

Des compétences pour maîtriser les SI

La définition des disciplines à partir de la méthodologie Praxeme

Dominique Vauquier

« ...métier, c'est ministère (ministerium, dans lequel minus s'entrevoit). Il est intéressant de noter que le langage a utilisé ce mot dans des locutions dont l'une en relève le sens : métier de roi; l'autre le réduit à désigner une machine : métier à tisser. »

Paul Valéry, *Regards sur le monde actuel*.

« ...ce n'est point un talent que je vous demande, c'est un métier, un vrai métier, un art purement mécanique, où les mains travaillent plus que la tête (...) je veux absolument qu'Émile apprenne un métier. Un métier honnête, au moins, direz-vous ? Que signifie ce mot ? Tout métier utile au public n'est-il pas honnête ? »

Jean-Jacques Rousseau, *Émile, III*

« Lege, lege, relege ; ora, labora et invenies » (« Lis, lis, relis ; prie, travaille et tu trouveras ») devise alchimique

A l'origine de l'initiative pour une méthode publique, il y a le souci d'articuler les expertises. D'emblée était donc posée la question des compétences et de la cohabitation d'univers cognitifs variés. Mais la méthodologie ne se contente pas d'une recension¹ des métiers établis : elle part de l'analyse de la réalité sur laquelle nous agissons, et elle en déduit les compétences nécessaires. Ce faisant, nous passons d'une logique de la capacité à une logique de la nécessité : de « ce que l'on aime faire » à « ce que l'on a à faire ». Cette approche nous conduit à revisiter les métiers et à distribuer différemment les responsabilités, en respectant à la fois les exigences du « Produit » et les possibilités des hommes.

La complexité, cause des évolutions de nos métiers ?

Les informaticiens sont confrontés à une complexité croissante. C'est, du moins, ce que nous nous répétons à l'envi, non sans une certaine complaisance. Certes, la combinatoire des techniques a augmenté, les systèmes s'interpénètrent et les services proposés s'immiscent de plus en plus dans la vie et dans la société. Par-dessus tout, le sentiment de la complexité provient de l'impuissance devant les systèmes existants. D'où, notamment, l'importance que revêt l'architecture. Nous constatons cependant que la discipline de l'architecture s'exerce comme sur la défensive et qu'elle ose à peine concevoir. Le poids de l'existant inhibe toute créativité.

Nos prédécesseurs – et certains d'entre nous qui étaient dans la profession il y a vingt ou trente ans ou plus – avaient également le sentiment de se coller à la complexité : technologie récente, machines peu fiables, nécessité de maîtriser toutes les couches de l'informatique jusqu'à gérer soi-même la mémoire... Quelle fut leur attitude face à ces difficultés ? Ils constituèrent le corpus du génie logiciel² ; ils

inventèrent les techniques de modélisation. Peut-il y avoir une réponse plus censée, plus rationnelle que celle-ci : « pour agir efficacement sur les choses, commençons par les représenter » ? Comprendre, d'abord ; agir, ensuite.

Cette introduction n'est sans doute pas nécessaire pour le lectorat de notre revue. Elle l'est, hélas, pour un plus large public. Si nous devons penser les évolutions des métiers de l'informatique, ce n'est pas pour créer telle ou telle spécialité nouvelle dans laquelle iront s'enfermer les esprits ; c'est bien plutôt pour rétablir, à la face d'un monde agité et superficiel, les dures exigences du grand Art. Il s'agit de reprendre en main les systèmes d'information, ces créatures qui nous échappent.

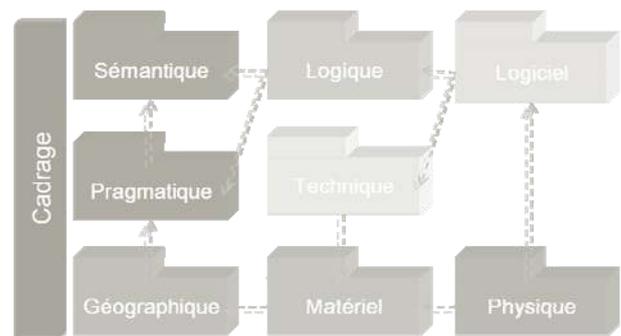


Figure 1: La Topologie du Système Entreprise

Les disciplines de la modélisation

Nous commencerons notre exploration par les disciplines de la modélisation, les plus évidentes dans la diffusion actuelle de la méthode publique Praxeme. Le socle théorique de Praxeme réside, pour l'essentiel, dans la « Topologie du Système Entreprise », déjà présentée dans ces pages. À la question « que faut-il représenter, de l'entreprise, pour tout en dire et agir rationnellement sur elle ? », Praxeme répond en identifiant huit aspects, tous justifiables de la modélisation. Nous nommons ici « modélisation » un effort de représentation qui obéit à certaines exigences formelles. Bien sûr, tout ne peut pas subir ce traitement, c'est pourquoi la

¹ recension : inventaire détaillé et critique (NDLR)

² Cette expression « génie logiciel » semble, d'ailleurs, devenue obsolète. Signe des temps ?

méthode prévoit un espace dit de « cadrage » (ou pré-modélisation) qui accueille les expressions « floues » trouvées dans l'entreprise : vocabulaires, objectifs, exigences...

La cartographie des compétences de modélisation

La Topologie du Système Entreprise fournit, avant tout, l'instrument pour analyser et ordonner la matière

de l'entreprise, du moins sa description : connaissances, représentations, informations et décisions. Elle peut aussi se lire comme une carte des compétences requises pour couvrir toute la chaîne de transformation de l'entreprise, de la stratégie au déploiement. La figure 2, ci-dessous donne une ébauche de cette carte.

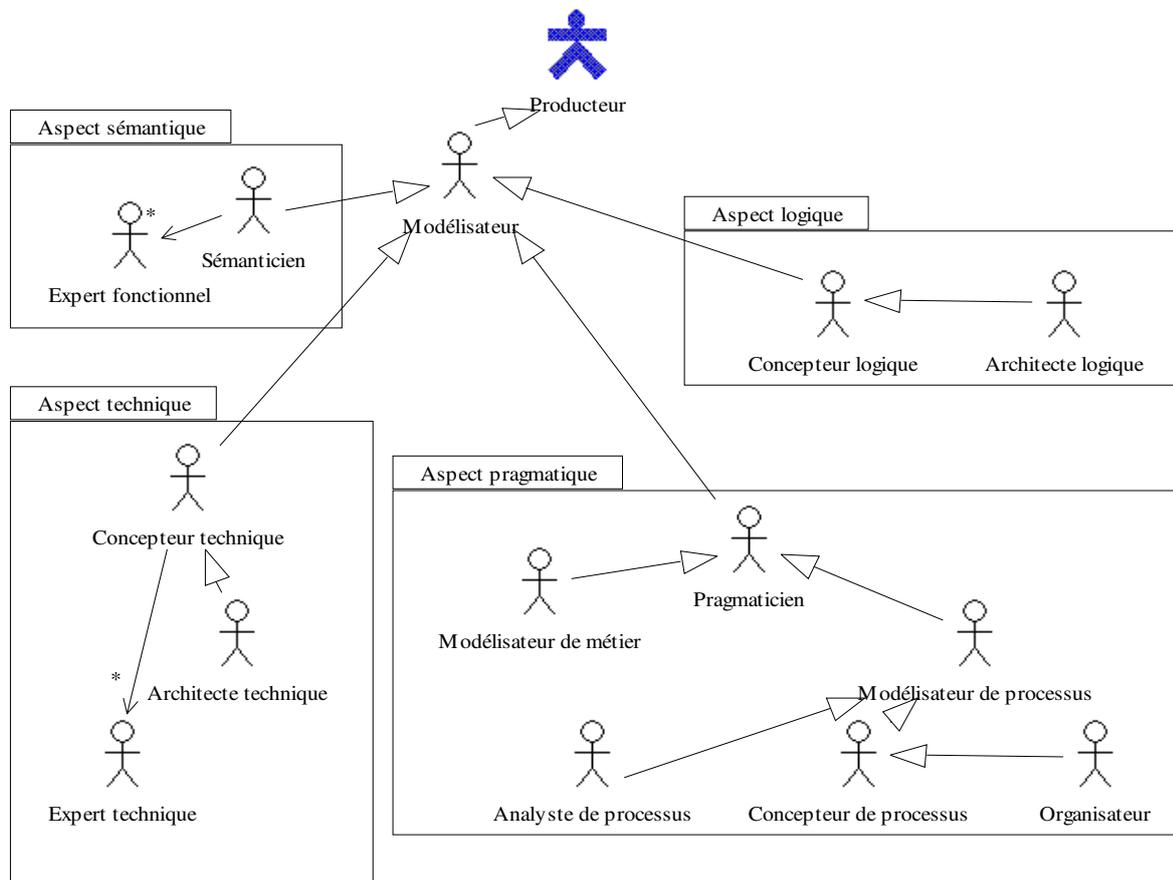


Figure 2: La carte des disciplines impliquées dans la représentation de l'entreprise

Ce schéma se lit de la façon suivante : le modélisateur est considéré comme un « producteur » (nous verrons plus loin la signification de cette racine). Est modélisateur celui qui possède les compétences et comportements d'un producteur à quoi s'ajoute un tronc commun de savoir-faire lié à la discipline générale de la modélisation. Ce métier se spécialise en fonction des aspects.

La modélisation sémantique

Un *modélisateur sémantique*¹ possède les connaissances minimales du modélisateur mais aussi une culture particulière qui le rend sensible à cet aspect. Cette culture se compose de linguistique, d'une orientation pour les sciences cognitives, d'une sensibilité aux problèmes que soulève l'expression rigoureuse des connaissances. S'il est Praxémien, le modélisateur sémantique a, en plus, intériorisé la

logique objet. Il analyse le monde comme un ensemble d'objets (objets « métier » ou objets physiques) plus que comme un récit d'actions. Le modélisateur sémantique interagit avec l'expert « métier ». Ce dernier n'est pas un modélisateur, ce qu'indique son exclusion de l'arbre d'héritage, dans la figure ci-dessus. Toute la difficulté consiste à coordonner le possesseur du contenu – l'expert – et le maître de la forme – le modélisateur. Cette dialectique se retrouve dans les autres aspects. Elle est particulièrement ardue dans la modélisation sémantique, quand il s'agit de capturer la connaissance « métier » fondamentale.

Conception, architecture et modélisation

Le schéma montre, pour les autres aspects, un point important : le *concepteur* est un modélisateur ; il doit donc posséder le bagage minimum qui lui permettra de produire une représentation correcte de sa

¹ Nous avons essayé parfois le néologisme malheureux de « sémanticien ».

solution. L'analyste (non représenté ici) peut tirer profit, également, des techniques de modélisation.

L'architecte, dans cette cartographie, est regardé comme un concepteur (relation d'héritage), ceci pour dire que l'architecte doit posséder la pratique d'un concepteur... et quelque chose en plus. La différence spécifique qui constitue l'architecte doit être analysée. Rappelons simplement que l'architecture est une conception de portée « système » : son effort embrasse la totalité du système, dans un aspect donné, alors que la conception est, le plus souvent, de portée locale (un besoin, une application, un projet...).

La modélisation pragmatique

Il est essentiel de spécialiser les modélisateurs selon les aspects. En effet, chaque aspect exige une approche et une culture bien précises. *L'organisateur et le concepteur de processus*, si nous les voulons créatifs et pertinents, doivent avoir été formés à la réalité de l'organisation. Ils ont plus besoin d'un bagage de sciences humaines que de technologie. Certes, nous observons que la modélisation des processus est souvent confiée à des informaticiens ou à des personnes qui ont une expérience en informatique. La raison en est conjoncturelle et non rationnelle : c'est uniquement parce que les compétences de modélisation, quand elles survivent dans les entreprises, ce sont réfugiées au sein des DSI. Elles y vivent, d'ailleurs, plutôt tapies dans la clandestinité. Cependant, il n'est pire conception de processus que celle faite en pensant à la solution informatique et en ignorant la dimension humaine et collective. Le concepteur et plus encore l'organisateur (le concepteur de l'organisation) puiseront dans le corpus abondant des théories de l'organisation, des études sur le management, la décision, etc. Ces connaissances les aideront à percevoir la relation entre leur description technique et réductionniste, d'un côté, et les hypothèses idéologiques qui les guident : style de management, modèle d'organisation, conditionnement culturel, etc.

La modélisation logique

Sur l'aspect logique – essentiel dans l'amélioration des systèmes informatiques –, nous retrouvons le partage de responsabilité entre *l'architecte logique*, qui veille à la structure du système dans son entier, et le *concepteur logique*, qui produit le détail de la spécification logique (des services, par exemple). Ce partage de responsabilité entraîne un mode de fonctionnement fort différent : autant le concepteur intervient comme à l'habitude en mode projet, autant l'architecte émerge dans la continuité, bien au-delà de l'horizon des projets. Ce changement nécessaire dans les pratiques est un changement majeur pour notre communauté, tellement habituée à travailler en mode projet qu'elle en oublie les autres modes d'organisation des activités. Sans la durée et la continuité de vision, il n'y a pas d'architecture. L'impact de ce changement se lira dans l'organigramme des DSI, avec un rééquilibrage entre les ressources gérées en mode projet et celles dévolues aux activités dites transverses¹.

¹ Ce point est détaillé par exemple dans le support de présentation XD7-08 (Journée exceptionnelle Praxeme 2007, « Praxeme appliquée à SOA », supports disponible sur le site www.praxeme.org).

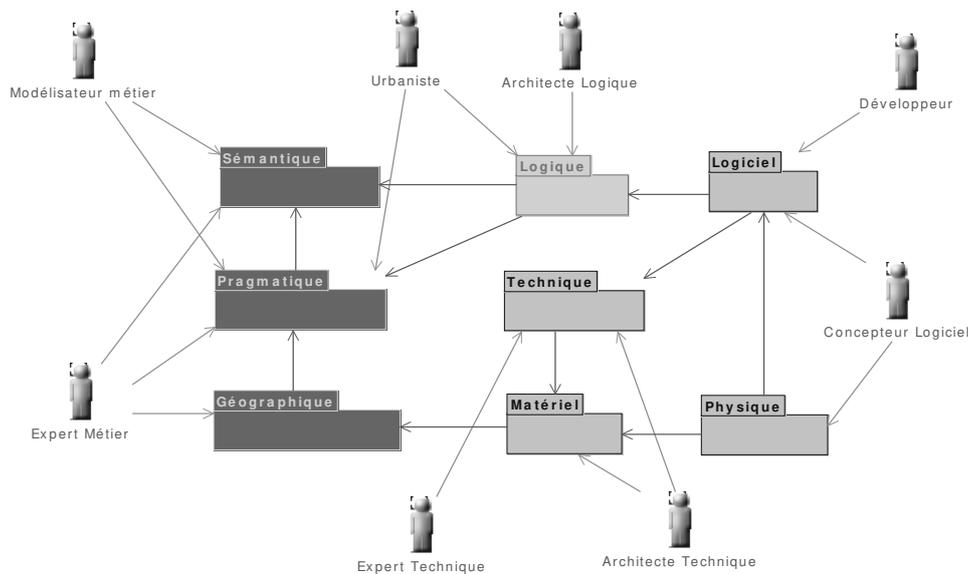


Figure 3: Attribution des responsabilités sur les aspects, selon Philippe Desfray

À propos de l'aspect logique, nous devons évoquer, aussi, le profil de l'urbaniste de SI (ou *Enterprise Architect*). Ce point, légèrement polémique, est abordé plus loin.

Quelques cas particuliers

La figure 2 donne une carte incomplète. Il faudrait mentionner bien d'autres métiers et faire figurer aussi les acteurs du « cadrage », en pré-modélisation. Par exemple, le *stratège* formule les objectifs de l'entreprise et ses décisions se reportent sur les aspects. Des décisions stratégiques se formulent sur l'aspect géographique, notamment...

Concernant les disciplines de la modélisation, nous donnons, ci-après, quelques indications pour montrer les évolutions soit par rapport à la tradition, soit par rapport aux pratiques répandues.

Le tronc commun des compétences en modélisation

Avec des compétences « abstraites » comme la modélisation, il est facile d'entretenir la confusion et de cacher l'incompétence derrière le rideau de fumée de la pédanterie. Se pose alors la question de l'évaluation objective des compétences de modélisation. Cette évaluation n'est pas seulement nécessaire pour l'affectation consciente des ressources aux missions ; elle est aussi une condition pour préserver la modélisation en tant que métier. Les managers étant incapables de juger de ces compétences, le risque est grand d'échouer dans les projets et, ensuite, de dévaloriser la modélisation alors que la cause de l'échec est justement l'insuffisance de la modélisation.

C'est pourquoi, pour faciliter l'appréciation des compétences de modélisation, Praxeme propose deux échelles d'évaluation :

- une pour la maîtrise d'UML (facile à évaluer mais de moindre portée) ;
- une pour la maîtrise de la modélisation.

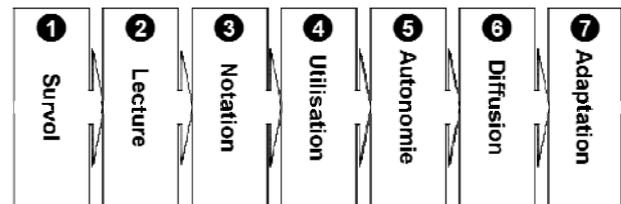


Figure 4: Échelle de la maîtrise d'UML

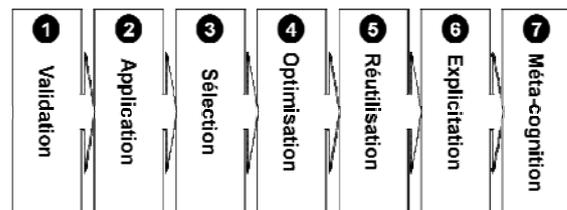


Figure 5: Échelle de la maîtrise de la modélisation

L'analyste et le concepteur

Praxeme rétablit le couple analyse/conception dans sa signification originelle : analyser, c'est décrire ; concevoir, c'est inventer¹. Ces deux postures s'appliquent à tous les aspects du Système Entreprise, parce qu'ils sont tous susceptibles d'une description passive et d'une transformation

¹ Cf. le Livre blanc, SLB-02.

volontariste. S'agit-il de deux métiers distincts ? Dans notre approche, l'analyste et le concepteur, sur un aspect donné, partagent un savoir constitué des catégories et des techniques de représentation propres à cet aspect. Ce qui les distingue, c'est la capacité du concepteur à imaginer une nouvelle solution. D'où ce renversement de statut : le concepteur sait plus et fait plus que l'analyste. Toutefois, l'analyste peut maîtriser des techniques d'enquête et d'évaluation que le concepteur ignore. Prenons l'exemple de l'aspect pragmatique (organisationnel) : l'analyste de processus décrit un processus existant, observe son fonctionnement, détecte ses dysfonctionnements et analyse ses performances. S'il est expérimenté, il peut formuler des recommandations d'amélioration. Le concepteur intervient ensuite pour élaborer une nouvelle solution d'organisation, redessiner les processus, changer les modes de régulation, etc. Bien sûr, un même individu pourra assumer ces deux rôles, sur la base de sa connaissance de l'aspect. Ce type de cumul s'opère plus sûrement que l'accumulation des rôles sur des aspects différents. L'architecte logique ne sera jamais, sérieusement, organisateur – et inversement. Mais, il est tout à fait normal qu'il adopte, sur un système informatique, un point de vue critique, celui de l'analyste. Ceci est même nécessaire avant qu'il se coule dans son rôle d'architecte proprement dit et qu'il élabore une nouvelle solution globale.

En conclusion sur ce point, analyste et concepteur dénotent des *postures* différentes qui s'exercent sur tous les aspects du Système Entreprise. Il y a, certes, des différences de compétences et, peut-être davantage, d'état d'esprit entre ces deux postures, mais pas au point de les dissocier comme des métiers séparés. Mieux vaut, pour l'analyse des compétences, en faire des étapes dans une carrière. Dans ce cas, la pratique de l'analyse doit précéder celle de la conception. Sur un projet et en termes de processus, les deux rôles doivent être distingués et confiés éventuellement à des personnes différentes en fonction de leur expérience ou de leur psychologie. Plus important, le processus de transformation ou de développement doit distinguer l'analyse et la conception comme deux moments. Une bonne illustration en est le dossier des cas d'utilisation (dans Praxeme, la « Vue de l'utilisation » opposée à la « Vue de l'organisation »). La première version de ce dossier, si elle est produite spontanément, apparaît toujours comme un produit d'analyse, capturant des pratiques en cours. L'erreur fréquente est de fonder la conception du logiciel sur ce document d'analyse, avant tout effort de conception pragmatique. À cette première version, il faut substituer une version de conception, laquelle introduit le souci d'une meilleure structuration et des aménagements dans les pratiques.

L'urbaniste et l'architecte

La figure 2 n'a pas positionné le rôle de l'urbaniste de SI, rôle que nous assimilons à celui d'architecte d'entreprise tel que défini dans le courant anglo-saxon *Enterprise Architecture*. Si nous suivons la littérature, l'urbaniste de SI a affaire à la partie « métier » et à la partie informatique. La partie « métier » est réduite aux processus, situés dans l'aspect pragmatique, l'aspect sémantique étant presque totalement omis. Quant à la partie informatique, il s'agit de l'aspect logique puisque l'on en parle par métaphore (l'urbanisation, justement). L'urbaniste semble donc à cheval entre plusieurs aspects. Sans doute n'est-il pas en mesure de réformer une organisation ou d'inventer de nouveaux processus : nous avons vu ci-dessus que cela requiert une véritable spécialisation. Sur l'autre versant, la structuration du système informatique, l'architecte logique possède des outils mieux affûtés et met en œuvre une plus grande rigueur que l'urbaniste. Il s'impose aussi de s'exprimer dans des termes – logiques – qui assurent, d'une part, la continuité avec la conception logique détaillée et, d'autre part, la convertibilité du modèle logique en logiciel.

D'où cette question quelque peu provocatrice : à quoi sert l'urbaniste de SI ?

S'il n'est pas qualifié pour la modélisation pragmatique (organisation, processus) et s'il n'est pas aussi bon que l'architecte logique pour structurer le système informatique, que lui reste-t-il ?

Et bien, dans la nouvelle distribution des rôles en vue d'une maîtrise rationnelle de la transformation du Système Entreprise, l'urbaniste de SI – l'architecte d'entreprise – assume une fonction essentielle : celle de passeur entre le versant « métier » et le versant « informatique ». Nous savons tous, par expérience, combien est difficile la communication entre ces deux mondes. C'est pourquoi la chaîne de transformation a absolument besoin d'une fonction de facilitation, de passage entre ces deux mondes. Sans être spécialiste d'un aspect en particulier, l'architecte d'entreprise va aider à la circulation des préoccupations et décisions entre métier et informatique. Plus précisément, il lui faudra prendre en charge les orientations et décisions stratégiques (cadre) ainsi que la structuration et la thématique des aspects « amont » :

- pragmatique (organisationnel) conformément à TOGAF ou à la littérature française ;
- sémantique, dont la prise en compte changera la physionomie du système informatique ;
- géographique, également, de façon à préparer l'impact de certaines décisions stratégiques sur l'architecture matérielle ;
- logique, enfin, point d'aboutissement de l'urbanisation de SI, avant de passer le relais à l'architecture technique et au développement.

La conception des tests

Dernier point particulier que nous aborderons au titre des évolutions de nos pratiques : les tests. Faut-il en faire un métier, avec l'avantage de professionnaliser cette activité ingrate ? Faut-il laisser les choses en l'état, en faisant confiance au développeur, pour réaliser cette tâche que personne n'apprécie ?

Sur ce thème, la méthodologie Praxeme part du constat suivant : la personne qui appréhende un aspect du système est la mieux placée pour dire comment vérifier cet aspect. La conception des cas de test impliqués par une règle de gestion revient donc au modélisateur sémantique. S'il s'agit d'une règle d'organisation, les cas de test appartiendront au modèle pragmatique. Ainsi de suite sur tous les aspects. En découle un précepte : *tout modèle doit contenir sa propre preuve*. Ce précepte se justifie doublement d'un point de vue économique :

- D'une part, pour la conception des cas de test, on mobilise la bonne compétence : le modélisateur qui connaît la teneur du modèle. On s'épargne ainsi le coût d'acquisition de la connaissance nécessaire à la conception des tests.
- D'autre part, ce précepte renforce l'exigence sur la qualité du modèle : en relisant son travail pour élaborer les cas de test, le modélisateur détecte les incohérences, les imprécisions, etc. et les corrige tout de suite, évitant des dépenses en cascade sur les étapes suivantes.

Ces considérations n'excluent en rien les dispositions habituelles concernant la vérification et la validation, notamment dans le cadre des relations client-fournisseur. Elles les complètent.

Les autres disciplines

La carte des compétences discutées ci-dessus se limitait aux métiers de la modélisation. L'arbre des compétences commençait avec le rôle de « Producteur ». Il est temps de revenir aux fondements.

Le modèle de production

Si nous voulons établir un recensement à peu près exhaustif des compétences nécessaires, notre filet doit être assez large et ses mailles solides. Nous le tissons à partir d'une représentation formalisée de la réalité des projets. Le « modèle de production » décrit la réalité des processus de transformation :

La notion se confond avec celle de compétence ou de discipline. Les quatre types de compétences déduits du modèle sont très génériques. Ce sont les racines de classifications plus pratiques. Aussi, les noms donnés – peu courants, je vous l'accorde – ne constituent pas un problème (voir figure 7). Ces

projets de développement, initiatives stratégiques, programmes de transformation, architecture d'entreprise ou urbanisation de SI, etc. Nous partons du schéma PRO³ qui établit les trois grands chapitres de la méthodologie :

1. « Produit » se tient pour tout objet que nous fabriquons ou sur lequel nous intervenons (application, composant logiciel, système, entreprise, réseau d'entreprises...);
2. « Processus » vaut pour la prescription de la production (description générique de la façon de travailler sur les missions ou les projets de transformation);
3. « Procédés » réserve l'effort de la méthodologie pour fixer le détail des procédures, méthodes, techniques, si l'on veut : les gammes opératoires.

Ces trois chapitres de la méthodologie correspondent à trois grandes catégories de préoccupation dans la vie des projets.

Le diagramme de classes ci-dessous (figure 6) n'est qu'un extrait du modèle de production. Il focalise l'attention sur la réalité des projets. Nous ne rencontrerons donc, par la suite, que des acteurs impliqués dans les projets et activités de transformation. Pour élargir la cartographie des compétences, il conviendra d'étendre le modèle de production considéré. Pour cela, on se reportera au méta-modèle de Praxeme. À titre d'exemple, ce méta-modèle inscrit les notions d'objectif et d'exigence. Il permet d'introduire d'autres classes d'acteurs, comme le stratège ou la maîtrise d'ouvrage.

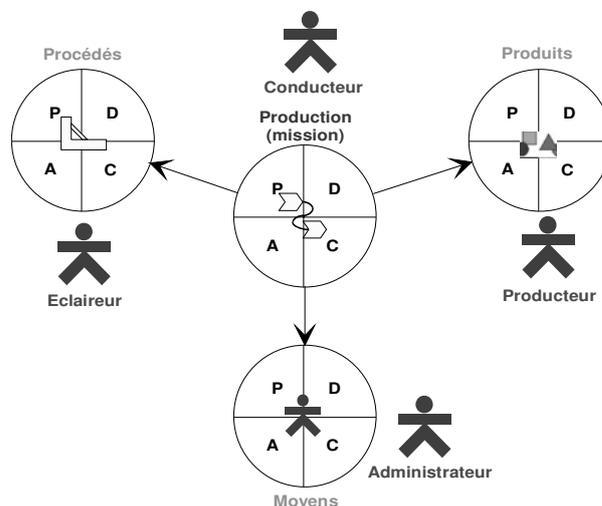


Figure 6 : Le modèle de production (extrait simplifié)

À chacune des composantes du modèle de production, nous associons une responsabilité, un type d'acteur. Ces noms répondent à notre tentative de théorisation des compétences requises pour la transformation du Système Entreprise.

Sur ce schéma, apparaissent quelques-uns des symboles du standard SPEM¹.

La figure suivante illustre le déploiement de la cartographie ainsi que les partages de connaissance qui peuvent s'établir. Par exemple, *l'éclaireur* est celui qui met au point et garantit les procédés sur une mission. Il se spécialise en *ingénieur Méthodes*, *outilleur* (personnalisation de l'outil, etc.), *lanceur*. Ce dernier est un ingénieur méthode ou un ancien conducteur que sa personnalité inquiète rend inapte à la conduite des projets mais particulièrement pertinent lors de la construction des projets : il mène des analyses de risques radicales. C'est un anxieux, doué pour anticiper les problèmes. Il n'est pas nécessairement le meilleur pour vérifier l'application de ces dispositions et suivre le projet au quotidien. Il laissera cette tâche à l'ingénieur méthode ou l'ingénieur qualité. Son anxiété lui interdit l'endurance nécessaire à la course de fond du projet.

Nous observons que le lanceur partage au moins un élément de connaissance avec le conducteur : la discipline de l'ingénierie des projets.

À partir des quatre rôles « racines », les métiers s'analysent en fonction des contextes d'intervention : mode d'organisation des activités, dimensionnement de l'activité, etc. Cette analyse peut – doit – prendre en compte des considérations de nature psychologique.

L'administrateur de projet « hérite » de l'administrateur. Il n'assume pas le rôle de conducteur d'un chef de projet, par exemple ; il ne prend pas de décision. Il suit les moyens. Sa présence permet de décharger le chef de projet des tâches administratives.

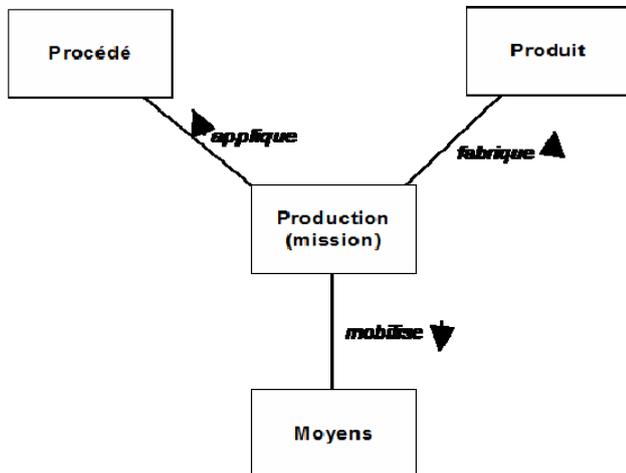


Figure 7 : Les rôles « racines » déduits du modèle de production

¹ Software Process Engineering Metamodel (SPEM) est un standard de l'OMG, établissant le méta-modèle pour la formulation des processus de développement logiciel.

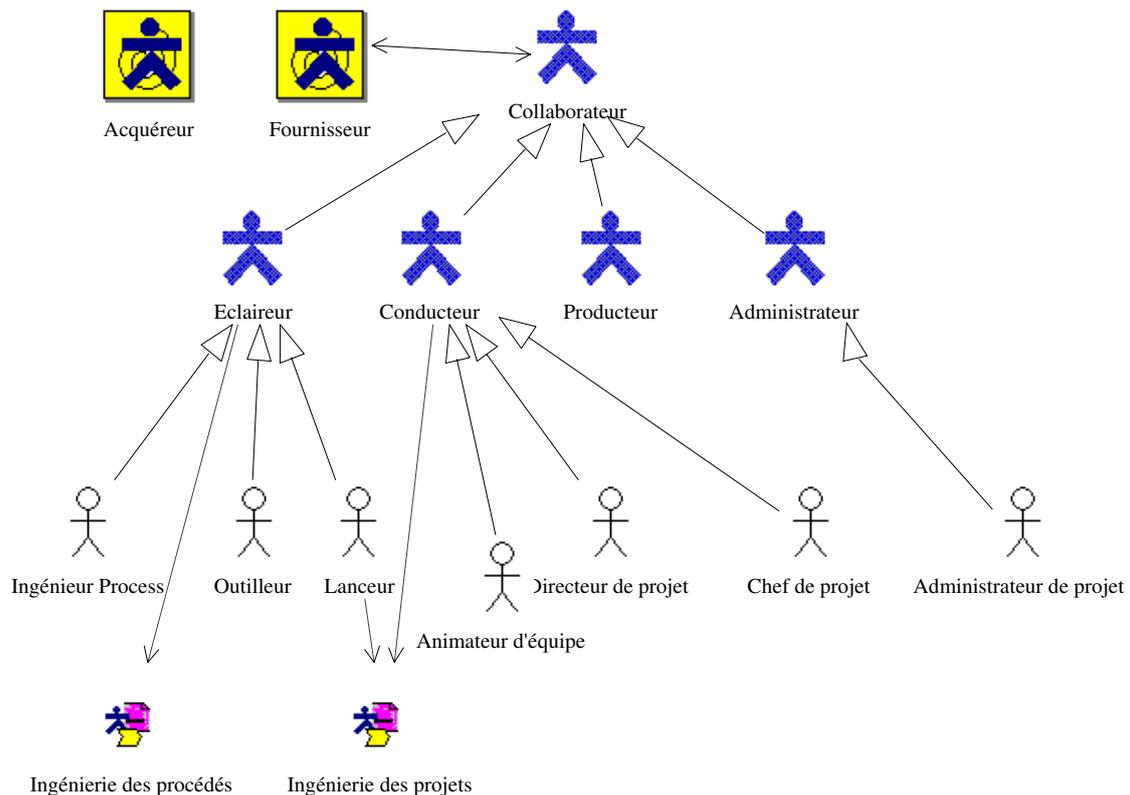


Figure 8 : Les rôles déduits du modèle de production

Conclusion

Ce rapide exposé dresse un tableau des réflexions sur les métiers, actuellement menées dans la communauté Praxeme. Il montre que la cartographie des compétences et la définition des métiers recourent à de nombreuses notions : compétence, discipline, métier, rôle, responsabilité... Nous ne nous sommes pas attardés ici à élucider ces notions. Ce travail est néanmoins nécessaire, dès lors que nous voulons donner au corpus méthodologique une forme rigoureuse et économique. Ces qualités rejailliront sur toute la gestion des compétences. Ce travail est entrepris à travers le méta-modèle de la méthode¹.

De la même façon que la méthode publique s'adresse au marché pour faciliter les travaux de transformation, la mise au point d'une carte des compétences rigoureuse et précise facilitera grandement le développement des compétences et leur allocation.

Les évolutions des métiers sont une affaire trop sérieuse pour être laissée aux praticiens (pour paraphraser Clémenceau). Nous cherchons à asseoir la classification des compétences sur une base théorique. Nos motivations sont :

- mettre en cohérence, d'un côté, la description et la prescription des métiers, de l'autre, la méthodologie, (ce continuum est essentiel pour ensuite donner du contenu aux postes) ;
- tirer les conséquences, en termes de compétences, des ambitions affichées (par exemple, SOA suppose une certaine exigence dans la pratique de l'architecture et de la modélisation ; des conséquences organisationnelles s'ensuivent) ;
- éliminer les étiquettes inutiles et résoudre les chevauchements de responsabilités (ex. entre architecte logique, architecte technique, urbaniste) ;
- alerter sur l'effort nécessaire pour acquérir certaines connaissances.

En filigrane se reconnaît le souci d'articuler les expertises, pour éviter les déperditions d'énergie et d'opportunités.

Il y a donc des évolutions auxquelles nous devons nous préparer, à la fois individuellement et collectivement. Pour autant, tout n'est pas nouveau sous le soleil et nous devons absolument préserver les disciplines qui ont atteint, par le passé, un bon niveau de maturité et qui, parfois, sombrent dans l'oubli parce qu'elles ne sont pas connectées au goût du jour. ▲

¹ Un projet est en cours, financé par l'armée de terre, pour élaborer le processus de développement qui enrichira le fonds public Praxeme. Ce projet fournit l'opportunité de travailler sur le méta-modèle de Praxeme, contribution des Caisses d'allocations familiales.

Bibliographie

La Lettre d'ADELI, n° 69, dossier sur la modélisation

Sur le site du Praxeme Institute (<http://www.praxeme.org>), voir notamment les supports de la formation « Journée exceptionnelle Praxeme pour SOA »
Présentation commentée « La définition des compétences selon Praxeme », réf. CND-02 (Collège des conducteurs, avril 2008)

Ouvrages :

« Le système d'information durable », Pierre Bonnet, Jean-Michel Detavernier, Dominique Vauquier, 2007, Hermes.

« Le Plan qualité du logiciel et des services Internet », Dominique Vauquier, 2003, AFNOR.

« Développement orienté objets, Principes, processus, procédés », Dominique Vauquier, 1993, Eyrolles.

« Les arbres de connaissance », Michel Authier, Pierre Lévy, 1992, Éditions La Découverte.