

Modélisation des prises de décisions dans les processus métier grâce à DMN (Decision Model and Notation)

THIERRY BIARD¹, JEAN-PIERRE BOUREY², MICHEL BIGAND², JEAN-CLAUDE BOCQUET¹

¹ LABORATOIRE GENIE INDUSTRIEL - CENTRALESUPELEC - UNIVERSITE PARIS-SACLAY
Grande Voie des Vignes, 92290 Châtenay-Malabry, France
thierry.biard@centralesupelec.fr, jean-claude.bocquet@centralesupelec.fr

² ÉCOLE CENTRALE DE LILLE - UNIVERSITE LILLE NORD DE FRANCE
Cité Scientifique - CS 20048, 59651 Villeneuve d'Ascq Cedex, France
jean-pierre.bourey@centralesupelec.fr, michel.bigand@centralesupelec.fr

Résumé - L'Architecture d'Entreprise est la discipline pour mener à bien les projets de transformation de l'Entreprise et notamment pour aligner son Système d'Information sur sa stratégie. L'Architecture d'Entreprise s'appuie sur une méthode, avec un cadre de représentation pour organiser ses modèles. La méthode Praxeme est détaillée, avec son cadre de représentation contenant 4 aspects d'architecture métier et 3 aspects d'architecture technique. La modélisation des processus métier de l'Entreprise fait généralement appel au standard BPMN (Business Process Model and Notation), mais cette notation se révèle limitée pour représenter des prises de décisions opérationnelles. En appliquant le principe de séparation des préoccupations et en utilisant le nouveau standard DMN (Decision Model and Notation), qui propose un diagramme spécifique pour représenter les prises de décision et des tables de décision pour formaliser les règles métier, cet article démontre à partir d'un exemple réel qu'il est possible de concevoir un modèle mieux adapté à la situation. Enfin, une démarche méthodologique est proposée pour positionner DMN dans la méthode Praxeme et pour traiter les règles métier. La conclusion s'intéresse à la pertinence de ce nouveau standard DMN et ouvre des perspectives d'enrichissement de la méthode Praxeme et d'automatisation des prises de décisions opérationnelles.

Abstract - Enterprise Architecture is the discipline to drive the transformation projects of the Enterprise and especially to align its Information System on its strategy. Enterprise Architecture relies on a method, with a representation framework for organizing its models. The Praxeme method is detailed, with its representation framework containing 4 aspects of business architecture and 3 aspects of technical architecture. The Enterprise business process modeling usually calls out BPMN (Business Process Model and Notation) standard, but this notation reveals itself limited to represent operational decision-making. When applying the separation of concerns as a principle and using the new DMN (Decision Model and Notation) standard, which offers a specific diagram to represent decision-making and decision tables for business rules formalization, this article demonstrates from a real example that it is possible to design a better model adapted to this situation. Finally, a methodological approach is proposed for positioning DMN into the Praxeme method and for treating business rules. The conclusion focuses on the relevance of this new DMN standard and opens perspectives for Praxeme method enrichment and operational decision-making automation.

Mots clés - prise de décision, modèle, notation, processus, architecture d'entreprise.

Keywords - decision-making, model, notation, process, enterprise architecture.

1 INTRODUCTION

Modéliser l'Entreprise dans sa globalité, afin d'explicitier son fonctionnement complexe, c'est un des buts de l'Architecture d'Entreprise. Il convient de proposer des points de vue différents, mais cohérents, de l'Entreprise, adaptés à chacun, qu'il soit spécialiste métier ou informaticien.

La méthode Praxeme propose une nouvelle approche de l'Architecture d'Entreprise. Elle propose un cadre de représentation qui accueille déjà des diagrammes UML (Unified Modeling Language), mais également des diagrammes BPMN (Business Process Model and Notation) pour la modélisation des processus métier. Encore faut-il que ces modèles soient compris par tous.

Car les diagrammes BPMN se révèlent parfois peu adaptés pour modéliser les prises de décisions opérationnelles et les règles dans les processus métier. En appliquant le principe de séparation des préoccupations, cet article vise à démontrer la pertinence (ou pas) de la nouvelle notation standard DMN (Decision Model and Notation) pour modéliser ces prises de décisions et ces règles métier différemment.

Après la présentation de l'Architecture d'Entreprise et de quelques cadres de représentation, dont celui de la méthode Praxeme, la nouvelle notation DMN sera introduite, afin de répondre aux besoins précis de modélisation des prises de décision opérationnelles et des règles métier, puis elle sera mise en pratique sur un cas réel.

2 ARCHITECTURE D'ENTREPRISE

2.1 Contexte de la Transformation

L'Architecture d'Entreprise est une discipline qui permet de concevoir l'Entreprise dans tous ses aspects, du métier à la technique. Elle permet notamment d'aligner son Système d'Information sur sa stratégie.

Dans un contexte économique difficile et pour s'adapter à l'ère numérique, l'Entreprise doit se transformer pour survivre. L'Architecture d'Entreprise est la discipline adéquate pour mener à bien son projet de transformation (en complément de la conduite du changement et du pilotage).

L'Entreprise moderne, qui veut être agile pour pouvoir se transformer constamment, a besoin d'un cadre méthodologique pour maîtriser sa complexité puis gouverner sa transformation.

2.2 Différents types d'Architectures alignées

L'Architecture d'Entreprise est composée de l'Architecture Technique de son Système d'Information (ses applications, ses serveurs, son réseau, etc.), mais surtout de son Architecture Métier (ses produits et ses services, son organisation, ses processus, ses règles, ses décisions, etc.). Idéalement, l'Architecture Technique doit être parfaitement alignée sur l'Architecture Métier, elle-même préalablement alignée sur la Stratégie de l'Entreprise [Henderson et Venkatraman, 1992].

2.3 Méthode, Cadre de Représentation et Modèles

L'Architecture d'Entreprise doit s'appuyer sur une méthode, un cadre de représentation, des modèles.

L'Architecture d'Entreprise s'appuie sur un cadre de représentation (*framework*) qui est constitué d'un socle de référence théorique, sur lequel une méthode plus ou moins aboutie peut être élaborée. Un cadre de représentation permet d'organiser la perception de l'Entreprise.

Selon les bonnes pratiques, un cadre idéal devrait respecter certains principes comme : la focalisation sur l'Entreprise ; la prise en compte de toute sa réalité ; l'unicité de tout élément à l'intérieur du cadre (et son corollaire, le bannissement de la redondance).

Toute démarche d'Architecture s'appuie également sur la modélisation de l'Entreprise et son lot de principes : la cohérence des modèles ; la mise en relation des éléments les uns avec les autres ; l'aide à la transformation d'un modèle vers un autre et enfin l'existence d'un métamodèle de référence.

2.4 Principaux Cadres de Représentation

2.4.1 Des approches différentes

Les cadres de représentation d'Architecture d'Entreprise les plus courants, comme Zachman et TOGAF, d'origine américaine, se révèlent bien souvent inadaptés à la réalité du terrain, lourds à mettre en œuvre et parfois peu cohérents.

Ces cadres sont en outre avares en recommandations concernant la modélisation, qui ne semble pas être leur préoccupation principale.

Mettant en pratique le concept de l'IDM - Ingénierie Dirigée par les Modèles - [Jézéquel et al., 2012] popularisée par la MDA – *Model Driven Architecture* [OMG, 2014], la méthode Praxeme, d'origine française, propose une alternative émergente.

Il ne s'agit pas de concurrencer les cadres de représentation précédents : la méthode Praxeme pourrait compléter les cadres de Zachman et TOGAF pour leur donner plus de consistances.

2.4.2 Cadre de Zachman

Le cadre de Zachman [Zachman, 2011], qui a le grand mérite d'avoir été le premier (1987), pose le problème de l'Architecture d'Entreprise avec une série de 6 questions : Quoi ? Comment ? Où ? Qui ? Quand ? Pourquoi ? Puis il tente d'y répondre selon 6 points de vue différents. Sa représentation matricielle donne l'illusion d'une approche cartésienne et rassurante. Mais au final, il propose donc 36 aspects différents (version 2011) de l'Entreprise, ce qui semble bien compliqué. Et la séparation des données (Quoi) et des fonctions (Comment) n'est plus adaptée à une approche moderne Orientée Objet.

2.4.3 Cadre TOGAF

Le cadre TOGAF - *The Open Group Architecture Framework* [Open Group, 2011], dont il faut constater la diffusion récente dans les grandes entreprises, est constitué d'une longue liste (de type taxonomique) d'activités à réaliser par les architectes, en respectant certains principes, que d'un véritable cadre de représentation pour l'Architecture d'Entreprise.

Il faudrait en effet que TOGAF se focalise plus sur l'Entreprise et propose des points de vue différents de celui des architectes. En outre, TOGAF s'intéresse peu aux modèles et encore moins à leur transformation. Cette lacune est toutefois comblée par un outil comme Modelio [Desfray et Raymond, 2014].

Finalement, la Figure 1 emblématique de TOGAF est l'ADM (Architecture Development Method), qui est en fait le processus d'Architecture, découpé en plusieurs phases :

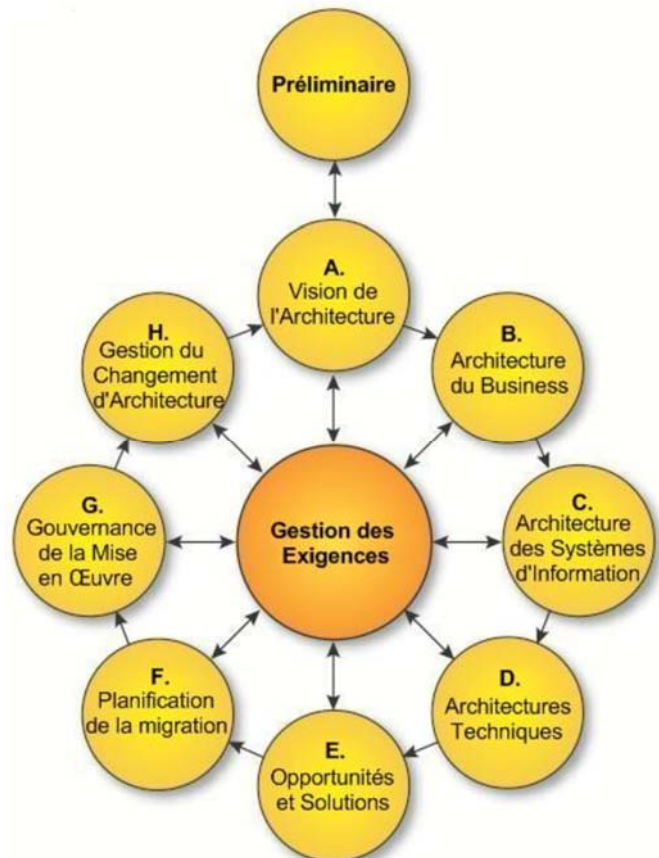


Figure 1: Architecture Development Method (TOGAF)

2.4.4 Méthode Praxeme

Selon son étymologie, Praxeme signifie « le sens de l'action » [Praxeme Institute, 2006]. Praxeme, tout en tenant compte des évolutions technologiques, s'inscrit depuis 2003 dans l'héritage de la méthode de référence Merise [Tardieu et al., 2000].

La méthode Praxeme est supposée complète, rigoureuse et très bien adaptée à l'Architecture Orientée Services – SOA [Caseau, 2011]. Elle a été utilisée avec succès pour la refonte du système d'information d'un assureur [Bonnet et al., 2007], l'urbanisation de stations de contrôle de drones et la modélisation de systèmes de transport, dont la signalisation.

Praxeme est une méthode publique et ouverte dont le développement, la promotion et la formation sont assurés, depuis 2006, par une association à but non lucratif, reconnue d'utilité publique, le Praxeme Institute, qui met son fonds documentaire librement à la disposition du public.

La période du financement public d'une méthode semble révolue. Toutefois, la Direction Générale de la Modernisation de l'État recommandait en 2009 la méthode Praxeme dans la version 1.0 de son Référentiel Général d'Interopérabilité.

3 TOPOLOGIE DU SYSTEME ENTREPRISE SELON PRAXEME

3.1 Les 7 Aspects de Représentation de l'Entreprise

La méthode Praxeme s'intéresse à l'Entreprise, considérée comme un système complexe, dans sa globalité (approche holistique). La méthode Praxeme, qui considère l'Entreprise comme un tout, est basée sur 7 aspects. Certains aspects peuvent être représentés par des modèles au standard UML. Une représentation structurée de ses 7 aspects est donnée dans la Topologie du Système Entreprise, fondement de la méthode Praxeme. Ces aspects sont similaires aux « niveaux d'abstraction » de Merise et au principe de séparation des préoccupations préconisé dans la MDA – Model Driven Architecture.

Sur la Figure 2 ci-dessous, les flèches de couleur grise symbolisent les liens de dépendance entre ces aspects.

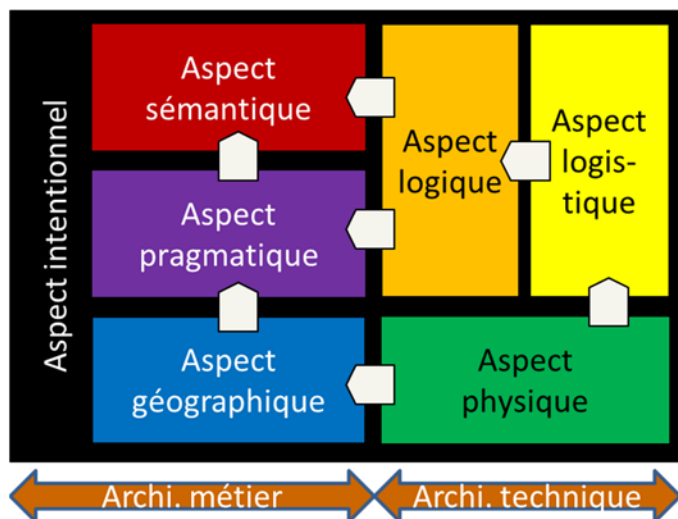


Figure 2: Topologie du Système Entreprise (Praxeme)

Ces 7 aspects de la méthode sont conformes au métamodèle Praxeme et semblent bien articulés les uns avec les autres. Des règles précises de dérivation entre ses différents aspects apportent à la méthode Praxeme de la rigueur dans la traçabilité.

3.2 Les 4 Aspects de l'Architecture Métier

Les aspects Intentionnel, Sémantique, Pragmatique et Géographique constituent l'Architecture Métier de l'Entreprise. Tout commence par l'omniprésent aspect Intentionnel. Puis, les aspects Sémantique et Pragmatique servent de base aux autres aspects.

L'aspect Intentionnel formalisera les intentions, la stratégie et les objectifs de l'Entreprise, voire son éthique et ses valeurs. Les indicateurs clefs seront regroupés dans cet aspect ; il est alors possible de les organiser en un ensemble cohérent, appelé Arbre de Performance.

L'aspect Sémantique est basé sur la modélisation des objets métier. Il convient souvent d'extraire la sémantique des processus métier, décrits dans l'aspect Pragmatique. Le paradoxe de la modélisation des objets métier, souvent appelés Fondamentaux ou Cœur du métier, est le suivant : c'est le travail le plus difficile à faire, mais c'est aussi le travail le plus durable, car indépendant de l'organisation.

L'aspect Pragmatique est basé sur la modélisation des processus métier et l'organisation nécessaire pour leur exécution.

La séparation des aspects Sémantique et Pragmatique est la spécificité principale de la méthode Praxeme. Cette séparation originelle, qui peut se révéler complexe à effectuer, est un gage de flexibilité pour faciliter les transformations à venir. Il est évident que ces deux aspects sont indépendants de toutes technologies - dont les cycles de vie sont bien plus courts - qui seront utilisées pour les mettre en œuvre.

Dans ces deux aspects Sémantique et Pragmatique, figurent les définitions des Règles Métier qui formalisent les différentes contraintes, exigences, mais aussi la connaissance et le savoir-faire de l'Entreprise. Si nécessaire, l'organisation pourra être modifiée, les processus métier reconfigurés et l'Entreprise transformée.

L'aspect Géographique est positionné du côté de l'Architecture métier, car il dépend bien souvent de l'histoire, la culture et l'organisation de l'Entreprise. Il s'agit d'envisager toutes les options de déploiement, puis de définir la localisation des activités, selon la mobilité du personnel et les moyens de communication.

3.3 Les 3 Aspects de l'Architecture Technique

Les aspects Logique, Logistique, et Physique constituent l'Architecture Technique de l'Entreprise.

L'aspect Logique est le plus important : il est l'intermédiaire entre l'Architecture Métier (ses aspects Sémantique et Pragmatique) et l'Architecture Technique ; il assure le découplage entre la solution conçue et les technologies utilisées pour la mettre en œuvre (déployées dans les aspects Logistique - logiciel & matériel informatique - et Physique), ce qui garantit sa pérennité. La discipline de l'urbanisation devra être appliquée dans l'aspect Logique, pour structurer le Système d'Information et le préparer à l'Architecture Orientée Services (SOA).

L'aspect Logistique décrit la solution logicielle et matérielle (technologies, composants, équipements). Grâce au découplage apporté par l'aspect Logique, les choix techniques de l'aspect Logistique peuvent être substituables.

L'aspect Physique décrit avec précision la réalité du système choisi dans l'aspect Logistique puis déployé selon l'aspect Géographique (instanciation et localisation).

4 MODELISATION D'ENTREPRISE

4.1 Différents Besoins de Modélisation d'Entreprise

Idéalement, chaque aspect de l'Architecture d'Entreprise devrait être représenté par un modèle, de préférence graphique. C'est le langage UML, un standard de l'OMG (Object Management Group), qui a apporté la plus forte contribution à la modélisation d'entreprise.

Parmi la quinzaine de diagrammes proposés par la version 2.5 d'UML, citons les plus utilisés pour l'architecture métier, selon Praxeme : le diagramme de Cas d'Utilisation pour l'aspect Pragmatique, le diagramme de Classes et de diagramme d'Etats-Transitions pour l'aspect Sémantique. D'autres diagrammes UML sont également utilisés pour l'architecture technique, notamment pour le génie logiciel et l'infrastructure.

Dans cet article, nous nous intéressons à la modélisation des processus métier et, après avoir séparé les préoccupations, des décisions opérationnelles et des règles métier.

4.2 Modélisation des Processus Métier

4.2.1 BPMN (Business Process Model and Notation)

Il y a une dizaine d'années, c'était le diagramme d'Activités UML qui était utilisé pour modéliser les processus métier. Depuis lors, le diagramme de Collaboration BPMN a largement pris le dessus, grâce à des possibilités de représentation plus larges des processus (plus de choix de passerelles par exemple).

La version 1.0 de la spécification BPMN (Business Process Model and Notation) a été publiée en 2003. Plusieurs versions se sont succédé ; la dernière version 2.0.2 [OMG, 2016], en plus d'être un standard de l'OMG, a également été publiée comme norme ISO/IEC 19510 en 2013.

A noter que BPMN propose également des diagrammes de Conversation et de Chorégraphie qui sont peu utilisés (et qui ne sont pas supportés par la plupart des outils de modélisation).

4.2.2 Passerelles BPMN en Cascade

Le diagramme de collaboration BPMN sert à représenter des séquences d'activités et les interactions entre les différents acteurs, représentées sous la forme de messages. Le diagramme de collaboration est supposé être suffisamment simple pour être compris par tous, y compris les parties prenantes du métier.

Pour emprunter des chemins différents selon la valeur de certains critères, c'est l'élément Passerelle (*Gateway*) qui est utilisé, le plus souvent de type OU exclusif (XOR). Dès qu'il y a beaucoup de critères, les passerelles en cascade nuisent à la clarté de la représentation (qui ressemble alors à un arbre de décision), comme nous le montrerons dans l'exemple qui suit.

De plus, cette représentation traditionnelle des décisions par des passerelles au cœur même des processus métier constitue un couplage fort entre ces deux types d'éléments de natures différentes. L'ajout d'un critère de décision supplémentaire peut avoir un fort impact sur le modèle de processus et sur son implémentation technique, si son exécution est automatisée.

Une représentation alternative serait d'utiliser une passerelle complexe, qui doit alors obligatoirement être annotée par un texte explicatif, qui est verbeux par définition et va à l'encontre du souhait initial d'obtenir une représentation graphique.

4.2.3 Exemple de Modélisation BPMN sans DMN

Au sein d'un processus plus global de recrutement dans un établissement public d'enseignement supérieur, il s'agit dans cet exemple de déterminer si un vacataire est recevable ou pas, avant de le recruter en continuant le processus. Voici l'extrait du diagramme BPMN de collaboration qui nous concerne :

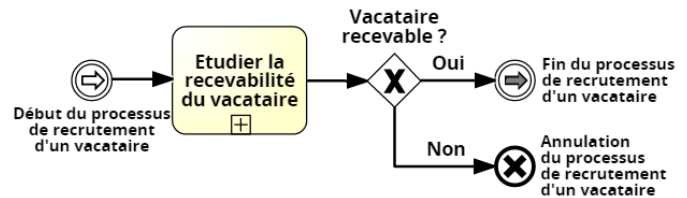


Figure 3: Extrait du processus de recrutement (BPMN) avec sous-processus replié "Etudier la recevabilité du vacataire"

Voici le sous-processus déplié « Etudier la recevabilité du vacataire », qui représente scrupuleusement les différents critères de recevabilité selon les Articles 2 & 3 du « Décret n°87-889 du 29 octobre 1987 relatif aux conditions de recrutement et d'emploi de vacataires pour l'enseignement supérieur ». A noter que l'Article 5 de ce décret fixe un nombre maximum d'heures de vacations, non représenté ci-dessous.

Puis appliquons le principe de séparation des préoccupations.

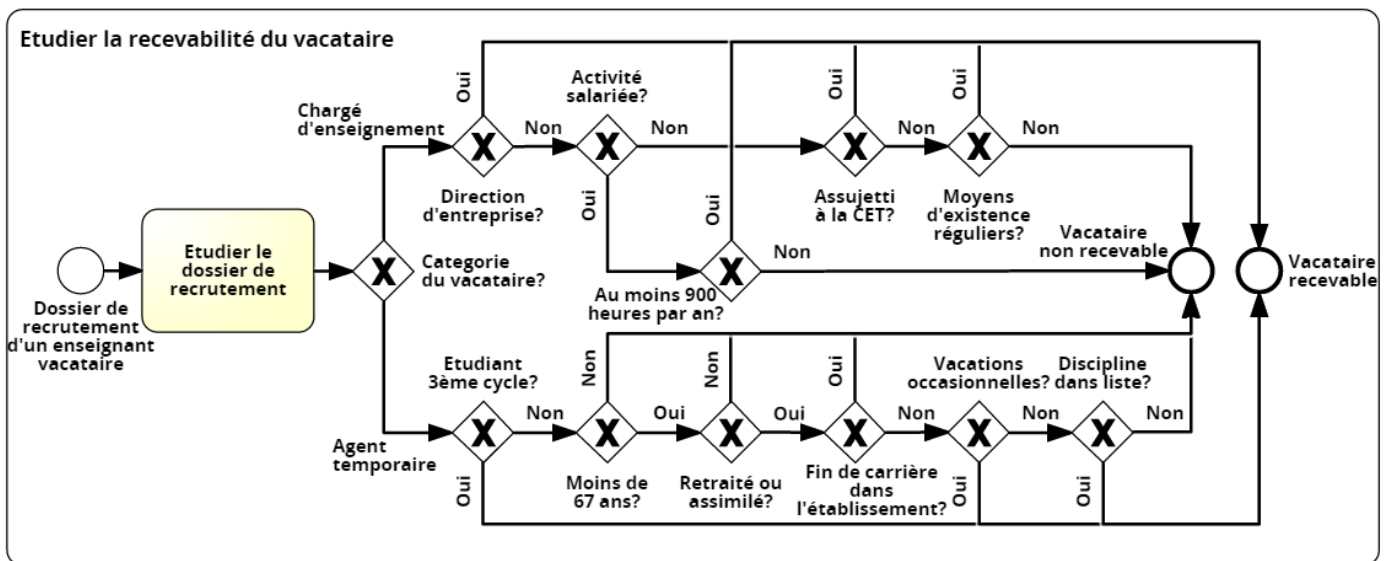


Figure 4: Sous-processus déplié "Etudier la recevabilité du vacataire" (BPMN)

4.3 Séparation des Préoccupations

4.3.1 Architecture Dirigée par les Modèles

La Séparation des Préoccupations (*Separation of Concerns*) est un bon principe provenant de l'informatique. Il date de 1974 [Dijkstra, 1974], mais s'agissant d'un principe indépendant de toute technologie, son application est toujours préconisée.

Dans le contexte qui nous intéresse, la Séparation des Préoccupations est notamment appliquée par l'Architecture Dirigée par les Modèles (*MDA : Model-Driven Architecture* [OMG, 2014]) pour concevoir un grand système, complexe par nature. Les préoccupations seront séparées, afin de réduire sa complexité. Chaque préoccupation sera traitée par un aspect du système, aussi indépendant que possible des autres aspects.

4.3.2 Evolution d'Architecture des Applications Métier

Ainsi, nous observons l'influence de la Séparation des Préoccupations sur l'évolution d'architecture des applications métier au fil des ans. Dans la Figure 5 ci-dessous, le symbole des Règles représente aussi les prises de décision :

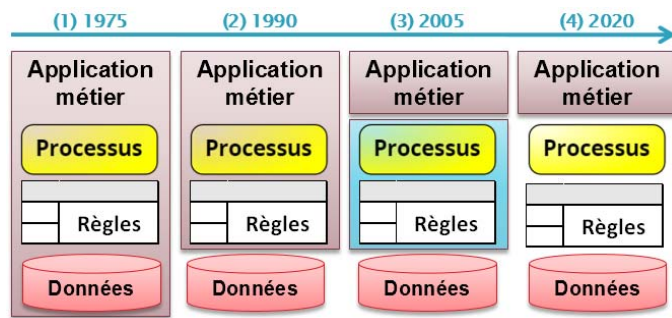


Figure 5: Evolution d'architecture des applications métier

Nous distinguons 4 grandes étapes dans l'évolution des applications métier (les dates sont estimées par les auteurs) :

- (1) Les premières applications monolithiques,
- (2) L'externalisation massive des données dans des SGBD (Systèmes de Gestion de Bases de Données),
- (3) L'extraction des processus et des règles métier, dans des suites BPM (Business Process Management) (les règles restant alors imbriquées dans les processus),
- (4) L'extraction des décisions et règles métier des processus.

Tandis que c'est le langage SQL qui a permis aux applications d'accéder aux SGBD, c'est l'Architecture Orientée Services (SOA) qui a permis aux processus d'orchestrer plusieurs applications métier et ensuite les moteurs de règles.

Les moteurs de règles existent depuis une quinzaine d'années déjà et il est possible d'automatiser les prises de décisions dans les processus métier [Fish, 2012]. Par contre, la modélisation standardisée des prises de décisions (objet principal de cet article) est récente et sera développée dans le chapitre suivant.

4.3.3 Application de la Séparation des Préoccupations aux Processus Métier et aux Prises de Décisions

Lorsqu'on applique ce principe à notre contexte, il s'agira alors d'extraire les prises de décision des modèles de processus métier, afin de modéliser leur structure séparément. Les règles métier elles-mêmes seront également modélisées à part.

L'avantage de la séparation de ces modèles faiblement couplés est leur relative indépendance facilitant les changements. Chaque modèle, autonome et consistant, peut évoluer séparément : c'est un gage d'agilité, voire de résilience.

Pour comprendre l'un des modèles, il n'y a pas besoin de connaître l'autre. Aussi, chaque modèle peut s'adresser à un public différent, comme celui des Analystes métier ou celui des Informaticiens.

4.4 Modélisation des Prises de Décisions

4.4.1 Principaux Types de Décisions

Le Tableau 1 présente les principales différences entre les décisions stratégiques-tactiques et les décisions opérationnelles. Dans un environnement prédéfini, il est possible d'associer les décisions opérationnelles aux processus métier :

Type de décision	Stratégique-Tactique	Opérationnelle
Environnement	Incertain	Prédéfini
Portée	Globale	Locale
Impact	A long-moyen terme	A court terme
Orienté processus	Faible	Fort
Prise de décision	Humaine avec Système d'Aide à la Décision	Automatisable
Fréquence	Petite-Moyenne	Grande
Période	Années-Mois	Secondes (Temps Réel)
Exemple de décision	Recrutement de vacataires	Recevabilité d'un vacataire

Tableau 1: Principaux types de décision

4.4.2 Proposition du Standard DMN par l'OMG

La première initiative marquante (d'une société privée) pour la modélisation des décisions avec une orientation métier fut la Decision Model [Von Halle et Goldberg, 2010]. C'est l'application des règles métier qui sert généralement à prendre des décisions.

En février 2014, l'OMG a publié un nouveau standard en version 1.0 bêta : DMN (Decision Model and Notation) [OMG, 2016], dédié à la modélisation des décisions.

La version 1.0 finale, qui n'était pas encore stable, a été publiée en septembre 2015, tandis que la dernière version 1.1 a été publiée en juin 2016. L'OMG ne s'est pas contenté de rédiger une spécification de 182 pages : un métamodèle, à l'attention des éditeurs de logiciel - et des chercheurs - est également fourni. Chaque standard de l'OMG est fourni avec un métamodèle sous forme de diagramme de classes UML.

4.4.3 Bouquet de Standards de l'OMG

Ce nouveau standard DMN n'est pas isolé. Au contraire, il s'appuie sur un bouquet de standards de l'OMG [OMG, 2016]:

- (1) SBVR (Semantics of Business Vocabulary and Business Rules) pour définir le vocabulaire utilisé dans les règles métier,
- (2) BMM (Business Motivation Model) pour définir les objectifs soutenus par les décisions,
- (3) BPMN (Business Process Model and Notation) pour associer les prises de décisions aux processus.

La relation entre SBVR et DMN n'est pas formalisée : c'est une simple recommandation [Linehan et de Sainte Marie, 2011].

L'extrait du diagramme de classes de la Décision (Figure 8 page suivante ; en haut à gauche) formalise, les relations avec BMM (une *Decision* peut soutenir un *Objective*) et avec BPMN (une *Decision* peut être rattachée à un *Process* et à une *Task*).

L'interface entre DMN et BPMN semble particulièrement intéressante [Debevoise et Taylor, 2014]. Les processus métier et les prises de décision peuvent désormais être modélisés séparément, avec leurs propres notations.

4.4.4 Présentation de DMN (Decision Model and Notation)

Graphiquement, il n'y a que 7 éléments de représentation dans DMN (en comparaison, BPMN propose près d'une centaine d'éléments graphiques). La Figure 6, appelée *Decision Requirements Diagram (DRD)* utilise tous ces éléments :

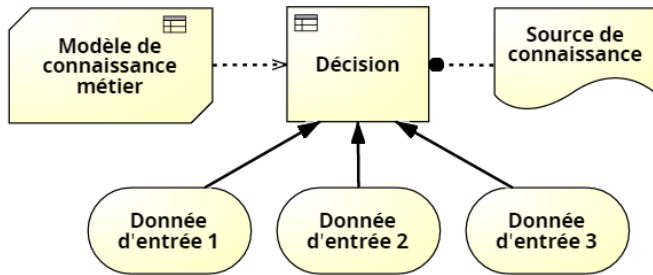


Figure 6: Tous les éléments graphiques de DMN dans un *Decision Requirements Diagram (DRD)*

Deux éléments sont obligatoires : les Données d'entrée (*InputData* ; il y en a généralement plusieurs) et la Décision (*Decision*), qui applique une logique sur ces Données d'entrée. Une Décision peut être décomposée en plusieurs sous-décisions.

Deux éléments sont optionnels : on ajoute souvent au moins une Source de connaissance (*KnowledgeSource*), qui représente une autorité, voire plus rarement un Modèle de connaissance métier (*BusinessKnowledgeModel*), qui représente une logique de décision réutilisable (l'équivalent de la *Call Activity* de BPMN).

Trois types de liens différents, les *Requirements*, permettent de représenter les relations entre ces quatre éléments. Ces *Requirements* sont explicitement représentés par des classes dans le métamodèle (Figure 8), ce qui est assez rare.

Les outils supportant DMN utilisent automatiquement le type de lien adéquat conformément au métamodèle. Une quinzaine d'outils supportent déjà DMN [OpenRules, 2016].

4.4.5 Principe d'Association entre BPMN et DMN

Au lieu de représenter un processus métier par un diagramme de collaboration BPMN, avec de nombreuses passerelles souvent imbriquées en cascade, il suffit désormais de les remplacer par une seule tâche de type décision métier (reconnaisable par la petite table de décision en haut à gauche), nommée par exemple « Prendre une décision » sur la Figure 7, qui renvoie un résultat qu'il suffit alors de tester par une passerelle.

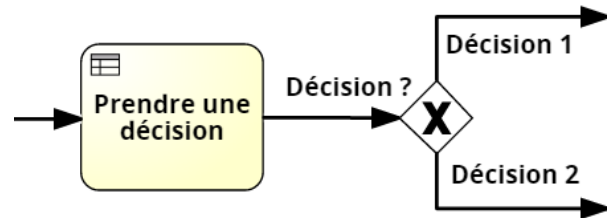


Figure 7: Tâche de type règle métier, suivie d'une passerelle exclusive (BPMN)

La passerelle qui suit une prise de décision n'est pas systématique : imaginons une tâche « Décider du pourcentage de remise », selon l'ancienneté du client et le montant de sa commande ; il suffit ensuite d'appliquer la remise qui a été décidée (cette remise peut être nulle).

Outre la simplification des diagrammes, il s'agit surtout d'un bel exemple de séparation des préoccupations, celles des analystes métiers et celles des informaticiens notamment.

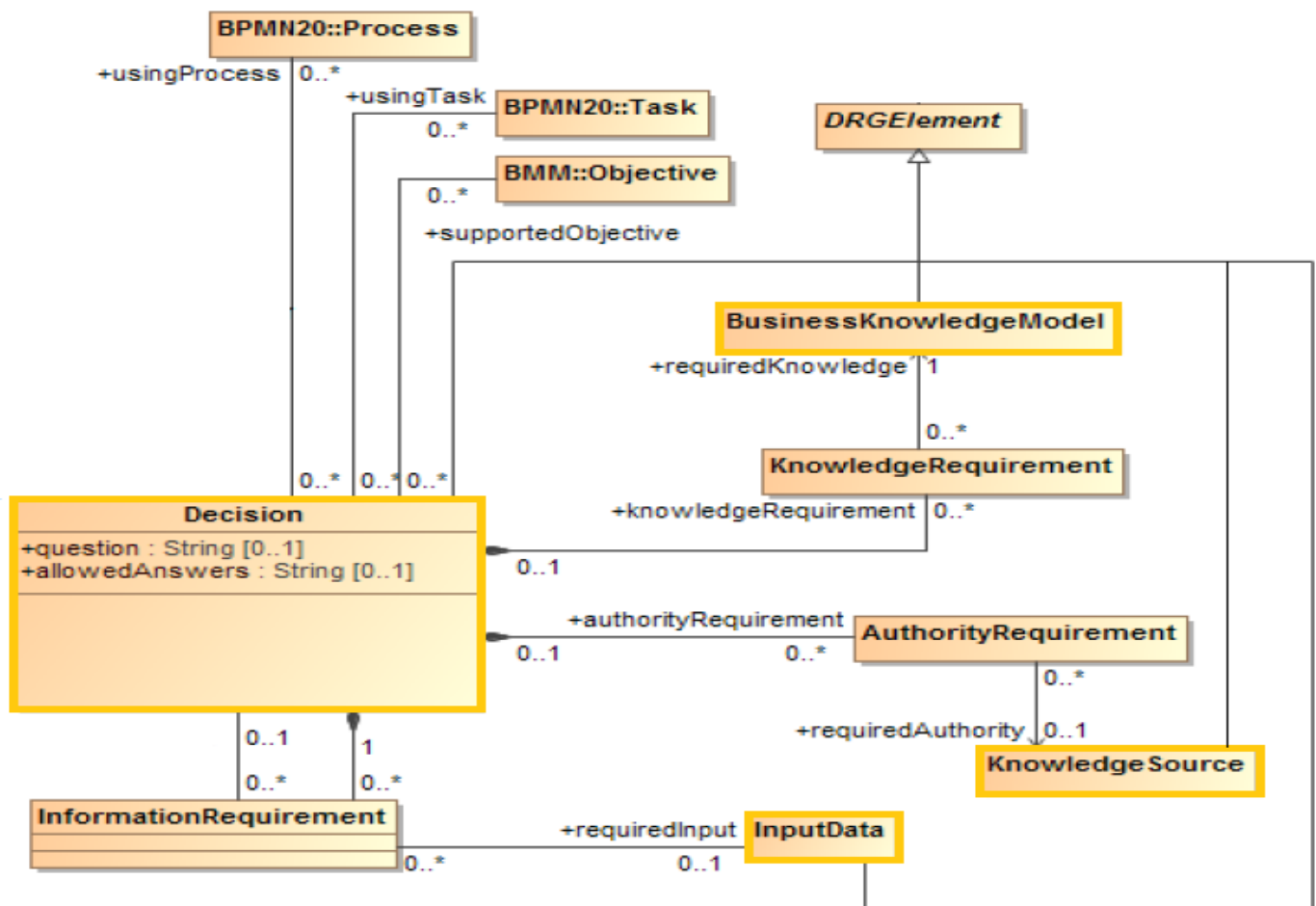


Figure 8 : Extrait du métamodèle DMN de l'OMG : diagramme de classes (UML) de la Décision

4.5 Modélisation des Règles Métier

4.5.1 Différentes Façons d'Exprimer la Logique de Décision

Il existe trois façons d'exprimer dans DMN la logique d'une décision (Comment une décision – la sortie – est-elle prise en fonction des données en entrée ?) :

(1) Une expression littérale libre : un simple texte descriptif pour définir les règles métier, difficilement automatisable, mais qui laisse une marge au décideur pour son interprétation,

(2) Une expression littérale formalisée, qui utilise un langage de programmation générique comme Java, ou mieux encore, le DSL (*Domain Specific Language*) FEEL (*Friendly Enough Expression Language*), qui fait partie de DMN. Exemple : `if Vacataire.Age < 67 then true else false`

(3) Une table de décision, qui est la façon la plus formelle et la plus courante, et qui est détaillée ci-dessous. Une table de décision est une forme particulière d'expression « en boîte » (*boxed expression*), dont l'usage doit être privilégié dans DMN.

4.5.2 Représentation des Tables de Décision

Tandis que DMN est un standard récent (2014), le formalisme des tables de décision est directement issu des travaux du CODASYL, qui datent de 1982 [Codasyl, 1982].

Dans les tables de décision dites horizontales (représentation recommandée, par défaut), les règles métier sont numérotées et listées sous forme de lignes. Les données d'entrée, qui sont donc les critères de décision, sont listées sous forme de colonnes, ainsi que la/les donnée(s) en sortie (le résultat de la décision).

Il doit y avoir une séparation claire entre les conditions (données en entrée) et les actions (décisions en sortie) [Vanthienen et Dries, 1994]. On peut préciser les types de données ou leurs domaines de valeur respectifs sur une ligne supplémentaire.

Décision				
U	Entrées			Sortie
	Donnée d'entrée 1	Donnée d'entrée 2	Donnée d'entrée 3	Décision
1	Valeur 11	Valeur 21	Valeur 31	Valeur A
2	Valeur 12	Valeur 22	Valeur 32	Valeur B
3	Valeur 13	Valeur 23	Valeur 33	Valeur C

Tableau 2: Table de décision représentant la logique d'une décision

4.5.3 DMN, un Style de Langage Déclaratif

Contrairement à la représentation des passerelles en BPMN qui impose par nature un séquençement particulier dans la modélisation des règles et l'évaluation des critères, l'ordre des règles et des critères dans une table de décision est arbitraire, car ceux-ci sont tous évalués en même temps, présageant une prise de décision plus rapide.

L'ordre des règles et des critères dans une table de décision n'influence pas leur interprétation et conduit à prendre exactement les mêmes décisions, quel que soit cet ordre. Aussi pouvons-nous dire que, contrairement à BPMN qui a un style de langage impératif, DMN a un style de langage déclaratif.

Les valeurs possibles en sortie d'une table de décision DMN doivent correspondre aux valeurs attendues dans le diagramme BPMN. Mais rappelons ici que DMN peut s'utiliser seul, sans être obligatoirement associé à BPMN. Cette indépendance est le résultat du principe de Séparation des Préoccupations.

4.5.4 Deux Propriétés Cruciales : Complétude et Cohérence

Bien que cela ne soit pas précisé dans le standard DMN, il a deux propriétés cruciales que doivent avoir les bonnes tables de décision : la complétude et la cohérence [Silver, 2016].

Tandis qu'il est possible de formaliser les tables de décision avec un simple tableur, les outils spécialisés dans la modélisation des décisions offrent une aide précieuse pour vérifier automatiquement leur complétude et leur cohérence.

4.5.5 Complétude d'une Table de Décision

La vérification de la complétude permet d'ajouter des règles implicites afin de couvrir toutes les combinaisons possibles des valeurs des données d'entrée. Il s'agit généralement des règles conduisant à une sortie de type « faux », les textes de spécifications se contentant généralement de décrire les règles (explicites, donc) qui conduisent à une sortie de type « vrai ». Le modélisateur devra souvent prendre quelques initiatives de simplification, afin de limiter le nombre de combinaisons et donc de règles possibles, tout en maintenant l'exhaustivité.

4.5.6 Cohérence d'une Table de Décision

La vérification de la cohérence détecte un chevauchement éventuel entre deux règles ou plus, où des valeurs identiques en entrée pourraient conduire à des valeurs différentes en sortie. La cohérence est obligatoire pour les tables de décision avec une « politique de succès » (*hit policy*) de type Unique : une et une seule règle doit s'appliquer aux données d'entrée.

4.5.7 Politiques de Succès des Tables de Décision

Il est recommandé d'utiliser des tables de décision de type Unique, où toutes les règles doivent être disjointes, bien qu'il existe 6 autres types de politique de succès simples (Any, Priority ou First) ou multiples (Output order, Rule order ou Collect), qui ne seront ni détaillés, ni même traduits, afin de conserver leurs lettres initiales. Notons toutefois que les politiques de succès First et Rule order, qui dépendent de l'ordre des règles, doivent être évitées, afin de conserver le style de langage déclaratif des tables de décision.

4.5.8 Exemple de Modélisation avec DMN

Le sous-processus modélisé dans la Figure 3 précédente en BPMN exclusivement est ici remplacé par une tâche de type règle métier :

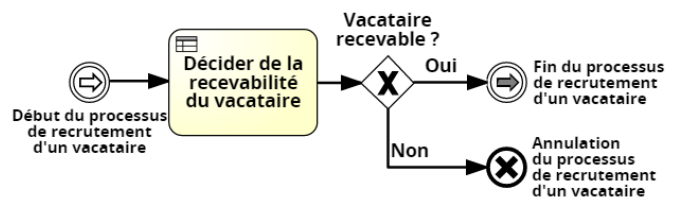


Figure 9: Extrait du processus de recrutement avec une tâche Règle "Décider de la recevabilité du vacataire"

Avec un outil de modélisation BPMN supportant également DMN, comme Signavio Process Editor, il suffit alors de cliquer sur la minuscule table de décision pour lancer l'outil de modélisation DMN, comme Signavio Decision Manager [Signavio, 2016a], et afficher le modèle DMN associé (Figure 10 page suivante).

Une recommandation est que la décision principale dans DMN porte le même « nom » (souvent un verbe d'action, suivi d'un complément d'objet) que la tâche qui l'appelle dans BPMN, car la relation entre les deux diagrammes DMN et BPMN est implicite (pas de lien graphique entre les deux).

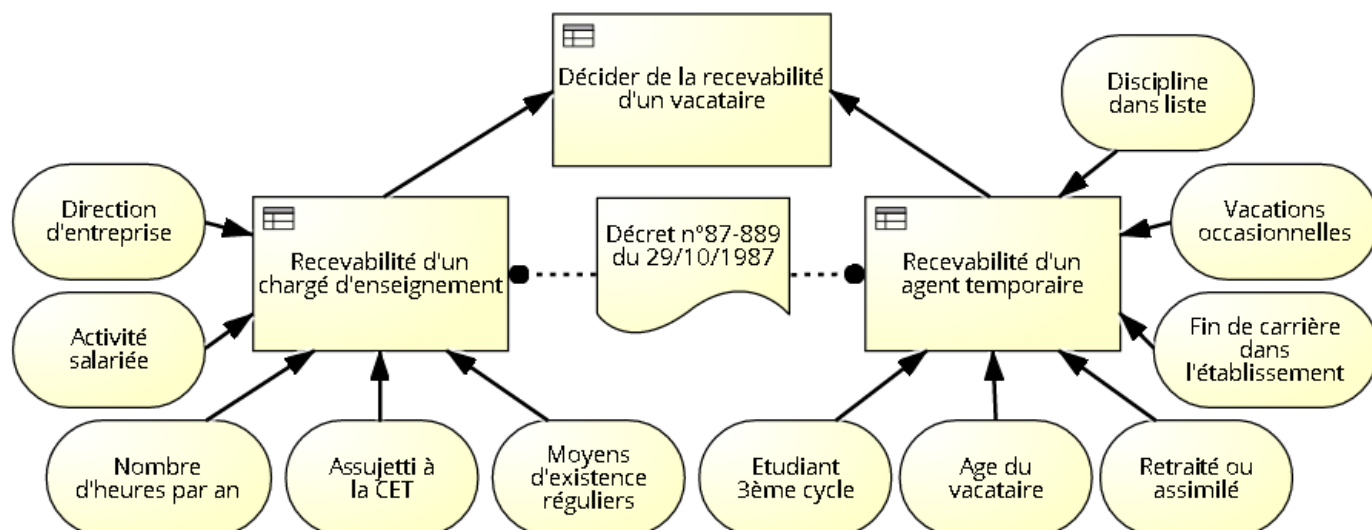


Figure 10: Diagramme de la Décision "Décider de la recevabilité du vacataire" (DMN)

La logique de chaque décision et sous-décision est représentée par une table de décision.

La logique de la sous-décision « Recevabilité d'un chargé d'enseignement » est représentée dans la table ci-dessous :

Recevabilité d'un chargé d'enseignement						
U	Entrées					Sortie
	Direction entreprise	Activité salariée	Nombre d'heures par an	Assujetti à la CET	Moyens existence réguliers	Recevabilité d'un chargé d'enseignement
	Booléen	Booléen	Nombre	Booléen	Booléen	Booléen
1	= vrai	-	-	-	-	vrai
2	= faux	= vrai	≥ 900	-	-	vrai
3	= faux	= vrai	< 900	-	-	faux
4	= faux	= faux	-	= vrai	-	vrai
5	= faux	= faux	-	= faux	= vrai	vrai
6	= faux	= faux	-	= faux	= faux	faux

Tableau 3: Table de décision "Recevabilité d'un chargé d'enseignement"

La logique de sous-décision « Recevabilité d'un agent temporaire » est représentée dans une table similaire :

Recevabilité d'un agent temporaire							
U	Entrées						Sortie
	Etudiant 3e cycle	Age du vacataire	Retraité ou assimilé	Fin de carrière établis.	Vacations occasion.	Discipline dans liste	Recevabilité d'un agent temporaire
	Booléen	Nombre	Booléen	Booléen	Booléen	Booléen	Booléen
1	= vrai	-	-	-	-	-	vrai
2	= faux	< 67	= vrai	= faux	= faux	= vrai	vrai
3	= faux	< 67	= vrai	= faux	= faux	= faux	faux
4	= faux	< 67	= vrai	= faux	= vrai	-	vrai
5	= faux	< 67	= vrai	= vrai	-	-	faux
6	= faux	< 67	= faux	-	-	-	faux
7	= faux	≥ 67	-	-	-	-	faux

Tableau 4: Table de décision "Recevabilité d'un agent temporaire"

Il aurait été possible techniquement de se passer de ces deux sous-décisions et d'associer les 11 données d'entrées directement à la décision principale, mais la table de décision aurait été plus compliquée et bien moins lisible. Ces deux sous-décisions et leurs tables de décision sont alors alignées sur les deux articles distincts du décret de référence qui fixent respectivement les critères de recevabilité des chargés d'enseignement et des agents temporaires.

Les règles métier sont listées sous forme de lignes, dans le même ordre que celui des articles du décret. Les données en entrée sont listées sous forme de colonnes, dans le même ordre que celui des articles du décret également. Comme nous l'avons expliqué précédemment (DMN est un langage de style déclaratif), cet ordre est arbitraire, mais facilite grandement la vérification de la bonne interprétation des règles métier originelles, souvent exprimées par du texte.

La logique de la décision principale « Décider de la recevabilité du vacataire » est une table beaucoup plus simple, qui utilise comme entrées les sorties des deux sous-décisions, que nous venons de décrire :

Décider de la recevabilité du vacataire			
U	Entrées		Sortie
	Recevabilité d'un chargé d'enseignement	Recevabilité d'un agent temporaire	Décider de la recevabilité du vacataire
	Booléen	Booléen	Booléen
1	= faux	= faux	faux
2	= faux	= vrai	vrai
3	= vrai	= faux	vrai
4	= vrai	= vrai	vrai

Tableau 5: Table de décision "Décider de la recevabilité du vacataire"

Il s'agit de représenter dans la table ci-dessus un opérateur logique de type OU, le vacataire recevable pouvant être chargé d'enseignement ou agent temporaire, selon les cours.

Il convient de préciser que le tiret « - » comme valeur d'entrée signifie qu'elle est « sans importance » et évite de représenter toutes les combinaisons possibles dans une table de décision. Toute démarche de simplification est toutefois discutable ; par exemple, un étudiant de 3^{ème} cycle, mais âgé de plus de 67 ans, est-il vraiment recevable ?

Les deux tables de décision présentées sur cette page sont cohérentes et complètes. Mais, comme tous les modèles, il faut s'assurer qu'elles représentent bien la réalité, c'est la préoccupation principale du modélisateur. Ou bien il faut aligner la réalité sur ces tables, c'est alors une décision tactique du manager. Car cet exercice de modélisation des décisions et de formalisation des règles dans des tables révèle souvent des différences entre théorie et pratique.

5 DEMARCHE METHODOLOGIQUE

5.1 Décisions et Règles dans les Cadres de Représentation

Les cadres de représentation comme TOGAF et Praxeme sont prévus pour prendre des décisions d'architecture d'entreprise, qui sont stratégiques et tactiques. Ces décisions de haut niveau ont un impact structurant sur l'organisation notamment et doivent pouvoir être justifiées. Pour les décisions opérationnelles, rien n'est prévu pour le moment. Quant aux règles, il s'agit de règles fondamentales, voire de principes généraux d'architecture, mais pas des règles métier.

5.2 Positionnement de DMN dans la Méthode Praxeme

Le positionnement le plus naturel des diagrammes DMN, représentant la structure des prises de décision, serait au plus près des diagrammes BPMN, c.-à-d. dans l'aspect Pragmatique qui présente les processus métier. Par contre, cette approche semble trop simpliste pour positionner les tables de décision, car elle ne permettrait de représenter qu'une partie des règles métier. Comme il est possible d'utiliser DMN seul, on pourrait l'utiliser dans l'aspect Sémantique également. Par contre, DMN qui s'appuie sur des Objets Métier, ne semble pas avoir sa place directement dans l'aspect Intentionnel.

5.3 Traitement des Règles Métier

La chaîne de traitement des règles métier peut être décomposée en plusieurs étapes : collecte des règles métier utilisées dans l'Entreprise et exprimées en langage naturel ; analyse ; qualification (selon leur légitimité et le niveau de contrainte qu'elles imposent) ; reformulation et enfin typage, pour les positionner dans les aspects adéquats de la Topologie du Système Entreprise, avant de les formaliser.

Les règles dites de gestion agissant sur les objets métier seront positionnées sur l'aspect Sémantique, tandis que les règles agissant sur les organisations seront positionnées sur l'aspect Pragmatique. Le respect de la réglementation introduit des exigences, type de règles particulières, obligatoires et contraignantes, à formaliser également.

La couverture complète du domaine métier par ces différentes règles devra être approchée, tout en évitant les recouvrements, et en garantissant la cohérence d'ensemble. La technique de vérification de l'exhaustivité des cas d'utilisation UML, basée sur l'étude du cycle de vie des objets, pourrait être transposée aux règles métier.

6 CONCLUSION ET PERSPECTIVES

Après avoir présenté les grands principes de l'Architecture d'Entreprise, les besoins en modélisation et les limites de BPMN pour la représentation des prises de décision, cet article a proposé une nouvelle approche. En séparant les préoccupations du modélisateur, comme l'a déjà fait l'architecte pour les applications métier, les processus, les prises de décision et les règles métier sont alors représentés par des modèles adéquats, cohérents entre eux, bien que faiblement couplés. La notation DMN sous forme de diagrammes et de tables de décision a été présentée.

L'objectif était de démontrer que DMN est un standard pertinent (ou pas) pour modéliser les prises de décisions dans les processus métier. Un exemple de modélisation sans DMN, puis avec DMN a permis de comparer l'approche traditionnelle avec cette nouvelle approche. Dans ce cas particulier, extrait de la réalité et à peine simplifié, les modèles avec DMN semblent bien plus simples à comprendre et éventuellement à modifier que les modèles BPMN. La pertinence semble donc démontrée.

Le standard DMN est très récent. Des améliorations sont déjà proposées pour la prochaine version [Taylor et Purchase, 2016].

Enfin, une ébauche de démarche pour enrichir la méthode Praxeme - rappelons qu'il s'agit d'une méthode ouverte - a été proposée. L'impact sur l'architecture en général et sur l'aspect Logique de la méthode Praxeme en particulier sera évalué, ainsi que le gain potentiel sur l'évolutivité grâce à la séparation des préoccupations (processus et décisions) et la factorisation des règles métier et donc la réduction du code.

La dernière phase sera l'outillage (réalisation d'un démonstrateur). L'automatisation des prises de décision (modélisées sous la forme de diagrammes DMN) dans les processus métier selon les règles métier (formalisées sous la forme de tables de décision) avec un outil informatique sera étudiée. Selon l'approche moderne de l'architecture des applications métier, le moteur de règles sera externalisé.

La principale perspective est de proposer un nouvel outil d'architecture qui permet à l'Entreprise d'aligner son Système d'Information sur sa stratégie, afin de relever le défi de ses transformations à venir.

7 REFERENCES

- Bonnet, P., Detavernier, J.-M., Vauquier, D. (2007). *Le système d'information durable*. Hermès science.
- Caseau, Y. (2011). *Urbanisation, SOA et BPM*. Dunod.
- Codasyl. (1982). A modern appraisal of decision tables. ACM.
- Debevoise, T., Taylor, J. (2014). *The MicroGuide to Process and Decision Modeling in BPMN/DMN*. ACR.
- Desfray, P., Raymond, G. (2014). *TOGAF en pratique*. Dunod.
- Dijkstra, E. W. (1974). On the role of scientific thought.
- Fish, A. (2012). *Knowledge automation: how to implement decision management in business processes*. Wiley.
- Henderson, J., Venkatraman, N. (1992). Strategic Alignment.
- Jézéquel, J.-M., Combemale, B., Vojtisek, D. (2012). *Ingénierie dirigée par les modèles*. Ellipses.
- Linehan, M., de Sainte Marie, C. (2011). The Relationship of Decision Model and Notation (DMN) to SBVR and BPMN. <http://www.brcommunity.com/b597.php>
- OMG. (2014). MDA (Model Driven Architecture) Specifications. <http://www.omg.org/mda/specs.htm>
- OMG. (2016). Business Modeling Specifications (BPMN, BMM, DMN, SBVR, etc.). <http://www.omg.org/spec/>
- Open Group. (2011). TOGAF Version 9.1. <http://www.opengroup.org/togaf/>
- OpenRules. (2016). Decision Model and Notation (DMN) Supporting Tools. <http://openjvm.jvmhost.net/DMNtools/>
- Praxeme Institute. (2006). Initiative pour une méthode publique. <http://www.praxeme.org/>
- Signavio. (2016a). Decision Manager. <http://www.signavio.com/products/decision-manager/>
- Signavio. (2016b). Process Editor. <http://www.signavio.com/products/process-editor/>
- Silver, B. (2016). *DMN method and style*. Cody-Cassidy Press.
- Tardieu, H., Rochfeld, A., Colletti, R. (2000). *La méthode Merise : principes et outils*. Ed. d'Organisation.
- Taylor, J., Purchase, J. (2016). *Real-World Decision Modeling with DMN*. Meghan-Kiffer Press.
- Vanthenien, J., Dries, E. (1994). Decision Tables: Refining the Concept and a Proposed Standard. *ResearchGate*.
- Von Halle, B., Goldberg, L. (2010). *The Decision Model*. Auerbach.
- Zachman, J. (2011). About the Zachman Framework™. <http://www.zachman.com/about-the-zachman-framework>