l'AIPL, contributeurs à la réception matières.

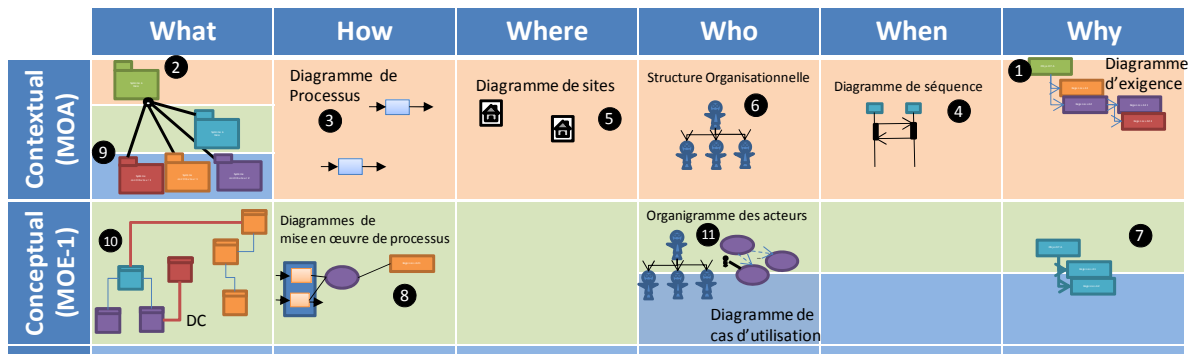


Figure 57. Zachman du SAF « système de traçabilité du produit à l'AIPL » (niveau "Conceptual").

2.2.1 (Conceptual-What du SAF), diagramme de classe

Notre proposition méthodologique se basant sur le paradigme SCP, notre modélisation est guidée par le cycle de vie du produit. Les travaux de thèse d'Angela Tursi (Tursi et al. 2007) ont pour objectif de proposer un modèle sémantique (ontologie « produit »), pivot des interopérations successives entre les modèles sémantiques des SI constitutifs d'un SAF. Ces travaux étant en cours, afin de valider notre proposition méthodologique sans attendre ce résultat, nous avons analysé de manière spécifique les modèles logiques des applications impliquées dans le processus de réception matière, et proposé une conceptualisation partielle des concepts qui leurs sont communs. Nous avons procédé à une conceptualisation d'une vue partielle des modèles logiques des applications (SAGE X3 & FlexNet93) restreint à notre scénario. À partir des modèles sémantiques (Figure 58, Figure 59, Figure 60), nous avons proposé une correspondance sémantique des concepts respectivement de SAGE X3 vers le Produit (Tableau 19) et du Produit vers Flexnet93 (Tableau 20). Ces correspondances nous ont permis de filtrer sur les modèles logiques correspondants, les champs implémentant les concepts modélisés et leurs interrelations (Tableau 22, Tableau 23, Tableau 24)

Tableau 19. Correspondance sémantique

entre l'application SAGE X3 et le produit

Concepts SAGE X3	Concepts Produit
Site commande	Nom
Numéro commande	NumeroBonDeCommande
Date commande	DateEmission
Fournisseur	CodeFournisseur
Inter-sites	
Inter-sociétés	
Condition paiement	
Acheteur	
Devise	Devise
Type cours	
Régime taxe	TVA
Site facturation	Site-Nom
Centre de coûts	Site-Nom
Section analytique	
Montant mini commande	
Total HT commande	PrixTotalHT
Article	NomProduit
Désignation	Description
Désignation standard	Description
Site réception	Nom
Adresse site	Adresse
Unité de mesure	Unite
Quantité commandée	QtyCommandee
Coefficient	
Unité achat	
Unité stock	
Quantité de stockage	
Date réception	
Prix net	PrixUnitaireHT
Montant ligne HT	
Compte	CompteBudgetaireInt
Nature	
Type achat	CategorieAchat
Soumis à contrôle	
Majoration CEE	
Pays	Pays
Soldée	
Numéro d'avenant	
Origine demande	
Taxe 1	TVA

Tableau 20. Correspondance sémantique entre le produit et l'application FlexNet93

Concepts Produit	Concepts FleNet93
NumeroBonDeCommande	Référence de la commande
	Type de commande
	Statut
DateEmission	Date de la saisie de la commande
	État d'avancement de la commande
	Numéro de ligne de la commande
Unite	Unité de la commande
QtyCommandee	Quantité commandée
Site-Nom	Site de production de destination
Site-Nom	Site de stockage de destination
NomProduit	Référence produit
	Numéro de produit
	Numéro d'Ordre
	Type d'ordre (Bon de commande)
	Priorité
DateEmission	Date de déclenchement Du bon de commande
	Date d'échéance de la commande
Unite	Unité
	Statut du bon de commande
CodeFournisseur	Référence du partenaire
	Rôle du partenaire (fournisseur)
CodeFournisseur	Référence fournisseur

Tableau 21. Tables, champs et concepts relatifs à la réception matières pour SAGE X3

SAGE X3		
Tables	champs	Concepts
POH0_1	POHFCY	Site commande
	POHNUM	Numéro commande
	REVNUM	
	ORDDAT	Date commande
	BPSNUM	Fournisseur
	BETFCY	Inter-sites
	BETCPY	Inter-sociétés
POH1_1	PTE	Condition paiement
	BUY	Acheteur
POH1_2	CUR	Devise
	CHGTYP	Type cours
POH1_3	VACBPR	Régime taxe
	INVFCY	Site facturation
POH1_4	CCEINT	Centre de coûts
	CCE	Section analytique
	ORDMINAMT	Montant mini commande
	WCUR	Devise
	TOTORD	Total HT commande
	WCURI	Devise
	ITMREF	Article
POH2_3	ITMDES	Désignation
	ITMDES1	Désignation standard
	PRHFCY	Site réception
POH2_1	FCYADD	Adresse site
	UOM	Unité de mesure
	QTYUOM	Quantité commandée
	UOMPUUCOE	Coefficient
POH3_1	PUU	Unité achat
	STU	Unité stock
	QTYSTU	Quantité de stockage
	EXTRCPDAT	Date réception
	NETPRI	Prix net
	LINAMT	Montant ligne HT
	LINACC	Compte
	LINCNA	Nature
	LINPURTYP	Type achat
	QUAFLG	Soumis à contrôle
	EECINCRAT	Majoration CEE
	ORICRY	Pays
	LINCLEFLG	Soldée
	LINREVNUM	Numéro d'avenant
	ORI	Origine demande
	VAT1	Taxe 1
	WCUR	Devise

Tableau 22. Tables, champs et concepts relatifs à la réception matières pour FlexNet93

FlexNet93		
Tables	champs	Concepts
OrderHeader	OrderNo	Référence de la commande
	OrderType	Type de commande
	OrderStatus	Statut
	OrderDate	Date de la saisie de la commande
	ProgressStatus	Etat d'avancement de la commande
	OrderDetail	OrderNo
OrderDetail	OrderType	Type de commande
	OrderLineNo	Numéro de ligne de la commande
	OrderStatus	Statut de la commande
	OrderedUomCode	Unité de la commande
	QuantityOrdered	Quantité commandée
	ToFacility	Site de production de destination
	ToWarehouse	Site de stockage de destination
	ProgressStatus	Etat d'avancement de la commande
	ProductID	Référence produit
	ProductNo	Numéro de produit
ProductID	WipOrderNo	Numéro d'Ordre
	WipOrderType	Type d'ordre (Bon de commande)
WipOrder	OrderQuantity	Quantité commandée
	Priority	Priorité
	ScheduledStartDate	Date de déclenchement Du bon de commande
	DueDate	Date d'échéance de la commande
	UomCode	Unité
	WorkOrderStatus	Statut du bon de commande
	OrderNo	Référence de la commande
	OrderType	Type de commande
	OrderLineNo	Numéro de ligne de la commande
	ProgressStatus	Etat d'avancement de la commande
	ProductID	Référence du produit
WipOrder	OrderNo	Référence de la commande
	OrderType	Type de commande (bon de commande)

OrderPartner	PartnerID	Référence du partenaire
	Account	Référence du partenaire
	PartnerRole	Rôle du partenaire (fournisseur)
PartnerID		Référence fournisseur

Tableau 23. Tables, champs et concepts relatifs à la réception matières pour le produit

Modèle Produit	
Tables	Champs
Produit	NomProduit Description Unite
Ligne de commande	Numéro de ligne QteCommandee QteRecue Status Description PrixUnitaireHT ProxTotalHT TVA Devise CategorieAchat
Zone stockage	Nom
Site	TypeSite
Bon de commande	NumeroBonDeCommande DateEmission Description DomaineFonctionnel CompteBudgetaireInvnt Fonds
Raison sociale	Nom Adresse CP Ville Tel Fax Email SiteWeb SIRET
Fournisseur	CodeFournisseur Type

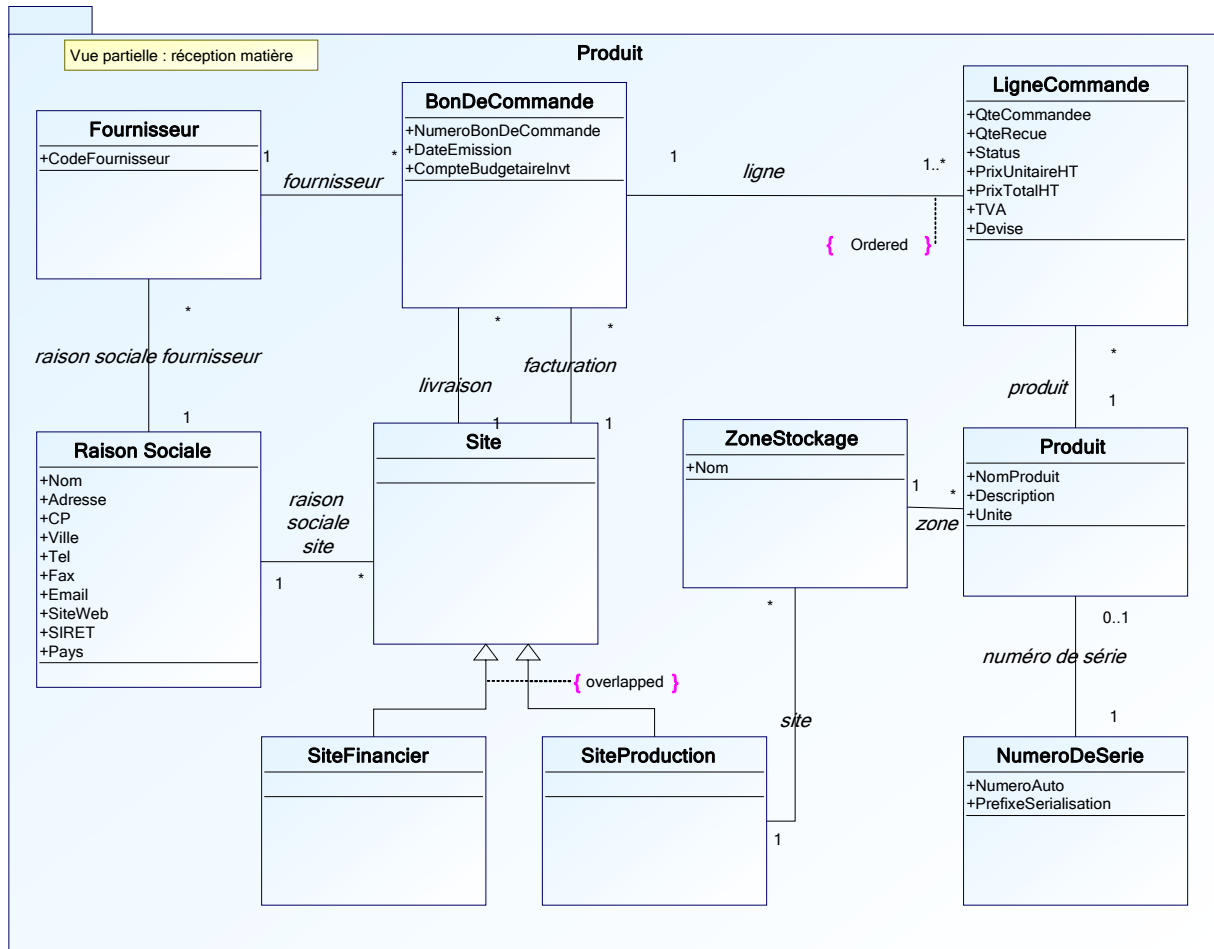


Figure 58. Conceptualisation partielle du modèle sémantique du produit relatif au SAF « système de traçabilité du produit à l'AIPL » focalisé sur la réception matières

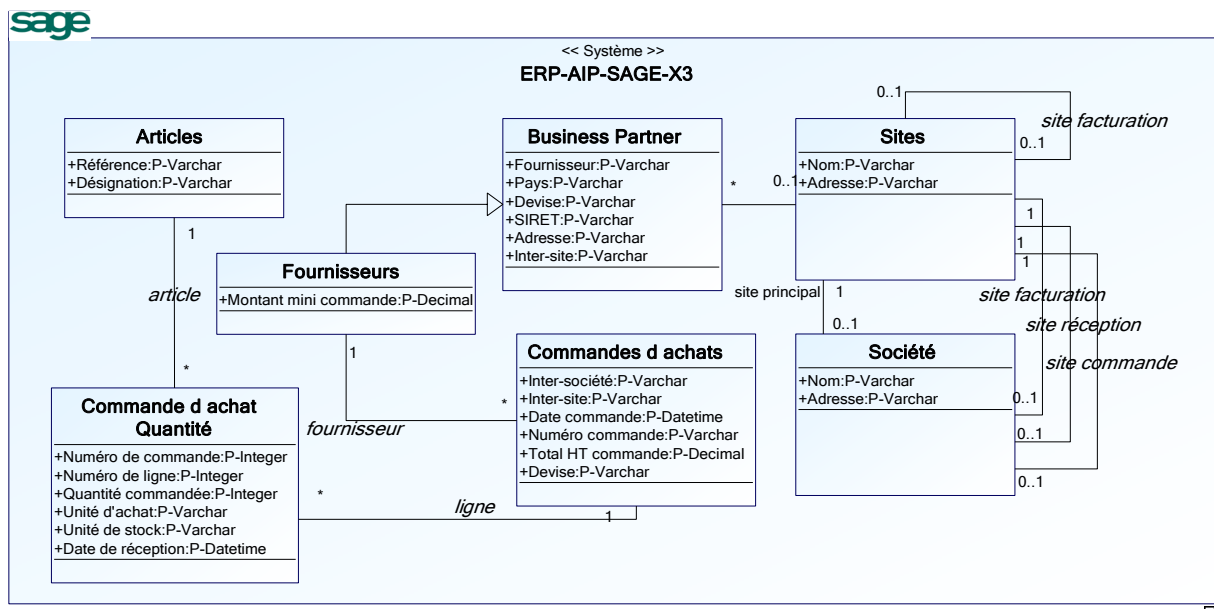


Figure 59. Conceptualisation partielle du modèle logique interne à SAGE X3

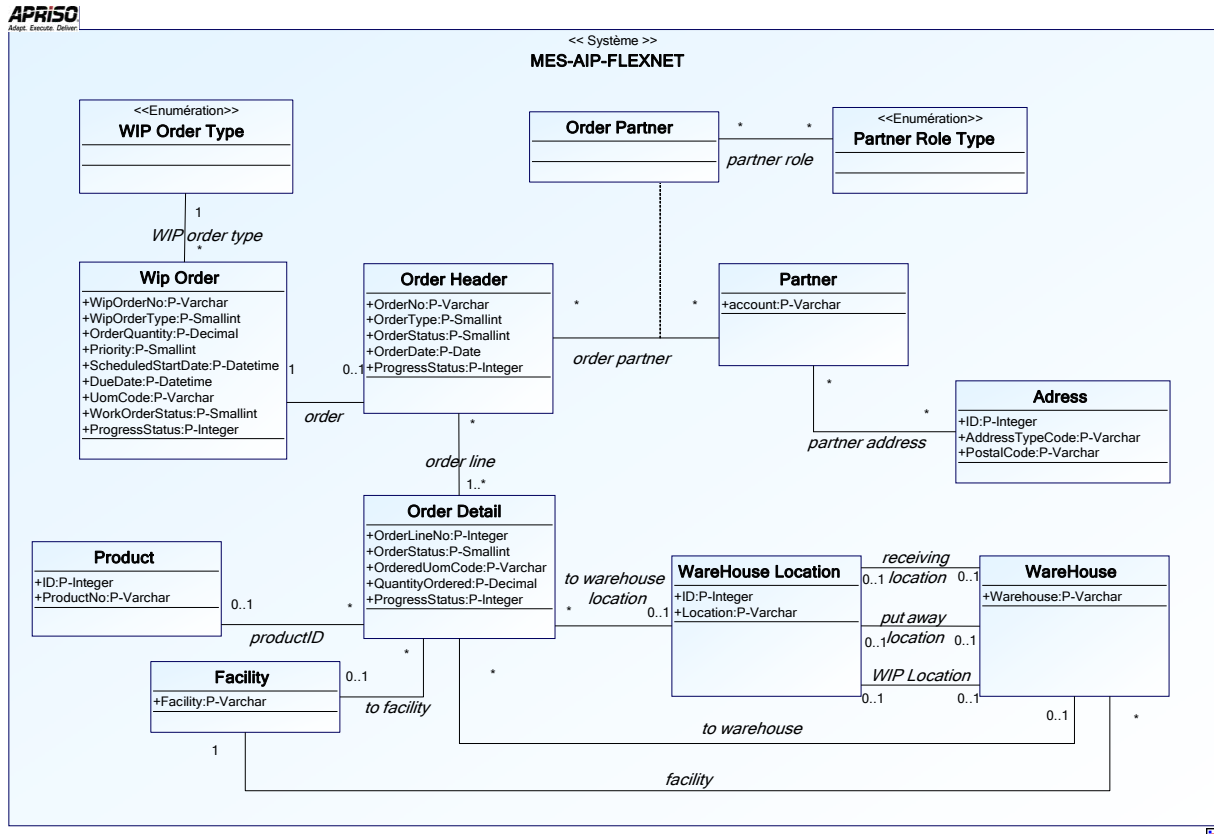


Figure 60. Conceptualisation partielle du modèle logique interne à FlexNet93

2.2.2 (Conceptual-Why du SAF), diagramme d'exigences

Pour modéliser le point de vue métier « réception matières », nous nous sommes inspirés des procédures formalisées par l'A IPL dans le cadre de sa démarche qualité ISO9001 :2000. Ainsi, nous avons modélisé une vue de l'activité « réception matières » par un diagramme d'exigences (Figure 61) qui nous a permis d'associer l'activité « réception matières » aux exigences héritées du « SAF_eProduction » qu'il a lui-même hérité du « SAF_SdSI_tracer_ produit ». En associant ces exigences à l'activité « réception matières », la MOE de rang 1 donne ainsi des contraintes à respecter pour la MOE de rang 2 lors des phases de développement et de réception des livrables.

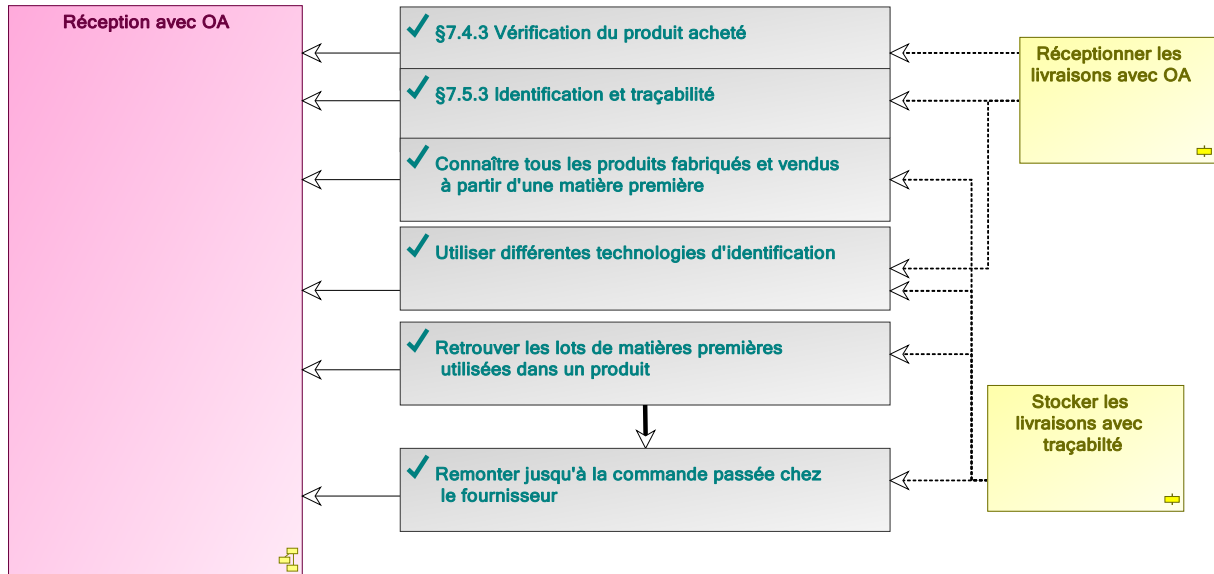


Figure 61. Diagramme d'exigence pour la "réception matières avec OA à l'AIPL"

2.2.3 (Conceptual-How du SAF), Diagramme de mise en œuvre du processus

Ce diagramme (Figure 62) décrit la mise en œuvre du processus dans l'organisation par un enchaînement de procédures du « SAF_eProduction » focalisé sur la réception matières . Chaque procédure est placée sous la responsabilité d'un acteur (de type structure) de l'entreprise. Ce diagramme a été bâti à partir des procédures déjà existantes de l'AIPL et décrit le processus de réception matières dans du « SAF_eProduction ». Ces procédures ont été spécifiées au préalable par la direction de l'AIPL en vue d'améliorer le fonctionnement de l'AIPL. Elles ont été par la suite le support de l'obtention de la certification ISO9001:2000.

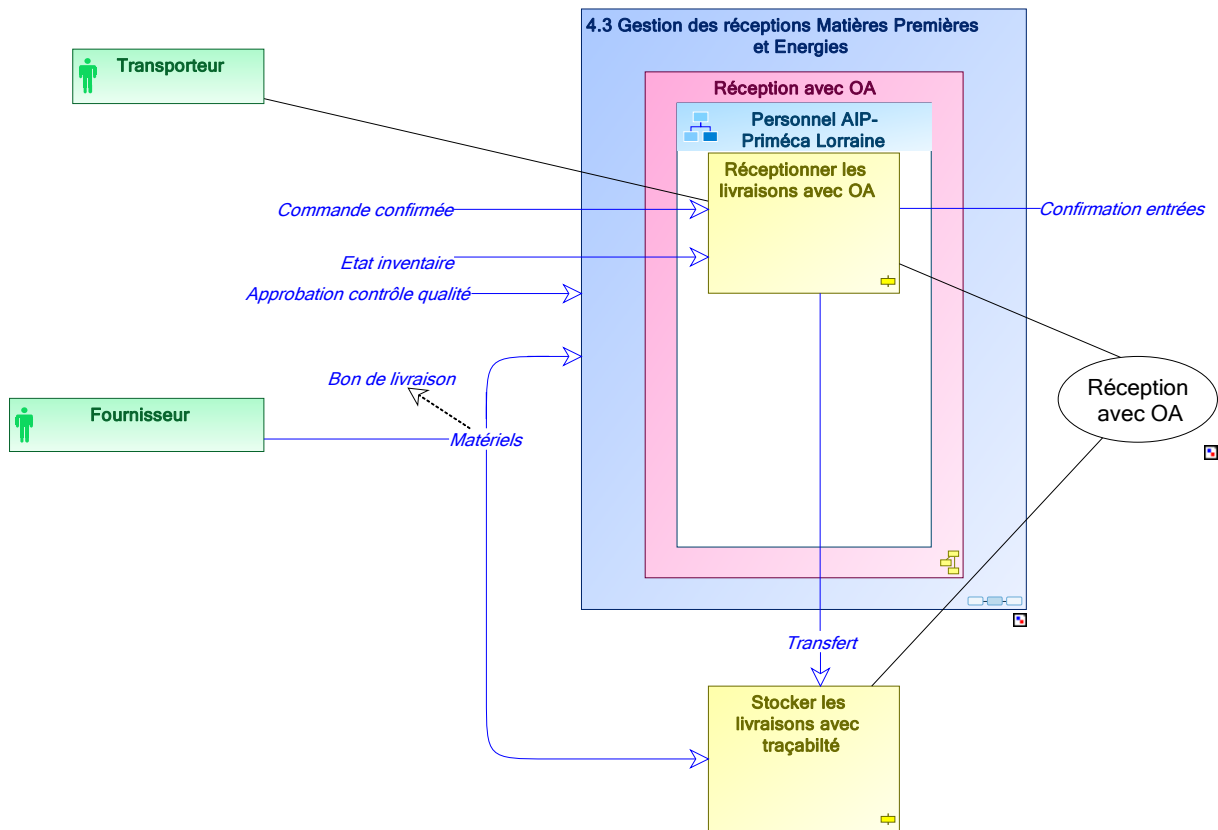


Figure 62. Diagramme de mise en œuvre de processus

2.2.4 (Conceptual-Who du SAF), organigramme des acteurs & diagramme de cas d'utilisation

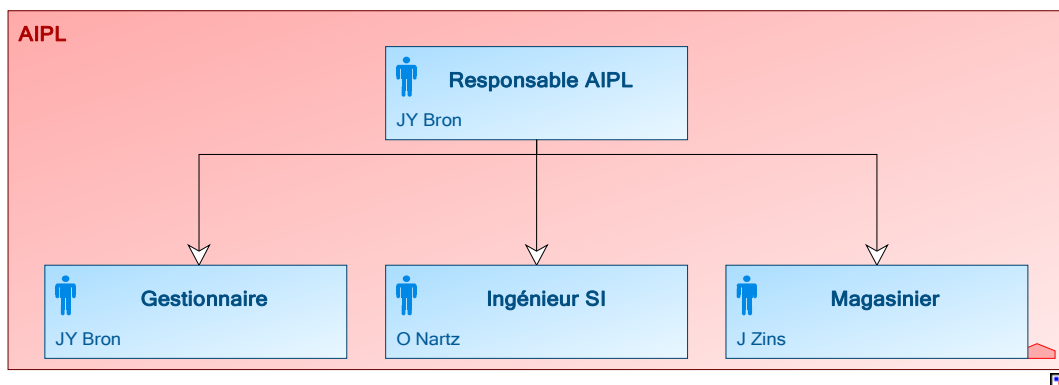


Figure 63. Organigramme opérationnel pour le «SAF_tracer_produit» à l'AIPL

Ce diagramme (Figure 63) fait suite à l'organigramme organisationnel de la MOA du « SAF_eProduction » focalisé sur la réception matières . La MOE de rang 1 du « SAF_eProduction » a simplement associé une personne physique de l'AIPL aux fonctions précitées.

Le diagramme de cas d'utilisation (Figure 64) spécifie les services (cas d'utilisation) et leurs interfaces qui devront être développés par la MOE de rang 2.

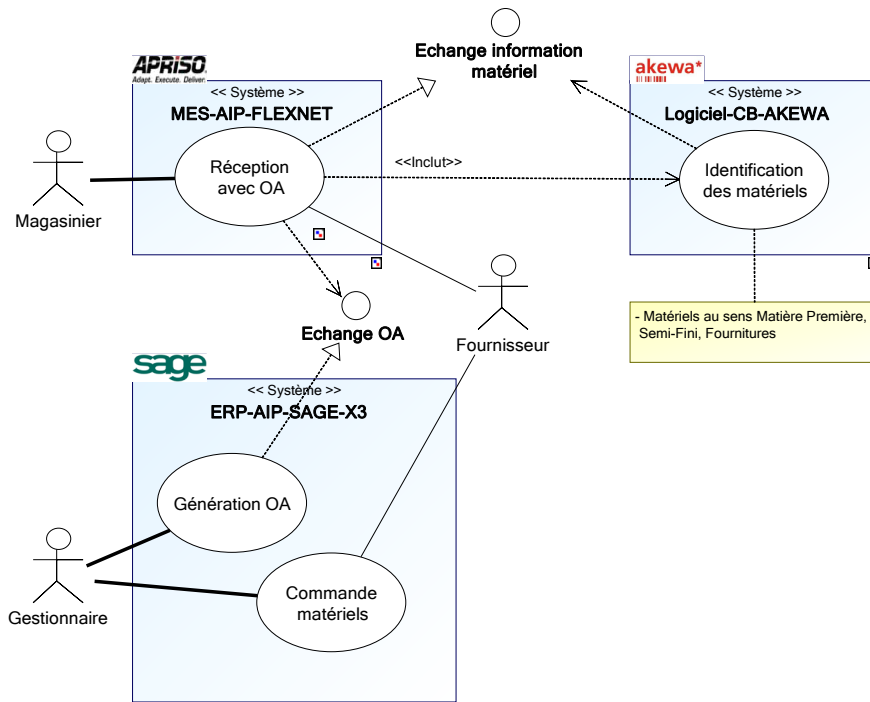


Figure 64. Diagramme de cas d'utilisation du SAF «SAF_tracer_produit» à l'AIPL

2.3 Spécification du SAF « système de traçabilité du produit à l'AIPL » : processus de la réception matières (MOE de rang 2)

Aussi, pour illustrer la partie « implémentation réelle sur site » du SAF de traçabilité d'un produit à la réception matières, nous proposons de montrer les diagrammes relatifs au MES (APRISO-FlexNet93).

Aussi nous nous plaçons dans l'hypothèse où la MOE de rang 1 communique les invariants de modélisation (niveaux « contextuel » & « conceptuel ») à la MOE de rang 2. La MOE de rang 1 devient récursivement en quelque sorte la « MOA de rang 2 » dont le rôle est de piloter dans notre cas 3 sous-projets : l'ingénierie de SAGE-X3 (ERP), l'ingénierie d'APRISO-FlexNet93 (MES) et l'ingénierie de l'application d'étiquetage. Chacun de ces 3 projets doit respecter les exigences de la MOA de rang 1 et reprendre les modèles (invariants) qui les concernent afin d'être implémenter. Le cadre de Zachman ci-dessous représente l'ingénierie du MES (FlexNet93) de l'AIPL (Figure 65). On y retrouve les invariants de modélisation de la MOA et de la MOE de rang 1.

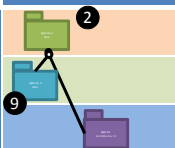
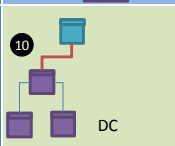
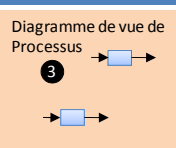
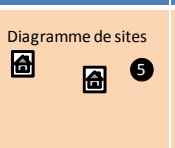
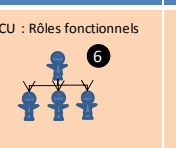
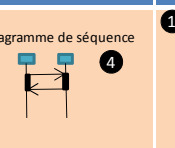
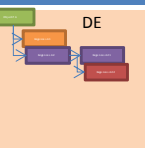
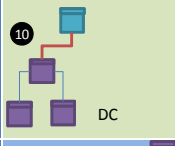
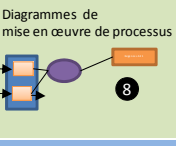
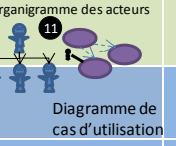
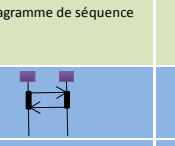
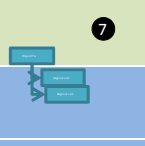
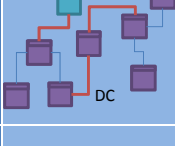

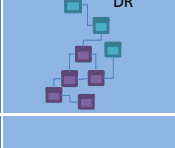
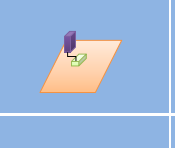
	What	How	Where	Who	When	Why
Contextual (MOA)	 	Diagramme de vue de Processus 	Diagramme de sites 	DCU : Rôles fonctionnels 	Diagramme de séquence 	
Conceptual (MOE-1)		Diagrammes de mise en œuvre de processus 		Organigramme des acteurs  Diagramme de cas d'utilisation 		
Logical (MOE-2)		Diagramme d'archi. applicative 				
Technical (réalisateur)			Diagramme d'archi. technique 			
Operational (Opérationnel)	DDL SQL	Fichiers plats XSLT XML				

Figure 65. Zachman du SAF « système de traçabilité du produit en réception matière» (niveaux "Logical", "Technical", "Operational").

Dans notre cas, le MES (APRISO-FlexNet93) hérite des invariants de modélisation du SAF global. La MOE de rang 2 instancie les cellules du Zachman particulier des niveaux « Logical » à « operational » (Figure 66).

Le MES (APRISO-FlexNet93) se voit donc associé à un cadre de modélisation de Zachman particulier avec les invariants de modélisation hérités des Zachman de la MOA et de la MOE de rang 1.

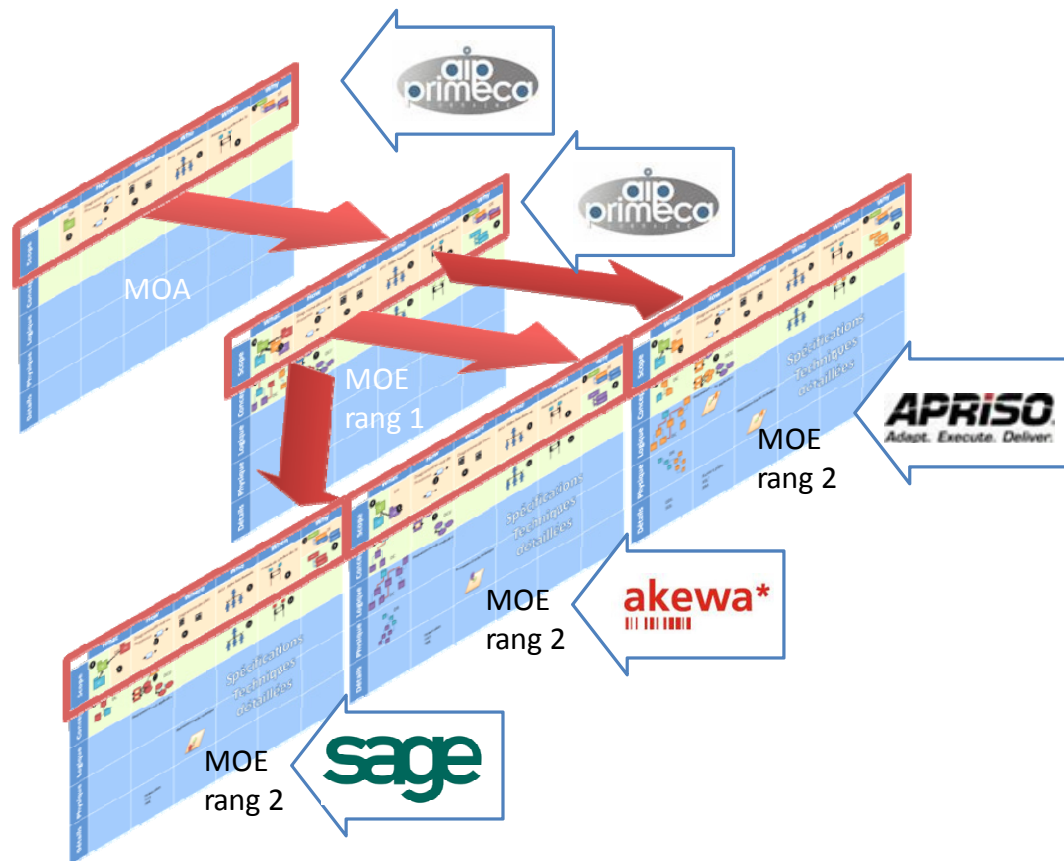


Figure 66. Exemple de projection des invariants de modélisation de la MOA du niveau "CONTEXTUAL" vers les l'ingénierie de la MOE de rang 1 qui le relaie à la MOE de rang 2.

2.3.1 (Contextual-Why du MES), diagramme d'exigence du MES

Ce diagramme des exigences du MES hérite des invariants de modélisation de la MOA qui le concerne (pour la réception matière) à savoir les objectifs et les exigences du diagramme des exigences du SAF (fond orange ; Figure 67). La partie sur fond vert de la même figure, représente les exigences d'interopérabilité intrinsèques au MES.

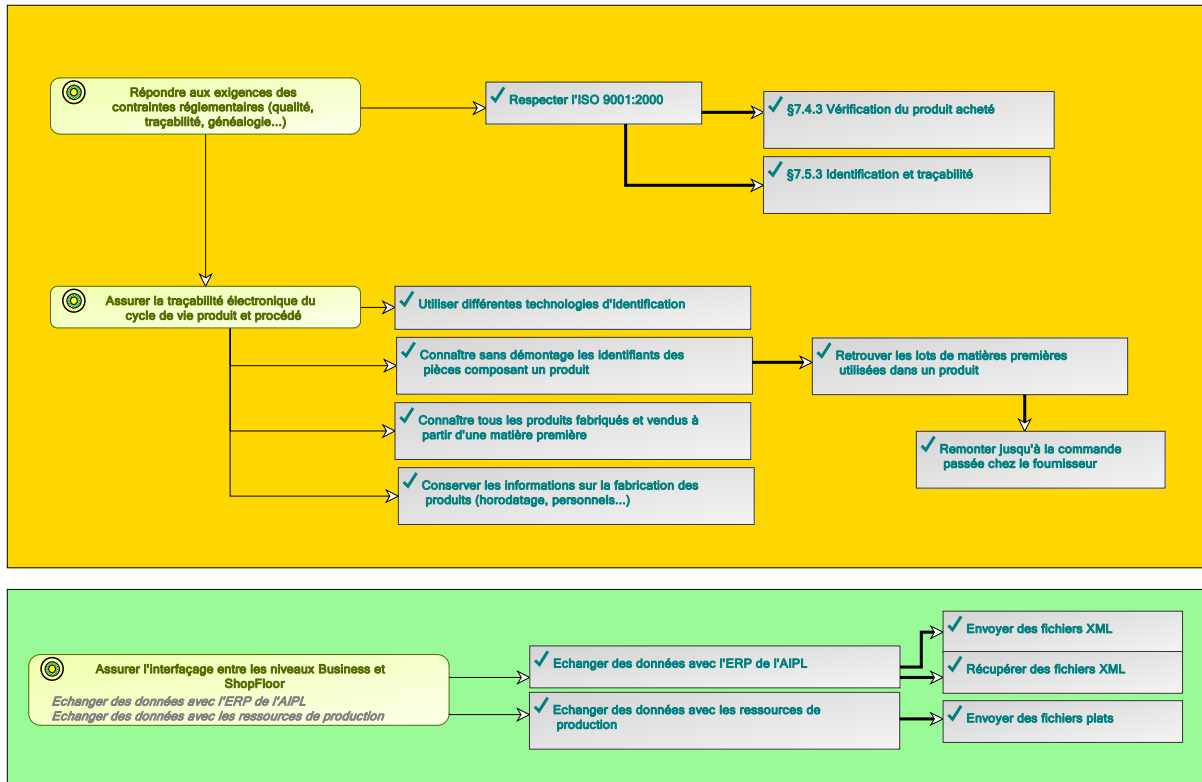


Figure 67. Diagramme d'exigences du SI MES pour la réception matières d'après l'héritage des exigences du SAF global

2.3.2 (Contextual-what du MES), diagramme de paquetage

Ce diagramme de paquetage est propre au MES APRISO-FlexNet93. Il représente les 3 classes d'objets qui permettent de créer des IHM⁴⁵ et des traitements adaptés à un besoin spécifique au travers du module de développement « ProcessBuilder » du MES APRISO-FlexNet93. D'un point « Flexnet », le process est un enchaînement d'opérations qui inclut des étapes dans lesquelles on retrouve entre autres des fonctions. Cette structure est propre au MES qu'utilise l'AIPL pour la mise en place de la traçabilité de la réception matières (Figure 68).

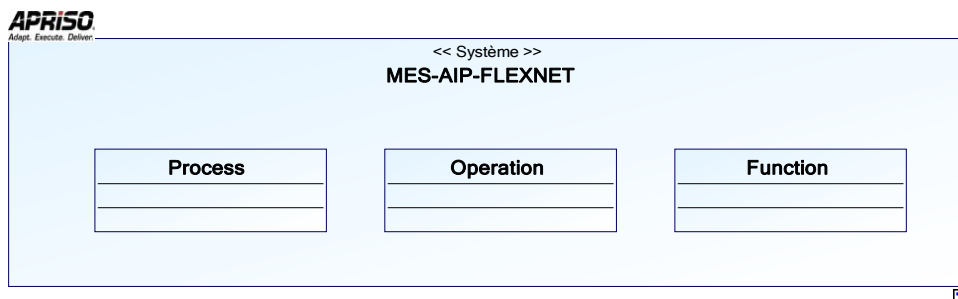


Figure 68. Diagramme de paquetage du SI MES pour la réception matières à l'AIPL

⁴⁵ IHM : Interface Homme Machine

2.3.3 (Conceptual-when du MES), diagramme de séquence

Ce diagramme de séquence (Figure 69) a comme objectif de spécifier les messages et l'ordre dans lequel ils sont échangés entre le magasinier et l'opération standard relative à la réception matières sur bon de commande. Ce diagramme nous a permis de construire les écrans proposés à l'opérateur suivant 3 scénarii distincts :

- Sérialisation par produit,
- Sérialisation pour un lot dans un container,
- Sérialisation pour un lot réparti dans plusieurs containers.

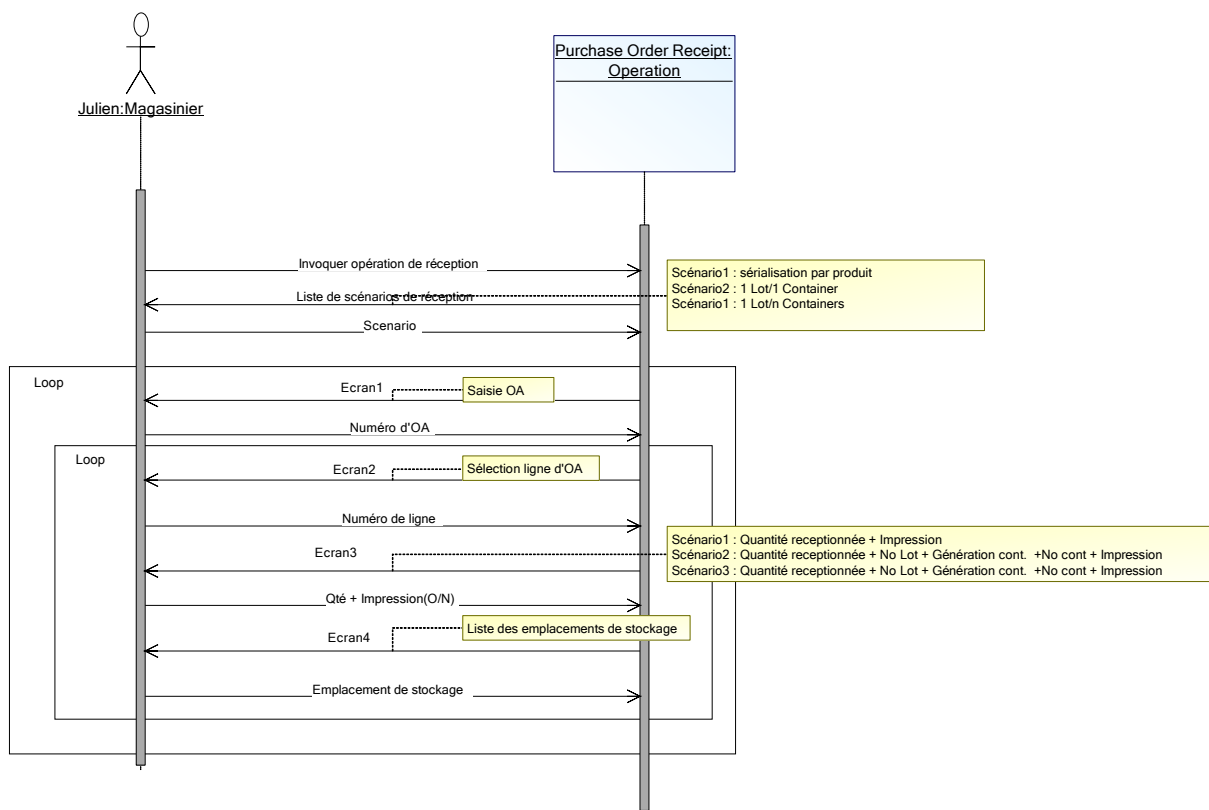


Figure 69. Diagramme de séquence du SI MES de la réception matières à l'AIPL

2.3.4 (Conceptual-who du MES), diagramme de cas d'utilisation

En rapport avec le diagramme de décomposition de processus précédent, le cas d'utilisation est détaillé dans un diagramme de cas d'utilisation (Figure 70) sur lequel on retrouve le SI concerné (packaging), le magasinier (acteur) et les cas d'utilisation mis en œuvre lors de la « réception matières ».

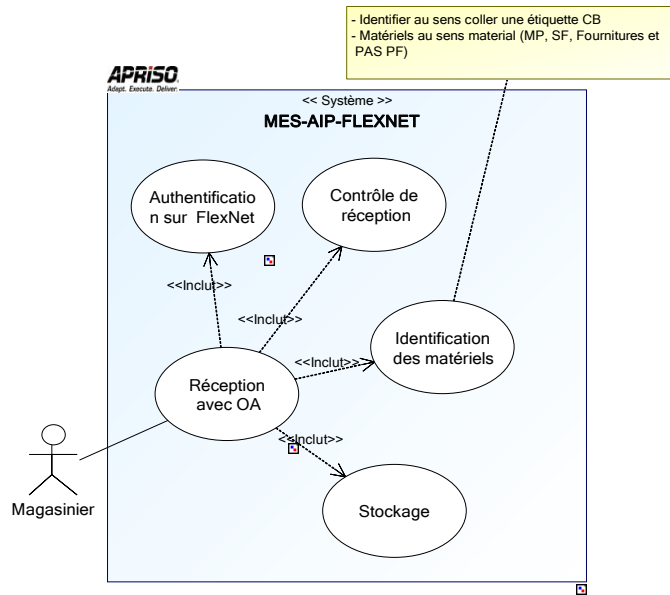


Figure 70. Diagramme de cas d'utilisation du SI MES de la réception matières à l'AIPL

2.3.5 (Logical-How du MES), diagramme d'architecture applicative

Dans ce diagramme (Figure 71), le modélisateur représente les applications concernées par le «SAF_tracer_produit» en réception matières (Flexnet et AKEWA) pour modéliser :

- l'impression d'une étiquette Code Barre au travers de messages échangés (informations étiquettes),
- les services à déployer (FlatFileProcessor, JobScheduler, AkewaTextASCIIService),
- les contraintes à respecter (périodicité « toutes les 5 sec. »).

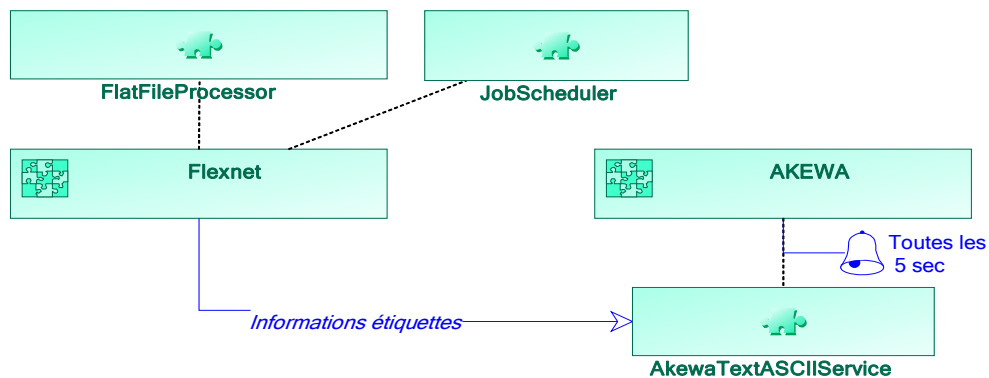


Figure 71. Architecture applicative entre le SI MES et le Logiciel AKEWA

2.3.6 (Technical-Where du MES), diagramme d'architecture technique

Par ce diagramme (Figure 72), le modélisateur décrit l'infrastructure informatique minimale qui supporte la procédure de « réception matières ».

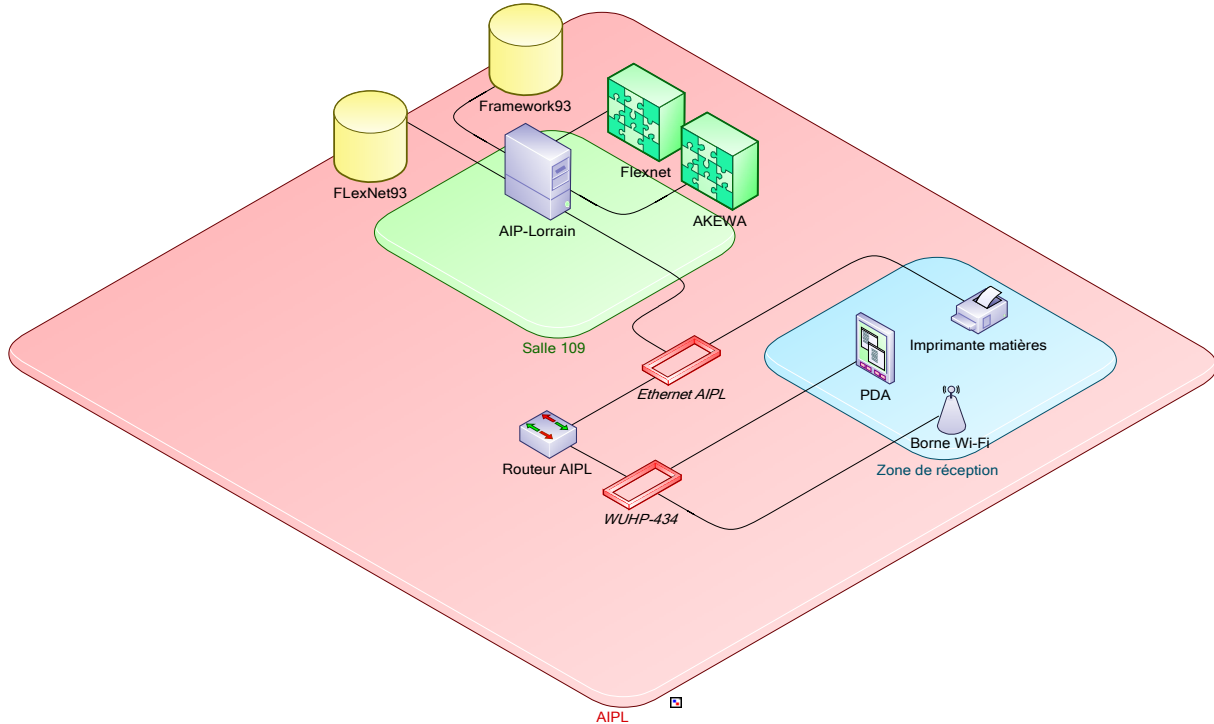


Figure 72. Diagramme d'architecture technique pour la réception matières à l'AIPL

2.3.7 (Operational-How du MES), fichiers plats, XML, XSLT,...

À partir des modèles sémantiques fournis par la MOE de rang 1, la MOE de rang 2 va faire correspondre les concepts relatifs à la réception matières entre le modèle sémantique de SAGE X3 et le modèle sémantique de FlexNet93. Ce travail de correspondance (ou mapping syntaxique) est facilité par le modèle sémantique du produit fournis lui aussi par la MOE de rang 1. Pour ce faire, la MOE de rang 2 doit créer le fichier de transformation **XSLT**⁴⁶ qui garantit la correspondance sémantique entre un flux **XML**⁴⁷ sortant d'un SI source (SAGE X3) et un flux XML entrant dans un SI cible (FlexNet93) (Figure 73).

La génération du code XSLT a été effectuée en créant tout d'abord le fichier XSD du fichier XML source de l'ERP avec le Logiciel XMLSpy©. Puis, le fichier de données XML source de l'ERP et son fichier XSD (Figure 74) ont été « mappés » avec le fichier XML cible du MES (Figure 75) dans le

⁴⁶ **XSLT** (eXtensible Stylesheet Language Transformation) : fichier faisant les correspondances (mapping) sémantiques entre les balises représentant les concepts dans les fichiers XML.

⁴⁷ **XML** (Extensible Markup Language) : format simple et très flexible utilisé pour stocker des informations structurées.

Logiciel MapForce© (Figure 76). Une fois les correspondances établies, MapForce© génère le fichier XSLT (§ Annexe 11).

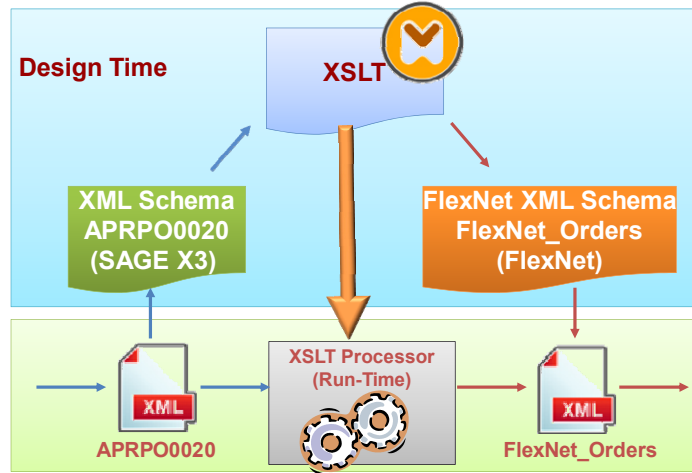


Figure 73. Mécanisme de transformation de fichier XML source (SAGE X3) vers un fichier XML cible (APRISO-FlexNet93)

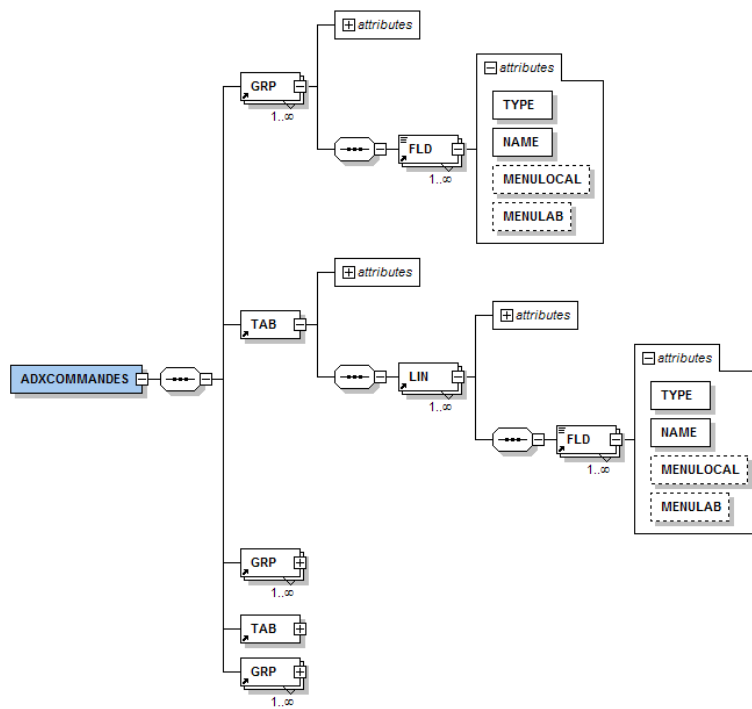


Figure 74. Structure XSD du fichier XML source pour la commande sous SAGE X3

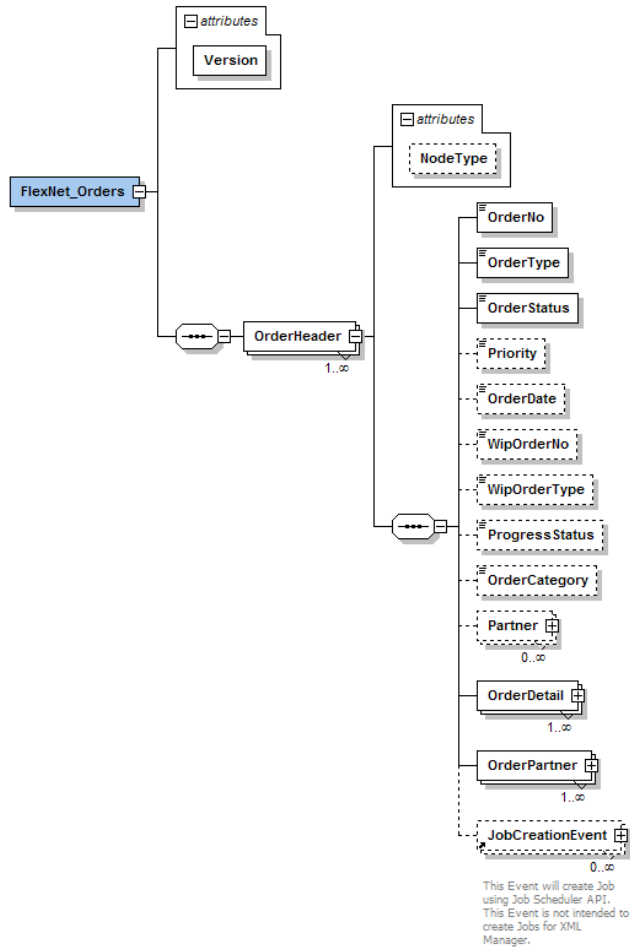


Figure 75. Structure XSD du fichier XML cible pour la commande sous Flexnet93

Tableau 24. Extrait des correspondances syntaxiques entre SAGEX3 et FlexNet93

SAGE X3	Opérateur	FlexNet93
Attributs		Attribut
POHNUM	=	OrderNo
ORDDAT	Fct Apriso_Time{[YYYYDDMM],[HHMMSS]}	OrderDate
BPSNUM	=	Account
POH2_1	=	OrderDetail, Job
QTYUOM	=	QuantityOrdered, OrderQuantity
UOM	=	OrderedUomCode, UomCode
PRHFCY	=	ToFacility
FCYADD	=	ToWarehouse
ITMREF	=	ProductNo
EXTRCPDAT	=	ScheduledStartDate, DueDate
POHNUM	Concat ("ADXCOMMANDES - " POHNUM)	Description
...

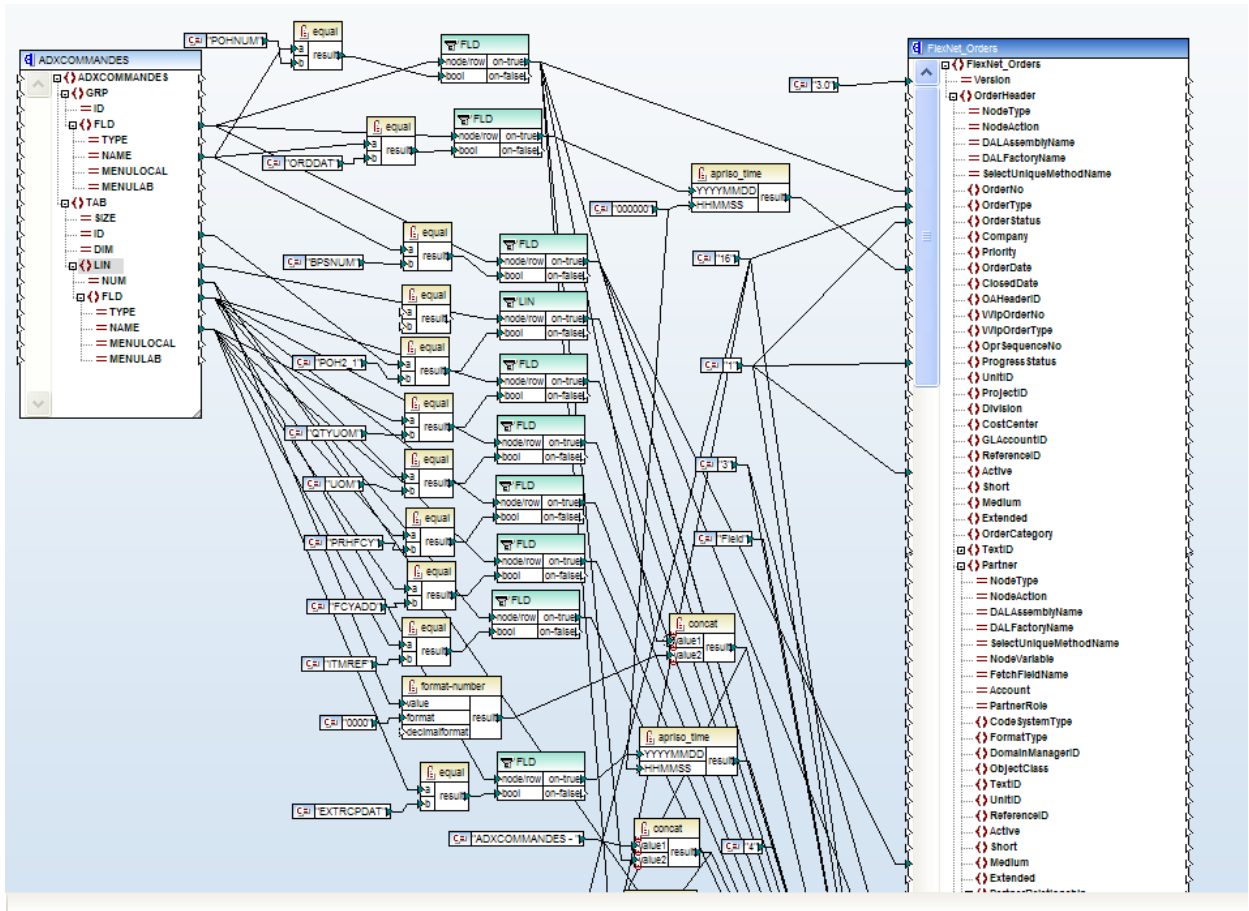


Figure 76. Extrait des correspondances entre les données des fichiers XML source (SAGE X3) et cible (FlexNet93)

Conclusion du chapitre 3

L'appropriation de la méthode a été relativement rapide par les modélisateurs. Il a fallu environ un mois temps plein pour assimiler la méthode et dégager les spécifications par la MOA et la MOE de rang 1. Les modélisateurs ont fait remarquer que la méthode leur a permis de les guider de manière rigoureuse dans le respect des exigences initiales et la cohérence des objets de modélisation. Le passage des exigences implicites aux exigences explicitement représentées dans le diagramme des exigences est une étape clef et fondamentale qui mérite d'y consacrer du temps. Tant que le diagramme des exigences n'a pas été approuvé par le donneur d'ordres, la MOA et les MOE ont été dans l'incapacité de spécifier quoique ce soit.

Une fois le diagramme des exigences approuvé, il a fallu environ 15 jours pour réaliser l'ingénierie de la réception matière. Il a été noté aussi que le travail de l'implémentation a grandement été facilité par le cadre rigoureux qu'impose le Zachman. Nous avons remarqué aussi que plus les modélisateurs progressaient dans leur modélisation, plus ils étaient efficaces et rapides dans leur tâche de modélisation. Les points de blocages ont surtout été relatifs aux problèmes techniques directement liés aux technologies particulières des SI et à leur configuration. Par contre, le fait d'avoir canalisé par les modèles le développement du SAF, la MOE de rang 2 a pu cibler plus facilement les points techniques sur lesquels ils devaient opérer. Ceci a permis entre autre de monter un stage de formation avec un fournisseur de progiciels sur des aspects techniques très ciblés et donc de spécifier plus facilement les besoins de formations internes.

Un des avantages de l'atelier Logiciel MEGA Modelling Suite est de fournir une plateforme de modélisation avec un contrôle des droits sur la base de données des modèles du SAF. Même si cet aspect n'a pas été abordé dans ces premières phases de modélisation, il semble opportun à l'avenir de mettre en place des droits particuliers suivant que l'on est MOA, MOE de rangs 1 ou 2. De même il serait souhaitable que certains modèles, comme les invariants de modélisation, soient « verrouillés » en modification et en suppression par la MOA et la MOE de rang 1. Dans le cadre d'une modélisation extra AIPL, les invariants de modélisation pourront ainsi être importés dans un environnement MEGA différent sans risque pour les modèles futurs. Par exemple, si une partie de la fabrication est délocalisée, la MOA enverra au MOE de rang 2 du nouveau site un fichier à importer dans son propre environnement MEGA sur son propre ordinateur en local. Une autre solution serait que chaque site constitutif se voit attribuer un accès via Internet à la base de données du projet du SAF sur MEGA. Dans la mesure où un projet met en activité des sites d'origines géographiques distribués internationalement, il sera préférable que les modèles de départ soient dans une langue consensuelle entre les parties prenantes. Ainsi, seuls les environnements de travail de l'atelier Logiciel MEGA sera traduit dans la langue maternelle du modélisateur.

CONCLUSION GÉNÉRALE & PERSPECTIVES

1 Conclusion générale

Les travaux de cette thèse se veulent être une contribution à l'**Ingénierie Système au travers d'une ingénierie basée** sur les modèles.

Cependant, nous avons noté que ce processus d'Ingénierie Système Basée sur les Modèles est relativement complexe car :

- Les SI se trouvent distribués dans un contexte d'entreprise étendue ou de réseaux d'entreprises. L'émergence et l'évolution des partenariats nécessitent la mise en œuvre d'interopérations de SI autonomes le plus rapidement possible.
- La multiplicité des ingénieries et de leurs SI, leurs aspects multi-métiers et multipoints de vue appartiennent à un même système d'entreprise ou réseaux d'entreprises, évoluant en permanence au sein d'un tout cohérent afin de réaliser un objectif commun et d'atteindre une performance globale,
- La coopération entre les SI internes à chaque entreprise ou entre des SI au sein d'un réseau d'entreprises se fait à différents niveaux d'échange, introduisant un facteur d'échelle (Kuras 2006) à prendre en compte pour la compréhension d'un système-entreprise ou d'entreprises, ainsi qu'une définition multi-échelles de ces systèmes (Bjerkemyr and Lindberg 2007b).

La complexité de l'ISBM et les problématiques présentées précédemment, nous amènent à proposer un cadre d'ingénierie dans lequel :

- Nous avons étudié le concept de SdS (Maier 1998; DeLaurentis and Callaway 2004; Boardman and Sauser 2006; Lane and Valerdi 2007) et tous les outils et modèles associés (Luzeaux and Ruault 2008b) afin de savoir modéliser l'ensemble hétérogène qu'est une entreprise ou un réseau d'entreprises ainsi que les interactions dynamiques se déroulant entre leurs constituants.
- Nous avons adopté une démarche de récursivité (Gamma et al. 1995) dans le but de considérer récursivement les coopérations au sein d'un système-entreprise ou d'un système d'entreprises, afin de réduire la complexité de compréhension de ces grands systèmes composés d'autres systèmes.
- Nous avons choisi une Ingénierie Système Basée sur les Modèles (ISBM) (Wymore 1993) pour la portabilité de la méthode sur de nouveaux projets lorsqu'il est potentiellement possible de reprendre les grandes étapes d'un processus de conception antérieure et d'en changer certaines données.
- Nous avons opté pour le cadre de modélisation de Zachman (Sowa and Zachman 1992) dans la mesure où cette pratique générique fournit un cadre d'architecture structuré, multi-points de vue qui guide l'architecte système dans son acte de modélisation de grands systèmes complexes.

- Et enfin, nous avons utilisé le paradigme du Système Contrôlé par le Produit (SCP) (Morel et al. 2003 ; Panetto 2006) afin de considérer ce produit, comme le pivot des unifications des différents modèles d'ingénierie du « Système Pour faire » ce « Système A Faire » (Berio et al. 2003; Berio et al. 2004).

Aussi nous nous sommes focalisés sur les problématiques d'interopérabilité entre des SI constitutifs de type COTS au service d'un sur-système bénéficiaire (un Système de Systèmes – SdS), un SdSI dans notre cas. Nous avons appliqué notre outil-méthode ISBM à l'ingénierie d'une relation d'interopération B2M pour définir un système de traçabilité de produit dans un contexte crédible de scénario PLM.

Nous comptons éprouver notre proposition dans le contexte plus large de la modélisation du SI de l'Université Henri Poincaré de Nancy dans le cadre de notre mission en tant qu'Ingénieur de Recherche. Cela nous permettra de mieux mesurer le chemin à parcourir pour rendre véritablement opérationnel notre outil-méthode, dans le cadre d'une relation d'interopération centrée sur l'étudiant.

Nous avons aussi été quelque peu frustré, bien que nous en ayant exploré certains aspects, de ne pouvoir concrétiser la relation d'interopération d'un produit-actif tangible synchronisant les applications de gestion (systèmes) contribuant à ses transformations successives.

Nous proposons en perspective une piste de réflexion et des résultats partiels sur ces trois aspects.

2 Perspectives

2.1 Méthodes et outils de l'interopérabilité de SdS

Nous avons mesuré par son absence dans notre mise en œuvre, l'intérêt de disposer d'un indicateur quantitatif d'un indice d'interopérabilité d'un SI (d'un SdSI) pour apprécier le degré de couplage entre le SI du Système-Entreprise et les Systèmes-COTS.

En 2008 le groupe de travail sur les SdS de l'AFIS⁴⁸, auquel nous contribuons, a proposé un outil qui permet d'établir une vue synthétique des relations d'interopérabilité entre les systèmes constituant un SdS : la matrice de couplage N² (Autran et al. 2008). La matrice N2 caractérise le SdS et les cellules en diagonale ses systèmes contributeurs (Tableau 25).

Cet outil recense les couplages qui existent entre les systèmes et met en évidence les systèmes qui sont des nœuds d'interopération les plus sensibles puisqu'ils interopèrent avec de très nombreux systèmes. Ces systèmes « critiques » ou « sensibles » devront faire l'objet d'une attention particulière en terme de défaillance car ils peuvent fortement perturber le SdS au comportement relativement stable (Figure 77). Néanmoins la particularité majeure d'un SdS est son auto adaptation face à des perturbations extérieures ou intérieures.

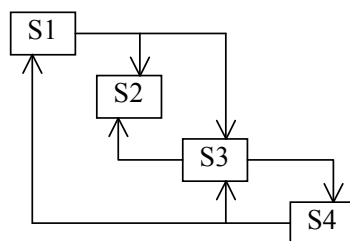


Figure 77 Systèmes couplés.

Tableau 25 Matrice de couplage N².

S1	Flow S1 → S2	Flow S1 → S3	
	S2		
	Flow S3 → S2	S3	Flow S3 → S4
Flow S4 → S1		Flow S4 → S3	S4

Cet outil est très utile car il donne une vision synthétique de l'état des lieux des couplages entre des systèmes qui coopèrent au sein d'un système beaucoup large. On peut imaginer des extensions à cette matrice. La première serait d'y ajouter une 3^{ème} dimension qui symboliserait les aspects récursifs des systèmes. C'est-à-dire que dans l'exemple (Figure 77) ci-dessus, n'importe quel système (S1,...S4) peut lui-même être composé de *n* systèmes qui doivent à leur tour coopérer pour répondre aux objectifs et

⁴⁸ AFIS : Association Française d'Ingénierie Système, <http://www.afis.fr>

exigences du système de niveau supérieur et faire l'objet de sa propre matrice N^2 . La deuxième extension serait d'associer une autre méthode qui quantifierait le degré d'interopérabilité du SdS en fonction du type d'interopérabilité comme le propose Ford avec son indice **i-Score** d'interopérabilité (Ford et al. 2007).

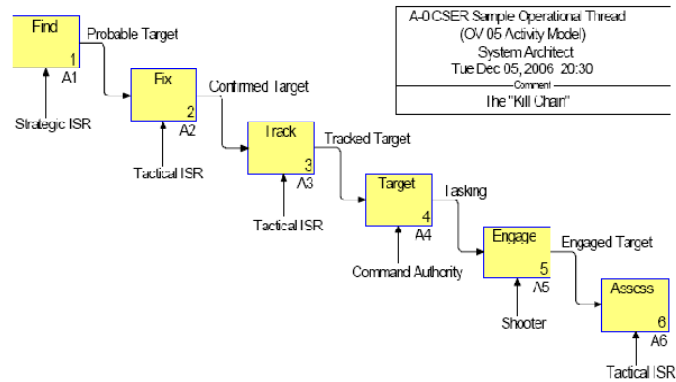


Figure 78 Exemple d'un scénario proposé par Ford et al. (IDEF0).

Les auteurs de **i-Score** ont probablement été les premiers à définir une méthode strictement quantitative pour mesurer l'interopérabilité d'un réseau hétérogène de systèmes dans le cadre d'opération d'appui de ces systèmes. **i-Score** calcul un degré d'interopérabilité absolu à partir d'un diagramme d'activité mettant en œuvre un ensemble de systèmes en interaction (Figure 78). Les auteurs proposent une méthode relativement simple basée sur des calculs matriciels. Leur méthode en 6 étapes, définit en premier le degré d'interopérabilité effectif, établit la différence entre un degré d'interopérabilité optimum et celui calculé précédemment. Cette différence permet de définir une valeur de l'effort nécessaire à fournir pour obtenir un niveau optimum. La matrice d'interopérabilité optimale est complétée en fonction des potentialités qu'on n systèmes à interopérer : « upgrade able » comme le citent ses auteurs. Les limites actuelles de **i-Score** se posent quand un système a de multiples activités en parallèle, mais aussi quand une activité est supportée par de multiples systèmes différents. De plus, les auteurs soulignent que **i-Score** ne permet pas encore de décrire le degré d'interopérabilité sur plusieurs niveaux : protocole, application, système, organisation.

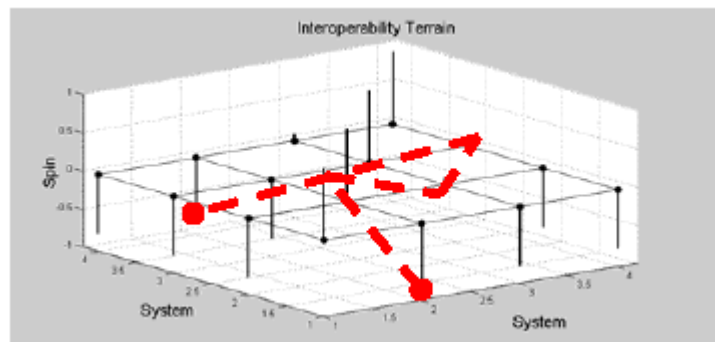


Figure 79 Exemple d'interopérabilité de terrain vu par (Ford, Colombi, Graham & Jacques, 2007)

Ainsi, suivant le type d'interopérabilité étudié du SdS (technique, syntaxique, sémantique,...), la matrice de couplage N^2 associée à la méthode de quantification du degré d'interopérabilité de (Ford et al. 2007) serait un outil très utile au pilotage de l'interopérabilité de grands systèmes complexes. Aussi, cette mesure du degré d'interopérabilité peut devenir un indicateur majeur lors de l'ingénierie du SdS ou lors de son pilotage pour son maintien opérationnel et mérite que l'on s'y intéresse plus en profondeur.

2.2 Le produit-actif pivot de l'interopération des ingénieries au sein d'un SdS

L'outil-méthode que nous proposons nous semble adapté pour les applications de type PLM (Terzi 2005) notamment dans le cadre de reconfiguration de réseaux d'entreprises qui impacte les exigences d'interopération entre systèmes existants.

La vision de la relation d'interopération via le produit-actif au sens du paradigme SCP dans le but de synchroniser les « Systèmes À Faire » est admise dans la communauté, notamment au travers du projet Européen 'FP6/PABADIS'PROMISE' (PABADIS'PROMISE 2006).

Dans ce sens, il nous a semblé qu'il serait intéressant de considérer un Produit-actif comme un système qui embarque non seulement l'ingénierie de son Système À Faire et l'ingénierie de Système Pour Faire.

Ainsi ce Produit-actif pourrait devenir porteur des invariants de modélisation qu'il pourrait diffuser **diachroniquement** de site en site. De la sorte, chacun des sites pourra **synchroniquement** se baser sur ces invariants de modélisation pour modéliser, concevoir ou configurer leur propre SI avec leur propre ingénierie. Aussi, pour représenter symboliquement ces invariants de modélisation portés par les niveaux « Contextual » et « Conceptual » de Zachman, nous les considérons comme deux couches supplémentaires enrobant le noyau du produit Holon qui compte déjà la couche correspondant à ses propriétés intrinsèques. Ainsi on retrouve le noyau de matière du produit recouvert par 3 couches : la première couche représentant les informations intrinsèques au produit, la deuxième et la troisième couches représentant respectivement les informations de modélisation de niveaux « Contextual » et « Conceptual » de l'interopérabilité du « Système A Faire » (Figure 80).

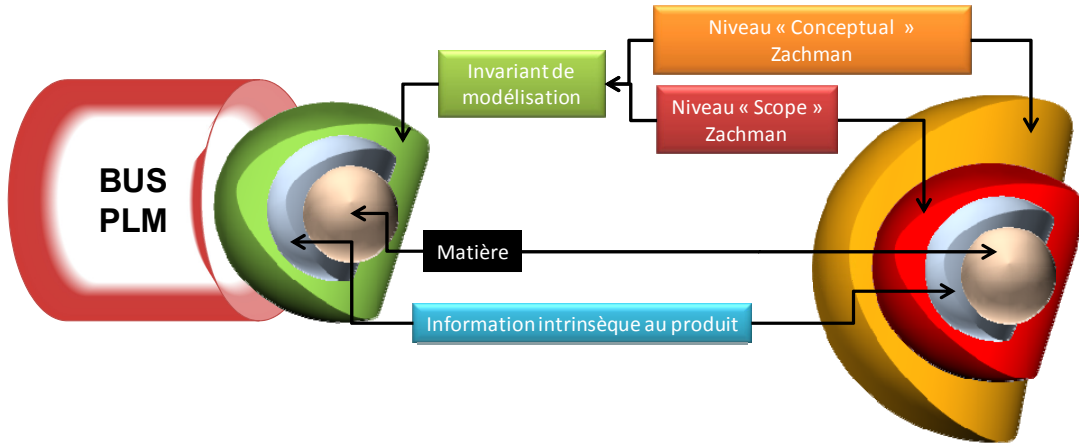


Figure 80. L'invariant de modélisation : niveaux "Contextual" et "Enterprise Model" du Zachman

De manière récursive, nous proposons que chaque SI soit associé à une ISBM basée sur un cadre de Zachman particulier dont les deux premiers niveaux (« Contextual » et « Conceptual ») seront hérités récursivement du cadre de Zachman du « Systèmes A Faire » parent. L'héritage s'opère à chaque fois que le produit doit être en relation avec un SI. Les invariants de modélisation sont alors la base de l'ingénierie des SI constitutifs. Ces ingénieries sont rarement associées à un cadre de modélisation. Voilà pourquoi nous proposons que cette ingénierie puisse être guidée par les niveaux « Logical », « Technical » et « Operational » des Zachman du « Système A Faire » (Figure 81).

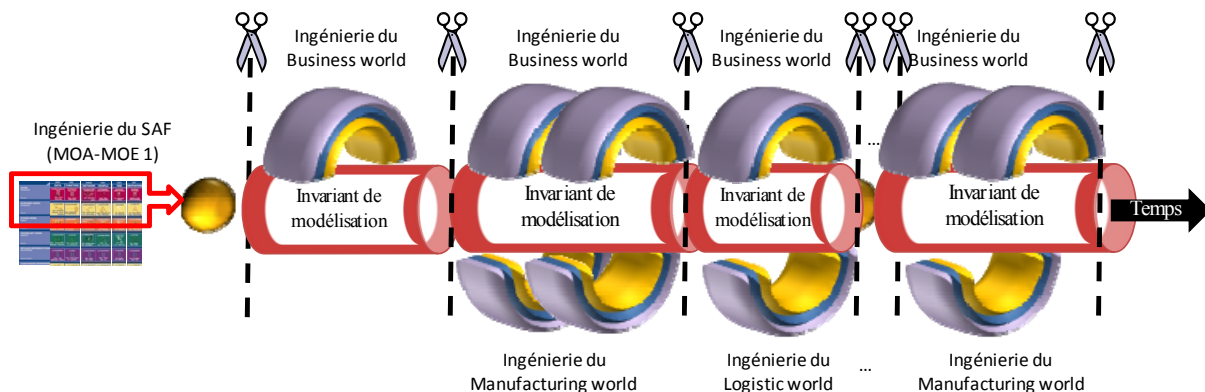


Figure 81. Modélisation diachronique de SdSI basée sur le cadre de modélisation de Zachman

Or, comme pour l'interopérabilité diachronique des SdSI, et toujours par analogie avec les problématiques d'interopérabilité des SdSI, l'ingénierie devient synchronique dans la mesure où chaque entreprise doit avant tout, individuellement, maîtriser les problématiques d'interopérabilité et donc sa propre modélisation (Figure 82). Ce tronçonnage synchronique de la modélisation du « Système A Faire », nous amène à considérer la **problématique d'interopérabilité dirigée par les modèles** (Baïna et al. 2008).

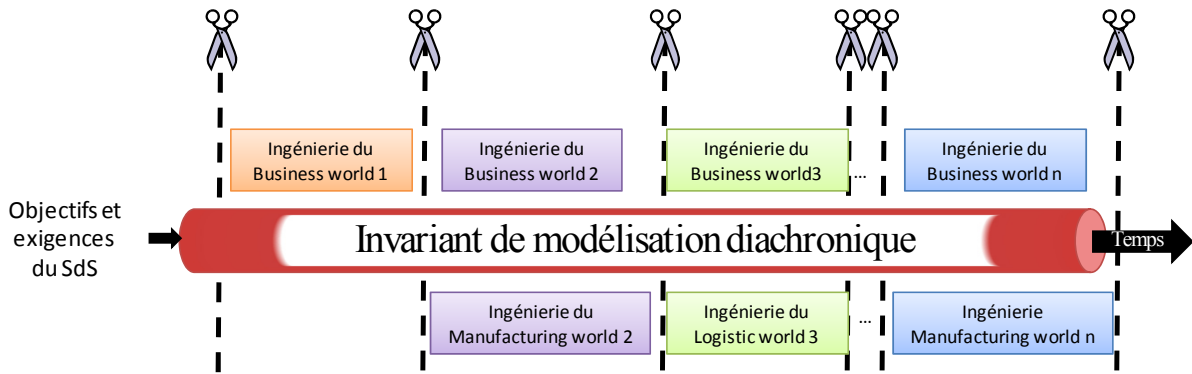


Figure 82. Ingénierie synchronique

Ainsi, nous nous retrouvons face à deux problématiques majeures : la **problématique d’interopérabilité du « Système A Faire »** et la **problématique d’interopérabilité du « système Pour Faire »**. Respectivement, l’une porte sur des problématiques d’interopérabilités entre les SdSI et l’autre porte sur des problématiques d’interopérabilité entre les modèles de ces mêmes SdSI.

Le rôle des modèles invariants, véhiculés par le produit au travers du BUS PLM, est de mettre à disposition des modélisateurs (MOE de rang 2) l’ensemble des objets de modélisation et l’ensemble des modèles sur lesquels ils vont se baser pour modéliser à leur tour leur SI (Figure 83).

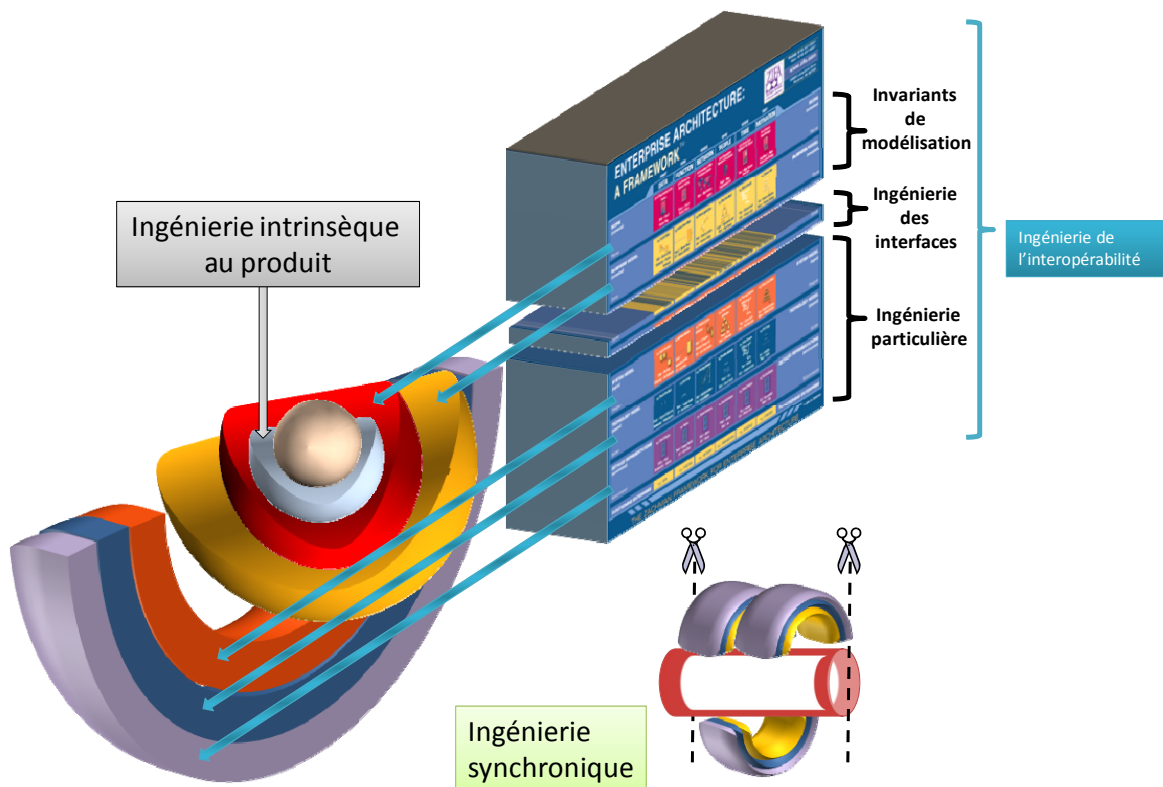


Figure 83. Couplage du produit-Holon et de l’ingénierie particulière d’un SI constitutif

De ce fait, l’apport principal de nos travaux de recherche dans cette thèse est de proposer le « Système Pour Faire » : une **Ingénierie Système Basée sur les Modèles** (ISBM). Son objectif est de favoriser

l'interopérabilité de SdSI avec le produit comme objet pivot des interopérabilités sémantiques et techniques.

Cette ISDM se base sur le cadre de modélisation **multi points de vue de Zachman** (Sowa and Zachman 1992) et **multi échelles** (Kuras 2006). Comme nous considérons les « Systèmes A Faire » comme des systèmes composés de sous-systèmes à l'image d'un système récursif, nous postulons que l'ingénierie de ces mêmes systèmes doit elle-même être récursive. De plus, comme nous associons la modélisation au cadre générique de modélisation de Zachman, nous nommons ce « Système Pour faire » : **une Ingénierie Système centrée sur le produit actif Basée sur les Modèles.**

2.3 Perspectives de mise en application de l'ISBM : L'Université Henri Poincaré de Nancy

Depuis la rentrée 2008 je suis en détachement à l'Université Henri Poincaré pour être l'Architecte des Systèmes d'Information de Nancy 1 et Chef de projet "sécurisation des systèmes d'information" dans le cadre de la loi du 10 août 2007 relative aux libertés et responsabilités des universités qui prévoit l'élargissement des compétences des universités.

Ma mission principale est de fournir un référentiel commun à l'ensemble des applications de gestion de l'Université et de garantir sa cohérence dans le temps.

Les problématiques rencontrées lors de mes travaux de recherche se retrouvent pleinement dans le SI de l'Université. Ainsi on dispose d'un grand nombre d'applications de gestion distribuées dans lesquels sont traitées, mémorisées, une quantité importante d'informations avec des doublons, des erreurs de saisie, des données manquantes, etc.. Bien que le SI contribue au bon fonctionnement de l'Université depuis plusieurs années, ce manque de cohérence induit pour le personnel une surcharge dans des tâches peut valorisantes et ne rend pas aux utilisateurs finaux un service optimum.

Un des objectifs de l'Université est de respecter le cadre législatif de la loi relative aux 'Établissements publics à caractère scientifique, culturel et professionnel' (loi n° 68-978 du 12 novembre 1968 d'orientation de l'enseignement supérieur) qui stipule qu'une Université a pour mission fondamentale l'élaboration et la transmission de la connaissance, le développement de la recherche et la formation des hommes. Voilà pourquoi nous avons choisi dans un premier travail (amorcé en septembre 2008) de modéliser les processus des grands métiers de l'Université : Scolarité, Recherche, Financier, etc. afin de mettre en évidence les fonctionnalités nécessaires autour du cycle de vie des utilisateurs impliqués dans la mission fondamentale de l'Université. En parallèle, une modélisation du SI est effectuée dans le but de lister le patrimoine applicatif et les services fonctionnels qu'il met à la disposition des personnels opérationnels et des usagers de l'Université (étudiants, fournisseurs, etc.).

L'étape suivante est de faire correspondre les besoins en services des utilisateurs avec les fonctionnalités des applications (réponses aux besoins) afin d'avoir un 'cliché instantané' du SI (le SI 'as-is' ad-hoc - Figure 84).

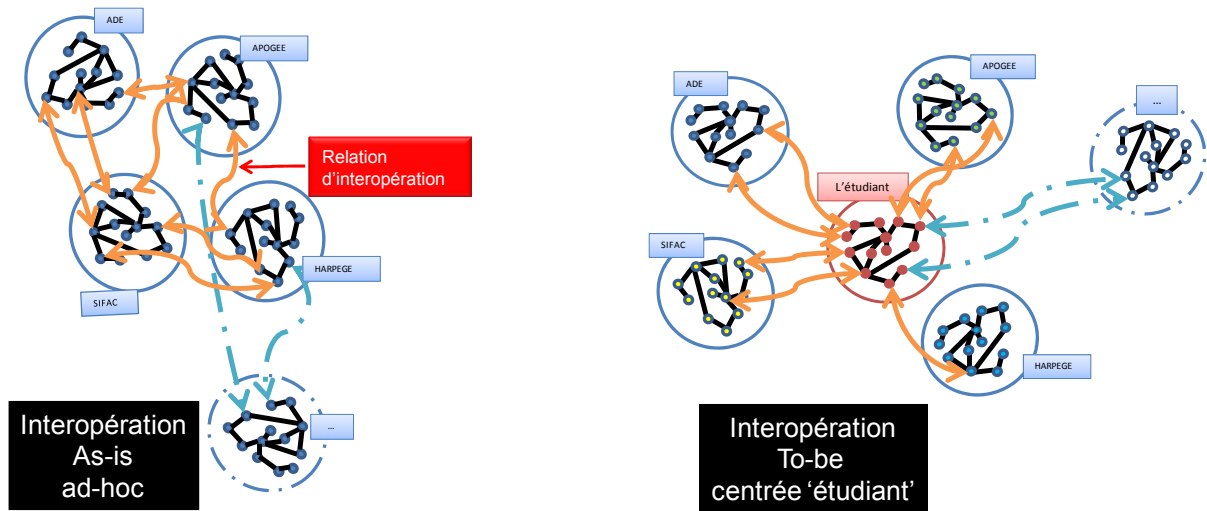


Figure 84. Paradigme SCP appliqué à l'Université Henri Poincaré de Nancy

Une solution envisageable pour l'optimisation des interopérations entre les applications de gestion du SI de l'Université serait de proposer un SI 'to-be' centré sur l'étudiant (Figure 84) au sens SCP du terme. Nous entendons par là que l'étudiant serait l'élément pivot de ces interopérations sur son cycle de vie d'étudiant (une sorte de 'Student Life cycle Management') à l'Université au travers des informations qui lui sont propres et qui sont partagées via des couplages faibles de type 'web services'. Le référentiel de données 'étudiant' serait ainsi un référentiel commun et central à l'ensemble des applications de gestion.

Acronymes

ADM : Architecture Development Method

AFIS : Association Française d'Ingénierie Système

AGATE : Atelier de Gestion de l'Architecture Technique des systèmes d'information de commandement

AIPL : Atelier Inter-établissements de productique de Lorraine

B2B : Business to Business

B2M : Business to Manufacturing

C4ISR : Computerized Command Control Communications Intelligence Surveillance

CAD : Computer Aided Design

CIM : Computation Independant modele

CIM : Computer Integrated Manufacturing

COTS : Commercial Off The Shelf

CRAN : Centre de Recherche en Automatique de Nancy

CRM : Customer Relationship Management

DCU : Diagramme de Cas d'Utilisation

DIMEG : Dipartimento di Ingegneria Meccanica e Gestionale – Politecnico di Bari, Italie

DoDAF : Department of Defense Architecture Framework

EAP : Enterprise Architecture Planning

ERP : Enterprise Resource Planning

FEAF : Federal Architecture Framework

FOV : Field of View

FOTS : Functions Off The Shelf

GERAM : Generalised Enterprise Reference Architecture and Methodology

IAM : Interoperability Assessment Model

IbM : Ingénierie basée sur les Modèles

IDM : Ingénierie Dirigée par les Modèles

IEEE : Institute of Electrical and Electronics Engineers

IHM : Interface Homme-Machine

INCOSE : International Council on Systems Engineering

IS : Ingénierie Système

ISBM : Ingénierie Système Basée sur les Modèles

IVVQ : Intégration du système, vérification du système, validation du système et Qualification

LCI : Layers of Coalition Interoperability

LCIM : Levels of Conceptual Interoperability Model
LISI : Levels of Information Systems Interoperability
MCISI : Military Communication & Information Systems Interoperability
MDA : Model Driven Architecture
MDE : Model Driven Engineering
MERISE : méthode MERISE est une méthode d'analyse, de conception et de réalisation de système d'information informatisé.
MES : Manufacturing Enterprise System
MOA : maître d'ouvrage.
MOE : maître d'œuvre.
MODAF : Ministry of Defence Architectural Framework
MOF : Meta-Object Facility
NIE : NATO Interoperability Environment
NMI : NATO Model Interoperability
NTI : Non - Technical Interoperability in Multinational Forces
NTIC : Nouvelles Technologies de l'Information et de la Communication
OA : Ordre d'Achat
OF : Ordre de Fabrication
OIAM : Organizational Interoperability Agility Model
OIM : Organizational Interoperability Model
OMG : Object Management Group
PDA : Personal Digital Assistant
PDM : Product Data Management
PIM : Platform Independent Model
PLM : Product Life cycle Management
PRM : Purdue Reference Model,
PSM : Platform Specific Model
QoIM : Quality of Interoperability Model
RAO : Réponse à l'Appel d'Offre
RFID : Radio Frequency Identification
SAF : Système A Faire
SPF : Système Pour Faire
SCP : Système contrôlé par le Produit
SdS : Système de Systèmes
SdSI : Système de Systèmes d'Information
SE : Système d'Entreprise
SI : système d'Information

SoSE : System of Systems Engineering
SoIM : System of Interoperability Model
SoSI : System of Systems Interoperability
SQL : Structured query language
SySML : SyStem Modeling Language
TEAF : Treasury Enterprise Architecture Framework
TISAF : Treasury Information System Architecture Framework
TLFi : Trésor de la Langue Française informatisé
TOGAF : The Open Group Architecture Framework
UML : Unified Modelling Language
UEML : Unified Enterprise Modelling Language
W3C : World Wide Web Consortium
XML : Extensible Markup Language
XSD : XML Schemas definition language
XSLT : eXtensible Stylesheet Language Transformation

Index des illustrations

Figure 1. Vision Ingénierie Système en Maîtrise d’Ouvrage de la modélisation en Entreprise adapté de [AFIS 2008]	9
Figure 2. Vision en Maîtrise d’Ouvrage de l’Ingénierie d’un Système-Entreprise	11
Figure 3. Product-driven manufacturing enterprise-wide control (Morel et al. 2007b)	13
Figure 4. Relationship between concepts in a Kuras’ multi-scale definition of a system by (Bjerkemyr and Lindberg 2007a)	15
Figure 5. Modèle sémantique : Pattern ‘Composite’ appliqué au Système-Entreprise	15
Figure 6. Ingénierie Basée sur les Modèles guidée par le cadre de modélisation de Zachman.....	16
Figure 7. Scénario général du cas d’étude	17
Figure 8. Approche didactique de l’IS : Ingénierie Système Basée sur les Modèles	17
Figure 9. Complexité de l’interopérabilité des Systèmes d’entreprise – des COTS (Auzelle et al. 2008).	27
Figure 10. Le produit vecteur de l’interopérabilité technique et sémantique entre le SI	29
Figure 11. SoS Operational Management Matrix (Boardman and Sausser 2006).....	32
Figure 12. Proposition de critères de SdS sur la base des travaux de (Maier 1998; DeLaurentis 2005; Boardman and Sausser 2006).....	33
Figure 13. Conceptualisation d’un système suivant des points de vus et des différentes résolutions (Kuras 2006).	42
Figure 14. Notion d’agrégation et de résolution d’un système.....	43
Figure 15. Modèle sémantique inspiré du pattern composite de (Gamma et al. 1995).	44
Figure 16. Conceptualisation des SI constitutifs à Faire "filtrés" par le pattern composite.....	45
Figure 17. Modèle sémantique : formalisation UML de systèmes « loosely coupled », « tightly coupled », « fully integrated ».....	47
Figure 18. Proposition d’un modèle sémantique de méta-modèle de Système de Systèmes centré sur le produit	48
Figure 19. Ingénierie Basée sur les Modèles	55
Figure 20. Couverture des normes relatives à la mise en œuvre d’une IS durant le cycle de vie d’un système. (AFIS : http://www.afis.fr/doc/normes/normes.html)	56
Figure 21. Cycle de vie générique d’un projet, inspiré de (Tucoulou 2008)	57
Figure 22. Formalisation du concept d’architecture proposé par (ISO/IEC-JTC1-SC7-N3160 2005).....	59
Figure 23. Influences et historiques des principaux cadres de modélisation (a) et langages de modélisation d’entreprise (b)	61
Figure 24. Cadre de Zachman (Sowa and Zachman 1992).	65
Figure 25. Vision Ingénierie Système en MOA de la modélisation en Entreprise, adapté de [AFIS 2008]	67
Figure 26. Modèle sémantique de l’ingénierie Système Basée sur les Modèles d’un réseau réticulaire de Système-entreprise.	70
Figure 27. Des modèles et cadres génériques vers une hiérarchisation de modélisation récursive. (Morel et al. 2007a)	73
Figure 28. Modélisation récursive multi-échelle de systèmes.	74
Figure 29. Relations entre les acteurs du système projet (SPF).	76
Figure 30. Zachman + MOA + MOE rang 1 + MOE rang 2	76
Figure 31. Types de recueil des exigences inspiré de ADN - R3 Management (Risk-Requirement-Regulatory Management)	78
Figure 32. Types de formalisme par cellule du cadre de modélisation de Zachman	79
Figure 33 Phases de l’ingénierie récursive basée par les modèles en relations avec la gestion du projet à faire.	81
Figure 34. Processus d’ISBM dans le Zachman	82
Figure 35. Spécification de la MOA dans le cadre de modélisation de Zachman	82

Figure 36. Spécification de la MOE de rang 1 dans le cadre de modélisation de Zachman.....	83
Figure 37. Spécification de la MOE de rang 2 dans le cadre de modélisation de Zachman	83
Figure 38. Alignement sémantique des concepts des SI par rapport aux concepts du produit	84
Figure 39 Zachman du SAF - spécifications de la MOA.	87
Figure 40 Zachman du SAF - spécifications de la MOE de rang 1.....	87
Figure 41 Zachman du système constitutif au SAF – exemple de spécifications de la MOE de rang 2.	88
Figure 42. Architecture de MEGA Modelling Suite (Source : <i>Mega International</i>).....	89
Figure 43. Extrait du Méta-modèle unifié de MEGA et nos extensions	91
Figure 44. Architecture applicative du prototype d'outillage de modélisation.....	92
Figure 45. Scénario global	98
Figure 46. Arborescence des systèmes bénéficiaires et constitutifs du cas d'étude.	99
Figure 47. Diagramme de collaboration du scénario global.....	100
Figure 48. Vue synchronique du SAF de traçabilité à l'AIPL.....	100
Figure 49. Zachman du SAF (niveau « contextuel »).....	103
Figure 50. Diagramme des exigences du projet "eProduction"	104
Figure 51. Diagramme de paquetage du SAF « e-Production »	105
Figure 52. Diagramme de processus du SAF « e-Production », dérivé du modèle fonctionnel PRM.....	106
Figure 53. Processus de réception matières dérivé du PRM	107
Figure 54. Diagramme de séquence relatif au sous processus 4.3 (Gestion des réceptions matières premières et énergies).....	107
Figure 55. Diagramme de sites du SAF « e-Production » focalisé sur la réception matières	108
Figure 56. Structure organisationnelle du SAF « e-Production » focalisé sur la réception matières.....	108
Figure 57. Zachman du SAF « système de traçabilité du produit à l'AIPL » (niveau "Conceptual").	109
Figure 58. Conceptualisation partielle du modèle sémantique du produit relatif au SAF « système de traçabilité du produit à l'AIPL » focalisé sur la réception matières	112
Figure 59. Conceptualisation partielle du modèle logique interne à SAGE X3.....	112
Figure 60. Conceptualisation partielle du modèle logique interne à FlexNet93	113
Figure 61. Diagramme d'exigence pour la "réception matières avec OA à l'AIPL".....	114
Figure 62. Diagramme de mise en œuvre de processus	115
Figure 63. Organigramme opérationnel pour le «SAF_tracer_produit» à l'AIPL.....	115
Figure 64. Diagramme de cas d'utilisation du SAF «SAF_tracer_produit» à l'AIPL.....	116
Figure 65. Zachman du SAF « système de traçabilité du produit en réception matière» (niveaux "Logical", "Technical", "Operational").....	117
Figure 66. Exemple de projection des invariants de modélisation de la MOA du niveau "CONTEXTUAL" vers les l'ingénierie de la MOE de rang 1 qui le relaie à la MOE de rang 2.	118
Figure 67. Diagramme d'exigences du SI MES pour la réception matières d'après l'héritage des exigences du SAF global.....	119
Figure 68. Diagramme de paquetage du SI MES pour la réception matières à l'AIPL.....	119
Figure 69. Diagramme de séquence du SI MES de la réception matières à l'AIPL.....	120
Figure 70. Diagramme de cas d'utilisation du SI MES de la réception matières à l'AIPL	121
Figure 71. Architecture applicative entre le SI MES et le Logiciel AKEWA	121
Figure 72. Diagramme d'architecture technique pour la réception matières à l'AIPL.....	122
Figure 73. Mécanisme de transformation de fichier XML source (SAGE X3) vers un fichier XML cible (APRISO-FlexNet93)	123
Figure 74. Structure XSD du fichier XML source pour la commande sous SAGE X3.....	123
Figure 75. Structure XSD du fichier XML cible pour la commande sous Flexnet93	124
Figure 76. Extrait des correspondances entre les données des fichiers XML source (SAGE X3) et cible (FlexNet93).....	125
Figure 77 Systèmes couplés.	131
Figure 78 Exemple d'un scénario proposé par Ford et al. (IDEFO).	132

Figure 79 Exemple d'interopérabilité de terrain vu par (Ford, Colombi, Graham & Jacques, 2007)	132
Figure 80. L'invariant de modélisation : niveaux "Contextual" et "Enterprise Model" du Zachman	134
Figure 81. Modélisation diachronique de SdSI basée sur le cadre de modélisation de Zachman	134
Figure 82. Ingénierie synchronique	135
Figure 83. Couplage du produit-Holon et de l'ingénierie particulière d'un SI constitutif	135
Figure 84. Paradigme SCP appliqué à l'Université Henri Poincaré de Nancy	138

Index des tableaux

Tableau 1. Récapitulatif des critères retenus de SdS sur la base des travaux de (Maier 1998; DeLaurentis 2005; Boardman and Sauser 2006)	33
Tableau 2. Types d'interopérabilité (Panetto 2006)	35
Tableau 3 Récapitulatif des principaux modèles d'interopérabilité avec leur référence bibliographique	36
Tableau 4. Étape 1 du diagramme commutatif : définition de S1 et S2.....	38
Tableau 5. Étape 2 du diagramme commutatif : définition du morphisme de plongement	39
Tableau 6. Étape 3 du diagramme commutatif : définition de la composition S1xS2	39
Tableau 7. Étape 4 du diagramme commutatif : définition du morphisme de projection.....	39
Tableau 8. Étape 5 du diagramme commutatif : définition du morphisme de plongement	40
Tableau 9. Étape 6 du diagramme commutatif : définition des propriétés des solutions envisagées.....	40
Tableau 10. Étape 7 du diagramme commutatif : définition du morphisme de projection.....	40
Tableau 11. Synthèse des cadres de modélisation et langages de modélisation.....	60
Tableau 12. Correspondance entre un système au sens de Kuras, un SAF et un SPF	67
Tableau 13. Proposition d'objets de modélisation pertinents du point de vue de l'IS et rattachés au cadre de modélisation de Zachman	86
Tableau 14 Séquencement de la modélisation des spécifications du SAF par la MOA.....	87
Tableau 15 Séquencement de la modélisation des spécifications du SAF par la MOE de rang 1	88
Tableau 16. Modification du méta-modèle de MEGA Modelling Suite	90
Tableau 17. Gestion du projet « eProduction » – extrait du Gantt du projet.	102
Tableau 18. Correspondances entre les exigences du SAF "e-Production" focalisé sur la réception matières et les applications informatiques existantes à l'AIPL	105
Tableau 19. Correspondance sémantique	110
Tableau 20. Correspondance sémantique entre le produit et l'application FlexNet93	110
Tableau 21. Tables, champs et concepts relatifs à la réception matières pour SAGE X3.....	111
Tableau 22. Tables, champs et concepts relatifs à la réception matières pour FlexNet93	111
Tableau 23. Tables, champs et concepts relatifs à la réception matières pour le produit.....	111
Tableau 24. Extrait des correspondances syntaxiques entre SAGEX3 et FlexNet93.....	124
Tableau 25 Matrice de couplage N ²	131

Références

- Ackoff, R.L. (1971). "Towards a system of systems concepts." *Management Science* (v17, n°11), pp 661-671.
- AFIS (2005). "Principaux aspect de l'Ingénierie Système." (Association Française d'Ingénierie Système, groupe de travail en IS), http://www.afis.fr/nav/gt/isys/prod/GTIS_4P_4FICHES_taille%20reduite.pdf
- AFIS (2009). "Découvrir et Comprendre l'Ingénierie Système. Version 3" (Association Française d'Ingénierie Système, groupe de travail en IS)p 126.
- AGATE (2002). "Atelier de Gestion de l'Architecture Technique des Systèmes d'Information de Commandement." (Délégation Générale pour l'Armement), <http://www.achats.defense.gouv.fr/article33349>.
- Alexander, C. (1999). "The Origins of Pattern Theory: The Future of the Theory, and the Generation of a Living World." *IEEE software* (v16, n°5), ISSN 0740-7459, pp 71-82.
- AMICE (1993). *CIMOSA: Open System Architecture for CIM* (2nd, rev. and ext. ed. edition). Springer-Verlag, ISBN 978-3540562566, 234 Pages.
- Anaya, V., Berio, G., Harzallah, M., Heymans, P., Matulevicius, R., Opdahl, A.L., Panetto, H. and Verdecho, M.J. (2008). "The Unified Enterprise Modelling Language: Overview and Further Work." in *Proceedings of the 17th IFAC World Congress, July 6-11, IFAC-PapersOnLine*, ed. (Seoul, Korea).
- ANSI/EIA-632 (1999). "Processes for Engineering a System." (Electronics Industries Alliance, Arlington, Virginia, USA).
- Architecture_Working_Group (1997). "C4ISR Architecture Framework version 2.0." D. Department of Defense, ed.
- ATHENA-IP (2007). "D.A4.2: Interoperability Framework and Services for Networked Enterprises." p 215.
- Autran, F., Auzelle, J.-P., Cattan, D., Garnier, J.-L., Luzeaux, D., Mayer, F., Peyrichon, M. and Ruault, J.-R. (2008). "Coupling Component Systems towards Systems of Systems." in *18th Annual International Symposium of INCOSE, 6th Biennial European System Engineering Conference, June 15-19, (Utrecht : Pays-Bas)*.
- Auzelle, J.P., Morel, G., Panetto, H. and Mayer, F. (2008). "Using Systems of Systems Engineering to Improve the Integration of Enterprise-Control Systems." *Insight Journal of INCOSE* (v11, n°3).
- Baïna, S. (2006). "Interopérabilité dirigée par les modèles : une approche orientée produit pour l'interopérabilité des systèmes d'entreprise." Thèse de Doctorat de l'Université Henri Poincaré. 7 décembre 2006: Université Henri Poincaré.

- Bařna, S., Panetto, H. and Benali, K. (2008). "Product oriented modelling and interoperability issues." in Enterprise Information Systems, Y. Manolopoulos, J. Filipe, P. Constantopoulos and J. Cordeiro, eds. (Berlin: Springer-Verlag Berlin), ISBN 978-3-540-77580 (Print) ISBN 978-3-540-77581 (Online), pp 293-308.
- Baptiste, P., Bernard, A., Bourrières, J.-P., Lopez, P., Morel, G., Pierreval, H. and Portmann, M.-C. (2007). "Comité d'Experts Productique : prospectives de recherche." 15 janvier 2007,
- Bar-Yam, Y. (2004). "A mathematical theory of strong emergence using multiscale variety." Complexity (v9, n°6), ISSN 1076-2787, pp 15-24.
- Bedau M (1997). Weak emergence,. Mind, causation and world, Blackwell Publishers, 375–399 Pages.
- Berio, G., Panetto, H. and Petit, M. (2004). "UEML : résultats et enjeux d'un Langage Unifié de Modélisation d'Entreprise. Conférence invitée. Actes de la conférence francophone de modélisation et simulation MOSIM'2004." 1-3 Septembre, (Nantes, France), ISBN 2-7430-0731-1, pp 17-29.
- Berio, G., Panetto, H., Petit, M., Benali, K. and al. (2003). "D3.2: Core constructs, architecture and development strategy, UEML TN IST - 2001 - 34229." March 2003,
- Bernus, P., Nemes, L. and Schmidt, G. (2003). Handbook on Enterprise Architecture. Springer, ISBN 3540003436, 9783540003434, 787 Pages.
- Berre, A. and al. (2007). "ATHENA deliverable A4.2, Specification of Interoperability Framework and Profiles, Guidelines and Best Practices." (Advanced Technologies for Interoperability of Heterogeneous Enterprise Networks and their Application), FP6-IST IP n°507849, March,
- Berry, B.J.L. (1964). "Cities as systems within systems of cities." Papers in Regional Science (v13, n°1), ISSN 1056-8190 (Print) ISSN 1435-5957 (Online), pp 146-163.
- Bjerkemyr, M. and Lindberg, B. (2007a). "The Effects of Limits to Human Abilities on System of Systems Properties." in The 7th International Conference on Complex Systems October 28-November 2, A. Minai, D. Braha and Y. Bar-Yam, eds. (Boston, MA).
- Bjerkemyr, M. and Lindberg, B. (2007b). "An engineering systems perspective on system of systems methodology." in in Proceeding of 1st Annual 2007 IEEE Systems Conference, 9-13 April, (Hawaii, USA), pp 1-7.
- Boardman, J. and Sauser, B. (2006). "System of Systems—the meaning of of." System of Systems Engineering, 2006 IEEE/SMC International Conference on, ISSN 1-4244-0188-7, pp 4-10.
- Bruneau, J.M., Pujos, J.F. and Bouyssonnie, J.P. (1992). Le management des connaissances dans l'entreprise: ressources humaines et systèmes d'information. Les Ed. d'Organisation, 122 Pages.
- C4ISR (1998). "Levels of Information Systems Interoperability (LISI)." (Washington, DC: Architectures Working Group report, Department of Defense), February 1998,

- Cariani P (1991). "Emergence and Artificial Life. Artificial Life II." in in Proceeding of SFI Studies in the Sciences of Complexity C. Langton, C. Taylor, J. D. Farmer and S. Rasmussen, eds. (Redwood City, CA: Addison-Wesley), pp 775-797.
- Carney, D., Fisher, D. and Patrick, P. (2005). "Topics in Interoperability: System-of-Systems Evolution." in Carnegie-Mellon Univ Pittsburgh Pa Software Engineering, Inst (Defense Technical Information Center), Technical Note, CMU/SEI-2005-TN-002,
- CEN/ENV-12204 (1995). "advanced manufacturing technology - systems architecture - constructs for enterprise modelling. – rapport technique, comité européen de normalisation (CEN), Bruxelles, Belgique, février."
- CEN/ENV-40003 (1990). "computer integrated manufacturing - CIM systems architecture framework for modelling. - comité européen de normalisation (CEN), Bruxelles, Belgique."
- CEN/ISO-11354-1 (2008). "Draft document. Requirements for establishing manufacturing enterprise process interoperability." in Part I: Framework for Enterprise Interoperability (N996 , CEN/TC 310/WG 1).
- CEN/ISSS (2005). "(European Committee for Standardization/Information Society Standardization System), http://www.support-eam.org/supporteam/CEN_ISSS_Workshop.asp." (Comité Européen de Normalisation), http://www.supporteam.org/supporteam/CEN_ISSS_Workshop.asp.
- Chapurlat, V. (2006). "Verification, Validation and Accreditation of Enterprise Models." in in Proceedings volume from the 12th IFAC International Symposium on Information Control Problems in Manufacturing (INCOM'2006), May 17-19, Elsevier-Science, ed. (St Etienne, France: Elsevier), ISBN 978-0-08-044654-7, pp 597-603.
- Chapurlat, V. (2007). "Vérification et validation de modèles de systèmes complexe : application à la modélisation d'entreprise." Habilitation à Diriger les Recherches de Université de Montpellier 2. 1er mars 2007.
- Chen, D. (2005). "Modélisation d'entreprise pour l'intégration et l'interopérabilité des systèmes industriels." Habilitation à Diriger des Recherches de l'Université Bordeaux 1. 4 décembre 2005.
- Chen, D., Dassisti, M. and Elvesaeter, B. (2006). "INTEROP-NoR FP6-IST : DI.2.Enterprise Interoperability Framework and knowledge corpus." InterOP-VLab, ed. (InterOP-VLab), December 15, p 210.
- Chen, D., Doumeings, G. and Vernadat, F. (2008). "Architectures for enterprise integration and interoperability: Past, present and future." Computers in Industry (v59, n°7), ISSN 0166-3615, pp 647-659.
- DeLaurentis, D. and Callaway, R. (2004). "A System-of-Systems Perspective for Public Policy Decisions." Review of Policy Research (v21, n°6), pp 829-837.
- De Rosnay, J. (1975). "Le microscope". Collection Points Ed. Le Seuil, 305 pages, 1975.

- Dieng, R., Corby, O., Giboin, A. and Ribiere, M. (1999). "Methods and tools for corporate knowledge management." *International journal of human-computer studies* (Elsevier ed.) (v51, n°3), ISSN 1071-5819, pp 567-598.
- DoD (2001 (As Amended Through 16 October 2006)). "The Department of Defense Dictionary of Military and Associated Terms." in *Joint Publication 1-02* (Department of Defense), <http://www.dtic.mil/doctrine/jel/doddict/>.
- Doumeingts, G. (1984). "La méthode GRAI." Thèse d'état, Université de Bordeaux I, France. Novembre 1984.
- Doumeingts, G., Girard, P. and Eynard, B. (1996). "GIM: GRAI Integrated Methodology For Product Development." in in "Design for X: Concurrent Engineering Imperatives" C. a. H. G.Q. HUANG (Eds.), London, ed., pp 153-172.
- EPAN (2004). "Official Report - Key Principles of an Interoperability Architecture." e. W. Group, ed. (European Public Administration Network - EU & Europe-wide – Interoperability), June 24, <http://www.epractice.eu/document/2963>.
- Estefan, J.A. (2007). "Survey of model-based systems engineering (MBSE) methodologies." (INCOSE MBSE Focus Group), Rev. A. 25 Mai 2007, pp 1-47.
- ETSI (2005). (European Telecommunications Standards Institute), http://portal.etsi.org/Portal_Common/home.asp.
- Favre, J.-M., Estublier, J. and Blay-Fornarino, M. (2006). *L'ingénierie dirigée par les modèles : au delà du MDA*, Traité IC2, série informatique et Systèmes d'Information. Lavoisier, 226 Pages.
- FEAF (2006). "Federal Architecture Framework." in *Draft Federal Enterprise Architecture Framework* (Federal CIO Council Architecture and Infrastructure Committee), <http://www.feacoinstitute.org/>.
- Ferrarini, L., Kalogeras, A., Lüder, A. and al. (2006). "Deliverable 6.1, PABADIS' PROMISE, Next Generation Control Devices." (PABADIS based Product Oriented Manufacturing Systems for Reconfigurable Enterprises), FP6-IST-016649, September 22, p 77,
- Fisher, D.A. (2006). "An Emergent Perspective on Interoperation in Systems of System." (Software Engineering Institute, Carnegie Mellon), March.
- Ford, T., Colombi, J., Graham, S. and Jacques, D. (2007). "The Interoperability Score." in *Proceeding of the 5th Conference on Systems Engineering Research*, March 14-16, (Hoboken, N.J.).
- Gamma, E., Helm, R., Johnson, R. and Vlissides, J. (1995). *Design Patterns Elements of Reusable Object-Oriented Software*. ISBN 0-201-63361-2, 396 Pages.
- GERAM (1999). "GERAM: Generalised Enterprise Reference Architecture and Methodology (Version 1.6.3)." (IFIP-IFAC Task Force on Architectures for Enterprise Integration), <http://www.cit.gu.edu.au/~bernus/taskforce/geram/versions/geram1-6-3/v1.6.3.html>.

- Gorod, A., Ryan, G., Brian, S. and John, B. (2007). "System of Systems Management: A Network Management Approach." in IEEE International Conference on System of Systems Engineering., April 16-18, (San Antonio, Texas, USA), ISBN 1-4244-1160-2, pp 1-5.
- Grundstein, M. (2004). De la capitalisation des connaissances au management des connaissances dans l'entreprise. In Imed Boughzala et Jean-Louis Ermine (Eds), 310 Pages.
- IDABC/EIF (2004). <http://ec.europa.eu/idabc/en/document/2358>.
- IEEE-Standard-1220 (2005). "IEEE Standard for Application and Management of the Systems Engineering Process." (Institute of Electrical and Electronics Engineers, Inc. New York,).
- IEEE (2000). "Standards Information Network. IEEE 100, The Authoritative Dictionary of IEEE Standards Terms." (New York, NY: Seventh Edition),
- INCOSE (2006). "Systems Engineering Handbook, a guide for system life cycle processes and activities." C. Haskins, ed. (INCOSE-TP-2003-002-03).
- ISO-15704 (2000). Industrial automation systems — Requirements for enterprise-reference architectures and methodologies (First edition 2000-06-01). 54 Pages.
- ISO/CEN-19439 (2002). "Enterprise Integration – Framework for enterprise Modelling, Final draft, CEN TC310/WG1 et ISO TC184 SC5/WG1."
- ISO/CEN-19440 (2002). "Language Constructs for Enterprise Modelling, Final draft, CEN TC310/WG1 et ISO TC184 SC5/WG1."
- ISO/IEC-7498 (1994). "Information technology -- Open Systems Interconnection -- Basic Reference Model." (Geneva, Switzerland: ISO/IEC FDIS Standard, IEC and ISO),
- ISO/IEC-15288 (2002). "System Life Cycle Processes and its Guide ISO." (Geneva, Switzerland: ISO TC 184/SC7/JTC1),
- ISO/IEC-19501:2005 (2005). "Information technology -- Open Distributed Processing -- Unified Modeling Language (UML) Version 1.4.2." (Geneva, Swizerland: ISO), April,
- ISO/IEC-JTC1-SC7-N3160 (2005). "IEEE Std 1471 -2000 IEEE Recommended Practice for Architectural Description of Software-Intensive Systems." (IEEE Computer Society), 2005-01-03, p 31.
- Johnson C.W. "What are emergent properties and how do they affect the engineering of complex systems?" Reliability Engineering & System Safety, Elsevier Ltd (v91, n°12).
- Koestler, A. (1967). The Ghost in the Machine. London, Arkana, ISBN 978-0140191929, 400 Pages.
- Kuras, M.L. (2006). "A Multi-Scale Definition of a System." M. T. Report, ed. (Bedford, MA: The MITRE Corp.), Report MTR 06B0000060, August, www.mitre.org.
- Lane, J.A. and Valerdi, R. (2007). "Synthesizing SoS concepts for use in cost modeling." Systems Engineering (v10, n°4), ISSN 1098-1241, pp 297-308.
- Le Gallou, F. and Bouchon-Meunier, B. (1992). Systémique : théorie et applications. Paris, Lavoisier, 2-85206-784-6, 341 Pages.

- Lea, D. (1994). "Christopher Alexander: an introduction for object-oriented designers." ACM SIGSOFT Software Engineering Notes (v19, n°1), pp 39-46.
- Le Moigne, J.L. (1994). La théorie du système général : théorie de la modélisation. mexapc.org, ISBN 2-13-043323-5, 352 Pages.
- Lievre, P. (2007). "La construction de savoirs pour l'action par intégration de connaissances pratiques 'tacites'et de savoirs scientifiques 'classiques'." in La construction de savoirs pour l'action, M.-J. Avenier and C. Schmitt, eds.,ISBN 978-2-296-02887-6.
- Lilleng, J. (2005). "Towards Semantic Interoperability." in preprints of eGov-Interop'05 Annual Conference (Geneva, Switzerland), February 23-25.
- Luzeaux, D. and Ruault, J.R. (2008a). Ingénierie des systèmes de systèmes - concepts et illustrations pratiques. Hermes Science, ISBN 978-2-7462-1875-8, 225 Pages.
- Luzeaux, D. and Ruault, J.R. (2008b). Ingénierie des systèmes de systèmes - méthodes et outils. Hermes Science, ISBN 978-2-7462-2045-4, 225 Pages.
- Maier, M.W. (1998). "Architecting principles for systems-of-system." Systems Engineering (v1, n°4), pp 267-284.
- Man-Sze. Li, Ricardo. Cabral, Guy. Doumeingts and Keith. Popplewell (Editors) (2006). "Enterprise Interoperability: Research Roadmap (Version 4.0)." (European Commission, Objective 1.3 ICT for Enterprise Networking), 31 July 2006,
- Mayer, F. and Auzelle, J.-p. (2007). "Is system of systems a candidate rationale artifact for entreprise information-intensive system modeling?" in 9th International Conference on The Modern Information Technology in the Innovation Processes of the Industrial Enterprise, MITIP 2007, (Florence : Italie).
- McConnell, S. (1996). Rapid Development: Taming Wild Software Schedules (1st ed.). Redmond, WA: Microsoft Press, ISBN 1-55615-900-5, Pages.
- Mellor, S.J., Scott, K., Uhl, A. and Weise, D. (2002). "Model-Driven Architecture." in Proceedings of the 8th Workshops on Advances in Object-Oriented Information Systems, 2 September 2002, (Montpellier , France), ISBN 3-540-44088-7, pp 290-297.
- MODAF (2007). "Ministry of Defence Architectural Framework, version 1.2 " (UK Ministry of Defence), Last Revision Version 1.2.003, 2nd September 2008.
- Morel, G. (1992). "Contribution à l'automatisation et à l'ingénierie des systèmes intégrés de production." Habilitation à Diriger des Recherches de l'Université Henri Poincaré. 28 janvier 1992.
- Morel, G., Panetto, H., Zarembo, M.B. and Mayer, F. (2003). "Manufacturing Enterprise Control and Management System Engineering: paradigms and open issues." IFAC Annual Reviews in Control, Elsevier (v27, n°2), pp 199-209.
- Morel, G., Panetto, H., Mayer, F. and Auzelle, J.P. (2007a). "System of Enterprise-Systems Integration Issues: an Engineering Perspective." in Invited conference at IFAC Conference on

- Cost Effective Automation in Networked Product Development and Manufacturing, IFAC-CEA'07 (Monterrey : Mexico), October 2-5.
- Morel, G., Valckenaers, P., Faure, J.-M., Pereira, C.E. and Diedrich, C. (2007b). "Special Issue on Manufacturing Plant Control: Challenges and Issues." INCOM 2004, 11th IFAC INCOM'04 Symposium on Information Control Problems in Manufacturing Control Engineering Practice (v15, n°11), ISSN: 0967-0661, pp 1321-1331.
- Morin, E. (1977). *La methode, Tome 1: la nature de la nature*. Le Seuil (coll. Points). Paris, France, ISBN 2-02-004634-2, 399 Pages.
- NAHIT (2005). "National Alliance for Healthcare Information Technology ", <http://www.nahit.org>.
- Oberndorf, P. (1998). "COTS and Open Systems." in SEI Monographs on the Use of Commercial Software in Government Systems. Software Engineering Institute. Carnegie Mellon University, Pittsburgh, Pennsylvania (Software Engineering Institute, Carnegie Mellon University), February, 1998,
- Office Of Force Transformation Washington, D.C. (2005). *The Implementation of Network-Centric Warfare*. Defense Technical Information Center, Pages.
- PABADIS'PROMISE (2006). "Deliverable 6.1, PABADIS'PROMISE, Next Generation Control Devices." (Project funded by the European Community under the "Sixth Framework Programme" (2002-2006) within the "Information Society Technologies" (IST) priority), FP6-IST-016649, September 22, p 77,
- Panetto, H. (2006). "Meta-modèles et modèles pour l'intégration et l'interopérabilité des applications d'entreprises de production." *Habilitation à Diriger des Recherches de l'Université Henri Poincaré*. 4 décembre 2006.
- Panetto, H., Berio, G., K., B., Boudjlida, N. and Petit, M. (2004). "A Unified Enterprise Modelling Language for enhanced interoperability of Enterprise Models." in proceeding of 11th IFAC Symposium on Information Control Problems in Manufacturing, INCOM'04, April 5-7, (Salvador-Bahia, Brazil).
- Penalva, J.-M. (1997). "La modélisation par les systèmes en situations complexes." *Thèse de Doctorat de l'Université de Paris Sud, Orsay*, 9 décembre.
- Rochet, S. (2007). "Formalisation des processus de l'Ingénierie Système : Proposition d'une méthode d'adaptation des processus génériques à différents contextes d'application." *Thèse de doctorat de l'Institut National des Sciences Appliquées de Toulouse*. 26 Novembre 2007, Toulouse.
- Sausser, B. and Boardman, J. (2007). "Complementarity: In Search of the Biology of Systems." in IEEE International Conference on System of Systems Engineering, 2007. SoSE apos07, April 16-18, (IEEE), pp 1-5.
- Scheer, A.-W. (1992). *Architecture of Integrated Information Systems: Principles of Enterprise Modeling*. Berlin, Springer-Verlag, 219 Pages.

- Schekkerman, J. (2007). "Enterprise Architecture Tool Selection Guide V4.2." I. F. E. A. Developments, ed., [http://www.enterprise-architecture.info/Images/EA_Tools/Enterprise Architecture Tool Selection Guide v4.2.pdf](http://www.enterprise-architecture.info/Images/EA_Tools/Enterprise_Architecture_Tool_Selection_Guide_v4.2.pdf).
- SESSI (2007). "Déploiement accru et diversification des TIC dans l'industrie manufacturière." <http://www.francetech.gouv.fr/sessi/4pages/231/principal.htm>, ed. (Ministère de l'Économie des Finances et de l'Emploi).
- Simon, H.A. (1996). *The Sciences of the Artificial*. MIT Press, ISBN 0262691914, 9780262691918, 231 Pages.
- Sowa, J.F. and Zachman, J.A. (1992). "Extending and formalizing the framework for information systems architecture." *IBM Systems Journal* (v31, n°3), ISSN 0018-8670, pp 590-616.
- Spewak, S.H. and Hill, S.C. (1992). *Enterprise Architecture Planning: Developing a Blueprint for Data, Applications, and Technology*. John Wiley & Sons, ISBN 978-0-471-59985-2, 392 Pages.
- Tardieu, H., Rochfeld, A. and Colletti, R. (1983). *La méthode Merise - Tome 1 Principes et outils*. Editions d'organisation (Paris), ISBN 2-7081-1106-X, 328 Pages.
- TEAF (2000). "Treasury Enterprise Architecture Framework, Version 1." (Department of the Treasury Chief Information Officer Council),
- Terzi, S. (2005). "Elements of Product Lifecycle Management: Definitions, Open Issues and Reference Models." Thèse de Doctorat de l'Université Henri Poincaré & du Politecnico di Milano. 27 Mai 2005.
- TISAF (1997). "Treasury Information System Architecture, Framework, Version 1.0 " (Washington, D.C., : Office of the Deputy Assistant Secretary for Information Systems and Chief Information Officer, Department of the Treasury), 3 January 1997.
- TMAB (2005). "Telemedicine Alliance Bridge " (<http://www.opengroup.org/architecture/togaf8-doc/arch/toc.html>).
- TOGAF (1995). "The Open Group Architecture Framework, version 1." (The Open Group), <http://www.opengroup.org/togaf/>.
- TOGAF (2000). "The Open Group Architecture Framework, version 8.1.1 (enterprise edition)." (The Open Group), p 500, <http://www.opengroup.org/architecture/togaf8-doc/arch/toc.html>.
- Tsagkani, C. (2005). "Inter-Organisational Collaboration on the Process Layer." in *In Proceedings of the IFIP/ACM SIGAPP INTEROP-ESA conference.*, February 23-25, Springer, ed. (Geneva, Switzerland), ISBN 1-84628-151-2.
- Tucoulou, J.-C. (2008). "Cycle de vie générique d'un projet en E&R : chronologies." (Management & Communication).
- Tursi, A., Panetto, H., Morel, G. and Dassisti, M. (2007). "Ontology-Based Products Information Interoperability in Networked Manufacturing Enterprises." in *Proceedings of the IFAC CEA'2007 conference on Cost Effective Automation in Networked Product Development and*

- Manufacturing, October 5-7, (Monterrey, Mexico: Elsevier - IFAC Papersonline), ISBN 978-3-902661-32-6.
- Tursi A., Panetto H., Morel G., Dassisti M. (2009). Ontological approach for Products-Centric Information System Interoperability in Networked Manufacturing Enterprises. IFAC Annual Reviews in Control. 33/3, September, Elsevier, ISSN: 1367-5788.
- Ueda K (2001). "Synthesis and emergence – research overview." Artificial intelligence in engineering, Elsevier Ltd (v15, n°4), ISSN 0954-1810, pp 321-327(327).
- Vallespir, B. (2003). "Modélisation d'entreprise et architectures de conduite des systèmes de production." Habilitation à Diriger des Recherches de l'Université Bordeaux 1. 19 Décembre 2003.
- Vallespir, B., Chapurlat, V., Braesch, C. and Crestani, D. (2003). "L'intégration en modélisation d'entreprise: les chemins d'UEML." in proceeding of MOSIM, April 23-25, (Toulouse).
- Vernadat, F. (1996). Enterprise Modeling and Integration: Principles and Applications. Springer, ISBN 0412605503, 9780412605505, 513 Pages.
- Vernadat, F. (2002). "UEML: towards a unified enterprise modelling language." International Journal of Production Research (v40, n°17), ISSN 0020-7543, Online ISSN: 1366-588X, pp 4309-4321(4313).
- Vernadat, F. (2006). " Enterprise Model For Integrating Process Control Into Cam System." in INCOM, 17,18,19 May, (Saint Etienne).
- Vernadat, F. (2007). "Interoperable enterprise systems: Principles, concepts, and methods." IFAC, Annual Reviews in Control, Elsevier (v31, n°1), ISSN 1367-5788, pp 137-145.
- Whitman, L., Santanu, D. and Panetto, H. (2006). "An Enterprise Model Of Interoperability." in Proceeding of the 12th IFAC Symposium on Information Control Problems in Manufacturing (INCOM'2006) (Saint Etienne, France: Elsevier), May 17-19.
- Wieggers, K.E. (2003). Software Requirements 2: Practical techniques for gathering and managing requirements throughout the product development cycle (2nd ed.). Redmond: Microsoft Press, ISBN 0-7356-1879-8, 544 Pages.
- Williams, T.J. (1992). "The Perdue Enterprise Reference Architecture." Instrument Society of America, Research triangle Park.
- Wymore, A.W. (1993). Model-Based Systems Engineering. CRC Press, Inc. Boca Raton, FL, USA, ISBN 0-8493-8012-x, 700 Pages.
- Zachman, J.A. (1987). "A Framework for Information Systems Architecture." IBM Systems Journal (v26, n°3), pp 276-292.
- Zacklad, M. and Grundstein, M. (2001). Management des connaissances: modèles d'entreprise et applications. Hermès science publications, ISBN 978-2-7462-0235-1, 245 Pages.
- Zanettin, M. (1994). "Contribution à une démarche de conception des systèmes de production." Thèse de Doctorat de l'Université Bordeaux 1, France. Mars 1994.

Titre : PROPOSITION D'UN CADRE DE MODÉLISATION MULTI-ÉCHELLES D'UN SYSTÈME D'INFORMATION EN
ENTREPRISE CENTRÉ SUR LE PRODUIT

Résumé : Notre thèse porte sur la Modélisation d'Entreprise et plus particulièrement sur une vision Ingénierie Système de celle-ci afin de faciliter son application « à minima » en entreprise. Alors que la modélisation d'entreprise arrive à maturité scientifique et technologique, force est de constater que sa pratique est encore trop peu courante. En outre, il n'existe pas à proprement parler de tâche de modélisation, de maintenance et de gestion de modèles dans l'entreprise. La question s'est alors posée de la relation d'interopération entre les systèmes, d'un Système-Entreprise évolutif, le plus souvent composé de façon ad-hoc de sous-systèmes (COTS), qui ont pour particularité d'encapsuler leur propre Savoir-Faire et leurs Ingénieries respectives, en incluant le Système-Produit (son besoin, son projet, ...) à la source de chaque recomposition « à la demande » du Système-Entreprise. Un des objectifs de nos travaux est donc de faciliter l'interopération entre les ingénieries de ces systèmes interopérants (autour du produit A Faire) et l'Ingénierie du Système-Projet d'Entreprise (Pour Faire). Nous nous sommes alors inspirés de la proposition de Kuras pour formaliser notre Système-Entreprise (en fait son Ingénierie) dans une sorte de « Système de Systèmes d'Information » (SdSI) en tenant compte de ses aspects récursifs et multi-échelles dans un contexte organisationnel évolutif. De plus, en nous appuyant sur la théorie des patterns, nous avons identifié puis dérivé le « composite-pattern » pour mieux décrire la nature « Tout et Partie » de l'Holon Système-Entreprise utilisé par Kuras pour extraire le patron de conceptualisation le plus adéquat. Parmi tous les points de vue d'un système, cette dérivation nous permet de représenter ceux du Système A Faire et ceux du Système Pour Faire avec son contenu d'Ingénierie associée guidée par le cadre de modélisation de Zachman. Nous avons outillé notre proposition de cadre d'Ingénierie Système Basée sur des Modèles (ISBM) en étendant le méta-modèle de l'outil « MEGA Modelling Suite » ainsi qu'en développant les interfaces utilisateurs nécessaires et nous l'avons appliqué à l'Ingénierie d'un Système de Traçabilité d'un Produit dans le contexte d'un scénario PLM plausible entre l'AIPL et DIMEG.

Mots clés : modélisation d'entreprise, interopération, intégration, systèmes d'information, système de systèmes, récursivité, multi-échelles, Zachman, ingénierie système, ingénierie basée sur les modèles, pattern.

Title : PROPOSAL OF A MULTI-SCALE MODELLING FRAMEWORK FOR A PRODUCT-CENTRIC ENTERPRISE INFORMATION SYSTEM

Abstract: Today's needs for more capable enterprise systems in a short timeframe are leading more organizations toward the integration of existing component-systems. In the domain of Enterprise Modelling and more generally on Systems Engineering, the key issue is related to aligning enterprise models with the composite structure of the modelled systems. While Enterprise Modelling matures both scientifically and technologically, it is clear that its practice is still uncommon in the enterprises. There is no modelling approach in enterprise, neither even capitalization of models. This paper aims at proposing a Model-Based System Engineering (MBSE) approach, by using the recursion principle, together with a "multiple scales system thinking", for capitalizing the engineering know-how, over time, during a project lifecycle. In this context, we advocate that, within an enterprise, the heterogeneous set of Information Systems may be considered as COTS (Commercial-Off-The-Shelf), encapsulating their own expertise (engineering), and contributing all together to a single System-of-Information System (SoIS). We are then proposing a SoIS engineering methodology resulting from deriving our MBSE approach with a recursive view of the Zachman framework. This approach, based on modelling rules and enterprise modelling constructs, is contributing to ensure the consistency and the completeness of the various heterogeneous models interoperating during systems engineering projects. To validate our MBSE proposal, a modelling tool has been prototyped. It helped at applying our SoIS engineering methodology in a use case scenario related to the requirement analysis and the specification of a product traceability system.

Keywords: Enterprise Integration, Systems Interoperability, Systems Engineering, System-of-Systems, Enterprise Modelling, Information Systems, recursion, Zachman, multi-scales, pattern, Model Based Systems Engineering