



## Composant

PxM-02 « Modus : La méthodologie Praxeme »

---

## Guide général

*Objectif* Ce document établit le socle sur lequel s'appuient les pratiques de conception du Système d'information.

Les choix méthodologiques majeurs sont : le standard UML pour la représentation, l'approche orientée objet pour la conception, l'architecture de services et l'urbanisation du SI.

- Contenu*
- Le socle : « Les principes structurants », « La notion de « service »
  - Les produits
  - La démarche
  - Les procédés de modélisation

*Rédacteur* Dominique VAUQUIER

*Version* 1.0, le 27 avril 2006

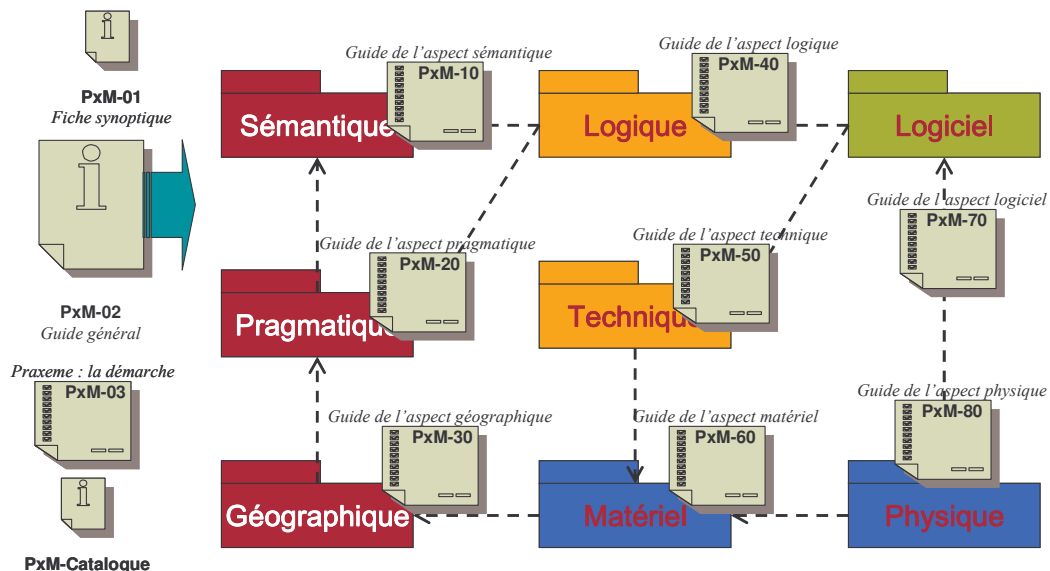
## Éléments de configuration

### Situation du composant

#### Positionnement dans la documentation



La méthodologie Praxeme est structurée selon les aspects de la topologie du Système Entreprise. Le Cadre méthodologique explique cette approche.



#### Propriétaire

Le Référentiel Praxeme a été élaboré dans le cadre du chantier Praxime. Les principaux contributeurs sont les sociétés SAGEM et SMABTP. Les contributeurs sont réunis dans l'initiative pour une méthode publique. L'institut Praxeme fait évoluer le fonds commun.

Toute suggestion ou souhait d'évolution sont les bienvenus (à adresser à l'auteur).

### L'historique

Indice	Date	Rédacteur	Contenu
	Mars 2004	DVAU	Première rédaction (Dromos : méthode Sagem pour l'urbanisation de l'informatique des systèmes de drones)
	Novembre 2005	DVAU	Version étendue (Amos : méthode de la DSI SMABTP ; approche SOA)
1.0	27/04/06	DVAU	Généralisation pour soumission au « cercle des experts »
1.0			Version actuelle du document

## Sommaire

	Situation du composant	2
	L'historique	2
Introduction.....		6
	Une nouvelle méthode pour de nouveaux enjeux	6
	Les trois dimensions de la méthodologie : Produit, Processus, Procédés	7
Le socle.....		8
	Fonder les pratiques sur une base solide	8
	La Topologie du Système Entreprise	9
	La définition des aspects	10
	Mettre de l'ordre dans les représentations	11
	Les relations entre les aspects du système	12
	Le rôle de l'aspect logique	13
	La notion de « service »	15
	La chaîne de production	16
Les produits.....		17
	Les termes généraux	17
	Le modèle sémantique : Aller à l'essentiel pour dégager un noyau stable	18
	Le modèle pragmatique : Exprimer le besoin – local – et garantir la cohérence – globale	19
	Le modèle logique : Structurer pour durer	20
	Le modèle logique : un langage à part	21
	Le modèle logique : la stratification du système	22
	L'élaboration de l'architecture de services	23
	La couverture du système par les modèles	24
	Les livrables	25
Le processus.....		27
	La mise en œuvre de la topologie	27
	Modéliser : entre analyser et concevoir	28
	Les niveaux de cibles	29
	La démarche : les possibilités de parallélisme	30
	La démarche : les travaux d'architecture	31
	Les activités de portée globale	32
	L'urbanisation	33
	L'organisation de l'urbanisation	34
	Les responsabilités sur les aspects	35
Les procédés de modélisation.....		36
	Représenter avant d'agir	36
	La modélisation sémantique : Aller à l'essentiel pour dégager un noyau stable	37
	La modélisation sémantique : quelques préceptes	38
	L'expression des besoins par les cas d'utilisation	39
	L'architecture logique	40
	L'identification des services logiques	41
	La documentation des services logiques	42
	L'architecture technique	43
La « topologie du Système Entreprise ».....		46
	Que faut-il représenter pour maîtriser le Système ?	46
	Le schéma général de la topologie	47
	Que faut-il représenter ?	48
	La solution passe par une vision claire de l'entreprise	49
La déduction des aspects.....		50
	Le fil conducteur	50

	« Pourquoi ? Pour quoi ? »	51
Le Système Entreprise	.....	52
	« Comment mettre en œuvre la stratégie ? »	52
	« Quoi ? » : la sémantique du Système Entreprise	53
	« Qui » : la pragmatique du Système Entreprise	53
	« Où » : l'aspect géographique	54
	Récapitulatif de la vue externe	54
Le Système de production	.....	55
	« Comment mettre en pratique les orientations ? »	55
	Le contenu	56
Le Système productique	.....	57
	Les derniers aspects dans la construction du système	57
Récapitulatif	.....	58
	Une ligne explicative pour définir les modèles à réaliser	58
Complément à la Topologie du Système Entreprise	.....	59
	Les questions quantitatives : « Quand » et « Combien »	59
	Contenu et forme des modèles	60
	L'apport d'UML pour représenter les aspects : vue externe du Système	61
	L'apport d'UML pour représenter les aspects : vue interne du Système	62
	Conclusion	63
Sur la modélisation	.....	64
	Les dimensions de la modélisation	64
	UML et les dimensions de la modélisation	65
Complément sur le processus	.....	66
	La dynamique globale	66

## Table des figures

Figure PxM-02_1. Structure du document .....	7
Figure PxM-02_2. Le schéma de la topologie du système .....	9
Figure PxM-02_3. La définition des aspects du Système Entreprise .....	10
Figure PxM-02_4. Illustration des aspects .....	11
Figure PxM-02_5. Le cycle de vie du service, rapporté aux aspects de la topologie du Système Entreprise. ....	16
Figure PxM-02_6. La terminologie pour la modélisation sémantique .....	18
Figure PxM-02_7. La terminologie pour la modélisation pragmatique .....	19
Figure PxM-02_8. L'emboîtement des agrégats logiques .....	21
Figure PxM-02_9. La stratification du système d'information, déduite des règles de l'architecture logique. ....	22
Figure PxM-02_10. Le méta-modèle Praxeme pour l'aspect logique .....	23
Figure PxM-02_11. L'articulation des modèles par aspects .....	24
Figure PxM-02_12. Tableau d'ensemble des livrables .....	25
Figure PxM-02_13. Les niveaux de cibles .....	29
Figure PxM-02_14. Les macro-activités en regard des niveaux de cibles .....	29
Figure PxM-02_15. Les axes de la conception.....	30
Figure PxM-02_16. L'approche Architecture .....	31
Figure PxM-02_17. Les activités de portée globale accompagnant les projets.....	32
Figure PxM-02_18. Les responsabilités en fonction de la portée .....	34
Figure PxM-02_19. Les rôles dans Praxeme.....	35
Figure PxM-02_20. La liste des procédés dans Praxeme .....	36
Figure PxM-02_21. Déduction des services logiques à partir des modèles externes .....	41
Figure PxM-02_22. Aperçu des thèmes de l'architecture technique.....	44
Figure PxM-02_23. Présentation synoptique de la Topologie .....	47
Figure PxM-02_24. UML pour les aspects de la vue externe .....	61
Figure PxM-02_25. UML pour les aspects internes du Système .....	62
Figure PxM-02_26. Les trois axes de la modélisation .....	64
Figure PxM-02_27. La distribution des diagrammes UML sur les trois axes de la modélisation.....	65
Figure PxM-02_28. Les macro-activités de la dynamique globale .....	66

### Exergue

*« La théorie sans la pratique est inutile ; la pratique sans la théorie est aveugle. »*

Immanuel Kant

## Introduction

---

### Une nouvelle méthode pour de nouveaux enjeux

---

#### Le contexte

*Contexte général* Soumises aux exigences de l'adaptation, les entreprises doivent tirer le plus grand profit des nouvelles technologies mais aussi libérer leur potentiel d'innovation en matière de savoir-faire et d'organisation. Parmi les difficultés qu'elles rencontrent, la plus grande, sans doute, est de mettre en synergie des expertises qui ont du mal à se reconnaître et à interagir. Pour surmonter cet obstacle, une condition réside dans la partage d'un cadre commun permettant de loger les compétences et de les articuler.

*Une méthodologie d'entreprise* Praxeme est une méthodologie d'entreprise qui se donne pour tâche de fournir un tel cadre général. La Topologie du Système, présentée ici, en est le socle théorique. Le corpus Praxeme contient, également, des procédés qui vont préciser la manière d'aborder tel ou tel aspect, sans perdre de vue la cohérence d'ensemble.

*Pour la maîtrise des systèmes d'information* L'axe actuellement le plus développé dans Praxeme concerne les systèmes d'information. Ces systèmes, réputés complexes, constituent un atout essentiel dans la lutte économique. Pourtant, cette complexité et le désordre des spécialités réduisent considérablement les pistes d'amélioration ou augmentent les coûts et les incertitudes. Praxeme a été élaboré, principalement, pour aider les directions informatiques à relever leurs nouveaux défis. Elle absorbe des courants comme l'approche objet, l'architecture de services (SOA) et le standard MDA (*Model Driven Architecture*). Elle exploite les techniques de représentation offertes par le standard UML (*Unified Modeling Language*) dont elle fournit un mode d'emploi.

---

#### Les objectifs

Le présent document fournit une vision d'ensemble sur le référentiel. Il explicite les fondements de la nouvelle approche et justifie les différentes méthodes ainsi que les livrables.

Le « Guide général » formule les principes structurants du référentiel méthode. Il les situe par rapport aux besoins. Il ne couvre pas tout le référentiel, mais en constitue une introduction. D'autres documents se focalisent sur un procédé ou un type de livrables (les formulaires, notamment).

---

#### Le contenu

La première partie fixe les principes. Ceux-ci se déploient, ensuite, à travers les trois dimensions de la pratique :

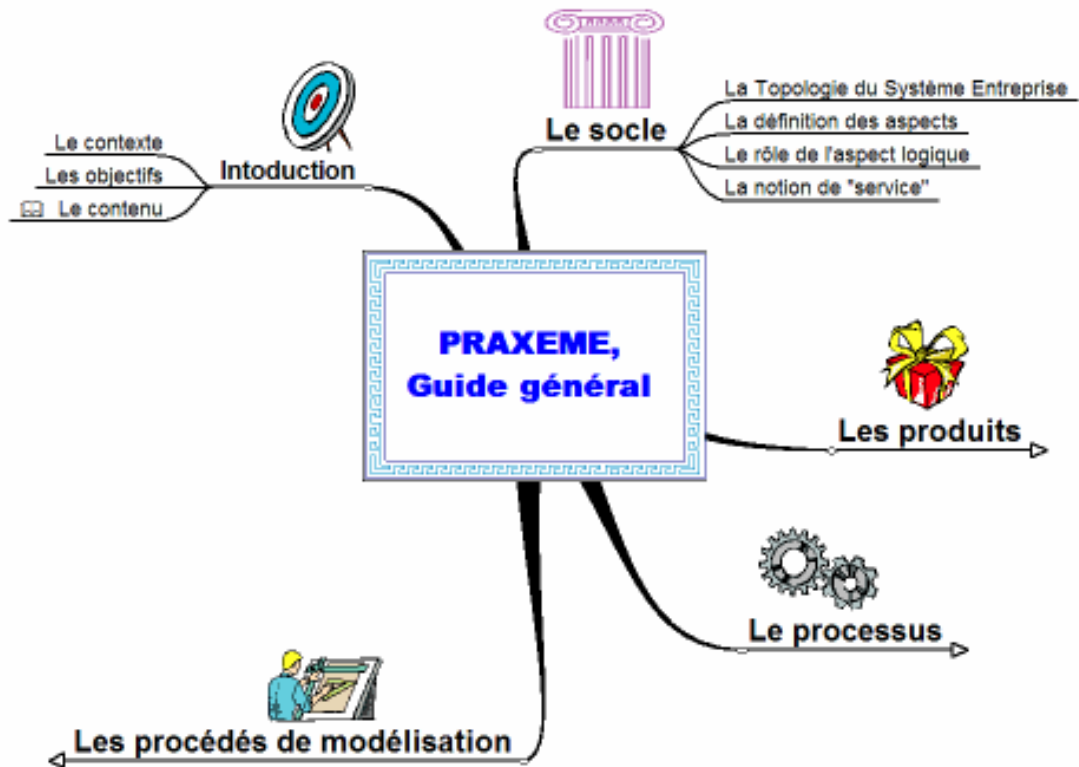
1. Produit (ce qu'il faut produire, les livrables),
2. Processus (comment s'organiser pour produire),
3. Procédé (comment travailler concrètement).

Dans la troisième partie, on insistera sur les procédés de modélisation.

## Introduction (suite)

### Les trois dimensions de la méthodologie : Produit, Processus, Procédés

Figure PxM-02\_1. Structure du document



#### À propos des procédés

Le document présente une sélection des procédés ordonnés dans la méthodologie :

- La modélisation sémantique, préalable à l'identification de services à forte valeur.
- La spécification du système à base de cas d'utilisation.
- L'architecture logique.
- La conception des services.

## Le socle

---

### Fonder les pratiques sur une base solide

---

#### Motivation

Les systèmes d'information sont des objets complexes qui mobilisent des expertises variées. On ne saurait guider l'action sur ces systèmes qu'une fois posé le cadre théorique qui situe ces expertises.

#### Les principes structurants

- Une première question surgit : « Que faut-il représenter pour agir sur le système ? ». Praxeme répond à cette question par la **Topologie du Système Entreprise**, recensement des aspects du système.
- Le principe de **séparation des aspects** structure l'approche du système : il isole différents modèles et conditionne le processus de développement.
- Parmi ces aspects, **l'aspect logique** revêt une importance particulière dans la perspective de l'architecture de services et de l'urbanisation du système d'information.
- Praxeme accorde une attention spéciale à la **notion de service**, grain élémentaire du système informatique et monnaie d'échange avec les systèmes partenaires.

Les pages suivantes exposent ces principes.

#### « Que faut-il représenter ? »

La Topologie du Système Entreprise s'inscrit dans l'héritage du Génie logiciel :

- Elle reprend la notion des niveaux d'abstraction, propre à Merise.
- Elle s'inspire de travaux comme le *framework* de Zachman.
- Elle actualise cet héritage en se conformant au standard MDA (*Model Driven Architecture*, standard de l'OMG<sup>1</sup>). MDA pose le principe de modèles dont certains sont indépendants de la technologie, d'autres dépendants (PIM : *Platform Independent Model* et PSM : *Platform Specific Model*).

---

<sup>1</sup> L'OMG (*Object Management Group*) regroupe la plupart des acteurs de l'industrie logicielle. Il a publié plusieurs standards dont CORBA et UML.



## Le socle (suite)

### La Topologie du Système Entreprise

**Le problème posé** S'il est entendu que la modélisation est un préalable indispensable à la conception des systèmes complexes et qu'elle constitue une des conditions de leur maîtrise, la question se pose de savoir précisément ce qu'il faut modéliser. UML ne répondant pas à cette question, elle est du ressort de la méthodologie.

Faute de répondre à cette question « Que faut-il représenter ? », les projets s'exposent à d'importants déboires et, en tout cas, à une réelle déperdition.

### La réponse

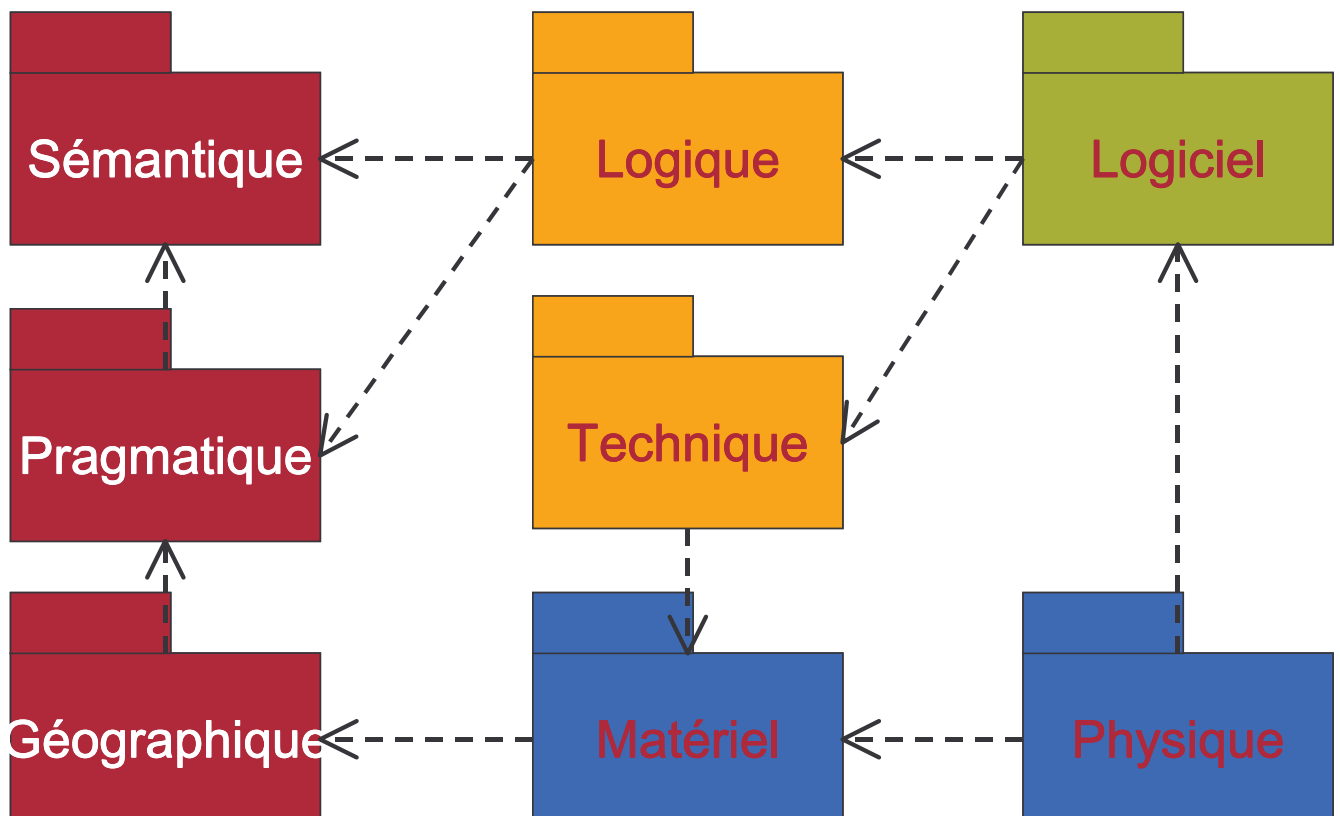
La réponse proposée consiste à recenser les « aspects » ou facettes de la réalité sur laquelle nous voulons agir. Ces aspects sont articulés rigoureusement dans la « topologie du Système Entreprise ». Cette dernière fournit, au référentiel Praxeme, sa base théorique.

Il s'agit de recenser les aspects à examiner pour décrire de façon pertinente et exhaustive la réalité sur laquelle on veut intervenir, en l'occurrence l'entreprise. Il est nécessaire de bien appréhender cet objet pour concevoir la solution informatique qui pourra l'outiller.

Le terme « topologie » est retenu pour évoquer le « lieu » (en grec : *topos*), l'endroit où on traite et conserve les différentes informations et décisions portant sur le système.

La figure ci-dessous résume la topologie du Système Entreprise. On trouvera, en annexe, une justification de cette structure (cf. pages 46 et suivantes).

Figure PxM-02\_2. Le schéma de la topologie du système



## Le socle (suite)

### La définition des aspects

#### La séparation des aspects

Les systèmes sur lesquels nous intervenons sont des réalités complexes. Elles impliquent un grand nombre d'informations et de questions qui intéressent des métiers différents. Pour maîtriser cette complexité, il est préférable de séparer les préoccupations en ensembles homogènes, chaque ensemble intéressant une discipline ou spécialité bien délimitée. Ces ensembles homogènes de préoccupation sont les aspects.

**Isoler les aspects contribuer à maîtriser la description et à faciliter l'évolution du système.**

*L'aspect* L'aspect (ou facette) est une vue du système. Le système est vu selon un type de préoccupation particulier. L'aspect, tout en étant une composante du système, a donc une nature relative : il est lié à un point de vue, un type de préoccupation, une spécialisation. Certains aspects n'ont rien à voir avec l'informatique.

Le tableau ci-dessous définit les huit aspects retenus dans la Topologie (pour leur justification voir l'annexe).

Figure PxM-02\_3. La définition des aspects du Système Entreprise

Aspect	Termes équivalents	Définitions
<b>Sémantique</b>	Conceptuel, essentiel, « Cœur de métier »	L'aspect sémantique ne retient que les objets au cœur de l'activité. On décrit le noyau fondamental indépendant de la manière de mener l'activité.
<b>Pragmatique</b>	Organisationnel	L'aspect pragmatique réunit les choix relatifs à la manière de mener l'activité : acteurs, responsabilités, actions sur les objets, processus, situations de travail.
<b>Géographique</b>	«Communication», «Contexte»	L'aspect géographique est celui de la localisation des objets et des actions. Il fait apparaître les notions de sites, d'emplacements et de besoins de communication.
<b>Logique</b>	«Fonctionnel»	Aspect intermédiaire permettant de fixer les grandes décisions de structuration du système d'information, dans une relative indépendance par rapport aux solutions techniques.
<b>Technique</b>	Technologique	L'aspect technique est celui des choix de technologies et des façons de les mettre en œuvre.
<b>Matériel</b>	Logistique	L'aspect matériel du système est l'ensemble des machines physiques composant le système, avec leurs propriétés (capacité...).
<b>Logiciel</b>	Applicatif, informatique	L'aspect logiciel couvre l'ensemble des composants logiciels qui automatisent une partie des actions du système.
<b>Physique</b>	Déploiement	À travers l'aspect physique, on décrit la localisation des composants logiciels (bases de données comprises) sur les matériels.

## Le socle (suite)

### Mettre de l'ordre dans les représentations

#### Motivation

Ordonner rigoureusement les informations à collecter au cours de la conception est une hygiène qui présente les retombées suivantes :

- **productivité** due à la séquence rigoureuse des travaux et des questions à examiner ;
- meilleure **exploitation des modèles** et de la documentation : chaque modèle a son propre cycle de vie (le modèle sémantique est très stable et servira de référence à long terme ; le modèle logique est indépendant de l'architecture technique et résistera, donc, aux changements de technologies, etc.).

#### Exemples

Le tableau suivant illustre les aspects prescrits par la topologie.

Figure PxM-02\_4. Illustration des aspects

Aspects	Illustrations	Principales catégories de représentation	Commentaires
<b>Sémantique</b>	Produit, Contrat, Sinistre, Objet (objet assuré ou sinistré)	Classes, automates à états	Le modèle sémantique capture et formalise les fondamentaux du métier. Très stables.
<b>Pragmatique</b>	Acteur, partenaire, règles d'organisation, habilitations, « Déclarer un sinistre », « Commander un produit »	Acteurs, cas d'utilisation, processus	Les pratiques et règles d'organisation sont isolées. Elles pourront évoluer plus facilement.
<b>Géographique</b>	Siège, directions régionales, agences, étranger, poste nomade	Type de sites, réseaux (au sens courant)	Le modèle géographique fournit les hypothèses et contraintes de localisation physique.
<b>Logique</b>	« domaines », ressources communes, « Structures », « blocs, quartiers... »	Machines logiques, services logiques	Cela permet de prendre les décisions de structuration du système.
<b>Technique</b>	Support de données, <i>middleware</i> , composants techniques, langages...	Choix techniques, <i>frameworks</i>	L'architecture technique explique comment dériver la description logique en logiciel, pour une cible donnée.
<b>Matériel</b>	Machines, processeurs, liaisons, réseaux	« nœuds » et connexions	
<b>Logiciel</b>		Composants logiciels, applicatifs	Les composants logiciels s'obtiennent par combinaison entre les unités logiques et les choix techniques.
<b>Physique</b>		Composants logiciels et matériels	Les composants logiciels sont localisés sur les matériels informatiques qui composent l'architecture matérielle.

## Le socle (suite)

---

### Les relations entre les aspects du système

---

#### Les relations

La topologie du Système Entreprise indique les possibilités de passage d'un aspect à l'autre au travers de liens. Ces liens expriment les dépendances entre les aspects (ou relations d'utilisation).

Articuler les modèles est une exigence pour ordonnancer les travaux. Les paragraphes suivants justifient les liens représentés sous la forme de flèches en pointillés dans la figure de la page 9.

*Lien du pragmatique au sémantique* L'action porte sur l'objet essentiel (conceptuel). Le modèle pragmatique (décrivant les acteurs et leurs actions) se réfère, donc, au modèle sémantique (décrivant les objets au cœur du métier).

*Lien du géographique au pragmatique* Les types de sites sont déterminés par la répartition des acteurs et des responsabilités. S'il y a une direction régionale, c'est parce que l'on y localise un certain type de responsabilité.

*Lien du logique au sémantique et pragmatique* Les termes du modèle logique proviennent d'une transcription des descriptions sémantiques et pragmatiques, transcription guidée par les règles de structuration. Ce point est détaillé plus loin.

*Lien du matériel au géographique* Cette relation permet de dire quel matériel se trouve à quel endroit. Cette détermination s'accompagne d'éléments quantitatifs.

*Lien entre technique et matériel* On peut choisir l'un en fonction de l'autre ou inversement. La topologie choisie accorde la priorité à l'architecture matérielle, plus contraignante. Même si le raisonnement se fait dans l'autre sens, la documentation restera les dépendances fixées par la topologie.

*Lien du logiciel vers logique et technique* Un composant logiciel, quel que soit son niveau, s'obtient par dérivation d'un composant logique, pour une architecture technique donnée.

*Lien du physique vers le matériel et le logiciel* La chaîne de production du logiciel aboutit à l'aspect physique. La dernière étape consiste à localiser les composants logiciels sur l'architecture matérielle.

---

#### L'importance des relations

L'articulation des aspects montre comment les informations, représentations et décisions s'enchaînent d'un bout à l'autre de la chaîne de production du logiciel.

Les relations entre les modèles résument les dépendances ou références au niveau plus fin des éléments de modélisation.

Sur ces relations, il est possible de définir des règles de dérivation<sup>2</sup> qui permettent de passer, parfois mécaniquement, d'un aspect à un autre. Par exemple, une opération sur une classe du modèle sémantique devient un « service logique », après quelques changements de sa signature (son interface).

---

<sup>2</sup> Nous préférons le terme « dérivation » à celui de « transformation ». Transformer, c'est prendre un objet et le changer ; alors que la dérivation part d'un objet pour en fabriquer un autre. Les transformations sont internes à un aspect ; les dérivations impliquent plusieurs aspects : elles alimentent un modèle à partir d'informations trouvées dans un modèle en amont.

## Le socle (suite)

### Le rôle de l'aspect logique

#### Une réalité intermédiaire

Il est difficile de situer l'aspect « logique ». Cela tient à son rôle intermédiaire et à la nature quelque peu arbitraire de son expression.

L'aspect logique est intermédiaire entre :

- La « vue externe » : le monde réel des objets du domaine et des acteurs du système.
- Le système informatique (choix techniques, composants logiciels, déploiement).

**On intercale ce niveau intermédiaire afin de faciliter les décisions de structuration du système logiciel.**

La métaphore d'une bibliothèque fait comprendre la finalité de l'architecture logique ainsi que le type de questions à traiter pour cet aspect (voir encart ci-après).

#### Le positionnement de l'aspect logique

L'aspect logique n'a pas de réalité propre. Il ne vaut que comme intermédiaire entre la vue externe (aspects : cœur de métier, organisation et géographie) et la vue interne (système logiciel) ou, si l'on veut, entre la réalité du champ d'intervention et les technologies de l'information et de la communication. Ceci explique pourquoi son mode d'expression repose sur la métaphore, par exemple :

- métaphore de l'urbanisation, le système d'information étant comparé à une ville à organiser ;
- métaphore du service, le système étant conçu comme un ensemble de réponses élémentaires à des demandes.

#### Deux portées

L'aspect logique donne lieu à deux types d'intervention, selon la portée « système » (portée globale) ou « application » (portée locale).

##### *L'architecture logique*

L'architecture logique est le premier niveau de description du système informatique<sup>3</sup>. Elle s'exprime par un graphe d'architecture, préexistant au système futur et guidant ses évolutions. La description ne s'arrête pas là : l'architecture logique doit être considérée comme un référentiel de description qui rassemble toute l'information au niveau logique et la met à la disposition des développeurs.

##### *La conception logique*

La conception logique s'applique, également, au niveau des applications. D'une part, le concepteur recherche dans l'architecture logique les services qui peuvent contribuer au fonctionnement de l'application. D'autre part, le développement contribue – en retour – à consolider et enrichir l'architecture logique, sous la pression des nouveaux besoins.

<sup>3</sup> Les « niveaux » antérieurs (sémantique et pragmatique) portent sur le Système Entreprise lui-même, non sur le système informatique. Celui-ci est une composante de celui-là.

### Qu'est-ce que la conception logique ? – La métaphore de la bibliothèque

Si on veut ranger au mieux des ouvrages dans une bibliothèque et faciliter son exploitation, on commencera par étudier le domaine de la bibliographie : classification des disciplines et des thèmes, analyse des relations entre thèmes, auteurs, livres, etc. Cette approche de la sémantique aboutit à un modèle très buissonnant, où tout peut être relié à tout. Il est essentiel de percevoir cette réalité, car, même si on ne peut pas la restituer physiquement (dans la bibliothèque) ou informatiquement (dans le logiciel), elle sous-tend les pratiques. Elle constitue donc le point de départ et l'horizon de la conception. Elle fait l'objet de la modélisation « **sémantique** ».

Dans un deuxième temps, avant d'organiser notre bibliothèque, nous devons étudier les comportements de ses visiteurs. Quelles sont les populations ? Quels sont leurs besoins et leurs motivations ? Comment agissent-ils ? Certains viennent avec un besoin précis ; d'autres sont là pour la flânerie. Parfois, l'approche sera pratique : trouver toutes les ressources sur un thème, sans distinction de discipline ou de support. D'autres fois, elle sera ludique : goût du livre, séduction des classements par collection, etc. Ces questions sont étudiées dans l'aspect « **pragmatique** ».

Ensuite, le concepteur de la bibliothèque se doit de considérer les contraintes et possibilités liées à la topologie des lieux : séparation entre espace public et espace réservé ; nombre d'étages ; communications, etc. Ces éléments devront être pris en compte dans la suite de la conception. C'est l'aspect « **géographique** ».

La bibliothèque comporte, également, un aspect « **technique** ». Les catalogues proposent des éléments de mobiliers avec des caractéristiques physiques à respecter. La hauteur et la longueur des étagères dépendent du matériau et du style choisi. On peut penser à des dispositifs mécaniques pour le transport et le rangement des livres ou pour l'accumulation des étagères dans un espace réduit...

En considérant l'ensemble de ces éléments :

- le modèle sémantique comme idéal, réel dans l'esprit du lecteur mais inaccessible quand on veut le transposer dans la solution physique ;
- les usages et attentes des utilisateurs ;
- les possibilités techniques ou les choix déjà imposés ;

le concepteur va **esquisser un schéma pour ranger** les ouvrages. Il cherchera à préserver le plus possible la prolifération de la sémantique et la fluidité de la pragmatique. Mais il est conscient des contraintes issues de la technologie et des conséquences de ses décisions sur le comportement du système qui sera mis en œuvre. Ce schéma, il en établit d'abord les principes, puis le mène progressivement jusqu'à un niveau de détail qui permettra au bibliothécaire de savoir, sans ambiguïté, où ranger chaque livre.

C'est cela, l'architecture logique !

Le concepteur est sans cesse en situation d'arbitrer entre différentes options. À chaque décision, il sait ce qu'il perd. Il peut, d'ailleurs, essayer de compenser ces sacrifices ou de limiter les désagréments occasionnés, en imaginant des dispositifs complémentaires<sup>4</sup>.

<sup>4</sup> Par exemple, dans les cas où un doute subsisterait sur le classement d'un ouvrage : soit acheter plusieurs exemplaires et en mettre un à chaque endroit candidat ; soit insérer entre les ouvrages des fiches qui renvoient à d'autres localisations...

## Le socle (suite)

### La notion de « service »

#### L'architecture de services

L'architecte logique a le choix entre plusieurs styles ou plusieurs approches d'architecture logique. Par exemple : l'architecture fonctionnelle adopte le critère de la fonction pour décomposer le système ; beaucoup d'architectures appliquent un schéma systémique ; le vocabulaire s'appuie, aussi, sur la métaphore de l'urbanisme.

Actuellement, c'est l'approche de l'architecture de services qui est le plus souvent retenue.

**Une architecture de système d'information est dite « architecture de services » ou « orientée services » si elle structure le système avec l'unité élémentaire du service logique.**

Dans une architecture de services, il n'existe pas de composants plus fins que le service logique : tout besoin d'information, d'action ou de transformation est satisfait par un service.

#### La définition du service logique

La notion de service est suffisamment importante pour la définir selon trois angles : l'être, le faire, le devenir.

*Qu'est-ce qu'un service ?*

**Le service est le composant de plus bas niveau d'une architecture logique.**

D'où l'expression « architecture de services ».

C'est l'atome, le grain élémentaire, dans la construction logique du système.

*Que fait un service ?*

**Un service est la réponse élémentaire donnée par le système à un besoin d'information, d'action ou de transformation.**

Cela évacue l'accès direct aux données ainsi que les manipulations « à distance ».

*D'où vient le service ? que devient-il ?*

Le service logique est une unité de conception, décrite dans l'architecture logique à partir de la modélisation amont (sémantique ou pragmatique).

Il se traduira en un composant logiciel, en application des choix techniques. Ce composant logiciel sera lui-même localisé sur une ou plusieurs machines et deviendra activable au cours d'une exécution.

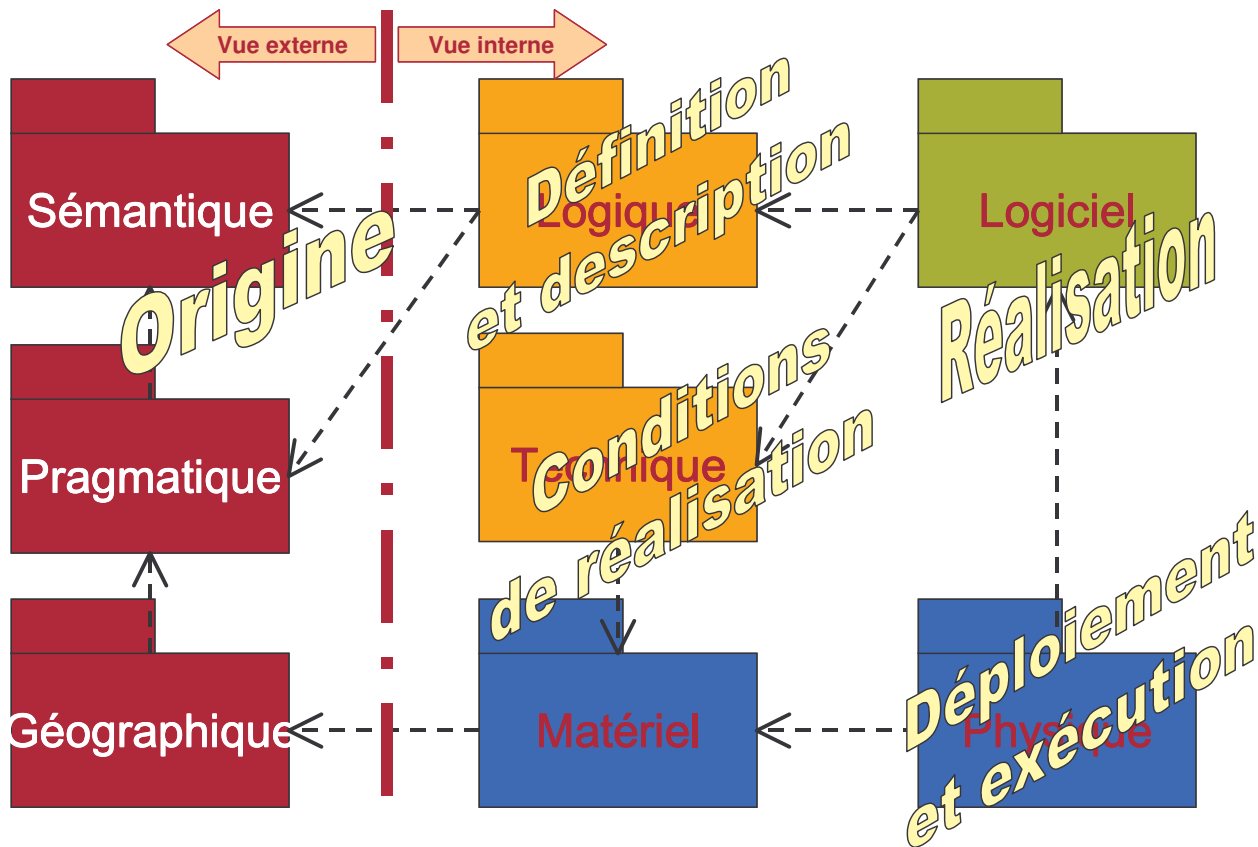
Ce « cycle de vie » du service logique fournit la trame de la chaîne de production, telle que montrée schématiquement page suivante.



## Le socle (suite)

### La chaîne de production

Figure PxM-02\_5. Le cycle de vie du service, rapporté aux aspects de la topologie du Système Entreprise.



#### Notions connexes

Les services logiques (des milliers) sont rangés dans des agrégats logiques sur trois niveaux d'agrégation :

- les machines logiques (représentées par des classes UML) ;
- les ateliers logiques (paquetages regroupant des machines fortement liées) ;
- les fabriques logiques (paquetages correspondant aux domaines).

La nature d'un service logique dépend de sa localisation (voir figure suivante) :

- soit dans une « machine logique métier » (MLM), dans le noyau du système (une MLM dérive d'une classe sémantique) ;
- soit dans une « machine logique organisation » (MLO), dans une couche intermédiaire.

La partie « Les produits » aborde ces notions plus en détail.

#### La justification

Le choix de l'architecture de services permet des gains en qualité sur la structuration du système, même en l'absence de technologies orientées objet. Il entraîne la production de composants logiciels publiés en tant que services, dans le droit fil des technologies *web services*. C'est donc un atout pour l'ouverture du système.



## Les produits

### Les termes généraux

#### La portée

La volonté d'urbaniser le système d'information conduit à distinguer :

- les objectifs de **portée locale** : la solution – spécifique – répond à un besoin exprimé par un client interne (application, processus métier, domaine...);
- les objectifs de **portée globale**, qui mobilisent l'entreprise sur le long terme.

La portée locale se caractérise par la connaissance du besoin, une réalisation spécifique, une échéance à court terme. C'est tout le contraire pour la portée globale.

Quelle que soit la portée, les aspects de la réalité à examiner sont les mêmes. Aussi, la topologie s'applique à ces deux niveaux d'action.

#### Le modèle

**Un modèle est une représentation pertinente d'une réalité.**

En génie logiciel, un modèle comporte :

- des représentations graphiques (en UML : des diagrammes) ;
- des informations sur les éléments représentés (définitions, descriptions, informations quantitatives) ;
- d'éventuelles justifications des choix de modélisation.

Chaque développement requiert un modèle, préalable à la réalisation.

#### Le modèle et les aspects

Plusieurs options se présentent, entre les deux extrêmes :

- chaque modèle ne traite qu'un aspect ;
- un modèle traite tous les aspects nécessaires à la satisfaction d'un besoin.

On pourra fondre dans un même modèle certains aspects connexes.

La formule adoptée dépend de l'ampleur du domaine. Dans la mesure du possible, elle devrait être la même pour tous les projets.

La solution préconisée est d'utiliser une seule base de modélisation, structurée en paquetages : chaque paquetage représente un des huit aspects de la topologie.

#### Le référentiel

**Un référentiel est un ensemble d'objets ou d'informations, partagés par une communauté d'acteurs.**

Cette notion de partage est essentielle à l'urbanisation. Elle s'inscrit dans le même mouvement que l'approche par composants, en faveur de la réutilisation.

Il est souhaitable de mettre en place un référentiel pour chacun des aspects : « Référentiel Assurance » pour le modèle sémantique étendu à toute l'entreprise, « Organisation » pour le modèle pragmatique, les architectures...

## Les produits (suite)

### Le modèle sémantique : Aller à l'essentiel pour dégager un noyau stable

#### L'attitude

Le modélisateur qui se donne pour tâche d'exprimer la sémantique, aborde le réel, sans *a priori*. Contrairement aux apparences, cette attitude n'est pas spontanée. Elle exige, du modélisateur, un effort particulier et sans cesse renouvelé pour faire abstraction des contingences organisationnelles et techniques. La qualité du modèle sémantique repose sur sa capacité à s'affranchir des pratiques actuelles et de la solution existante.

De plus, loin de s'embarrasser de l'apparente complexité du domaine, il doit capter l'essentiel et dégager le noyau fondamental.

Il lui faudra défendre la simplicité de son modèle face à la tendance générale à la complication. Une de ses réponses consistera à montrer comment ce modèle essentiel restitue la réalité et comment il peut se « déplier », s'instancier pour prendre en compte la diversité des situations concrètes.

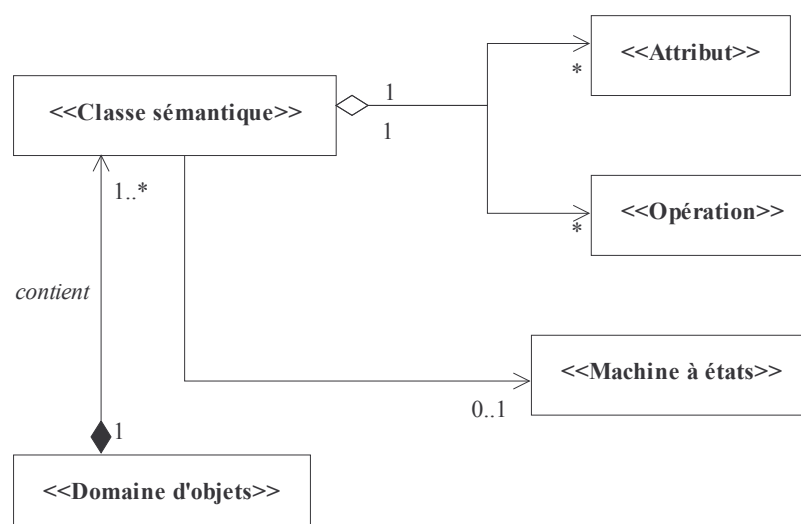
#### Les termes

Les termes de la modélisation sémantique, telle que pratiquée dans Praxeme, sont ceux de l'approche orientée objet. En effet, le but étant de représenter l'univers mental de l'utilisateur, le modélisateur recherche un moyen d'expression qui soit le plus naturel possible. La notation objet, aujourd'hui standardisée avec UML, fournit un bon moyen, à condition de le maintenir au bon niveau d'expression.

On parlera, donc, en termes de classes, de propriétés de classes (informatives : les attributs ; actives : les opérations ; associatives : les relations). Les propriétés structurelles sont assurées par les relations : héritage, association. Le modèle se structure en domaines d'objets, représentés par les paquetages (voir dans la partie « Les procédés », la décomposition en domaines d'objets).

Ces notions étant les catégories de représentation standard, le schéma ci-dessous n'est qu'une vue simplifiée du méta-modèle UML.

Figure PxM-02\_6. La terminologie pour la modélisation sémantique



## Les produits (suite)

### Le modèle pragmatique : Exprimer le besoin – local – et garantir la cohérence – globale

**L'attitude** L'attitude du modélisateur est très différente selon qu'il élabore la Vue de l'utilisation ou la Vue de l'organisation.

*La Vue de l'utilisation* Dans le premier cas, il adopte une approche fonctionnelle et cherche à restituer les demandes et préoccupations de l'utilisateur. Il regarde le système, en quelque sorte, par le « petit bout de la loupe ». Son attitude est marquée par l'empathie à l'égard de l'utilisateur. Ceci peut le condamner à rester collé aux pratiques actuelles.

*La Vue de l'organisation* Dans le second cas, il s'intéresse à la circulation des informations et au devenir des objets dans l'ensemble du système. Il dépasse le cloisonnement imposé par la première approche et embrasse du regard tout le système. La conception des processus peut pousser à la simplification.

### Les termes

La modélisation de l'aspect pragmatique recourt à des catégories UML classiques :

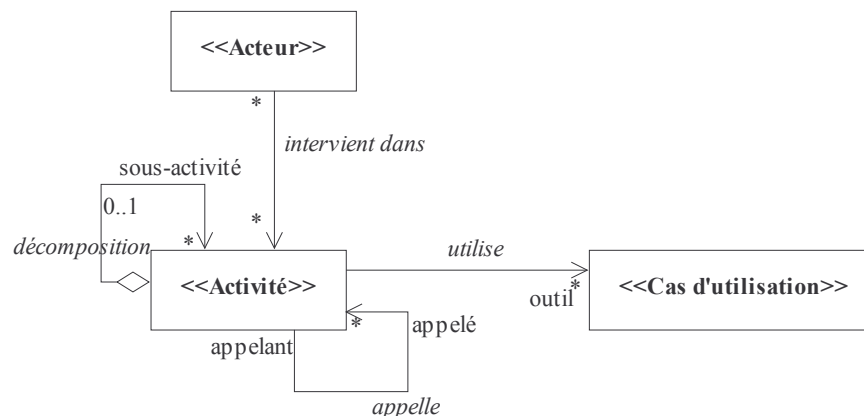
- L'acteur (en fait, le type d'acteur).
- L'activité : notion à maille variable qui peut couvrir le processus métier et descendre jusqu'au niveau de détail de l'opération (ou en-dessous), en passant par le cas d'utilisation.
- Le cas d'utilisation : interaction élémentaire entre un acteur et le système (assimilable à une « transaction fonctionnelle » : l'exécution d'un cas d'utilisation implique un seul acteur, et se déroule de façon insécable, à moins qu'elle avorte).

Cette définition du cas d'utilisation impose une restriction par rapport à UML, ce qui doit guider la mise en œuvre. Le critère pour identifier un cas d'utilisation est la motivation de l'acteur.

Par ailleurs, le système peut nécessiter de modéliser des objets de nature organisationnelle (dossiers, structures...). Dans ce cas, on utilise les catégories évoquées pour la modélisation sémantique.

Le diagramme ci-dessous est, lui aussi, une lecture simplifiée du méta-modèle UML.

Figure PxM-02\_7. La terminologie pour la modélisation pragmatique



## Les produits (suite)

---

### Le modèle logique : Structurer pour durer

---

#### L'attitude

La finalité de l'architecture logique et de la conception logique est de structurer le logiciel :

- en tenant compte des évolutions possibles ;
- en gardant présent à l'esprit, l'objectif stratégique.

Parmi les attendus, on trouvera, par exemple :

- la volonté d'urbaniser le système d'information pour réduire sa redondance et partager des composants ;
- l'ouverture du système et son intégration avec des systèmes partenaires<sup>5</sup> ;
- l'agilité de l'organisme ou, au moins, la facilité de reconfiguration organisationnelle ;
- la possibilité de multiplier, à moindre coût, les types d'interfaces offertes...

L'attitude par rapport à la technologie se définit par le principe d'indépendance logique, exposé plus loin.

---

#### L'enjeu

L'architecture logique donne une vision unifiée par-dessus la diversité des réalisations techniques.

Il y a, et il y aura toujours, un décalage entre la façon de représenter la réalité (concepts, objets, processus...), d'un côté, et la réalité du logiciel, de l'autre. Il est possible, cependant, de prendre certaines décisions de structuration du système logiciel au niveau logique. Ces décisions se soustraient, au moins en partie, aux contraintes et changements de nature technique.

L'architecture logique répond à des choix généraux de structuration du système logiciel, définis par la politique logiciel ou la stratégie de l'entreprise.

---

#### L'objectif de la conception logique

**L'architecture logique a pour but d'élaborer la structure optimale pour le logiciel, indépendamment des choix techniques définitifs.**

La conception logique apporte les précisions nécessaires sur les composants logiques.

---

<sup>5</sup> On parlera, alors, de SIIO : système d'information inter-organisationnel.

## Les produits (suite)

### Le modèle logique : un langage à part

#### La terminologie

Du fait de sa position intermédiaire entre, d'un côté, la vue externe du système (la vraie vie) et, de l'autre, le système informatique, le modèle logique doit se doter d'un vocabulaire qui lui est propre. Ce vocabulaire doit permettre de restituer la réalité (sémantique et pragmatique) dans le système, tout en facilitant les décisions de structuration de celui-ci.

Dans notre cas, la terminologie de la modélisation logique s'élabore à partir de la notion de service logique (cf. définition p. 15).

#### Les termes

Le service logique est le grain élémentaire du système, décrit sur le plan logique.

Les milliers de services s'ordonnent dans différents niveaux d'agrégats logiques.

*Figure PxM-02\_8. L'emboîtement des agrégats logiques*

Le tableau ci-dessous en donne les définitions, assorties d'exemples de règles.



<b>Les agrégats</b>	<b>Machine logique</b>	<b>Atelier logique</b>	<b>Fabrique logique</b>
<b>Leur définition</b>	Ensemble cohérent de services logiques	Ensemble de machines logiques	Ensemble d'ateliers logiques
<b>Le critère de délimitation</b>	Les services portent sur la même classe (la même notion). On sépare les services élémentaires et les services ensemblistes.	Les notions encapsulées par les machines de l'atelier sont proches.	Les fabriques correspondent sensiblement aux domaines d'objets, augmentés des dispositifs transverses.
<b>Leur contenu<sup>6</sup></b>	Correspond à la manipulation d'une ligne dans une table (un objet).	Correspond à une table (tous les services imaginables, synchrones et asynchrones).	Correspond à une base de données.
<b>Leurs relations</b>	Des machines d'un même atelier peuvent (par dérogation) entretenir des relations d'utilisation mutuelle.	Les machines attaquant une même table sont rangées dans un même atelier.	Les ateliers puisant dans une même base de données (ou un ensemble cohérent de BD) sont installés dans une même fabrique.

<sup>6</sup> Ces remarques relatives aux données ne sont qu'indicatives. En fait, l'architecture des données peut être plus compliquée, voire totalement découplée de l'architecture des services. C'est le cas quand l'architecture de services se déploie en conservant des bases de données existantes. Une seule chose demeure, dans tous les cas de figure, quand on choisit l'approche SOA : le plan des services surplombe et masque le plan des données.

## Les produits (suite)

### Le modèle logique : la stratification du système

#### Les types de machines logiques

Selon leur provenance, on distinguera plusieurs catégories de machines logiques :

- Les machines qui traduisent une classe sémantique sont dites « **Machines Logiques Métier** » (MLM) ou « distributeurs ».
- Les machines qui expriment des choix d'organisation sont dites « Machines Logiques Organisation » (MLO) ou « orchestrateurs ». Elles sont délimitées sur un critère qui peut être le cas d'utilisation, le type d'acteur, le processus, etc.
- D'autres machines émergent de considérations logiques ou techniques. On les appellera, parfois, « **Machines transverses** » ou « utilitaires ». Elles fournissent des dispositifs généraux tels que la gestion des événements ou les codifications.

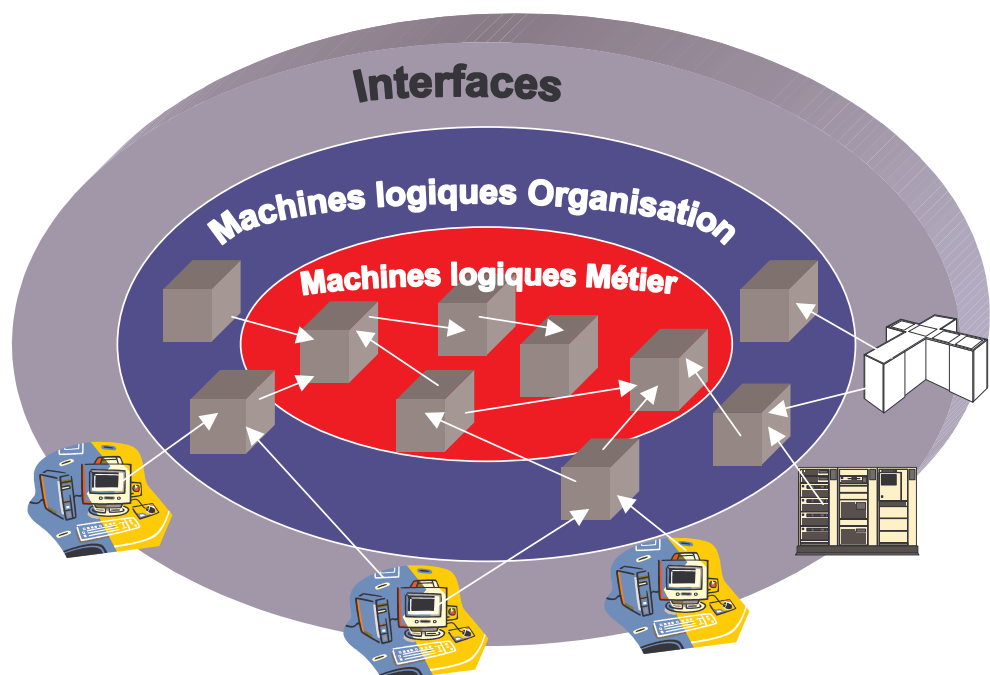
L'architecture logique fixe les règles d'identification, de structuration et de conception qui s'imposent à tous les éléments de l'aspect logique. La limitation du couplage et la maîtrise du système motivent ces règles.

#### La stratification

Par application des règles d'architecture, le système d'information se décompose en trois cercles :

1. Un noyau constitué des « machines logiques métier ».
2. Un cercle intermédiaire isolant les choix d'organisation avec « les machines logiques organisation ».
3. Une périphérie qui fournit les points d'entrée dans le système avec un assortiment de postes de travail ou autres interfaces (éventuellement, vers d'autres systèmes).

Figure PxM-02\_9. La stratification du système d'information, déduite des règles de l'architecture logique



## Les produits (suite)

### L'élaboration de l'architecture de services

#### Les termes

En plus des termes présentés jusqu'ici, la conception logique distingue :

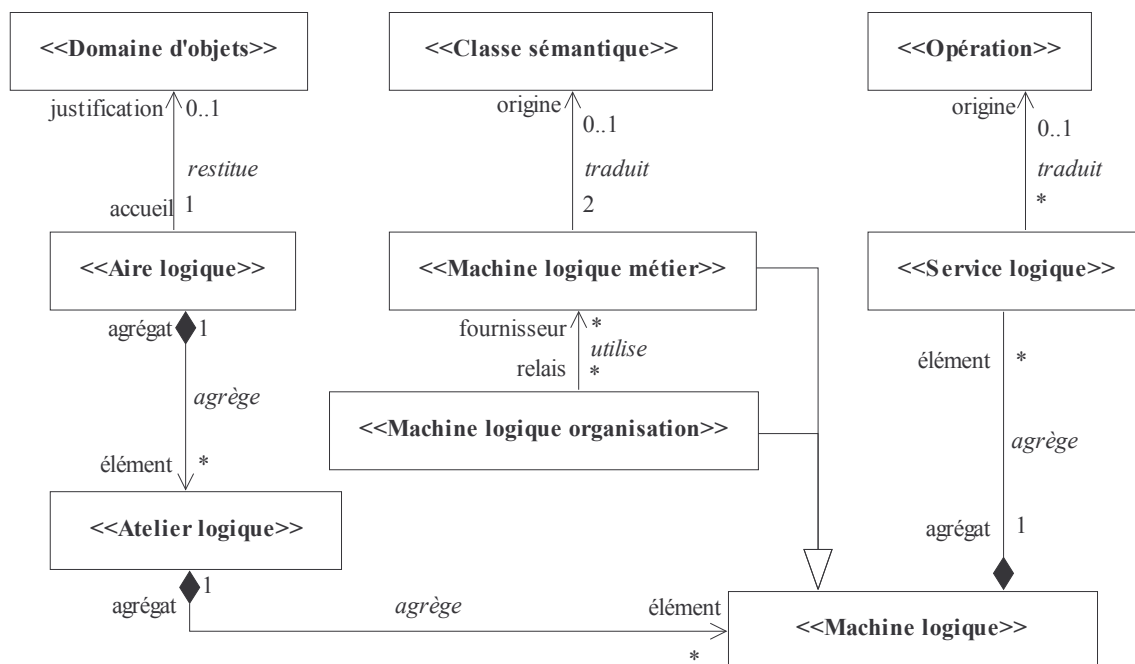
- les **machines logiques individuelles**, dont les services traitent une seule instance (un seul objet ou occurrence) ;
- les **machines logiques ensemblistes**, qui rassemblent les services tels que l'instanciation, les requêtes, les statistiques.

L'architecture logique traduit les comportements d'une classe sémantique en deux machines logiques, une de chaque catégorie. Les propriétés de la classe (y compris les associations) sont distribuées sur ces machines en fonction de leur portée.

#### Les correspondances

Le méta-modèle ci-dessous établit des correspondances simples entre les éléments de modélisation des aspects amont et les éléments logiques.

Figure PxM-02\_10. Le méta-modèle Praxeme pour l'aspect logique



#### Commentaire

Le diagramme de classes ci-dessus ne présente qu'une partie du méta-modèle. Il faut savoir, surtout, que des services logiques peuvent être déduits autrement qu'à partir des opérations des classes sémantiques : à partir des attributs (services accesseurs), mais aussi à partir des associations, dans certains cas de figure.

NB : « aire logique » équivaut à « fabrique logique ».

## Les produits (suite)

---

### La couverture du système par les modèles

---

#### Le principe

La topologie du Système Entreprise fournit le principe structurant auquel Praxeme se réfère pour définir les modèles et les livrables. Elle offre l'avantage de définir les modèles comme des représentations rigoureuses et complètes, limitées à un aspect clairement identifié. Elle montre, également, les dépendances entre les modèles. Ces dépendances indiquent comment les informations se relient les unes aux autres, d'un modèle à l'autre. Chaque métier de la modélisation connaît ainsi ses responsabilités.

*Figure PxM-02\_11. L'articulation des modèles par aspects*



## Les produits (suite)

### Les livrables

#### Le principe

Les livrables se définissent, en fonction du processus, aux points de visibilité créés par les fins de phases. Ils répondent à un besoin de communication et de décision. Placée dans la dimension du Produit, la définition des livrables obéit à une exigence : l'ensemble des livrables doit couvrir tout le champ de l'étude. Praxeme quadrille ce champ en croisant la liste des aspects et la portée. C'est ainsi que les livrables se définissent, comme le montre le tableau ci-dessous.

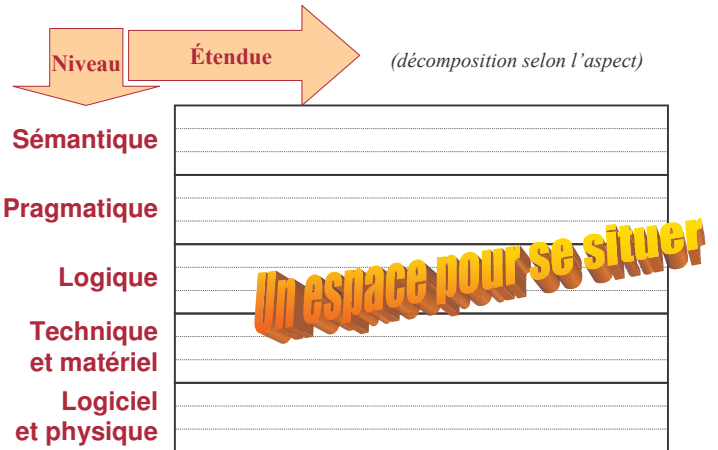


Figure PxM-02\_12. Tableau d'ensemble des livrables

Aspect	Portée locale (Projet)		Portée globale (Système)	
10 - Sémantique	FRM-02 Dossier de Conception des Scénarios	Modèle Sémantique	Référentiel Assurance	
20 - Pragmatique		FRM-24 Dossier des cas d'utilisation	FRM-24 Fiches C-U Dossier des processus	
40 - Logique			FRM-45... Fiches sv FRM-40 Dossier d'Architecture de Services	
30 - Géographique				
50 - Technique				
60 - Matériel				
70 - Logiciel		Dossier de maquette	<b>Maquette</b>	
80 - Physique		Composants		

## Les produits (suite)

---

### Les livrables (suite)

---

**Dossiers et fiches** Parmi les livrables, on distingue :

- **Les dossiers de projet** : ils fixent les décisions dans la progression d'un projet et n'ont de valeur que dans le temps de ce projet.
- **Les dossiers de portée globale** : ils constituent une référence pour l'ensemble de l'entreprise. Ils peuvent évoluer mais sous contrôle d'une autorité centrale.
- **Les fiches** : elles sont éventuellement produites à l'intérieur d'un projet mais leur format est conçu pour en faire des éléments communs et partageables. Elles acquièrent, donc, une visibilité qui dépasse le projet. Elles obéissent à un cycle de vie qui tient compte de leur exploitation par de multiples acteurs.

---

### Positionnement par rapport aux pratiques

Par définition, les spécifications fonctionnelles expriment des exigences en termes de fonctions. Pour aller à l'essentiel et exprimer les besoins par rapport à l'activité des utilisateurs, le dossier des cas d'utilisation convient parfaitement. Il ne traite pas des « écrans » ; cet aspect, susceptible de décisions de conception, est laissé à la maquette et au dossier qui l'accompagne.

L'opposition général/détail (SFG/SFD) n'est pas structurante dans Praxeme. En effet, chaque aspect demande une modélisation formelle et complète. Le niveau de détail et d'exhaustivité de chaque livrable peut être ajusté en fonction du contexte du projet et des besoins du moment, mais un modèle n'est réputé définitif que s'il décrit pleinement le champ d'étude, dans le périmètre assigné et pour un aspect de la topologie.

En revanche, la notion de portée permet d'introduire des livrables dont dépend la continuité de l'urbanisation et des choix au niveau du système.

---

### La conception des scénarios

Dans la documentation du projet, Praxeme prescrit le Dossier de conception des scénarios. Ce dernier offre un outil pour :

- appliquer la topologie à l'échelle d'un projet,
- stimuler l'invention des scénarios, donc l'effort de conception,
- exprimer et évaluer les options par rapport aux choix globaux (urbanisation, architecture) et, par voie de conséquence, faire prévaloir l'intérêt général sur la logique de projet.

Ce dossier constitue un foyer de décisions à examiner lors des revues (voir plus loin, dans la partie « Processus »).

---

### La description des livrables

Les livrables sont décrits à travers des formulaires, assortis d'un commentaire. Ils peuvent, aussi, faire l'objet de profils UML installés dans un outil de modélisation.

## Le processus

### La mise en œuvre de la topologie

#### Les conséquences sur les travaux

La séparation des aspects conduit à isoler des éléments de la solution qui évoluent à des rythmes différents.

Ceci débouche sur les avantages suivants :

- Faciliter la définition des travaux (cf. p. 30).
- Rendre plus lisibles les décisions, en les sériant (cf. p. 28).
- Ordonner rigoureusement les informations dans la base documentaire du Produit (le Système dans son entier ou un domaine ou une application).
- Isoler un noyau stable (sémantique et logique), en le préservant des variations organisationnelles et des innovations technologiques.
- Gérer les compétences grâce à la spécialisation des disciplines (cf. 34).

La séparation des aspects engage, donc, à introduire, dans la structure de la solution, le souci de maîtrise à long terme et les exigences qualité telles que l'adaptabilité du Produit, la robustesse et l'indépendance par rapport aux technologies. L'urbanisation du Système d'information y trouve une solide base de départ (cf. p. 33).

#### Les retombées

La topologie constitue un premier niveau de réponse aux préoccupations de la DI et à l'inscription du SI dans la cible stratégique de l'entreprise :

- Elle aide à concevoir le Système d'information de façon modulaire pour permettre de l'adapter aux technologies, de l'ouvrir aux partenaires et de multiplier les interfaces.
- La topologie sépare des éléments de conception qui évoluent selon des rythmes et des logiques différentes. Elle incite, donc, à structurer le logiciel pour qu'il puisse absorber plus facilement les changements.

#### L'orientation « services »

L'orientation « services » renforce les bienfaits de la séparation des aspects. Le développement ou l'acquisition de services (plutôt que de simples classes) impose des règles de conception supplémentaires. La conception vise un *middleware* intégrant des composants, dont certains peuvent être publiés, trouvés sur le marché ou échangés avec des partenaires.

La notion de composant vaut pour les aspects :

- logiciel (tout particulièrement en recourant aux technologies portant le courant web services) ;
- logique (c'est son domaine natif).

## Le processus (suite)

---

### Modéliser : entre analyser et concevoir

---

#### **Analyse *versus* conception**

La dichotomie « analyse / conception » est un grand classique du génie logiciel (et, plus généralement, des sciences de l'ingénieur).

Cependant, les évolutions de ces dernières années (notamment, la généralisation des processus itératifs) ont eu tendance à brouiller cette opposition simple.

Les démarches de développement traditionnelles identifiaient l'activité (d'analyse ou de conception) et la phase (segment temporel, unité de découpage des travaux). Elles obéissaient au modèle de la cascade. Les noms des phases désignaient les activités. Il nous faut distinguer, désormais, entre ces deux notions, une phase pouvant mêler ces deux types d'activités. Dans les cycles de développement itératif, une itération comporte souvent, à la fois, de l'analyse et de la conception.

Pourtant, même si les outils mobilisés peuvent être les mêmes, l'analyse et la conception supposent deux attitudes radicalement différentes.

---

#### Les définitions

*Analyser* **Analyser, c'est observer.** Le terme évoque la décomposition en éléments plus fins, le souci du détail.

La posture du modélisateur qui analyse est caractérisée par :

- la passivité (il ne prend pas d'initiative ; il se « contente » de restituer ce qu'il observe) ;
- le souci du détail et de l'exhaustivité ;
- éventuellement, la traque des dysfonctionnements en vue d'alimenter la réflexion critique (analyse de l'existant, audit).

*Concevoir* **Concevoir, c'est inventer.**

La posture du modélisateur est alors très différente. On attend de lui qu'il imagine une ou plusieurs solutions à un problème posé ou à un besoin exprimé.

Les caractéristiques de cette activité sont :

- la prise en compte des exigences (qui ont été, préalablement, formulées et analysées) ou couverture du besoin ;
- l'inventivité, appuyée sur la connaissance de l'état de l'art et des possibilités ;
- l'évaluation économique des solutions proposées...

## Le processus (suite)

### Les niveaux de cibles

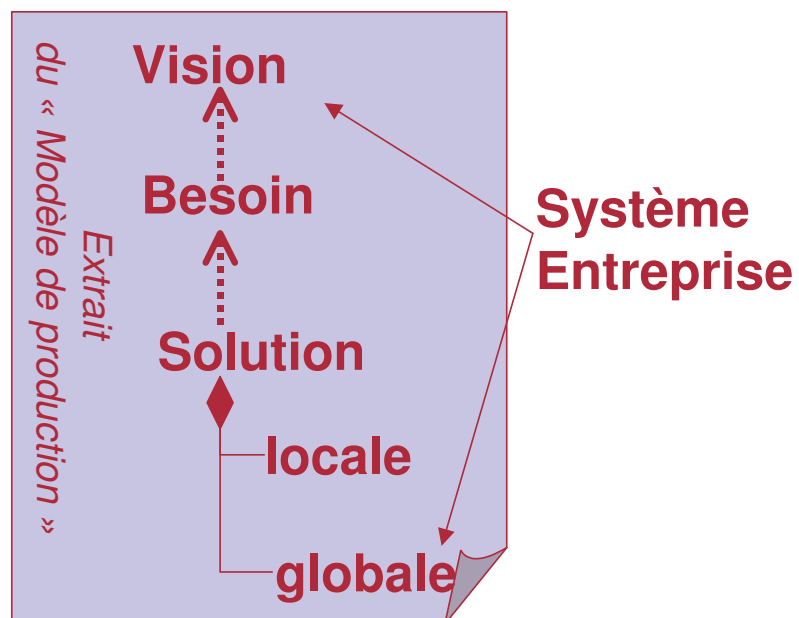
#### Les éléments qui polarisent les activités

En plus du besoin et de la solution locale (réponse spécifique à un besoin), l'urbanisation ajoute deux grandes catégories d'objectifs :

- le « **Système** » lui-même, c'est-à-dire la solution globale, maîtrisée dans sa structure, ses évolutions et son contenu ;
- la « **vision** » ou l'analyse et l'anticipation des besoins (regard tourné vers l'extérieur : la stratégie de l'entreprise, le marché, les possibilités et préoccupations futures).

Autrement dit, le même mouvement d'urbanisation concilie la conception globale du système, d'un côté, et l'analyse prospective du marché, de l'autre.

Figure PxM-02\_13. Les niveaux de cibles



#### Les conséquences

Figure PxM-02\_14.  
Les macro-activités en regard des niveaux de cibles

Pour servir l'entreprise sur tous ces niveaux de cibles et contribuer à ses ajustements opérationnels, organisationnels ou stratégiques, les processus de la DI incluent les activités nécessaires. Le schéma ci-dessous en donne un aperçu (détail en annexe).



## Le processus (suite)

### La démarche : les possibilités de parallélisme

#### Les fondements de la démarche

Le schéma de la page 24 montre comment les éléments de description du système s'articulent. La démarche de développement respecte cette logique de dépendance entre les produits. On déduit de ce schéma :

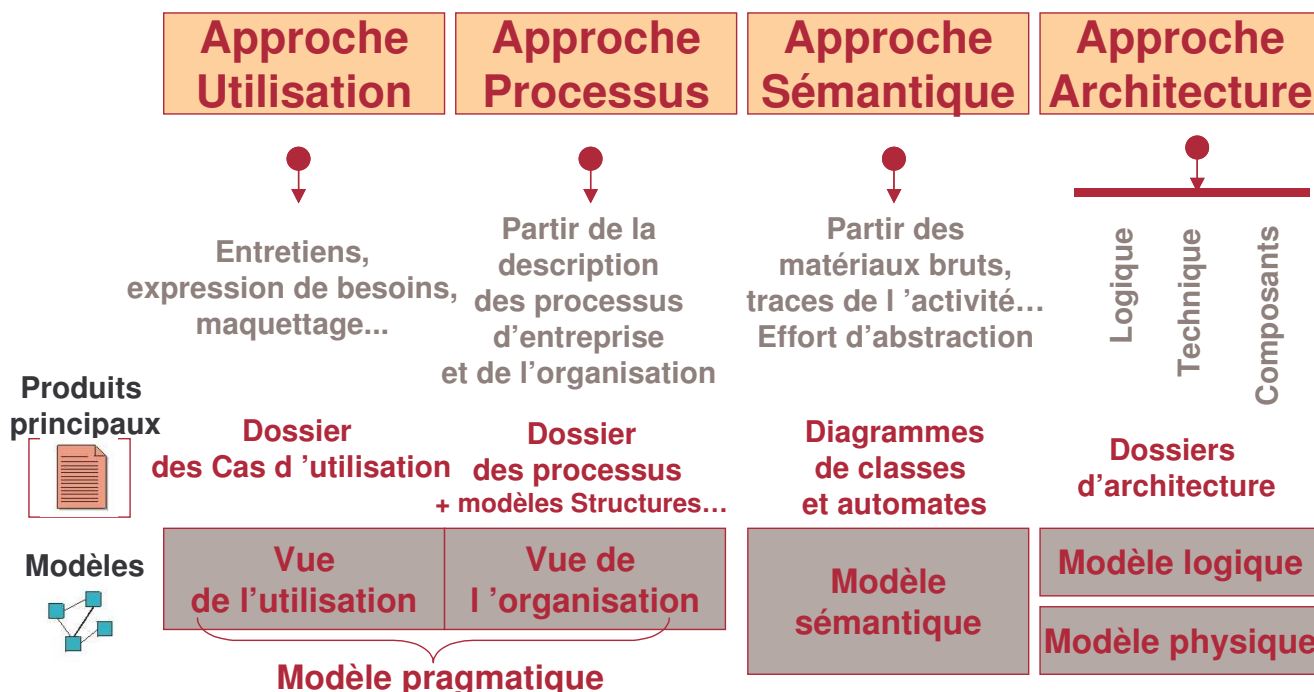
- La possibilité d'un parallélisme entre les travaux de modélisation : entre les axes sémantique et pragmatique.
- La source de la conception des services (la conception logique s'alimente des modèles sémantique et pragmatique).
- La prise en compte des contraintes et objectifs par l'architecture logique.
- La nécessité de points de synchronisation entre les axes.

#### Le parallélisme

Sans interdire le parallélisme, les relations de dépendances entre les modèles imposent quelques précautions. Le schéma de la page 24 signale une dépendance entre la vue de l'utilisation (les dossiers des cas d'utilisation) et le modèle sémantique. Précisément, cette dépendance résulte de la valeur accordée à la sémantique dans Praxeme et de notre souci de fonder la documentation sur le substrat du Référentiel Métier (sémantique).

Ceci entraîne une conséquence pratique : la démarche peut mettre en parallèle les travaux sur les modèles sémantiques et pragmatiques à condition de ménager des points de synchronisation réguliers. Ces points fourniront l'occasion de vérifier la convergence entre les produits élaborés sur les deux axes. Le bon déroulement d'une telle démarche repose sur les **réunions de coordination**.

Figure PxM-02\_15. Les axes de la conception



## Le processus (suite)

### La démarche : les travaux d'architecture

#### Les contraintes et objectifs

La première étape de cette approche est l'analyse architecturale. Elle recueille l'expression des contraintes et des objectifs, perçus avant tout aux niveaux de cibles les plus élevés.

Ce travail permettra d'ajuster le système d'information aux intérêts de l'entreprise et d'anticiper les transformations futures. Cette matière entrera, également, dans les arbitrages qui ne manqueront pas entre différents scénarios de conception ou entre des intérêts locaux divergents. De plus, l'analyse architecturale prépare les discussions budgétaires, sur le long terme.

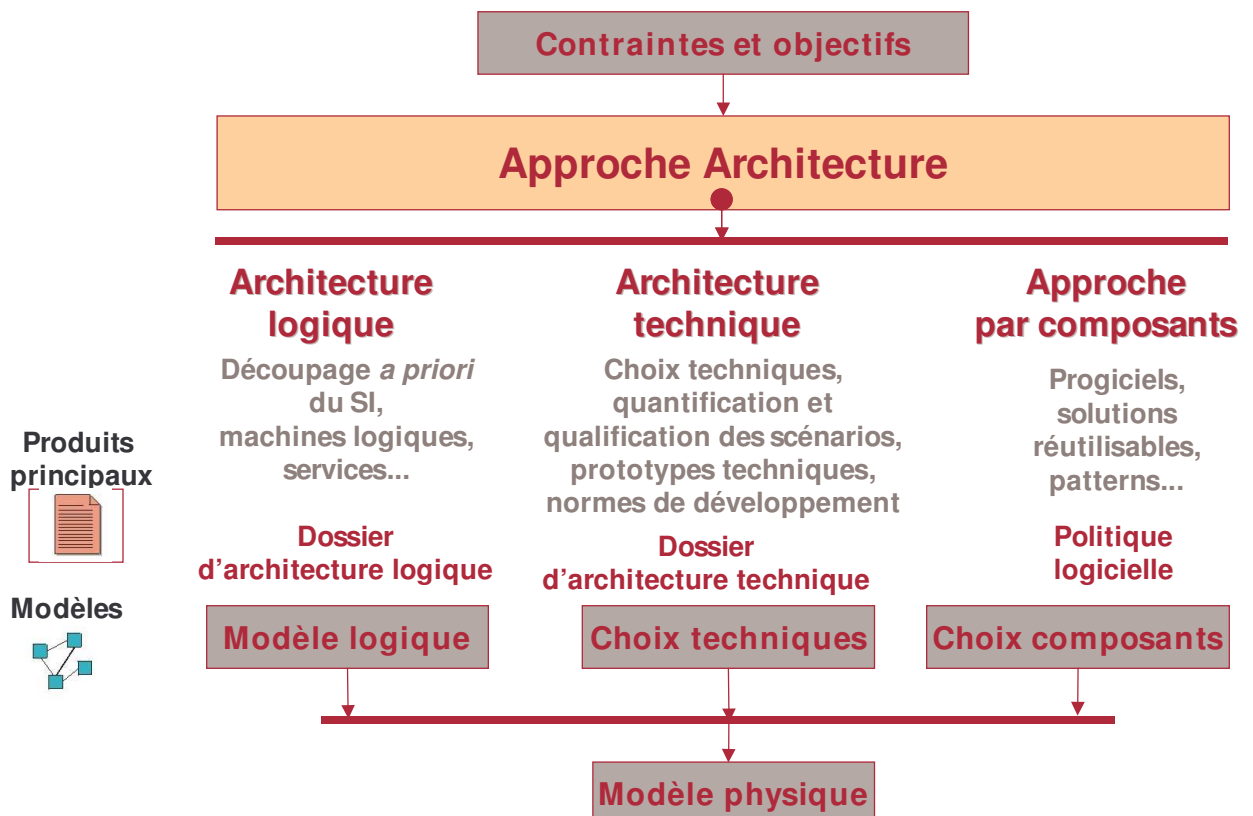
#### Trois sous-approches

L'architecture est la discipline de conception qui fixe les choix globaux, c'est-à-dire qu'elle s'intéresse au système dans son entier. Elle se scinde en trois sous-approches :

- L'architecture logique qui sera investie par le discours de l'urbanisation (voir pp. 33 et sq.).
- L'architecture technique (cf. le dossier type FRM-50).
- L'approche par composants pour les parties du système qui peuvent être couvertes par des solutions du marché.

Ces sous-approches peuvent être menées en parallèle, à condition de prendre quelques précautions, tout particulièrement la négociation logique/technique (voir p. 43).

Figure PxM-02\_16. L'approche Architecture



## Le processus (suite)

### Les activités de portée globale

#### Leurs caractéristiques

À partir des niveaux de cibles, la dynamique globale (p. 29) identifie des activités nécessaires. La volonté de structurer le Système introduit de nouvelles activités, propres à assumer les objectifs et exigences de portée globale. En effet, les nouveaux objectifs requièrent des activités particulières qui se distinguent des activités courantes de développement par : leur visée, leur portée et leur échéance.

*La visée* Les objectifs de ces activités s'inscrivent dans la stratégie de l'entreprise.

*La portée* Ces activités considèrent l'ensemble du Système Entreprise (par opposition au processus de développement classique)<sup>7</sup>.

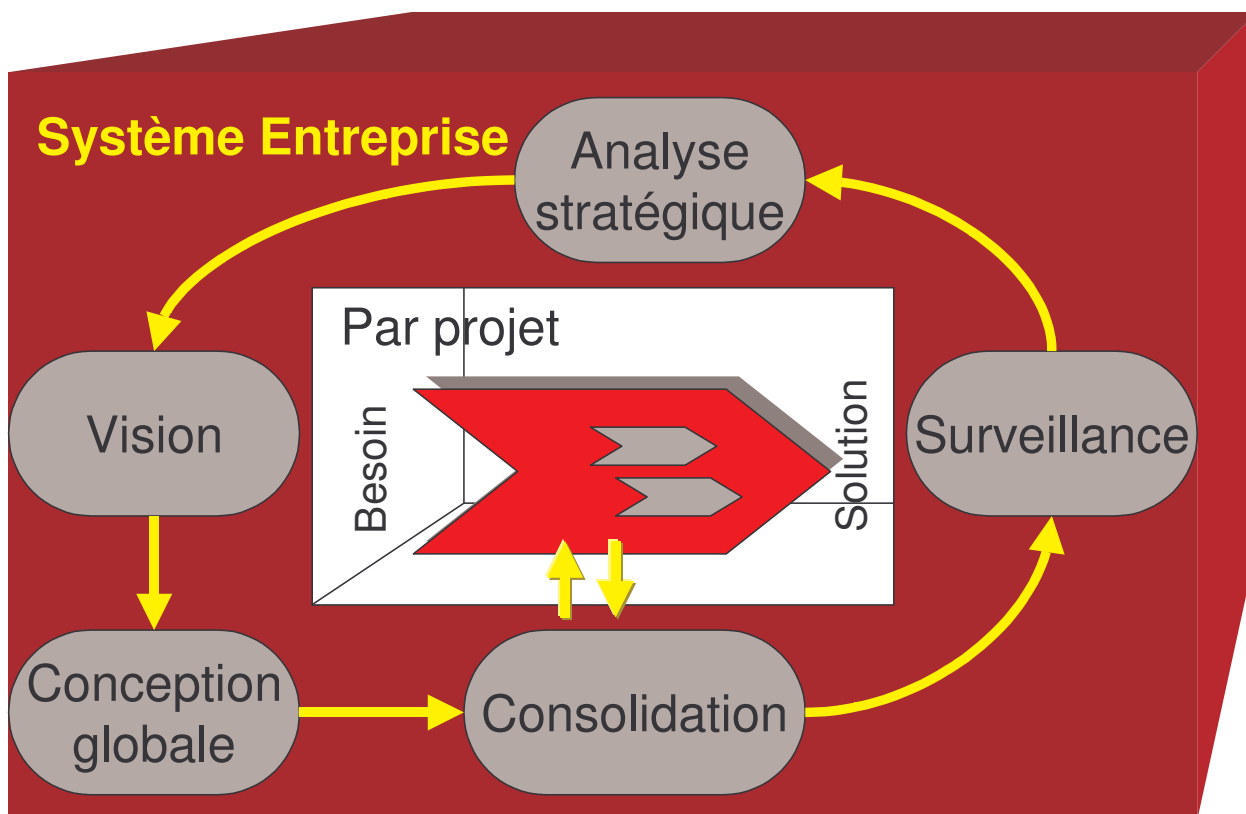
*L'échéance* Ces activités se déploient sur la longue durée.

#### Leur organisation

Nous entrons dans une nouvelle dynamique qui doit coordonner des activités au niveau du projet et d'autres au niveau du système.

Le schéma ci-dessous recense ces activités et montre leur positionnement par rapport au fonctionnement ordinaire de l'activité de développement.

Figure PxM-02\_17. Les activités de portée globale accompagnant les projets



<sup>7</sup> Même pour une demande mineure, il s'agit d'évaluer l'impact sur l'ensemble du système logiciel.



## Le processus (suite)

### L'urbanisation

#### L'urbanisation du système d'information

Au fil des années, l'empilement des développements logiciels a débouché sur un système très complexe, assemblage de fonctionnalités s'exécutant sur des matériels différents, répondant à des besoins perçus localement et soumises à l'obsolescence et aux changements de toutes sortes.

Afin de réduire cette complexité et de mettre les nouvelles possibilités technologiques au service de l'entreprise, il est nécessaire de réorganiser le système en une architecture plus économique. Cette architecture doit tenir compte des possibilités technologiques – en aval – mais surtout – en amont – de la vision d'entreprise, des contraintes et objectifs évoqués précédemment.

Une discipline vient au service d'une telle ambition : l'urbanisation.

**L'urbanisation est une discipline de conception qui a pour but de : structurer le système d'information pour favoriser son adéquation à la stratégie et aux besoins des entreprises**

L'urbanisation des systèmes d'information consiste, principalement, à adopter une attitude qui privilégie le global sur le local, le long terme sur le court terme.

#### Les notions liées à l'urbanisation

Pour l'essentiel, les procédés employés dans l'urbanisation sont ceux de la cartographie applicative (pour l'analyse de l'existant) et de l'architecture logique (pour la conception). Aujourd'hui, UML vient à l'appui de cette discipline et la transforme de l'intérieur en la dotant d'un langage précis. Cette technique de représentation assure la **continuité entre les représentations** globales de l'architecte et les représentations détaillées du concepteur. Cette continuité entraîne une véritable révolution dans la distribution des rôles et la définition des responsabilités.

*La cible d'urbanisation* L'acte fondateur de l'urbanisation, dans l'entreprise, est la publication d'une « cible d'urbanisation ». Il s'agit simplement d'un **graphe d'architecture logique** qui représente la « cité idéale », le système tel que l'on voudrait le construire. Cette cible sera d'autant meilleure que l'urbaniste sera capable de s'affranchir de l'existant, de simplifier le système et de le redessiner sur des critères qui en assurent la stabilité.

*La trajectoire d'urbanisation* La cible d'urbanisation ne peut être atteinte qu'après un parcours de plusieurs années, à condition qu'elle soit portée au plus haut niveau de la hiérarchie, seul capable de lui assurer la continuité de vision et de moyen nécessaire. Au-delà de la maîtrise opérationnelle, le succès d'une urbanisation repose sur le *leadership*.

La « trajectoire d'urbanisation » se compose d'une succession d'états stables du système, pré-définis, qui le conduisent, années après années, de l'existant à la cible. Chaque état fait l'objet d'un graphe d'architecture intermédiaire qui loge les investissements de l'année en les situant le plus possible dans la cible. De cette manière, l'investissement fructifie davantage...

## Le processus (suite)

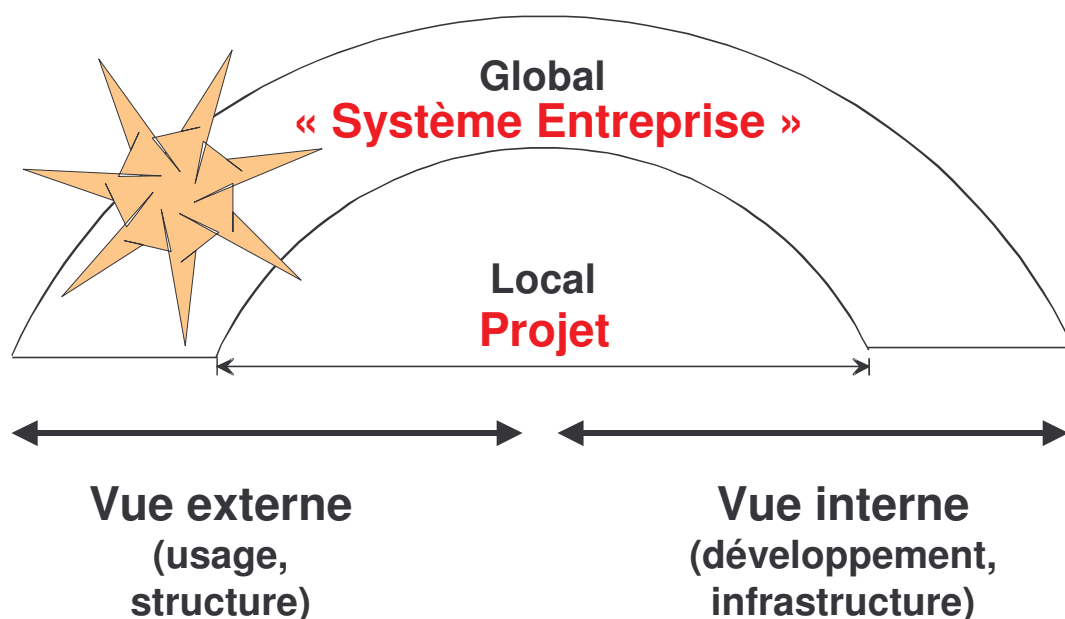
### L'organisation de l'urbanisation

#### La distribution des objectifs

Chacun des objectifs doit être assumé par un acteur. En ce qui concerne les objectifs de portée locale, c'est-à-dire ceux d'un projet, une méthode de conduite classique précise les rôles nécessaires et leur implication dans le processus de développement.

Reste à fixer les rôles associés à la visée long terme et à affecter les personnes. C'est ce que tente le tableau suivant.

Figure PxM-02\_18. Les responsabilités en fonction de la portée



Coordonnées	Global * Externe	Local * Externe	Local * Interne	Global * Interne
Fonction	Direction Générale	Directions métier	Développement	Direction Informatique
Rôle	Stratège	Maître d'ouvrage, utilisateurs	Chef de projet	Urbaniste et architectes
Individu	À renseigner dans un Plan de Production			
Interface avec :	Actionnaires	Clients		Fournisseurs informatiques

#### L'urbaniste

Si la plupart des rôles cités sont relativement courants, il faut insister sur celui de l'urbaniste, propre à la dynamique d'urbanisation des systèmes. L'urbaniste est celui qui porte l'objectif d'urbanisation ; il est le gardien de la cible d'urbanisation, ajuste la trajectoire à la vision d'entreprise ainsi qu'aux contingences rencontrées, assure l'interface avec l'organisateur et l'architecte technique. Il participe activement aux revues de conception où il fait prévaloir le point de vue global et stimule la réutilisation et le partage.

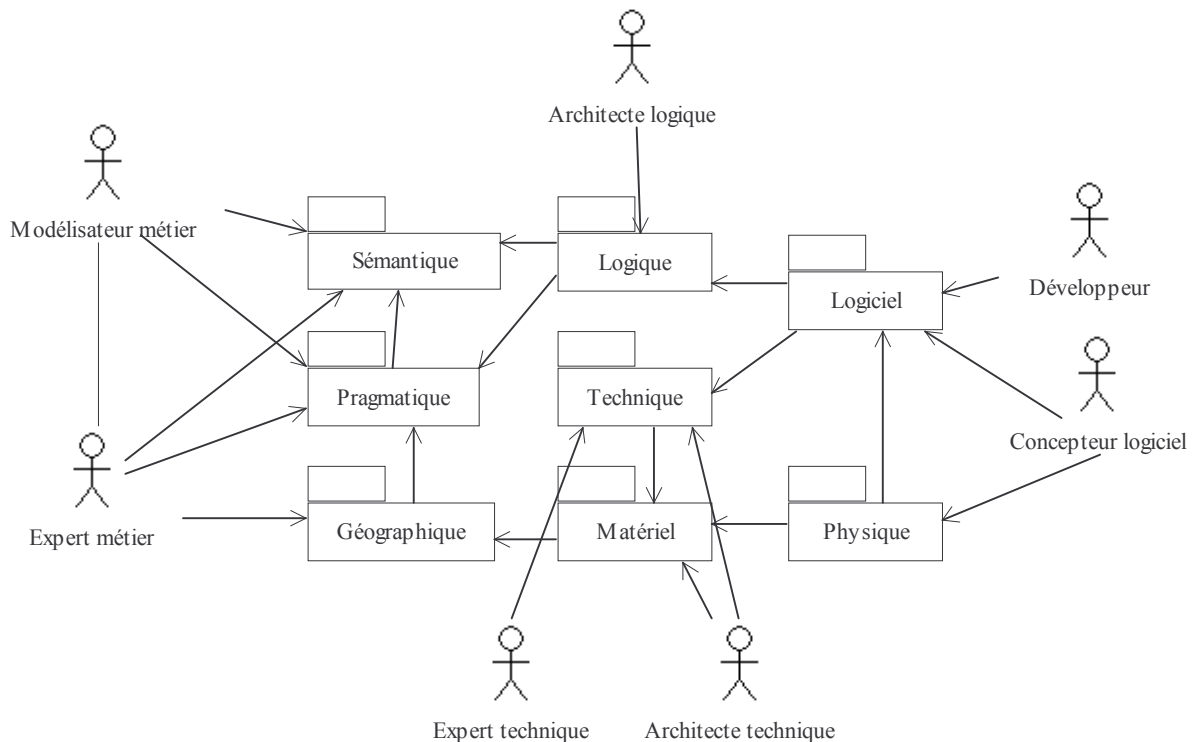
## Le processus (suite)

### Les responsabilités sur les aspects

#### La gestion des compétences

Le schéma ci-dessous propose une typologie des rôles, appuyée sur la topologie.

Figure PxM-02\_19. Les rôles dans Praxeme



#### Les revues

La revue se range parmi les procédés d'évaluation. Elle consiste à réunir des personnes habilitées à porter un jugement sur un produit ou un élément de produit. La dynamique d'urbanisation donne un relief particulier à certaines des revues de conception que le processus de développement doit absolument prévoir.

Les modèles sémantiques, pragmatiques et logiques produits au sein de chaque projet doivent être homologués :

- d'abord, pour la bonne fin du projet qui obtient ainsi des assurances sur ses objectifs et ses modalités (convergence vers la cible, réutilisation maximale, conformité des procédés, gestion des risques...);
- ensuite, afin d'apporter leur contribution à l'œuvre commune, chaque modèle homologué pouvant se déverser dans le référentiel ad hoc et enrichir le patrimoine commun.

Le Dossier de conception des scénarios entre tout à fait dans cette logique. Il balaie tous les aspects et fixe les options conformes aux objectifs du projet. La revue de ce dossier permet de jauger chaque scénario par rapport aux objectifs et enjeux globaux.

## Les procédés de modélisation

### Représenter avant d’agir

#### Définition

Les procédés sont les modes opératoires à appliquer dans les travaux. Par opposition au processus, ils se situent à un niveau individuel : ils guident le producteur dans une tâche précise. Ils peuvent se rattacher à une discipline (par exemple, la conception interne) mais peuvent aussi être « instrumentalisés » dans plusieurs disciplines (par exemple, un procédé de documentation). Parfois, ils s’appuient sur un outil.

#### Les procédés majeurs

Liste des procédés majeurs, qui peuvent s’insérer dans Praxeme.

Figure PxM-02\_20. La liste des procédés dans Praxeme

Procédé	Aspect concerné	Définition (objectif du procédé)
<b>Modélisation sémantique</b>	Sémantique	Représentation de la réalité étudiée, telle qu’en elle-même, c’est-à-dire abstraction faite des contingences organisationnelles et techniques
<b>Expression des besoins</b>	Pragmatique (Vue de l’utilisation)	Formulation des exigences fonctionnelles et opérationnelles par la technique des cas d’utilisation
<b>Preuve d’exhaustivité des cas d’utilisation</b>	Pragmatique (Vue de l’utilisation)	Vérifier l’exhaustivité du recensement des cas d’utilisation
<b>Conception des processus</b>	Pragmatique (Vue de l’organisation)	Conception des flots d’activités, non pas à partir des pratiques actuelles, mais à partir du cycle de vie des objets
<b>Architecture logique</b>	Logique	Conception de la structure d’ensemble d’un système logiciel, avec une relative indépendance par rapport aux choix technologiques
<b>Urbanisation (planification)</b>	Logique et Logiciel	Élaboration d’une cible pour canaliser les développements à long terme et les faire tendre vers une structure idéale. Élaboration de la trajectoire qui doit conduire à cette cible, en fonction des investissements.
<b>Architecture technique</b>	Matériel et Technique	Conception de l’infrastructure technologique : choix techniques, composants techniques, règles de développement et outillage associés aux choix techniques
<b>Élaboration des scénarios</b>	Tous les aspects	Détermination des scénarios pour la conception d’une solution (à un besoin exprimé peuvent correspondre plusieurs réponses qu’il s’agit d’évaluer pour retenir la « meilleure »)
<b>Conception interne</b>	Logiciel	Conception d’un élément logiciel à partir d’une spécification, selon l’approche objet et en respectant les règles de conception imposées par les objectifs généraux (urbanisation, réutilisation, etc.)
<b>Conception des tests</b>	Tous les aspects	Détermination des cas de test
<b>Simulation des modèles</b>	Tous les aspects	Vérification de la qualité d’un modèle en le réalisant et en l’exécutant dans un environnement technique <i>ad hoc</i>

---

## Les procédés de modélisation (suite)

---

### La modélisation sémantique : Aller à l'essentiel pour dégager un noyau stable

---

#### La définition

Dans son aspect sémantique, la représentation du Système vise :

- les notions, concepts et objets du domaine,
- leurs relations,
- les règles associées.

Cette représentation est débarrassée des contingences organisationnelles et techniques. Sa valeur ajoutée réside dans l'abstraction qui mène à la simplicité. Le modèle sémantique saisit l'essentiel ; cela le rend simple et stable.

---

#### L'enjeu

La simplicité de cette description libère l'imagination et permet au concepteur, ensuite, d'élargir sa palette de choix quant à l'organisation, à la logistique et aux technologies.

L'effort d'abstraction permet de retrouver l'essentiel, de s'affranchir de la diversité des pratiques et de dégager le champ pour la simplification du système.

La constitution du Référentiel sémantique (le Référentiel métier ou Référentiel Assurance) permet, également, de capitaliser la connaissance sur le domaine, les modalités d'action et la solution même. C'est un outil de gestion des connaissances.

---

#### L'attitude

Le modélisateur qui se donne pour tâche d'exprimer la sémantique, aborde le réel, sans *a priori*. Contrairement aux apparences, cette attitude n'est pas spontanée. Elle exige, du modélisateur, un effort particulier et sans cesse renouvelé pour faire abstraction des contingences organisationnelles et techniques. La qualité du modèle sémantique demande qu'il s'affranchisse des pratiques actuelles et de la solution existante.

De plus, loin de s'embarrasser de l'apparente complexité du domaine, il doit capter l'essentiel et dégager le noyau fondamental.

La modélisation sémantique peut se révéler inventive.

Il lui faudra défendre la simplicité de son modèle face à la tendance générale à la complication. Une de ses réponses consistera à montrer comment ce modèle essentiel restitue la réalité et comment il peut se « déplier », s'instancier pour prendre en compte la diversité des situations concrètes.

---

## Les procédés de modélisation (suite)

---

### La modélisation sémantique : quelques préceptes

---

#### Identifier les classes principales

Les classes principales jouent un rôle essentiel dans la chaîne de production du Système d'information. En effet, elles fournissent le point de départ pour construire les noyaux durs autour duquel le système va cristalliser.

*Critères* La première justification du caractère central d'une classe est celle de son importance dans le discours. C'est donc un critère purement sémantique, difficile à formaliser.

On peut, toutefois, s'aider de critères formels pour désigner une classe principale :

- Cardinalités des associations à partir de la classe (elle est liée à plusieurs classes satellites avec une cardinalité multiple ; cardinalité maximale = 1, de son côté).
- Agrégation (elle assemble plusieurs autres classes).
- Présence d'un automate à états (avec des états significatifs du point de vue utilisateur).

Le critère le plus fort est celui-ci : il existe au moins un cas d'utilisation qui l'utilise comme objet central. Ce critère rejoint le procédé de vérification d'exhaustivité pour le recensement des cas d'utilisation.

Autre critère (proche du critère sémantique) : la classe principale est une classe sur laquelle se localisent les opérations (les services) les plus significatives pour le métier. Ces derniers sont forcément publics, visibles de l'extérieur. *A contrario*, une classe satellite ou secondaire peut rester confinée à la partie privée d'un atelier logique et n'offrir aucune interface à l'extérieur.

---

#### Encapsuler les règles

Le modèle sémantique, quoique positionné au plus haut niveau d'abstraction, obéit à toutes les exigences d'un vrai modèle : il est complet, détaillé et valide. Il décrit complètement les notions du champ d'étude. Tout particulièrement, le modèle doit contenir la description des informations et des opérations attachées à la sémantique des notions et objets. Les règles d'organisation sont exclues de ce modèle (elles appartiennent à l'aspect pragmatique) ; mais les règles de gestion doivent impérativement être encapsulées. Le modélisateur conçoit les classes sémantiques comme des micro-machines, responsables de leur état interne et capables de fournir tous les services sous-tendus par leur sémantique.

---

#### Respecter le devenir des objets

Les objets de gestion évoluent, se transforment et se comportent différemment en fonction de leur état, du moins pour les principaux d'entre eux. Le cycle de vie des objets constitue une part importante du modèle. Le système, conçu à base d'automates, se révélera à la fois plus simple et beaucoup plus robuste.

---

#### Documenter les opérations

L'encapsulation des règles de gestion et la conception des cycles de vie font émerger des opérations à valeur sémantique. Le modèle décrit complètement ces opérations : signature (nom, paramètres d'entrée/sortie avec leur nature), contrat (pré et post-conditions), description interne (algorithme quand c'est nécessaire).

---

#### Décomposer le système

On est amené à décomposer le système, sur le plan sémantique. Cette décomposition applique le critère de l'objet (par opposition à la fonction) ; il s'agit d'une décomposition en domaines d'objets.

## Les procédés de modélisation (suite)

### L'expression des besoins par les cas d'utilisation

#### Qu'est-ce qu'un cas d'utilisation ?

Rappel de la définition dans Praxeme :

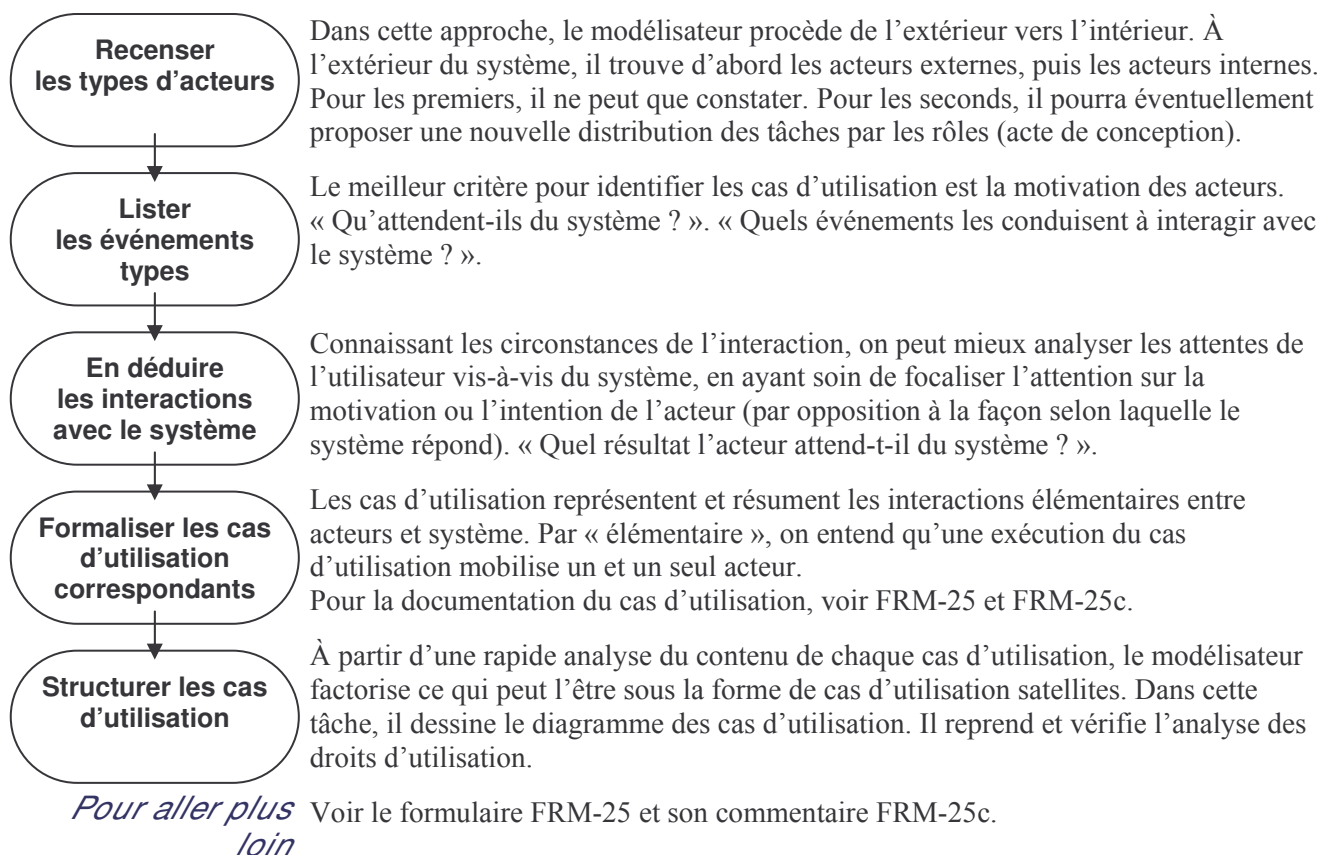
**Un cas d'utilisation est une interaction élémentaire entre un acteur et le système.**

L'insuffisance de la définition donnée dans le standard UML entraîne un problème d'interprétation. Sur le terrain, les modélisateurs sont confrontés à la difficulté de s'accorder sur le niveau de granularité de cette notion. La définition que nous retenons prétend aider à maîtriser cette difficulté. Elle porte, en filigrane, un critère d'identification dont les conséquences pratiques sont les suivantes :

- Un seul utilisateur participe à l'exécution d'un cas d'utilisation (une occurrence)<sup>8</sup>. Cette exécution est in-interruptible (en dehors des causes d'échec).
- Le cas d'utilisation répond à une intention précise de l'utilisateur, vis-à-vis du système.

#### La procédure

NB : cette procédure se veut prospective et va donc au-delà de la simple expression de besoin. Elle se caractérise par l'attention portée sur les motivations des acteurs.



<sup>8</sup> Si l'activité doit impliquer plusieurs acteurs ou se dérouler en plusieurs étapes avec interruption, on est en présence d'un processus.



---

## Les procédés de modélisation (suite)

---

### L'architecture logique

---

#### Résumé de la méthode

La terminologie a été exposée p. 21. Elle exprime et conditionne l'approche de l'architecture logique.

#### *Machines logiques*

Les MLM (machines logiques Métier) proviennent des classes du modèle sémantique. L'effort de modélisation porté sur la sémantique (le cœur de métier, abstraction faite des choix organisationnels et techniques) présente plusieurs avantages :

- Formaliser la connaissance du métier.
- Simplifier le modèle, le rendre à la fois plus compact et plus « naturel ».
- Découvrir des **services à valeur ajoutée**.

Les MLO (machines logiques Organisation) sont déduites du modèle pragmatique (organisationnel). Elles servent de relais sur le noyau du système, c'est-à-dire sur les Machines Métier.

#### *Stratification*

- **Noyau** : cœur de métier. Stable, protégé. Les MLM. Sur les MLM, les « services métiers » restituant toute la sémantique du Référentiel Assurance.
- **Strate Organisation** : MLO isolant les règles d'organisation dont les habilitations et conventions de visibilité avec les partenaires. Sur les MLO, les « services activité ». On peut avoir plusieurs ces services comme relais sur un même « service métier » : par exemple, un pour utilisation interne par le personnel de l'entreprise, un autre pour contrôler l'accès extérieur.
- **Périphérie, strate Présentation** : les postes de travail (selon choix techniques pour l'IHM) et les autres interfaces pouvant donner accès au système.

#### *Services logiques*

Les machines fournissent des services. Les services peuvent s'appeler les uns les autres, conformément au principe de coopération de l'approche objet.

Ce positionnement du terme **service** est conforme à la tradition du Génie logiciel (métaphore du client-serveur, méthode TACT, architecture de services de La Poste et ailleurs<sup>9</sup>). Il n'est pas incompatible avec les catégories techniques (notamment des solutions *Web Services*).

#### *Les règles de dérivation*

Des règles de dérivation quasi-mécaniques permettent de passer des modèles amont (sémantique et pragmatique) à l'architecture de services. Elles sont édictées dans le guide de l'aspect logique (référence : « PxM-40 »). L'architecture logique accueille le cœur de métier en préservant le plus possible sa structure. En ce qui concerne le modèle de l'activité (cas d'utilisation, processus, Structure...), la transformation dépend de plusieurs décisions dont certaines sont conditionnées par la négociation logique / technique (cf. ci-après).

---

<sup>9</sup> Dans le secteur des assurances, on peut citer : la CNP (machines logiques), le groupe Azur-GMF (Plan de Convergence de l'Informatique ; machines logiques ; services logiques et, au niveau logiciel : Modules de services = programmes Cobol).



## Les procédés de modélisation (suite)

### L'identification des services logiques

#### Introduction

Le grain le plus fin du modèle logique est le « service logique ». L'architecture logique fournit le cadre qui permettra d'ordonner les milliers de services. Reste la question : « Comment identifier un service ? ». Le tableau ci-dessous propose plusieurs moyens pour déduire les services.

On peut recourir à deux approches complémentaires :

1. **Par le haut** : les composants logiques se déduisent des modèles sémantique et pragmatique.
2. **Par le bas** : en relisant le détail des fonctions attendues (cas d'utilisation ou activités dans les processus), on complète le modèle logique.

#### Règle d'identification

**Tout service possède un identifiant logique, unique dans le système** (on s'efforce de le rendre le plus expressif possible).

#### Recommandations

Figure PxM-02 21. Déduction des services logiques à partir des modèles externes

Origine	Terme de départ	Dérivation	Remarque
Sémantique	Opération d'une classe sémantique	Service de base	1 pour 1, en général
	Attribut d'une classe sémantique	Service de base (de type accesseur), service « lire » sur machine ensembliste	À décider en relation à l'architecture technique : soit sous la forme d'attribut (si technologie objet), soit sous la forme de service programmé. À considérer : protection en écriture et principe du référent uniforme.
	Classe sémantique	Service de contrôle de l'invariant de classe Services ensemblistes	La plupart des classes sémantiques entraînent une classe ensembliste. Les classes ensemblistes proposent les services de création, de conservation, d'administration (comptage, statistique...). Exception : le singleton.
	Association	Services de navigation	À placer en fonction des cardinalités et de l'orientation de l'association ou des relations d'utilisation entre les machines.
	Domaine d'objets	Fabrique logique	Important outil de réduction du couplage et d'organisation de l'urbanisation.
Pragmatique	Classe pragmatique (acteurs, objets organisationnels)	Machines logiques de type « Métier » Machine « Organisation »	Deux options possibles : <ul style="list-style-type: none"> <li>• Même traitement que ci-dessus.</li> <li>• Ou sous forme de paramètres dans une « Machine Organisation » (tables des pouvoirs, par exemple)..</li> </ul>
	Cas d'utilisation	Peut être un point de départ pour la conception des MLO	Relire les scénarios pour détecter les besoins d'interrogation ou d'action sur le système. Ces besoins doivent être couverts par les services.

---

## Les procédés de modélisation (suite)

---

### La documentation des services logiques

---

#### Les informations à rassembler

Les services logiques font l'objet :

- d'un formulaire de demande de service (FRM-44) qui structure la communication entre les domaines de la DSI ;
- d'une fiche descriptive, plus complète (FRM-45).

Ces formulaires peuvent être intégrés dans un outil de modélisation, sous la forme d'un profil UML.

*Définition du service* La définition d'un service logique doit permettre, « de l'extérieur », de comprendre à quoi il sert et comment il se comporte. Elle se compose des rubriques suivantes :

- Le nom du service (voir p. précédente).
- La signature du service avec les paramètres, leur nature et leur fonction « entrée / sortie ».
- L'objectif du service (une phrase).
- Les conditions d'exécution du service.
- Les signaux qu'il peut émettre.

Les conditions d'exécution couvrent les pré-requis et garanties de services (pré- et post-conditions), et aussi la modalité temporelle : synchrone ou asynchrone.

*Description du service* Il peut être nécessaire, pour des services complexes, de fournir un algorithme. Dans tous les cas, la spécification du service est sans ambiguïté et on doit pouvoir lister les services qui seront appelés par le service documenté ainsi que les objets manipulés. En effet, ces informations permettent de vérifier la pertinence de l'architecture et la valeur du couplage.

D'autres informations sont nécessaires pour consolider l'architecture : la fréquence d'utilisation du service ; le volume des échanges (flots logiques) qu'il génère.

*Représentation du service* Le service logique est représenté, dans Praxeme, en tant qu'opération sur une classe stéréotypée « Machine ». Ces opérations pourraient être, elles-mêmes, stéréotypées « Service ».

---

#### À propos de la modalité temporelle

Le comportement du service – instantané ou différé – est une donnée essentielle de l'architecture. On conseille parfois de regrouper, dans les ateliers, des services de même nature (règle d'homogénéité).

Dans le souci de quantifier l'architecture logique, on pourrait chercher à évaluer la durée d'exécution de chaque service. À ce stade, ceci ne paraît pas réaliste. En revanche, le concepteur logique peut s'astreindre à évaluer les volumes (nombre d'instructions au niveau logique, nombre d'appels à d'autres services et pondération par la distance...). Ces informations pourront s'agréger et fournir une base de réflexion pour les choix d'architecture physique.

## Les procédés de modélisation (suite)

### L'architecture technique

#### Négociation logique / technique

La méthodologie prévoit une action préalable à l'élaboration en parallèle des architectures logique et technique. Cette action, la négociation logique / technique, est une précaution qui ne remet pas en cause le principe d'indépendance logique selon lequel il est utile de concevoir le système sur un plan logique, indépendant des choix techniques.

La négociation permet :

1. de fixer les termes dans lesquels pourra s'exprimer l'architecture logique (pour nous les services) et de s'assurer que l'on pourra aisément traduire ces termes en composants logiciels ;
2. de répartir les questions transverses entre la conception logique et la conception technique ;
3. de déterminer les unités qui seront manipulées par les activités ultérieures (livraison, déploiement...).

*Questions transverses* Exemples : communication dans le système ; gestion des incidents ; gestion des transactions ; moteur de règles ; traduction des automates à états ; gestion des événements ; etc.

Le principe pour le partage des questions transverses est le suivant : quand l'architecte technique peut proposer une solution du marché et que cette solution répond aux exigences, on la retient (c'est toujours plus économique) ; dans le cas contraire, la conception revient à l'architecte logique.

*Unités* L'architecture logique range les services selon plusieurs niveaux d'agrégation : machines logique, ateliers logiques, fabriques logiques.

Entre ces niveaux d'agrégats, quel est la bonne unité pour le déploiement et pour la maintenance ?

- **L'unité de déploiement** : c'est celle que l'on ne peut pas scinder pour localiser les morceaux sur des machines différentes. Le candidat actuel est l'atelier logique (associé à une structure de données).
- **L'unité de livraison** : unité d'œuvre pour les interventions de maintenance. On peut penser au service. En revanche, on a intérêt à s'imposer les tests de non régression au moins au niveau de la machine.

## Les procédés de modélisation (suite)

### L'architecture technique (suite)

#### Les aspects couverts

L'architecture technique, en tant qu'art, est une discipline de conception qui porte sur le système, dans son entier, et qui couvre les aspects suivants :

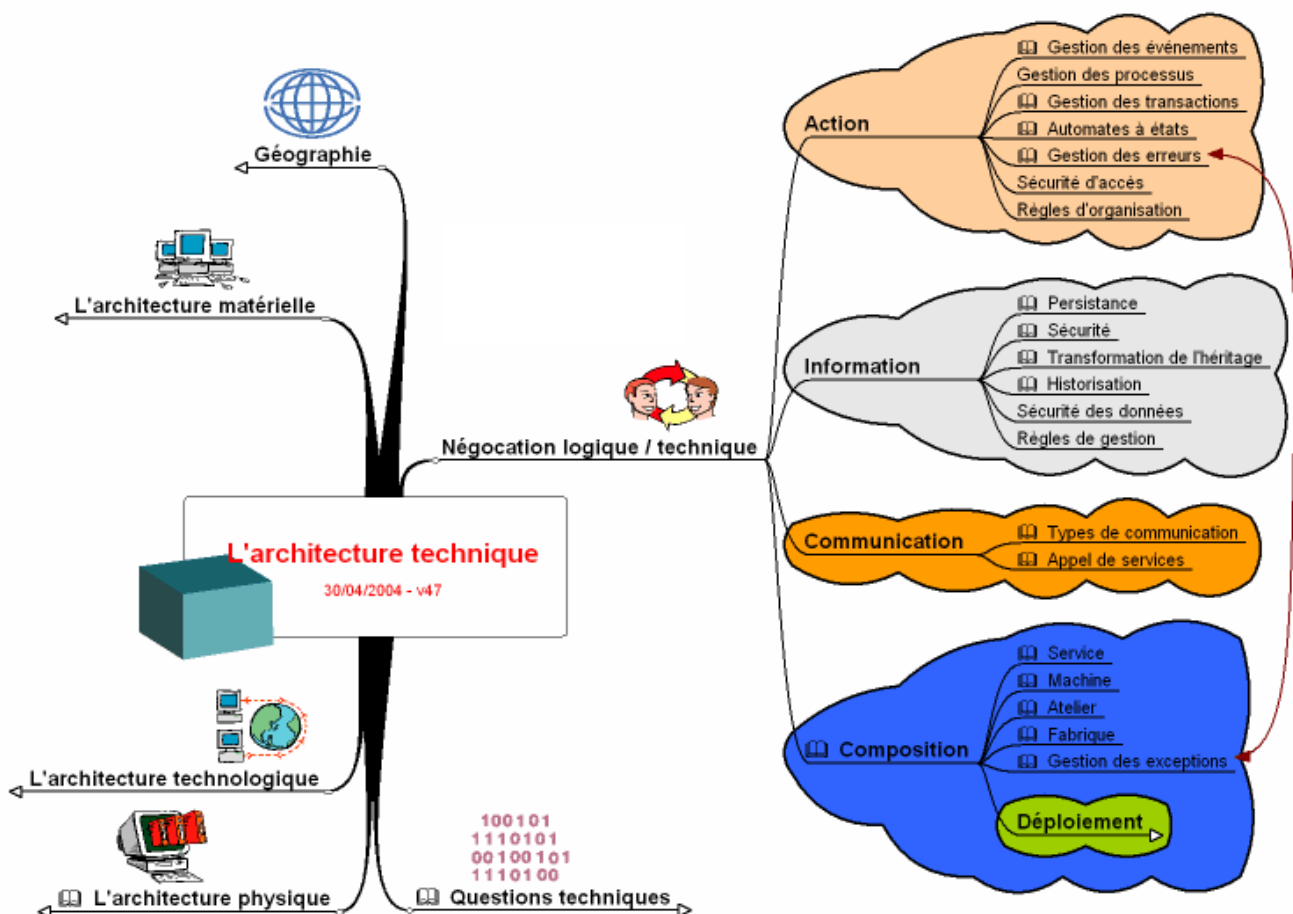
1. **L'aspect matériel** : ensemble des machines et autres moyens matériels, composants l'infrastructure (capacités, coûts...).
2. **L'aspect proprement technique ou technologique** : logiciels de base, composants techniques et règles de développement qui en découlent.
3. **L'aspect physique** : l'architecte technique se borne à fixer les règles de localisation des composants logiciels sur l'architecture matérielle (il ne saurait décrire complètement l'architecture physique, car, pour cela, il faudrait connaître la totalité des composants logiciels).

#### L'attitude

L'architecture technique se réalise en deux temps :

1. l'analyse, qui fait le bilan de l'existant, rassemble les contraintes et objectifs (analyse architecturale).
2. la conception, qui explore les possibilités technologiques, examine les options et les combine en scénarios d'architecture.

Figure PxM-02\_22. Aperçu des thèmes de l'architecture technique





## Composant

**PxM-02 « Modus : La méthodologie Praxeme »**

---

## Annexes

*Objectif* Les annexes rassemblées ici fournissent des détails pour :

- la justification du socle théorique de Praxeme ;
- l'application de Praxeme.

- Contenu*
- La « topologie du Système Entreprise », p. 46
  - Sur la modélisation, p. 64
  - Complément sur le processus, p. 66

## La « topologie du Système Entreprise »

---

### Que faut-il représenter pour maîtriser le Système ?

---

#### Introduction

L'utilisation du langage UML bute sur une difficulté : UML propose des outils de représentation – les diagrammes – mais n'en définit pas rigoureusement le contenu, ni les conditions d'utilisation. De plus, cette notation ne respecte pas le principe d'exclusivité, applicable aux langages formels, qui prescrit que pour un type d'élément à représenter, la syntaxe doit proposer une et une seule catégorie de représentation<sup>10</sup>.

Par ailleurs, l'approche orientée objet, pour puissante qu'elle soit, soulève aussi de graves questions :

- À partir de quel moment le modèle peut-il être considéré comme complet ?
- Que faut-il représenter à tel ou tel moment d'un projet ?
- Comment garantir la qualité d'un modèle (y a-t-il des « formes normales »<sup>11</sup>) ? Comment être certain de retirer tous les fruits espérés de l'approche objet ?

---

#### Des univers mentaux qui cohabitent

Dans la chaîne qui va de la perception d'un besoin à la fourniture d'une solution satisfaisante, les parties prenantes sont confrontées à un large éventail de questions et de décisions. Tout projet convoque des expertises fort différentes : entre l'expertise du domaine métier, la connaissance des technologies en perpétuel renouvellement et les objectifs des architectes... les compétences s'enchevêtrent. Cette situation entraîne les risques suivants :

- oubli des objectifs et contraintes initiaux ; confusion dans les priorités ;
- insuffisante expression de tel ou tel point de vue ;
- incompréhension entre les parties prenantes ;
- négligence de certaines considérations ;
- etc.

---

#### Le principe de la solution

La solution retenue pour clarifier les compétences et définir les modèles est, tout simplement, le questionnaire du Quintilien – le QQQQC (quoi, qui, où, quand, comment). On l'applique, dans une forme étendue, sur la réalité que l'on cherche à analyser. Ceci permet de dégager et d'articuler les aspects ou facettes de cette réalité, chaque aspect donnant lieu à un modèle.

---

<sup>10</sup> Pour la catégorie grammaticale courante qu'est le verbe, UML propose plusieurs types d'éléments de modélisation : le cas d'utilisation, l'opération, l'activité, voire l'événement. À l'inverse, différentes interprétations actuelles font correspondre des choses fort différentes à la notion de « cas d'utilisation ». Ce flou laisse la part belle aux interprétations personnelles et aux errements. Les conséquences pratiques sont un surcoût des activités de modélisation et, plus grave, le caractère inexploitable de nombreux modèles.

<sup>11</sup> À l'instar de la modélisation des données (dans Merise, par exemple).

## La « topologie du Système Entreprise » (suite)

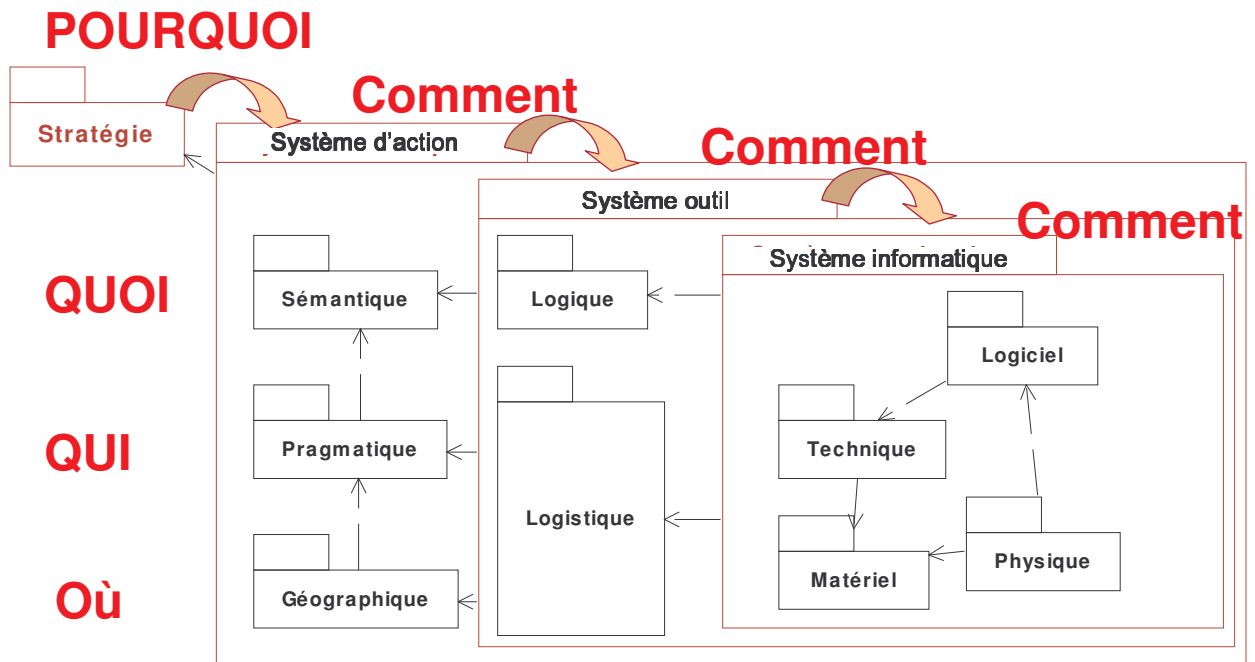
### Le schéma général de la topologie

#### Introduction

Le schéma ci-dessous donne le principe utilisé pour identifier les aspects (ou facettes) de la topologie. On pose les questions et on examine la réponse au Comment, sur plusieurs niveaux.

Cette logique d'exposition sera simplifiée ensuite : on ne retiendra que huit aspects, qui suffisent à répartir proprement l'ensemble des informations nécessaires pour décrire le système.

Figure PxM-02\_23. Présentation synoptique de la Topologie



## La « topologie du Système Entreprise » (suite)

---

### Que faut-il représenter ?

---

**Le point de départ** La transition méthodologique vers UML, les projets d'urbanisation des systèmes d'information et les programmes de progrès au niveau des entreprises butent sur de grandes difficultés. Elles sont, en grande partie, liées à la communication entre des points de vue différents.

Il est devenu urgent d'établir des règles de coexistence entre ces points de vue dans l'entreprise. L'affaire est de nature psychologique et sociologique, au départ, mais elle débouche sur des dispositions techniques précises : représentations à échanger, techniques de modélisation, outillage, démarche de projet, etc.

Le développement qui suit, propose un fil conducteur pour traiter ces problèmes et harmoniser les points de vue, **de la stratégie de l'entreprise à la réalisation informatique**, en passant par la conception des processus et l'architecture des SI. Cette approche de la complexité des entreprises se prolonge par l'utilisation régulée du standard UML.

---

### L'analyse du problème

Comment maîtriser le changement ? Comment garantir que l'on n'oublie rien dans la conception d'une nouvelle solution ? Ces questions – et bien d'autres encore – font l'objet d'une discipline : la méthodologie. À travers l'histoire du génie logiciel, les réponses apportées ont été surtout techniques. Le fond du problème, toutefois, est profondément enraciné dans l'humain. En effet, il s'agit de faire cohabiter des populations différentes (des utilisateurs aux informaticiens en passant par les sponsors, architectes, etc.) ; il s'agit, malgré cette diversité nécessaire, de mobiliser les énergies et de les faire concourir vers un but commun. On se heurte, aussitôt, au problème des différences de représentation et de culture : les divers métiers voient le monde de l'entreprise de façon fort différente, à travers le prisme de leur spécialité.

Nous laissons de côté les implications psychologiques de cette situation et s'intéresse au problème cognitif qui est posé. Il jette les bases de la topologie du Système Entreprise.



---

## La « topologie du Système Entreprise » (suite)

---

### La solution passe par une vision claire de l'entreprise

---

**Une vision unifiée** Afin de conduire toutes les ressources de l'entreprise vers un but commun, il est indispensable de leur présenter une vision unique. Cette vision doit être assez simple pour être communicable, mais assez complexe pour prendre en compte la diversité des compétences. L'utilisateur, selon sa spécialité ou son rôle dans l'entreprise, a quelque chose à dire ; de même l'organisateur, le manager, le directeur des ressources humaines, etc. L'informaticien a une contribution à apporter ; lui réserver sa juste place permet à l'entreprise d'accéder aux possibilités de la technologie. Mais, quelle est cette juste place ? Elle ne doit être ni trop confinée (sinon on sous-exploite les technologies de l'information), ni trop proéminente (car alors on fabrique des systèmes inadaptés). Par ailleurs, le terme « informaticien » est confus ; il recouvre de nombreux métiers qui ont du mal à se reconnaître et à se comprendre.

---

### Le respect des compétences

En l'absence d'une représentation qui reconnaît la contribution de chaque spécialité, surgissent la frustration, les conflits, les dysfonctionnements. C'est ainsi que l'on voit circuler des dossiers qui ne peuvent être validés, des modèles qui ne sont pas exploités, des décisions dont on a perdu les motivations... Cette représentation doit rendre lisible les expertises convoquées dans le fonctionnement ordinaire de l'entreprise et dans ses évolutions extraordinaires. Elle doit combiner les points de vue propres à toutes les populations impliquées.

---

### De quoi parle-t-on ?

Pour établir cette représentation commune (on pourrait dire « communautaire »), une approche consisterait à partir des *acteurs* et à étudier leur univers. Cette approche, cognitive, est extrêmement lourde et compliquée.

Nous choisissons une approche plus simple, qui consiste à observer *l'objet* commun, sur lequel on agit. De quel objet s'agit-il ? Rien de moins que le réel ! Du moins la portion de réel circonscrite par l'activité sur laquelle on veut agir. Compte tenu du mode de déclenchement et de financement de la plupart des opérations, cette réalité, dans sa plus grande extension, est celle d'un groupe humain organisé et tendu vers un but professionnel, de profit ou de service. Bref : **l'entreprise**.

Cet objet est complexe. En fait, il présente **plusieurs facettes** qui, toutes, ont leur cohérence et leurs règles. Elles s'expriment, chacune, à travers un vocabulaire particulier. Il faut donc, en premier lieu, isoler ces aspects. Ensuite, nous pourrions les articuler ; enfin, il sera nécessaire de dire comment les représenter le plus efficacement possible pour bâtir la représentation complète de l'organisme.

## La déduction des aspects

---

### Le fil conducteur

---

#### Les tentatives

Comment isoler les aspects de notre objet, l'entreprise ? Nous pourrions, là encore, adopter une approche par acteurs : l'aspect vu par le management, l'aspect qualité, l'aspect commerce, l'entreprise vue par ses comptables, etc. Cette approche souffre de deux défauts majeurs :

- d'une part, elle encourt le risque d'oublier quelqu'un, donc de manquer une des facettes ;
- d'autre part, elle ne permet pas de dégager un ordre facile à communiquer.

Le génie logiciel a proposé, au cours de son histoire, plusieurs solutions : les niveaux d'abstraction de Merise, le *framework* de Zachman, repris par Melissa Cook notamment. Ces solutions restent marquées par l'ancien paradigme où domine le postulat de la séparation données-traitements.

---

#### Le QQQQC

Nous optons pour une structure vieille comme la philosophie et largement partagée : la liste des questions, le bon vieux QQQQC, plus exactement une forme étendue du questionnaire de Quintilien.

- **Pourquoi** : vers quoi se tendent les efforts de l'entreprise ? qu'est-ce qui justifie les décisions ?
- **Quoi** : quels objets au cœur de l'activité ?
- **Qui** : qui agit ? qui fait quoi ?
- **Où** : où sont les ressources ? se déplacent-elles ?
- **Quand** : le temps imprègne tous les aspects, que ce soit l'organisation, l'informatique (les temps de réponse) ou le rythme de transformation des objets.
- **Comment** : comment aider l'activité ?
- **Combien** : quantité, fréquence, coûts.

## La stratégie de l'entreprise

### « Pourquoi ? Pourquoi ? »

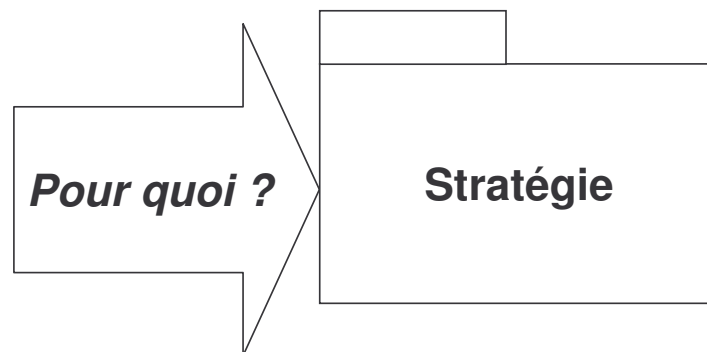
#### Définition de cet aspect

La première grande question à poser pour étudier l'organisme est celle de sa finalité et de ses buts : « Pourquoi ? ». Ce but vers lequel l'organisme tend ou doit tendre est l'objet privilégié de la stratégie d'entreprise. Une fois établi, il se retourne, en quelque sorte, et est vu par tous les autres aspects de l'organisme comme « cause première », réponse à la question lancinante « Pourquoi ? ». Toute décision de plus bas niveau (choix d'organisation, choix techniques, etc.) doit se justifier par rapport à un objectif de la stratégie, directement ou indirectement.

La stratégie elle-même comporte deux dimensions :

- l'une, statique, désigne la cible (modèle économique et orientations qui peuvent concerner les autres facettes de l'entreprise) ;
- l'autre, dynamique, où s'inscrit la trajectoire qui doit mener l'organisme vers l'état visé.

#### Positionnement



#### Représentation

En fait, la stratégie ne dispose pas, pour s'exprimer, d'un langage particulier. La cible s'exprime dans les termes des autres aspects. Une stratégie peut comporter des décisions sur n'importe lequel des autres aspects :

- Elle peut désigner un nouveau marché à conquérir (Quoi ?).
- Elle peut changer l'organisation, opter pour un style différent, réformer les processus... (Qui ? Qui fait quoi ?).
- Dans un mouvement de fusion-acquisition ou d'internationalisation, elle répond à la question « Où ? ».
- Fondée sur une exploitation approfondie des possibilités technologiques, elle peut se définir autour des choix informatiques...

En conséquence, l'expression de la cible stratégique se fait à travers une somme de décisions réparties sur les différents aspects du Système Entreprise.

---

## Le Système Entreprise

---

### « Comment mettre en œuvre la stratégie ? »

---

**Après le pourquoi** Cette finalité dégagée et, parfois, ajustée par la stratégie s'applique à un objet global, complexe, que nous nommerons le Système Entreprise. Celui-ci est une réponse volontariste à la question « Comment mettre en œuvre la stratégie ? ».

Il y a, donc, renversement de l'approche : ce n'est pas la stratégie qui vient après l'entreprise ; c'est l'entreprise qui est conçue comme un instrument adaptable pour réaliser la stratégie.

---

### Définition

Il n'est pas besoin de définir l'entreprise : qu'il s'agisse d'une entreprise privée ou publique, d'une administration, d'une ONG... bref de tout organisme, chacun en a une compréhension directe. Pourtant, cette réalité est extrêmement complexe. Si nous voulons la maîtriser et agir sur elle, il nous faut l'examiner selon différents angles : ressources humaines, organisation, processus, système informatique, connaissances, infrastructure, etc.

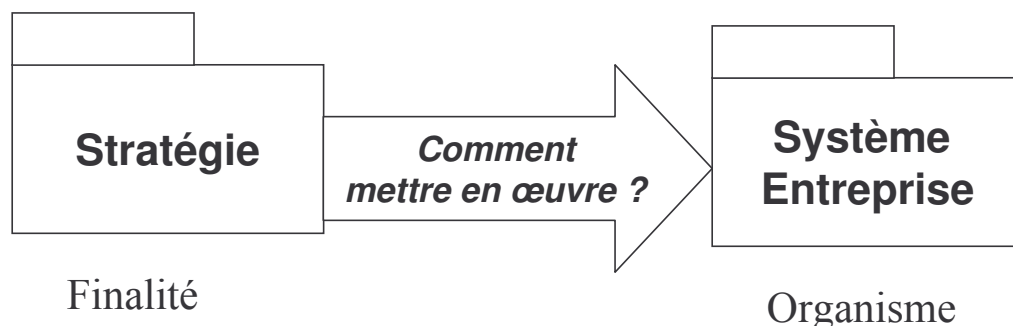
La réussite d'un Programme de Progrès passe, bien sûr, par la vision stratégique ; mais elle exige, également, de maîtriser les détails. Ceci nous amène à considérer la complexité : pour nous aider, nous avons besoin d'une structure qui identifie et articule tous les aspects pertinents de l'entreprise. Une telle représentation nous garantit de ne rien négliger, d'une part. Elle constitue une grille qui facilite la communication entre les parties prenantes, d'autre part.

L'expression « Système Entreprise » caractérise cette notion de **l'entreprise perçue comme un tout et examinée dans tous ses aspects**. C'est l'organisme, conscient de lui-même, maîtrisant ses finalités et qui se représente pour mieux évoluer et contrôler son destin.

Nous appellerons « Système Entreprise » l'objet complexe qui présente ces caractéristiques. Il ne recouvre pas forcément la totalité de l'organisme mais doit en constituer une partie significative, douée d'autonomie de décision et d'auto-organisation.

---

### Positionnement



---

### Représentation

Le contenu concret de cette représentation se manifeste à travers des modèles. Il s'ordonne en posant les questions suivantes : quoi ? qui ? où ? et, de nouveau, comment ?

---

## Le Système Entreprise (suite)

---

### « Quoi ? » : la sémantique du Système Entreprise

---

#### Définition

La question « Quoi ? » pointe sur les objets fondamentaux de l'activité « naturelle » de l'organisme et sur les concepts associés. Ces objets et concepts expriment le noyau de l'activité, l'essentiel. Un changement à ce niveau, même mineur, équivaut à un changement de nature de l'organisme. À titre d'exemple, on peut évoquer l'évolution des banques vers l'assurance. Cet ensemble d'objets et de concepts qui constituent la substance dont se nourrit l'organisme (le cœur de métier), nous le nommerons : aspect sémantique. La qualité des objets sémantiques est leur indépendance par rapport aux choix qui sortiront des questions suivantes.

#### Représentation

Au niveau des projets, on parlera de modèles sémantiques. Ils valent pour l'effort d'abstraction qui conduit à s'affranchir des contingences organisationnelles et techniques et à dégager les notions essentielles. Nous sommes sur le plan conceptuel. Les techniques de la modélisation objet permettent une expression plus naturelle que ce qui se faisait avec Merise.

Au niveau global de l'entreprise, il s'agit du **Référentiel Métier** (par exemple : le Référentiel Assurance). C'est un puissant outil de gestion des connaissances.

---

### « Qui » : la pragmatique du Système Entreprise

---

#### Définition

Le modèle sémantique présente les objets qui sont au cœur de l'activité. L'étape suivante est l'examen des actions sur ces objets. L'étude introduit alors les acteurs. C'est le rôle du modèle pragmatique qui montre comment l'entreprise s'organise, comment les responsabilités se distribuent, bref : qui fait quoi.

#### Représentation

Sur ce plan, on peut distinguer deux catégories de modèles, selon que la portée est globale (l'entreprise dans son ensemble) ou locale (une fonction, une application) :

- la vue de l'organisation, où apparaissent les processus (modélisés, par exemple, par les diagrammes d'activité d'UML<sup>12</sup>) ;
- la vue de l'utilisation, qui exploitera, en autres, le diagramme des cas d'utilisation.

---

<sup>12</sup> Notons que le diagramme d'activité d'UML offre une syntaxe plus riche que les techniques traditionnelles (modèle organisationnel des traitements de Merise). Il permet d'introduire les objets et, qui plus est, leurs états. Cette simple remarque de détail suffit à fonder une nouvelle approche des processus.

## Le Système Entreprise (suite)

### « Où » : l'aspect géographique

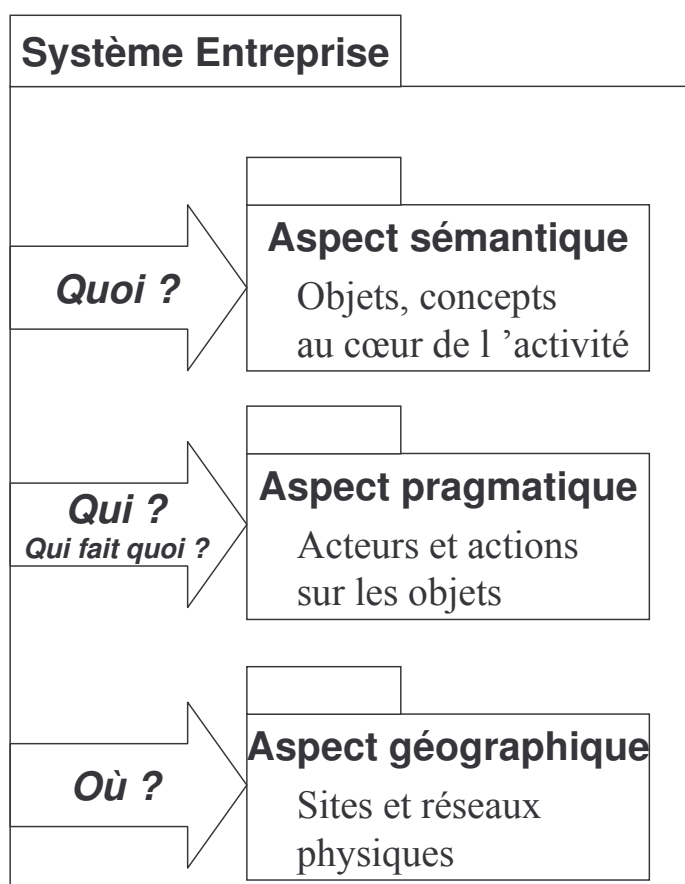
#### Définition

Même si, la plupart du temps, la localisation des activités est implicite ou évidente, elle peut introduire des contraintes fortes ou, au contraire, faire l'objet d'aménagements. La représentation de l'organisme ne saurait être complète sans cet aspect géographique. Il conditionne des choix d'infrastructure. Modéliser cette facette ne demande pas un effort considérable.

Les scénarios envisagés sous cet angle sont, par exemple :

- les activités multi-sites ;
- la redondance voulue ;
- l'absorption de systèmes externes ;
- la délocalisation ;
- le support du travail nomade...

### Récapitulatif de la vue externe



## Le Système de production

### « Comment mettre en pratique les orientations ? »

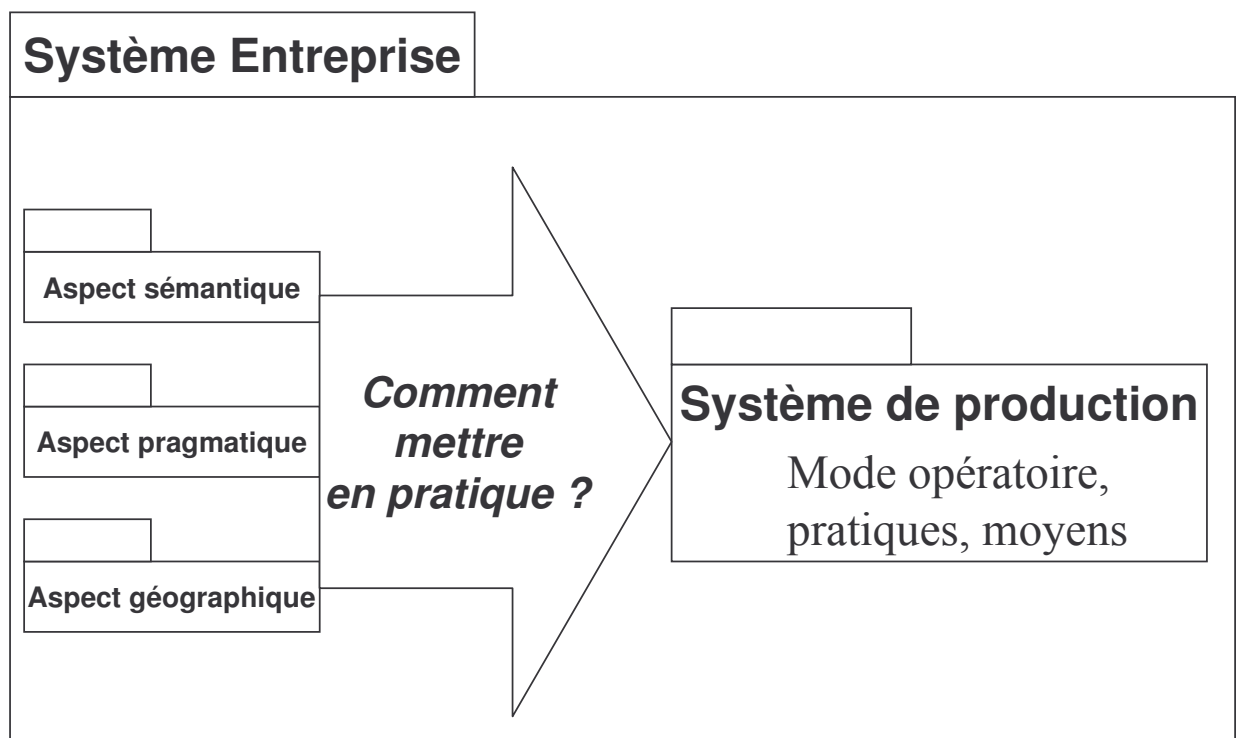
#### Définition

Dans une démarche de progrès, on commence par traiter les questions relatives aux aspects sémantique, pragmatique et géographique. On prend les décisions sur chacun de ces aspects et on les met en cohérence avec les orientations stratégiques. Par exemple :

- facette sémantique : un changement de marché, une évolution du catalogue...
- facette pragmatique : une réforme des processus, un changement de style de management...
- facette géographique : équipement nomade, fusion...

De nouveau, se pose la question du mode opératoire : « comment réaliser ces changements ? ». La réponse massive à cette question est le système de production.

#### Positionnement



## Le Système de production (suite)

### Le contenu

#### QQOQC

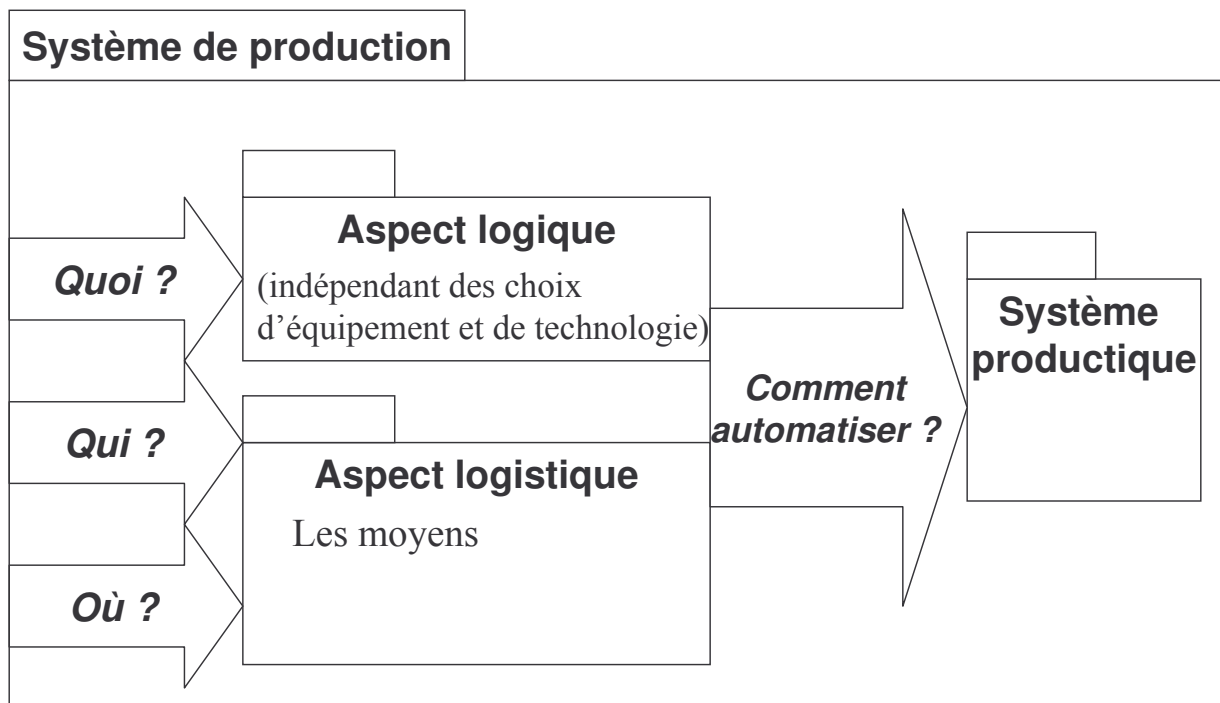
Le Système de production s'analyse à travers trois facettes :

Le **Quoi** : c'est l'aspect logique qui fait la synthèse de la sémantique et de la pragmatique. Le modèle logique reste indépendant du détail des choix techniques, mais s'impose une syntaxe qui le rend moins « naturel », moins communicable que les aspects précédents. C'est déjà une affaire de spécialistes. Il joue un rôle intermédiaire dans la chaîne de production qui permet de construire le nouveau système.

Le **Qui** et le **Où** se combinent dans l'aspect logistique, en même temps que le **Avec Quoi** (locaux et équipements). Cet aspect concentre l'information nécessaire pour déployer et administrer les ressources matérielles nécessaires au fonctionnement de l'entreprise.

Le **Comment**, troisième occurrence de la question du mode opératoire, introduit maintenant l'ultime détail de l'outillage : il mobilise les technologies de l'information et de la communication et débouche sur le Système productique, infrastructure intelligente qui décuple les capacités de l'organisme.

### Les aspects résultants





## Le Système productique

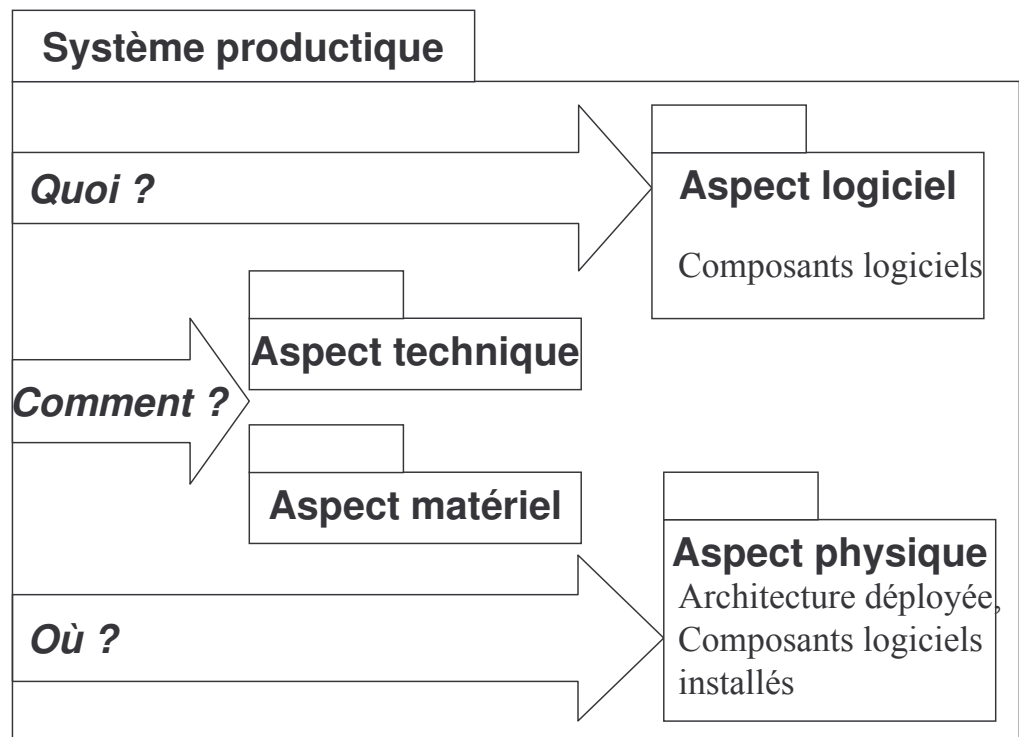
### Les derniers aspects dans la construction du système

#### Contenu

Le Système productique doit se conformer aux déterminations établies à travers les facettes précédentes.

Sa substance (son Quoi) est faite de composants logiciels. Un composant logiciel réalise un objet logique, en appliquant les choix techniques. Reste ensuite à le localiser, c'est-à-dire à le placer sur une des machines de l'architecture matérielle. En répondant à cette question, on aboutit à l'architecture physique.

#### Les aspects informatiques



## Récapitulatif

### Une ligne explicative pour définir les modèles à réaliser

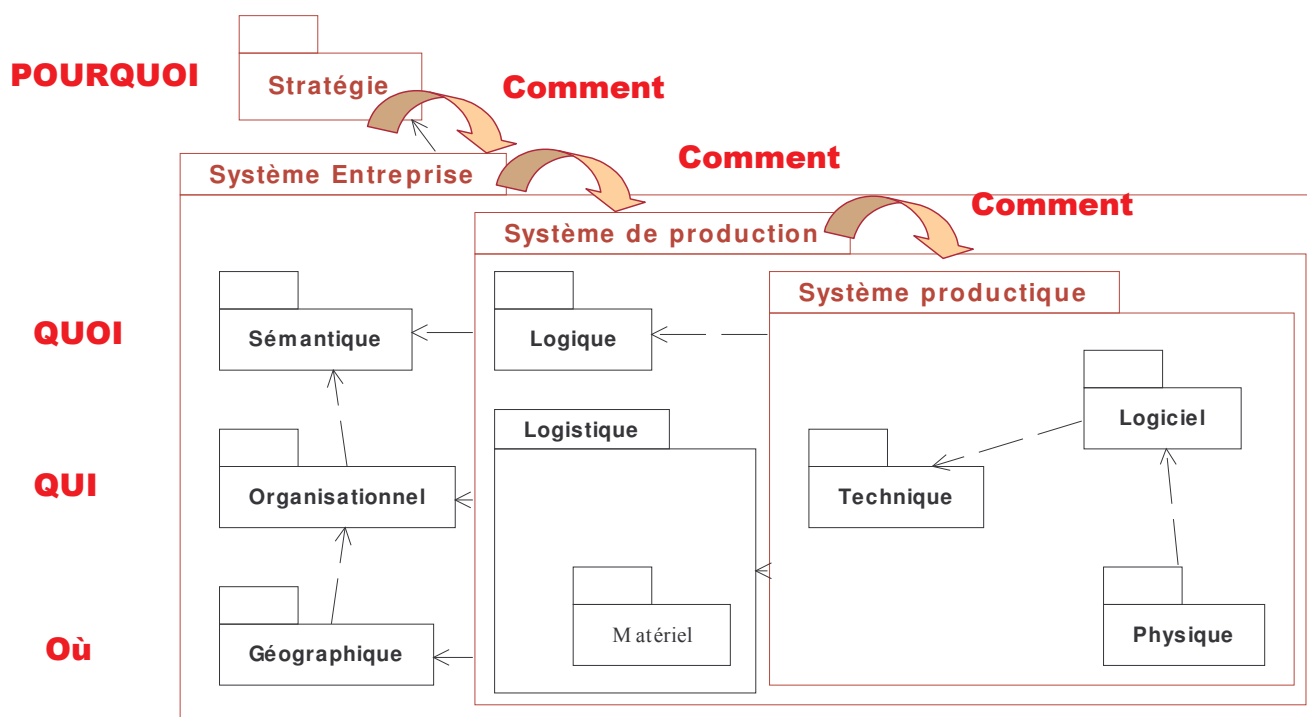
#### Des aspects isolés

En appliquant cette ligne explicative du QQQC, nous avons isolé les différents aspects du Système Entreprise qui demandent à être représentés. Les modèles à réaliser pour construire et faire évoluer le système se définissent par rapport aux aspects qu'ils traitent. La page suivante montre qu'il est possible de simplifier et de traiter plusieurs aspects dans un même modèle.

#### Des aspects articulés

Un point important de la topologie est que les aspects ne sont pas suspendus dans le vide. La topologie clarifie leurs articulations. Les articulations (représentées par des flèches ou dépendances dans le schéma ci-dessous) sont, volontairement, en nombre limité.

Ces relations de référence conditionnent la démarche de réalisation. Elles laissent entrevoir des possibilités de transformations, éventuellement automatisables.



## Complément à la Topologie du Système Entreprise

### Les questions quantitatives : « Quand » et « Combien »

#### Une omission ?

Deux questions du questionnaire du Quintilien n'apparaissent pas dans le recensement des facettes, du moins en apparence : « Quand ? » et « Combien ».

*Le temps* En fait, le temps est une dimension dans laquelle se situe n'importe quel élément de n'importe quelle facette. Il y a, bien sûr, le temps de l'entreprise, la longue durée à travers laquelle se déploie la stratégie, la construction du Système. Mais les objets sémantiques vivent aussi selon une temporalité qui leur est propre (le cycle de vie des objets). La facette « pragmatique » voit se distribuer les événements et les activités : les processus se déroulent, également, dans le temps (c'est une tautologie). À l'autre bout de la chaîne, surgit la question des performances et des temps de réponse. Donc, il n'est pas possible d'isoler le temps comme un aspect séparé. Tout phénomène est temporel, même si la vie de chaque facette peut avoir son rythme propre. En conséquence, le modèle qui décrit une facette, quelle qu'elle soit, devra toujours comporter une représentation temporelle.

Qui plus est, l'information de nature temporelle entre dans un réseau de dépendance, couvrant l'ensemble des aspects. Ainsi, on exprimera, dans la vue de l'utilisation, des exigences de temps de réponse. Dans l'architecture technique, après prototypage expérimental, on dégagera les valeurs types pour les performances. Elles permettront d'éclairer les choix de conception du logiciel et d'anticiper les comportements du système.

*La quantité* Tout comme la question « Quand », la question « Combien » se pose sur tous les aspects de la topologie. Dès le niveau sémantique, il convient de réunir des informations quantitatives telles que :

- Combien existe-t-il d'objets de la classe X ?
- Quelle est la fréquence d'instanciation de la classe ?
- Quelle est la fréquence d'apparition de tel événement ? Y a-t-il un profil d'occurrence ?
- En combien de temps l'objet parcourt-il son cycle de vie ?

L'architecture logique transporte cette question à l'échelle du système. Elle dimensionne les flots d'information et le volume des sollicitations au niveau de ses unités logiques.

Les architectures matérielle et technique abordent la question du « Combien » sous l'angle des capacités (nominales ou extrêmes).

*Le coût* La question se spécialise en « Combien ça coûte ? ». Dans les scénarios de conception, c'est un des éléments à considérer, avec – toujours – deux éléments : coût d'acquisition et coût d'exploitation.

## Complément à la Topologie du Système Entreprise (suite)

---

### Contenu et forme des modèles

---

#### Des outils de représentation

Le tableau des représentations (page suivante) propose une utilisation de la notation UML, en regard des aspects du Système d'action. Ce ne sont que des indications des principaux diagrammes, associés aux aspects. Les guides détaillés apportent davantage de détails et nuancent ces indications.

#### Le recours à UML

UML est un standard international qui présente les avantages suivants :

- il est l'aboutissement d'un consensus international sur les techniques de modélisation ;
- il est appelé à une large diffusion ;
- il est parfaitement outillé, non seulement pour produire les diagrammes mais également pour étendre la notation (particulièrement grâce au standard sur les profils UML).

Dans notre approche, nous tirons cette notation vers les techniques de communication, usage auquel elle se prête bien, sous réserve de prendre quelques précautions.

Les diagrammes mentionnés n'excluent pas le recours aux autres diagrammes de la notation. Ce sont les principaux, sur chaque aspect.

## Complément à la Topologie du Système Entreprise (suite)

### L'apport d'UML pour représenter les aspects : vue externe du Système


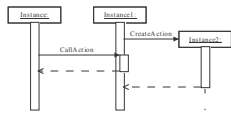
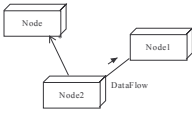
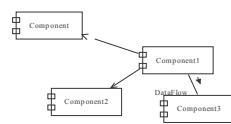
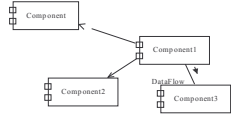
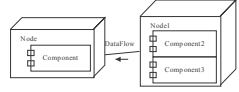
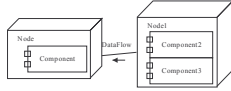
Figure PxM-02\_24. UML pour les aspects de la vue externe

Aspect	Portée locale (programme, solution spécifique)			Portée globale (Produit)		
	Qui	Pourquoi	Comment	Qui	Pourquoi	Comment
Sémantique	Utilisateur	Exprimer la connaissance, les représentations	diagramme de classes  diagramme d'états-transitions 	Utilisateur Décideur	Donner une vue simple des fondamentaux du domaine  Aller à l'essentiel et s'affranchir de l'existant	Référentiel Produit ; Mêmes diagrammes + diagrammes de paquetages 
Pragmatique	Utilisateur	Montrer comment mener l'activité	diagramme des cas d'utilisation 	Organisateur Décideur	Définir les rôles et dessiner les processus  représenter l'organisation	diagramme d'activité ; diagramme de classes montrant la hiérarchie des rôles ; diagramme de collaboration 
Géographique	L'étude de cet aspect ne présente d'intérêt qu'au niveau global.			Décideur	Exprimer les contraintes spatiales	diagramme d'objets ou de collaboration 

## Complément à la Topologie du Système Entreprise (suite)

### L'apport d'UML pour représenter les aspects : vue interne du Système

Figure PxM-02\_25. UML pour les aspects internes du Système

Aspect	Portée locale (programme, solution spécifique)			Portée globale (Produit)		
	Qui	Pourquoi	Comment	Qui	Pourquoi	Comment
Logique	Concepteur de système d'information	Recenser les services et composants de la solution informatique	diagramme de classes ; diagramme de paquetages 	Architecte logique	Maîtriser l'architecture de la solution informatique	diagramme de paquetages diagramme de séquence 
Technique	Informaticien		diagramme d'activité (algorithme) diagrammes dynamiques	Architecte technique	Élaborer l'infrastructure informatique	diagramme de déploiement 
Logiciel	Informaticien	Décrire le détail de la solution informatique	diagramme de composants 	Administrateur des composants	Gérer la configuration	diagramme de composants 
Physique	Informaticien	Installer la solution informatique	diagramme de déploiement 	Architecte technique et support	Maîtriser le déploiement de la solution informatique	diagramme de déploiement 

---

## La topologie dans la méthodologie Praxeme

---

### Conclusion

---

#### Le socle théorique de Praxeme

En conclusion, la topologie du Système Entreprise recense les aspects ou facettes que l'on doit considérer pour se faire une représentation complète de la solution globale et pour agir sur elle avec le minimum d'incertitude. La déduction des facettes s'opère en suivant un procédé didactique qui permet de faire converger les points de vue des différentes expertises impliquées.

La topologie permet, donc, de clarifier les responsabilités de chaque métier et d'articuler les univers. L'articulation entre les facettes doit être établie avec soin : elle doit exprimer les déterminations naturelles entre les aspects, tout en réduisant le plus possible les couplages.

Les applications pratiques de la topologie du Système d'action sont de deux ordres :

- D'une part, la topologie devient un instrument de gestion des compétences et de spécialisation au sein de la chaîne de production du logiciel.
- D'autre part, elle fournit le socle pour fonder les démarches de projets ou de programmes. Venant en complément du standard de représentation UML, elle permet de définir, sans ambiguïté, les modèles nécessaires et d'en fixer le contenu précis.

---

#### Les retombées de la séparation des aspects

Les retombées sont les suivantes :

- Définition rigoureuse des modèles.
- Sériation des questions à se poser.
- Spécialisation des expertises (les procédés de conception sont plus faciles à appréhender au sein d'un aspect ; notamment, l'utilisation d'UML s'éclaire quand on la situe pour un aspect seulement).
- Contribution à la qualité et l'évolutivité du système (chaque aspect évolue à un rythme différent)...

## Sur la modélisation

### Les dimensions de la modélisation

#### Le principe

L'approche orientée objet forme un tout. On ruinerait son intérêt à vouloir la désarticuler, par exemple en répartissant, entre les phases, différentes composantes de la description d'un même phénomène.

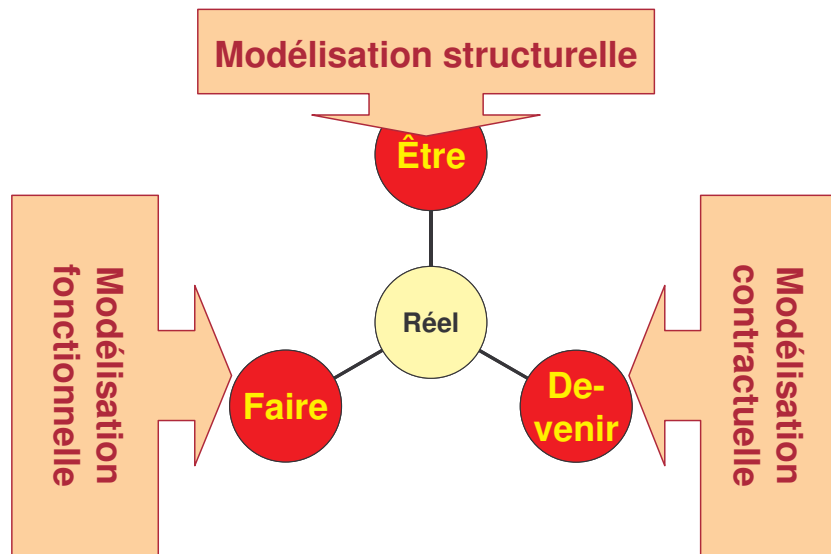
Sur chacun des aspects, la logique objet trouve à s'appliquer. Donc, l'arsenal complet de la boîte à outils UML peut trouver à s'employer à différents moments du processus.

Inversement, chaque objet retenant l'attention du modélisateur doit être examiné dans toutes les dimensions de notre perception. Notre approche du réel peut se résumer par les trois dimensions du réel (en fait, il s'agit plutôt de trois axes de notre connaissance) : ÊTRE, FAIRE, DEVENIR.

**Quel que soit l'aspect sur lequel il porte, un modèle doit envisager ces trois dimensions.**

#### Les dimensions

Figure PxM-02\_26. Les trois axes de la modélisation



*La modélisation structurelle* La modélisation structurelle identifie les classes et objets, et les organise en structures stables.

*La modélisation fonctionnelle* La modélisation fonctionnelle s'applique à différents niveaux : du comportement global du système jusqu'à l'algorithme des opérations. L'opération fait la jonction entre structure et fonction : elle est un morceau de la dynamique et aussi, elle est inscrite dans la structure.

*La modélisation contractuelle* L'approche objet fournit les outils pour décrire le système comme un ensemble de micro-machines, responsables de leur état et de leurs transformations. Ce troisième angle d'attaque est, sans doute, le plus innovant.



## Sur la modélisation (suite)

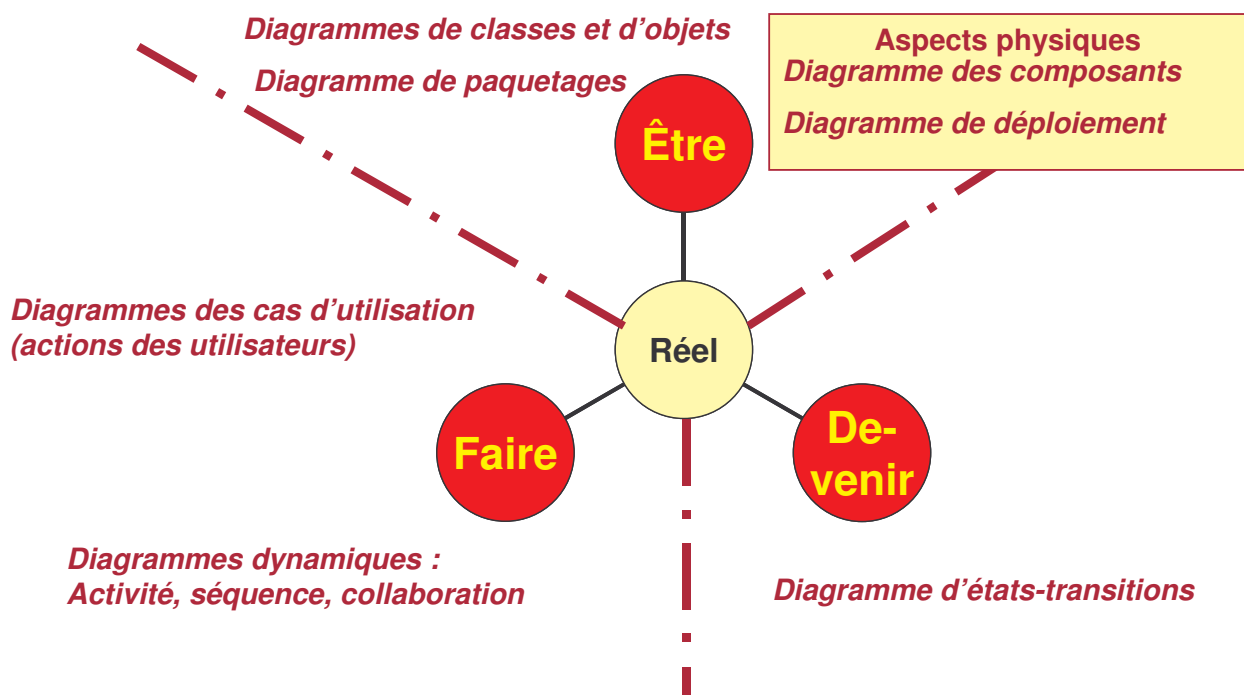
### UML et les dimensions de la modélisation

#### Statique / dynamique

L'opposition classique entre modèle statique et modèle dynamique ne convient pas à l'approche orientée objet. Elle est un héritage de l'approche traditionnelle qui oppose données et traitements. La logique objet agrège ces aspects. Aussi, le diagramme des classes, par exemple, se présente comme statique si on ne le voit que comme distribution des propriétés sur une structure ; mais il contient, en puissance, la dynamique du système puisqu'il en ordonne toute la substance (dont les opérations qui sont les briques élémentaires de la dynamique).

Cette considération conduit à adopter une autre figuration, plus conforme au paradigme objet. La disposition en trois axes offre l'avantage de faire ressortir le diagramme d'états-transitions, lequel – trop souvent négligé –, fournit un outil essentiel de l'approche objet en nous incitant à concevoir la classe comme une micro-machine autonome, responsable de son état interne et des services qu'elle rend.

Figure PxM-02\_27. La distribution des diagrammes UML sur les trois axes de la modélisation



#### Commentaire

Le schéma ci-dessus montre comment les diagrammes UML contribuent aux trois approches du réel.

Les diagrammes de composants et de déploiement constituent des cas à part, puisqu'ils se limitent au logiciel.

Tous les deux décrivent la structure du logiciel. Dans le diagramme de déploiement, certains éléments de notation permettent de décrire des phénomènes dynamiques qui se produisent dans l'architecture informatique.

## Complément sur le processus

### La dynamique globale

#### Les macro-activités

La notion de niveaux de cibles a été introduite (cf. p. 29) afin de sensibiliser aux conséquences de l'urbanisation. Il en découle la nécessité d'une dynamique globale pour accompagner le fonctionnement et l'ajustement de l'entreprise.

Le schéma ci-dessous apporte des précisions en recensant les « macro-activités » de la dynamique globale. Les macro-activités s'étagent selon les niveaux de cibles.

Figure PxM-02\_28. Les macro-activités de la dynamique globale

