



Composant

PxM-50

« Modus : la méthodologie Praxeme »

Guide de l'aspect technique

Objectif

L'effort de modélisation s'investit lourdement sur les aspects amont : de l'aspect sémantique à l'aspect logique. Pour poursuivre dans la chaîne de production, il reste à introduire les choix techniques.

Ce guide montre comment documenter les choix techniques, tant au niveau global (architecture technique) qu'au niveau local des programmes.

Contenu

- L'architecture technique : domestiquer la technologie
- Le positionnement dans la chaîne de production
- Les questions à traiter
- La transformation technique

Rédacteur

Dominique VAUQUIER

Valideurs

Collège des architectes technique

Version

0.9 - 2008-10-27

Éléments de configuration

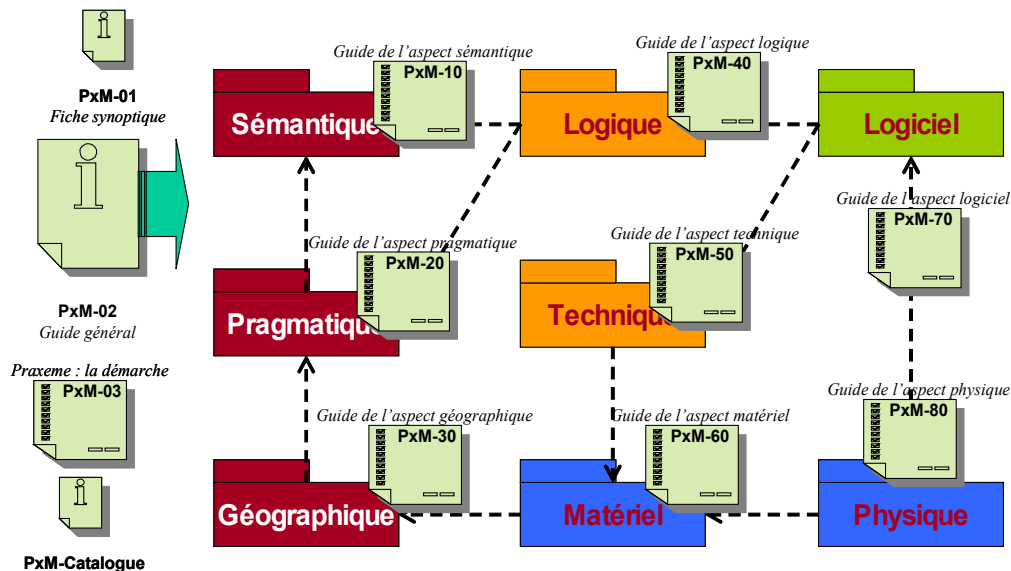
Situation du composant

Positionnement dans la documentation



La méthodologie Praxeme est structurée selon les aspects de la Topologie du Système Entreprise. Le *Guide général* explique cette approche.

Figure PxM-50_1. Architecture du référentiel méthodologique



Validation

Ont participé à la revue de ce document :

<< revue en cours au sein du Collège des concepteurs et architectes logiques ; voir le site : www.praxeme.org >>

Disponibilité

Ce document est disponible sur le site Praxeme et utilisable dans les conditions définies page suivante. Les sources (documents et graphiques) peuvent être obtenues sur demande.

Propriétaire

Le référentiel Praxeme a été élaboré dans le cadre du chantier Praxime. Les contributeurs se réunissent au sein du « Collège des contributeurs » qui oriente les travaux en fonction des préoccupations des entreprises et organismes. L'association *Praxeme Institute* fait évoluer le fonds commun. Le « Collège des architectes et concepteurs logiques (SOA) » prend en charge la validation et l'enrichissement de ce guide et des documents associés¹.

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

L'historique

Indice	Date	Rédacteur	Contenu
0.8	03/03/2004	DVAU	Version issue directement des travaux Sagem
0.9	27/10/2008	Fabien Villard	Mise à la forme du Praxeme Institute. Validation par le collège des Architectes technique.

¹ Pour information sur le fonctionnement : <mailto:college-technique@praxeme.org> .

Licence

Conditions d'utilisation et de diffusion

Droits et devoirs

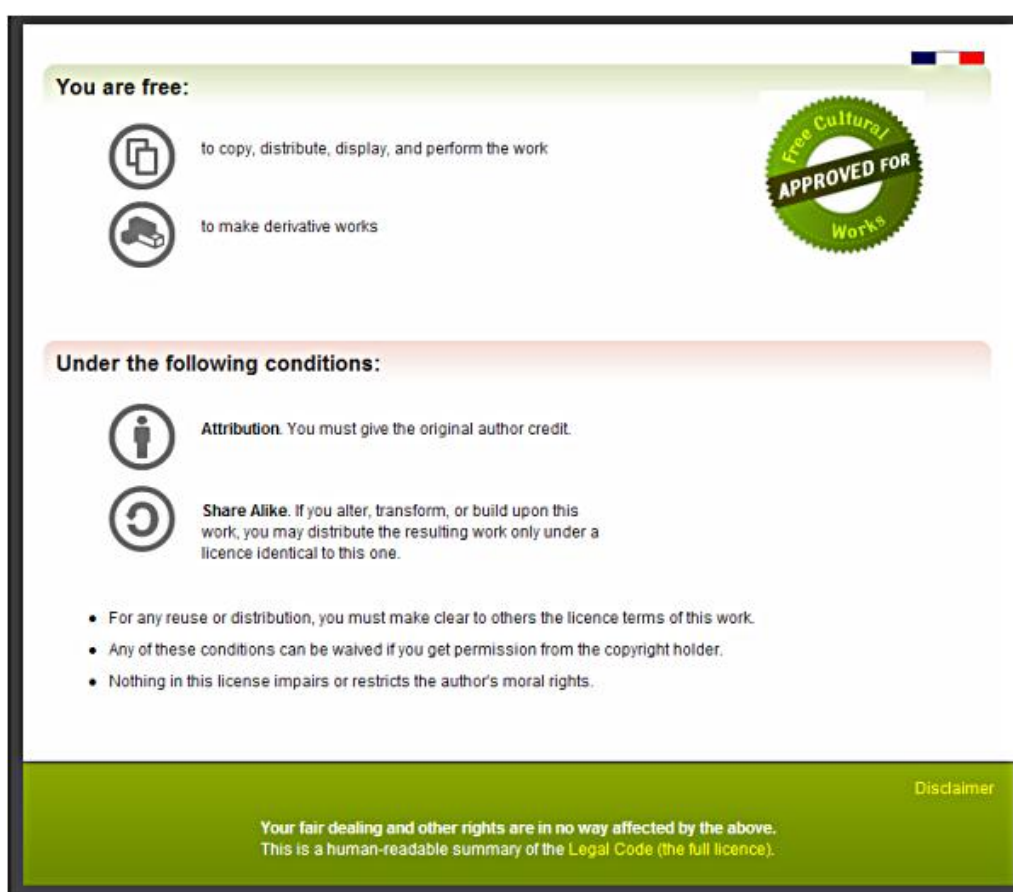
Ce document est protégé par une licence « [Creative Commons](#) », résumée ci-dessous. Le terme « création » s'applique au document lui-même. L'auteur original est :

- Dominique VAUQUIER, pour le document ;
- l'association *Praxeme Institute*, pour l'ensemble de la méthodologie Praxeme.

Nous vous demandons de citer l'un et/ou l'autre, selon que vous extrayez une citation directe ou que vous vous référez aux principes généraux de la méthodologie Praxeme.

Sommaire

GUIDE DE L'ASPECT TECHNIQUE.....	1
Situation du composant.....	2
L'historique.....	2
Introduction.....	5
L'architecture technique : domestiquer la technologie.....	5
La modélisation technique : ses produits et ses procédés.....	6
Définition et objectif de la conception technique.....	7
Expertise et architecture : deux métiers.....	7
Les termes de la modélisation technique.....	8
Les produits : référentiel et modèle.....	9



La documentation de l'architecture technique.....	9
Deux préoccupations générales.....	10
Les questions à traiter.....	11
La formalisation des choix techniques.....	13
Le récapitulatif de la démarche : la correspondance topologique.....	14
Le positionnement dans la chaîne de production.....	15
Les conditions pour le processus de développement.....	15
La négociation logique/technique.....	16

Les procédés de la modélisation technique.....	17
Les techniques de représentation.....	17
La transformation technique.....	18
La communication dans le système : typologie formelle.....	19
La communication dans le système : typologie de solution.....	20

Table des figures

FIGURE PxM-50_1. ARCHITECTURE DU RÉFÉRENTIEL MÉTHODOLOGIQUE.....	2
FIGURE PxM-50_2. TABLE DE CORRESPONDANCE LOGICO-TECHNIQUE.....	14
FIGURE PxM-50_3. TABLE DE CORRESPONDANCE LOGICO-TECHNIQUE POUR UNE ARCHITECTURE HÉTÉROGÈNE. 14	
FIGURE PxM50_4. PRINCIPE POUR LE TABLEAU RÉCAPITULATIF DE LA CORRESPONDANCE TOPOLOGIQUE.....	15
FIGURE PxM-50_5. TYPOLOGIE DES ÉCHANGES ENTRE LES COMPOSANTS.....	20
FIGURE PxM-50_6. TYPOLOGIE DES DISPOSITIFS TECHNIQUES ASSURANT LA COMMUNICATION ENTRE LES COMPOSANTS.....	21

Remerciements

Introduction

L'architecture technique : domestiquer la technologie

Objectif du guide

Ce guide rassemble les questions à examiner pour établir les choix techniques.

Domaine et circonstances d'application

L'architecture technique est du ressort de la conception globale du système. Son élaboration est donc associée à la visée Ligne de Produit et inscrite dans le chantier d'urbanisation.

À ce niveau, l'architecture technique peut proposer un catalogue de solutions compatibles avec les orientations de la Ligne de Produit. Le cas échéant, le concepteur a en charge de préciser les choix retenus pour le programme. Dans le cas contraire, c'est-à-dire si l'architecture technique se fige sur un seul scénario de solution technologique, le concepteur du programme est dispensé de toute conception technique.

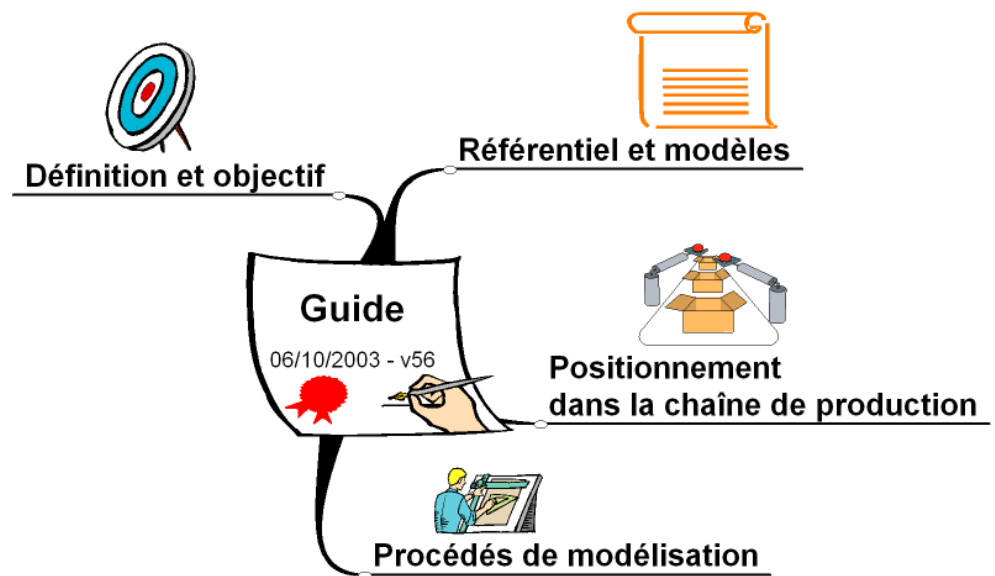
Introduction (suite)

La modélisation technique : ses produits et ses procédés

Contenu du guide

Après la définition de la modélisation technique et la sensibilisation à ses enjeux, le guide aborde les trois dimensions :

- produits ;
- processus ;
- procédés.



Produits

- Les choix techniques et leur justification.
- Les normes de développement.

Processus

Un point important à considérer pour mettre en place la dynamique de développement est la question de la « négociation » entre logique et technique.

On examinera, également, la distribution des activités relatives aux technologies, sur le processus de développement.

Procédés

Le besoin de représentation pour la conception technique se limite à des compléments apportés à l'architecture matérielle.

Définition et objectif de la conception technique

Expertise et architecture : deux métiers

L'attitude

De l'analyse à la conception

Les travaux sur l'architecture comportent deux moments :

- l'analyse architecturale, préalable nécessaire par lequel l'architecte saisit les contraintes, exigences et enjeux qui vont guider la conception ;
- la conception, proprement dite, qui consiste à arrêter – ou proposer – les choix techniques pour la nouvelle solution.

Les compétences

L'élaboration de l'architecture technique mobilise deux types de compétences, difficilement compatibles dans le même cerveau :

- **L'architecte technique** est celui qui, sans pouvoir être expert dans tous les domaines, peut se faire une vue assez générale et embrasser assez de sujets pour composer et évaluer différents scénarios, sans rien négliger des possibilités du marché.
- **L'expert technique** n'a pas l'étendue de champ de l'architecte mais peut apporter les réponses sûres aux points particuliers liés à une technique ou un scénario.

La distribution des rôles doit être claire. L'architecte mobilise les experts en fonction des besoins d'approfondissement pour élaborer et juger les scénarios d'architecture. L'expert technique s'engage sur la faisabilité des éléments de solution. Il doit percevoir les limites au-delà desquelles il convient soit de passer le relais à un autre expert, soit déclencher une étude de faisabilité.

Le rapport entre ces deux types de métier est celui de l'étendue et de la profondeur.

L'enjeu

L'architecture compte parmi les arts. Elle formule des choix généraux, mais tout l'art – et le risque – est d'assurer que le système résultant fonctionnera dans le détail.

L'objectif du modèle technique

Avec l'architecture technique, nous atteignons le cœur de la méthode Praxeme, qui se révèle comme un jeu de correspondances. Les termes propres aux différents aspects se renvoient les uns aux autres, jusqu'à se traduire dans la solution informatique.

Les choix techniques réalisent la dernière de ces transformations.

Le dossier d'architecture technique fixe les choix de technologie et de produits commerciaux qui entreront dans la solution ou permettront de la bâtir. Il explicite, également, la façon de mettre en œuvre ces technologies.

Définition et objectif de la conception technique (suite)

Les termes de la modélisation technique

Les termes

Dans quels termes le modélisateur – analyste ou concepteur – exprime-t-il l'aspect technique ? Quelles unités utilise-t-il pour décrire l'utilisation du système et son organisation ?

Les rubriques suivantes répondent à ces questions.

On entend, ici, par 'termes', les catégories syntaxiques de la modélisation.

Les constituants de l'architecture technique

- le système d'exploitation (multi-tâches ou non) ;
- les langages de programmation (leur structure – objet ou non - ; leur puissance d'expression ; l'outillage associé ; les possibilités de programmation parallèle, événementielle, etc.) ;
- les solutions de persistance (SGBD ou fichier) ;
- les gestionnaires de transaction (intégré ou non dans un middleware) ;
- la communication dans le système (middleware ou plus rustique ; bus...).

Les termes décrivant l'exécution

- processus (*process*) et tâches (*threads*) ;
- appels, envois de message, flots ;
- charges mémoire ; gestion de la mémoire (ramasse-miette²...) ;
- trafic...

Les préoccupations

L'architecture technique contribue largement à atteindre des objectifs généraux tels que :

- les performances ;
- l'interopérabilité ;
- la portabilité...

Ces exigences non fonctionnelles et générales (en ce sens qu'elles s'étendent à la totalité du système) sont examinées et pondérées à travers l'analyse architecturale.

² Le *garbage collector* des technologies objet.

Les produits : référentiel et modèle

La documentation de l'architecture technique

La différence entre logique et technique

Au niveau logique, on a fait l'hypothèse qu'un composant logique A pouvait, sous condition, communiquer avec un autre composant logique B (par exemple, A et B sont des « Machines Logiques » liées par une relation de dépendance).

Au niveau technique, ce simple appel fait l'objet de toutes les attentions. Il se réalise, sans doute, en « traversant » les couches techniques (descente et remontée). L'architecte technique ne s'intéresse pas au contenu et à la vocation des machines, mais il démonte le mécanisme qui leur permettra de fonctionner de concert.

C'est ici, dans l'aspect technique seulement, que se place le Modèle en couches (*pattern* d'architecture). Il structurera, par voie de conséquence, l'aspect logiciel.

Le contenu

L'architecture technique comporte deux volets bien distincts :

1. Tout d'abord, elle répond au « quoi » en exprimant les solutions techniques générales.
2. Mais aussi, il faut qu'elle légifère sur le « comment », les développeurs ayant besoin de savoir, précisément, comment mettre en œuvre les solutions techniques.

Ce deuxième volet se formule à travers les **normes de développement** et les règles d'utilisation des environnements de développement.

Les deux volets sont forcément corrélés.

On a tendance à omettre le deuxième volet, or il est aussi nécessaire que le premier à la bonne marche des développements. Les règles prescrites par l'architecte technique se conforment à la philosophie générale de la méthodologie et concrétisent les recommandations par le biais de l'atelier de développement.

Le prototypage

La conception sur table laisse subsister des zones d'incertitude. L'architecte technique est confronté à une profusion d'offres en ébullition permanente. Une même solution générale (par exemple, le middleware CORBA normalisé par l'OMG ou les Web Services et l'ensemble des standards WS-*) reçoit plusieurs réalisations commerciales, lesquelles évoluent elles-mêmes. Le rythme de rafraîchissement de l'information technique est plus proche de six mois que de plusieurs années. Tout particulièrement, lorsque l'architecte technique a pour mission de préparer les solutions de l'avenir, il se trouve face à un imbroglio d'hypothèses, de projections, de supputations...

Après décanation des hypothèses et stabilisation du plan général, **le seul moyen à la disposition de l'architecte pour réduire les zones d'incertitude et garantir la faisabilité de la solution est le prototype.**

Pour vérifier les choix d'architecture, on parle de « prototypage expérimental ». Il se caractérise par son indifférence au contenu fonctionnel et son intention focalisée exclusivement sur la vérification des hypothèses techniques³.

³ Pour une typologie des prototypes, voir DRM-10, à la fin de la partie sur les procédés.

Les produits : référentiel et modèle (suite)

Deux préoccupations générales

Avant de parcourir les différentes questions de nature technique, les deux paragraphes ci-après considèrent deux points généraux qui orientent la recherche des technologies.

La logique objet

Les procédés formulés dans Praxeme se mettent au niveau de l'état de l'art, sinon celui des pratiques, du moins celui d'une conversion radicale de l'approche. Le choix du paradigme objet n'est pas anodin : il se justifie par des visées de longue portée.

Arguments endogènes

Tout d'abord, du point de vue de l'informaticien, l'état des technologies ainsi que le génie logiciel conduisent à opter pour l'approche orientée objet, en raison des retombées de productivité et de qualité⁴.

Arguments exogènes

Les avantages de l'approche objet sont sensibles, également, en dehors de l'informatique. En effet, la restructuration opérée sur le produit logiciel, voire sur les processus « métier », parfois la reformulation du savoir-faire, préludent à l'innovation. La logique objet, menée à l'extrême, transmute le regard porté sur le « cœur de métier » et peut produire des effets désastreux.

Conséquences

Ces arguments légitiment l'effort de la modélisation menée sur les aspects amont. Le défi pour l'architecture technique est de **préserver, le plus possible, les qualités objets** de ces modèles en les transposant dans la solution informatique.

La vertu attendue de l'architecture technique est, donc, sa conformité au modèle objet (proximité sémantique). Dans les informatiques « lourdes » où les idées préconçues, le poids de l'existant et la force des habitudes réduisent considérablement la marge de manœuvre, les orientations techniques peuvent aller jusqu'à imposer à l'architecture logique de s'exprimer dans des termes étrangers à la structuration objet. On en reste alors à l'urbanisation fonctionnelle.

Même dans les cas où le contexte interdit le recours au « tout objet », l'architecte technique doit rechercher un cocktail technologique qui évitera de mutiler la structuration objet du modèle logique et en préservera les vertus jusque dans le logiciel.

Le principe d'indépendance logique

Une autre orientation du génie logiciel depuis le début des années 90 est l'indépendance entre la structure logique du système et les choix techniques. Ce principe revêt un caractère d'importance puisqu'il permet, en théorie, de profiter des évolutions technologiques en limitant l'impact sur le système logiciel.

Peut-on le réaliser complètement ? Cette question est du ressort de l'architecture technique. Elle se rapproche du principe au prix, parfois, d'un compromis établi lors de la négociation logique/technique. Ce compromis consiste à réduire la liberté formelle de l'architecte logique en lui imposant des contraintes de structure ou de syntaxe, les termes de la conception logique⁵.

⁴ Même si ces retombées ne se manifestent pas toujours, faute d'une véritable conversion culturelle et d'un effort suffisant de modélisation.

⁵ C'est la posture adoptée par Praxeme en ce qui concerne l'aspect logique : la modélisation logique utilise une théorie restreinte de l'approche objet ; la modélisation logique se dote de son propre vocabulaire pour tenir les deux bouts de l'équation : sémantique pur objet, d'un côté ; technique, de l'autre.

Les produits : référentiel et modèle (suite)

Les questions à traiter

La persistance

L'architecture technique comprend les solutions de stockage. Quand elle retient un système de gestion de bases de données orienté objet, on échappe à la « fracture sémantique » : le modèle objet obtenu sur les aspects externes est conservé tel quel et devient, à quelques retouches près, le schéma physique de la base de données.

Si le SGBD ou la solution de stockage (qui peut être plus rustique) ne sont pas orientés objet, le modèle objet subit une transformation importante qui affecte également la distribution des traitements.

L'architecture technique fixe les règles de transformation du modèle des données. La question la plus délicate est celle de la transformation des relations d'héritage.

La transformation des relations d'utilisation

Le graphe d'architecture logique montre les relations de dépendance entre les paquetages. Ces relations résument les liens entre les objets, rangés dans les différents paquets. Ces liens sont de deux natures :

- **liens statiques** : établis par les associations entre les classes ;
- **liens dynamiques** : correspondant à la connexion temporaire entre deux objets, par l'envoi de message.

Une des questions à traiter systématiquement dans l'architecture technique est la transformation de ces liens :

- La transformation des liens statiques fait partie de la transformation plus générale du modèle logique des données.
- Les liens dynamiques appellent un dispositif si possible homogène au sein de tout le système (Comment réaliser les appels de services ?).

La transformation des relations d'extension

Les relations d'extension entre les paquetages de l'architecture logique peuvent recevoir plusieurs traductions techniques. En fonction des technologies retenues, l'architecture en sélectionne une et une seule, valable dans tout le système. Là encore, la présence du mécanisme d'héritage est déterminante. Il faut considérer, également, les performances. Ce souci peut justifier la duplication de code, même avec la possibilité d'héritage.

L'allocation des ressources

Les possibilités architecturales d'allocation des ressources aux processeurs se classifient selon les degrés suivants :

1. L'allocation fixe : définitive et inscrite « en dur » dans le logiciel, elle rigidifie le système et ne permet pas son ajustement en fonction des performances observées ou des évolutions.
2. L'allocation statique : appuyée sur un annuaire des services, elle permet la redistribution de la charge, par un acte d'administration.
3. L'allocation dynamique : elle exploite, en temps réel, les ressources disponibles et choisit la meilleure distribution au moment de l'exécution.

Les produits : référentiel et modèle (suite)

Les questions à traiter (suite)

La transformation des composants

Le propos général de l'architecture technique est de trouver les meilleurs équivalents logiciels pour les catégories de représentation utilisées dans les modèles externes. L'intention est de réduire le plus possible la distorsion introduite par l'informatique par rapport au modèle naturel.

Ainsi, l'architecture technique dit comment traduire en logiciel les composants logiques du système, à chaque niveau d'agrégation : service (opération), machine (classe), atelier (dll, exe, ...), aire (espace de nommage).

Le paragraphe « La transformation technique », dans la dernière partie, reprend ce sujet.

La transformation des catégories exotiques

Les modèles amont, jusqu'au logique, recourent, entre autres, aux catégories suivantes⁶ :

- automate à états ;
- événement ou signal ;
- processus (ordonnancement d'activités).

L'architecte technique peut s'intéresser à ces sujets que des solutions du marché peuvent couvrir en tout ou en partie, selon le degré d'exigence et le style d'architecture que l'on souhaite.

Notamment, si la dimension des processus organisationnels prédomine dans le système, les solutions de *workflow* s'offrent à l'attention.

La gestion des événements peut trouver des réponses techniques dans la programmation événementielle (au moins, métaphoriquement) ou les dispositifs à base de messages.

Gamma et al. donnent une solution pour la transformation des *state machines*, sous la forme d'un pattern générant une classe par état. On peut s'en contenter : le pattern devient alors la solution technique pour transformer tous les automates. Si, au contraire, on trouve coûteuse ou insatisfaisante cette génération, alors deux directions s'ouvrent :

- soit, l'architecte technique recherche une autre solution du marché ;
- soit, la conception d'un dispositif propre revient à l'architecte logique.

⁶ Ces catégories sont réunies sous l'appellation « exotique » car elles sont externes à l'informatique, du moins externes aux langages de programmation.

Les produits : référentiel et modèle (suite)

La formalisation des choix techniques

Principe d'exposition

Outre les comptes rendus de recherche et d'expérimentation de technologies ainsi que la justification des solutions retenues, la partie utilisée par les développeurs peut se présenter sous la forme d'un simple tableau de correspondance.

Figure PxM-50_2. Table de correspondance logico-technique

Aspect	1	2	3	4	5			n
Logique	Service	Appel	Mach ine	Atelier				
Technique	Opérati on JAVA		Classe	Library				

Cas d'une architecture hétérogène

L'architecture peut être plus complexe et prévoir différentes solutions pour différentes parties du système. Les « sous-ensembles techniques » se composent de choix cohérents (système d'exploitation, langages, SGBD disponibles...). L'architecture hétérogène soulève la question de la connexion des sous-ensembles techniques, voire de leur cohabitation dans les mêmes nœuds du système.

Figure PxM-50_3. Table de correspondance logico-technique pour une architecture hétérogène

Aspect	1	2	3	4	5	Langage	Syst. exploit	Transaction	n
Logique	Servi ce	Appe l	Mach ine	Ateli er					
Sous-ensemble technique n°1		RPC				C	Unix		
Sous-ensemble technique n°2			Com po sant			Java	Unix		
Sous-ensemble technique n°3		COR BA					Win.	COR BA	

Les produits : référentiel et modèle (suite)

Le récapitulatif de la démarche : la correspondance topologique

Présentation complète

Si les choix techniques ont pu préserver la continuité formelle d'un aspect à l'autre, le concepteur peut résumer la méthode opératoire (méthodologie et technologie) dans un tableau récapitulatif unique. Celui-ci servira de boussole dans l'espace multidimensionnel de la topologie, pour tous les participants aux travaux, des modélisations amont jusqu'au déploiement.

Figure PxM50_4. Principe pour le tableau récapitulatif de la correspondance topologique

Cet exemple est très incomplet. Il est proposé pour illustrer le principe de construction du tableau.

Aspect					
Sémantique	Concept	Propriétés			
Pragmatique			Processus	Rôle	
Géographique	↓	↓		↓	Site
Logique	Machine logique	Services logiques		Machine « Organisation »	↓
Matériel	(types)		↓		Serveur
Technique	Langage Java	↓	Solution de workflow		
Logiciel	Classe Java	Opérations			
Physique	Partie du composant installé				process

Le positionnement dans la chaîne de production

Les conditions pour le processus de développement

Par rapport à l'architecture logique

La négociation logique/technique est un préalable à l'effort de conception logique (détail page suivante).

Le processus de conception doit la prévoir et ménager les points de rencontre nécessaires pour que les deux conceptions d'architecture puissent converger sur l'aspect logiciel, sans surprise.

Par rapport à l'architecture matérielle

Dans le processus de développement, architecture matérielle et architecture technique peuvent se fondre dans la même phase. Ceci facilite le jeu d'équilibrage entre ces deux aspects.

En fait, l'aspect technique peut fusionner avec l'aspect matériel.

→ produit = diagramme de déploiement avec nœuds, connexions et choix techniques sur chacun

La conception technique détaillée est assez différente de la conception d'architecture technique.

La conception technique détaillée

Architecture technique : choix généraux, d'abord à partir des exigences et des indications de volume (on se place en amont des développements ; on ne connaît pas le contenu)

La conception technique détaillée se place à un stade plus avancé.

Elle embrasse les aspects logiciel et physique et peut même, dans certains cas, envisager l'aspect matériel. Elle ajuste la solution.

Dans la conception technique détaillée, l'aspect physique est la vue du fonctionnement.

Une bonne architecture technique doit prendre en compte, également, tous ces aspects ; mais le travail est limité par l'état d'incertitude. On n'a d'autres parades que les prototypes et une estimation a priori des comportements du système.

Le positionnement dans la chaîne de production (suite)

La négociation logique/technique

Le besoin

Le guide de l'aspect logique (référence PxM-40) introduit ce moment crucial de la négociation entre l'aspect logique et l'aspect technique⁷.

Effectivement, une **zone trouble** subsiste entre ces deux aspects ; de nombreuses questions peuvent faire l'objet de l'une ou l'autre des approches. Surtout, on prendra soin d'éviter les confusions « technico-fonctionnelles », véritables « perversions » des architectures informatiques.

Le partage des responsabilités

L'enjeu de la négociation logico-technique (ou technico-logique) n'est pas le partage de responsabilité entre l'architecte logique et sa contrepartie technique mais bien une **distribution des questions à traiter**. Sur chacune des questions présentées ci-après, la négociation doit indiquer un et un seul aspect dans lequel elle sera prise en charge. Aucune question ne doit rester dans la zone trouble où les deux approches pourraient prétendre récupérer le sujet.

Pour être en mesure de distribuer les questions, la connaissance précise des solutions ne constitue pas un préalable. Il suffit d'avoir une idée a priori des possibilités technologiques. Elles feront l'objet de vérifications ultérieures.

Bien sûr, dans une phase d'exploration, on peut envisager de mener en parallèle une estimation du coût de la solution maison (via la conception logique) et une étude du marché (conception technique). Mais, cette phase terminée, chaque question doit être attribuée à l'aspect où son traitement paraît le plus intéressant.

La ligne de partage

La tendance est au glissement des questions « transverses », du logique vers le technique. L'idée de bon sens est celle-ci : il est inutile d'investir sur la conception de dispositifs spécifiques quand l'architecture technique ou logicielle peut intégrer des solutions clefs en main.

Ainsi, des questions comme la sécurité, la gestion des transactions ou des exceptions, la communication, la gestion des flux (*workflow*), etc. sont-elles couvertes par des solutions du marché, intégrant de nombreuses exigences qu'une informatique spécifique serait bien en peine de satisfaire. Cependant, des exigences spécifiques très fortes justifient de repousser la solution du marché pour élaborer une réponse appropriée. La conception logique reprend alors ses droits.

⁷ Cette activité est certainement datée historiquement. Au début de l'informatique, elle n'avait pas cours puisque le fonctionnel n'avait d'autre choix que de s'aligner sur le technique, du fait des limitations de celui-ci. Dans le futur, le rapprochement des technologies vers la sémantique ou une expression plus directe de celle-ci rendra, peut-être, la négociation caduque.

Les procédés de la modélisation technique

Les techniques de représentation

Le diagramme de déploiement

L'architecture matérielle apparaît sous la forme d'un diagramme de déploiement ne comportant que les nœuds et les connexions, sans précisions de technique, ni de protocole. La conception technique reprend ce diagramme et le complète avec les choix techniques :

- types de communication indiqués sur les connexions (protocoles réseaux, bus...);
- types de systèmes sur les nœuds.

Le diagramme des composants

Le diagramme des composants est utilisé pour inscrire dans le référentiel, les composants de nature technique.

Dès lors que l'on parle des composants logiciels, leur vraie place est dans l'aspect logiciel. Ce point de pratique est important car il conditionne la possibilité d'établir les relations entre les composants dans l'architecture logicielle.

En conséquence, on doit distinguer :

1. La documentation propre à l'aspect technique : cette documentation mentionne, bien évidemment, les composants techniques (qui sont des composants logiciels d'un genre particulier : ils ne sont pas générés, ni même associés, à un composant logique).
2. La base de modélisation : en tant que logiciels, ces composants techniques s'inscrivent dans le paquetage qui couvre l'aspect logiciel (et non dans le paquetage « Technique »).

Les diagrammes dynamiques

Le modèle technique doit contenir quelques diagrammes dynamiques : séquence, collaboration ou activité.

Le **diagramme de séquence** est un bon outil pour illustrer les choix techniques qui assurent la traversée du système. En général, les recommandations bornent la largeur du diagramme de séquence à une demi-douzaine d'objets. La modélisation technique fait exception, puisque le but est de montrer, indépendamment du contenu, comment le système fonctionne « en épaisseur ». De larges diagrammes de séquence permettent de pénétrer en profondeur dans le système et de montrer ses entrailles.

Ainsi, contrairement à tout autre aspect, il n'est pas choquant de montrer – au titre de l'architecture technique – un diagramme de séquence ou de collaboration comportant de nombreuses instances, encore qu'il soit toujours possible de le présenter par morceaux.

Les procédés de la modélisation technique (suite)

La transformation technique

Définition

C'est l'architecture technique qui fixe les règles de transformation des composants logiques en composants logiciels.

Le niveau de maille pour la transformation technique

Le niveau de maille de la transformation technique est établi lors de la négociation avec l'architecte logique. Normalement, ce niveau correspond, dans les termes de l'aspect logique, à l'atelier logique.

Cette transformation fournit un composant logiciel, lui-même décomposable. Les services logiques deviennent des opérations, si tant est que l'on dispose d'un langage orienté objet. On peut considérer une opération comme un composant et la rentrer en configuration.

Quand l'architecture technique est multiple ou hétérogène (elle propose plusieurs sous-ensembles de choix techniques, compatibles avec la cible), il peut arriver qu'un même composant logique produise plusieurs composants logiciels. Ils font la même chose mais tournent dans des environnements différents.

La localisation physique

Cette question de la maille pour la transformation doit être abordée selon un autre angle : celui de la localisation du composant logiciel sur l'architecture matérielle.

L'unité retenue servira aussi à placer le composant sur un nœud du système. Il n'est pas question de casser un composant pour le placer par morceaux dans plusieurs processeurs.

L'atelier logique assume cette condition : délimité par le souci de la cohérence et de l'autonomie des services, il forme un bloc insécable que l'on peut placer avec une relative indépendance.

La prise en compte des performances

Le souci des performances de la solution future alimente l'angoisse de l'architecte technique. On a vu que ce sujet concerne, déjà, d'autres aspects – logique, notamment –, mais c'est de l'architecture technique que l'on attend le plus de lumières sur le sujet.

Une première consigne est la séparation des composants *time-critical*. Cette consigne doit remonter vers l'architecture logique, puisque c'est là que l'on délimite les composants.

L'architecte technique peut rechercher une solution qui permette l'allocation flexible des exécutables aux processeurs :

- Statique = placer les composants exécutables dans l'architecture matérielle, sans impact sur les sources.
- Dynamique = allocation au moment de l'exécution (contingence).

C'est sur le plan physique que l'on traitera finement et avec rétroaction immédiate les décisions d'allocation des exécutables sur les processeurs.

Les procédés de la modélisation technique (suite)

La communication dans le système : typologie formelle

Les types de communication

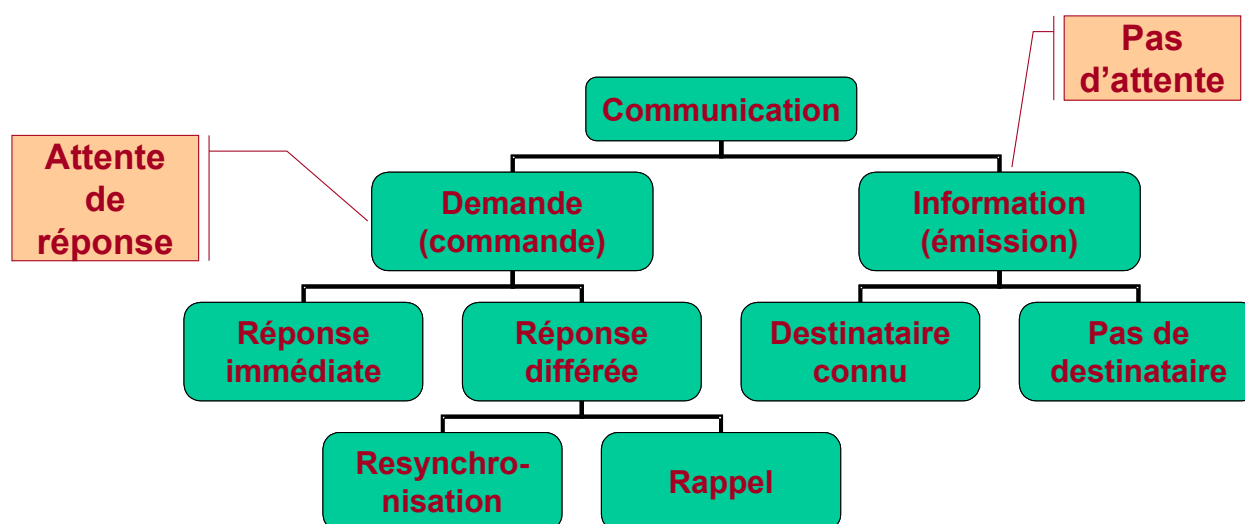
Architecturer, c'est isoler des éléments selon des règles et, dans le même mouvement, les relier par des communications.

Les solutions pour la communication conditionnent fortement le fonctionnement du système et son comportement à long terme.

C'est pourquoi nous nous attardons sur ce sujet et proposons une typologie des communications entre les composants du système informatique.

Ces communications se jouent au niveau le plus fin du système : celui de l'appel de services. Le sujet est d'autant plus important que l'approche objet généralise le principe de coopération et conduit à une augmentation significative des échanges au sein du système.

Figure PxM-50_5. Typologie des échanges entre les composants



Commentaire

Les services, par leur signature même ou au moins par leur contrat, révèlent leur mode de communication.

Certains modes de communication passent par le concours de plusieurs services (sur le modèle de la souscription).

Le type de communication entraîne, pour partie, la modalité temporelle de fonctionnement : synchrone dans le cas de la réponse immédiate ; asynchrone dans tous les autres cas.

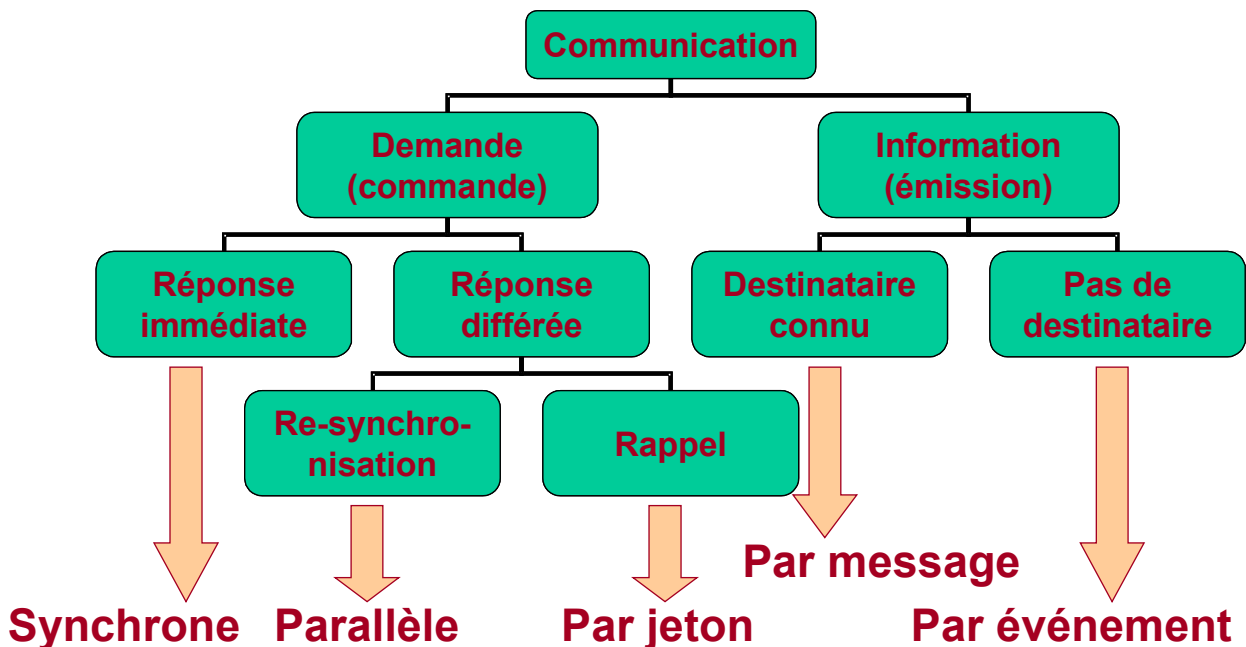
Les procédés de la modélisation technique (suite)

La communication dans le système : typologie de solution

Le schéma ci-dessous désigne les dispositifs techniques correspondant à ces divers types de communication.

Il faudrait aussi considérer le contenu de l'échange : message (information directe) ou fichier.

Figure PxM-50_6. Typologie des dispositifs techniques assurant la communication entre les composants



Commentaire

•Attente de réponse :

- La réponse est immédiate pour l'exécution.
- La réponse différée avec re-synchronisation en cours d'exécution.
- La réponse différée avec rappel au cours d'une autre exécution.

•Sans attente de réponse :

- Dans le cas où le destinataire est connu : il s'agit d'un message posté.
- Dans le cas où le destinataire n'est pas connu : un événement est émis (publication). Dans ce cas, le couplage est absolument nul.

== Page intentionnellement laissée blanche ==