



HAL
open science

Capitalisation et réutilisation sous la forme de patrons dans une démarche d'Ingénierie Système Basée sur les Modèles (ISBM) - Proposition de l'approche Minage-Maturation-Implémentation (MMI)

Quentin Wu, David Gouyon, Éric Levrat, Sophie Boudau

► To cite this version:

Quentin Wu, David Gouyon, Éric Levrat, Sophie Boudau. Capitalisation et réutilisation sous la forme de patrons dans une démarche d'Ingénierie Système Basée sur les Modèles (ISBM) - Proposition de l'approche Minage-Maturation-Implémentation (MMI). 16ème colloque national S-mart/AIP-PRIMECA, Apr 2019, Les Karellis, France. hal-02128210

HAL Id: hal-02128210

<https://hal.science/hal-02128210>

Submitted on 14 May 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Capitalisation et réutilisation sous la forme de patrons dans une démarche d'Ingénierie Système Basée sur les Modèles (ISBM)

Proposition de l'approche Minage-Maturation-Implémentation (MMI)

Quentin Wu, Sophie Boudau

Zodiac Aero Electric
7 rue des Longs Quartiers
Paris – France

quentin.wu@zodiacaerospace.com,
sophie.boudau@zodiacaerospace.com

David Gouyon, Éric Levrat

Université de Lorraine, CNRS, CRAN
Faculté des Sciences et Technologies, BP 70239
Vandœuvre-lès-Nancy – France

david.gouyon@univ-lorraine.fr, eric.levrat@univ-lorraine.fr

Résumé—*La maîtrise du savoir-faire au sein d'une entreprise est essentielle afin de disposer d'une base commune de « bonnes pratiques » disponible pour toutes les équipes d'ingénierie. Pour cela, il est nécessaire d'entreprendre une démarche de capitalisation afin d'encapsuler ces pratiques. Cependant, il est tout aussi important de rendre disponible l'accès à ce savoir-faire et d'en faciliter sa réutilisation afin que chaque ingénieur puisse se l'approprier. Afin de répondre à ce besoin de capitalisation et de réutilisation, nous proposons une approche méthodologique qui permet d'entreprendre cette démarche dans un cadre d'Ingénierie Système Basée sur les Modèles (ISBM), et qui entend s'appuyer sur le concept de « patron » afin d'encapsuler le savoir-faire à capitaliser. En effet, la flexibilité des patrons lors de la réutilisation est un avantage qui participera à l'efficacité d'une ISBM où les équipes d'ingénierie sont capables de s'appuyer sur le savoir-faire de l'entreprise.*

Mots-clés— *Capitalisation, Réutilisation, Savoir-Faire, Méthodologie, Ingénierie Système Basée sur les Modèles (ISBM), Minage, Maturation, Implémentation*

I. INTRODUCTION

Il est essentiel pour chaque organisation de pérenniser son savoir-faire afin de créer une base commune de savoir-faire clé à partager entre ses équipes d'ingénierie. Cela permet une meilleure compréhension du système à développer ainsi que des processus et méthodes propres à l'entreprise. Par cette capitalisation de savoir-faire, l'objectif est d'améliorer l'efficacité de l'ingénierie. En effet, les avantages sont nombreux face aux risques de : départ de salariés possédant le savoir-faire, syndrome de la page blanche ou de copier-coller, déviation par rapport aux besoins, répétition des mêmes erreurs, « réinventer la roue »,... De plus, ce savoir-faire clé, initialement « statique », puisque individuel et difficilement partageable, se retrouve désormais « dynamique » à la portée de tous et permet de mettre en avant une base commune des systèmes d'intérêts développés et des activités d'Ingénierie Système.

L'Ingénierie Système Basée sur les Modèles (ISBM) est définie par l'AFIS comme « l'utilisation formalisée de modèles comme support aux activités de spécification, conception, analyse, vérification et validation des systèmes ». Dans ce cadre, la mise en place d'une démarche de réutilisation de savoir-faire, pose plusieurs questions :

1. Que faut-il capitaliser ? Il s'agit d'identifier les éléments de savoir-faire qu'il est pertinent de capitaliser [1], et de se demander si l'effort nécessaire à cette capitalisation aura une rentabilité suffisante.
2. Sous quelle forme capitaliser ce savoir-faire clé en vue de sa réutilisation ?
3. Quelle démarche de réutilisation entreprendre dans un cadre d'ISBM ?

La première question portant sur la capitalisation, pour laquelle de nombreux travaux ont proposé des approches issues de l'ingénierie des connaissances telles que MASK [2], CommonKADS [3],... se situe en dehors du périmètre de cet article qui considère que la connaissance est déjà formalisée à l'aide de modèles métiers. Il adresse plus particulièrement les questions 2 et 3. Pour cela, l'article présente dans la partie II un état de l'art des approches de réutilisation en Ingénierie Système, en se focalisant en particulier sur l'utilisation de patrons en ISBM. La partie III propose une classification de ces patrons en quatre niveaux d'abstraction sur laquelle repose l'approche MMI de réutilisation sous la forme de patrons, introduite en partie IV et illustrée sur un exemple en partie V. Pour des raisons de confidentialité, l'exemple est fourni avec des éléments de modélisation dont la dénomination a été modifiée.

II. ÉTAT DE L'ART

A. Approches de réutilisation en Ingénierie Système

La réutilisation permet de reprendre et d'appliquer du savoir-faire et des connaissances sans avoir besoin de retraiter l'ensemble des informations nécessaires ayant permis de produire ce savoir-faire ou ces connaissances. En Ingénierie Système, de nombreux travaux de recherches se sont intéressés à cette pratique [4]–[12], et il apparaît à la lumière de ces travaux qu'il existe différentes manières de capitaliser du savoir-faire et des connaissances en vue de leur réutilisation. Ces approches appartiennent à un processus dit de « transfert de connaissances » [13] et on peut distinguer trois approches différentes de réutilisation, qui ne sont pas exclusives :

- Opportuniste (O) : lorsque le projet n'est pas prévu initialement avec des capacités de réutilisation. C'est l'ingénieur qui est à l'initiative de la démarche, il s'agit

donc du niveau le plus bas de fréquence de réutilisation [14] car toute la démarche repose sur la bonne volonté de l'utilisateur ;

- Planifiée (P) : lorsque le projet est développé avec des capacités de réutilisation. Dans ce cas, la démarche de réutilisation a été intégrée dans le processus d'ingénierie de manière à ce que les éléments développés soient réutilisables [15] ;
- Variante (V) : lorsque le projet définit un ensemble de solutions avec des variantes ou options possibles. Par exemple, sur une ligne de produit [16], il y a un modèle « cœur » commun et chaque produit est composé d'un ensemble d'options différentes [17].

Pour laisser la liberté de réutilisation à l'ingénieur tout en lui permettant d'être efficace dans son travail, il est nécessaire de le guider afin de rendre les éléments réutilisables pour les prochains projets et d'éviter que la réutilisation soit effectuée en « *one-shot* » sans capitalisation. De même, il est important de dissocier les artefacts d'ingénierie qui seront impliqués dans le processus [18]. En effet, les activités de réutilisation peuvent concerner :

- le Système d'Intérêt (SI) : le système dont le cycle de vie est considéré
- les Activités d'Ingénierie Système (AIS) : qui ont pour objectif de développer le Système d'Intérêt.

A des fins d'illustration, dans le cas d'étude de la partie V, nous considérerons une réutilisation « planifiée », dans laquelle des patrons du Système d'Intérêt seront créés et réutilisés. De plus, nous nous placerons dans un cadre d'Ingénierie Système Basée sur les Modèles afin de profiter des avantages de la modélisation pour la réutilisation de patrons [19].

B. Patrons en Ingénierie Système Basée sur les Modèles

Nous utilisons le concept de « patron », tel que défini par [20] comme étant « la description d'un problème qui ne cesse d'apparaître encore et encore dans notre environnement, et la description de l'essence de la solution à ce problème, d'une manière à ce que l'on puisse utiliser cette solution des millions de fois, sans jamais avoir à l'appliquer deux fois de la même manière », et que l'on peut donc associer *a minima* au triplet {Contexte, Problème, Solution} défini par [21]. L'objectif est ici d'utiliser ce concept afin d'améliorer l'efficacité du travail d'ingénierie par l'introduction de capacité de réutilisation dans un cadre d'ISBM [22]. En effet, le processus d'apprentissage étant très exigeant, il semble nécessaire d'aider les ingénieurs à « identifier rapidement les solutions optimales pour la mission voulue et non plus les solutions architecturales valides » [23]. Constituant une réponse adaptée, les patrons peuvent être introduits dans un cadre d'ISBM afin de jouer un rôle de guide à la fois pour la conception et pour la modélisation, et cela dans les différentes phases du cycle d'ingénierie.

Ainsi, afin d'aider les ingénieurs à se concentrer sur l'essentiel lors du processus de conception dans un cadre de modélisation, les travaux de [24] proposent un processus de développement de systèmes mécatroniques basé sur un patron de conception exprimé en langage SysML (*System Modeling Language*). L'objectif des auteurs est de démontrer que des instructions claires et bien définies pour la modélisation apportent un avantage considérable au processus de développement et notamment sur l'aspect « traçabilité » des informations. Cela permet notamment de faciliter les analyses

d'impact puisque la traçabilité est assurée entre l'exigence cliente (haut niveau) et l'exigence organique (bas niveau), qu'il s'agisse d'une évolution descendante (*top-down*) ou bien remontante (*bottom-up*). De plus, la structuration proposée facilite la réutilisation et l'intégration à un même niveau d'abstraction d'un ensemble d'éléments provenant d'autres projets puisque la structure de modélisation est la même pour tous les projets. Dans la même idée d'empêcher la déviance par rapport aux besoins au cours d'une modélisation, [25] décrit des patrons de « constructions » comportementales pour les diagrammes d'activités. Au lieu de réfléchir à un niveau atomique, ces « constructions » permettent de structurer la réflexion des ingénieurs de manière algorithmique, leur accordant la possibilité d'entrer dans un niveau d'abstraction supérieur pour la modélisation en se concentrant sur le comportement attendu et non sur l'esthétique des diagrammes.

L'intérêt des patrons réside également dans l'exploration et l'analyse des systèmes. Les travaux de [26] s'intéressent ainsi aux patrons d'architectures pour l'analyse des Systèmes de Systèmes. En effet, la complexité de tels systèmes rend difficile l'analyse des choix entre les différentes solutions possibles. Les auteurs proposent ainsi de miner des patrons du Système d'Intérêt exprimés en SysML, afin de représenter un modèle du Système de Système. Cela permet de mieux comprendre son comportement et de conduire les changements nécessaires à son optimisation en s'assurant que l'on ne compromet pas le reste du système ou bien que l'on fasse apparaître un comportement indésirable. Ainsi, des recommandations apparaissent pour la réutilisation de ces architectures dans les futurs projets.

Certains travaux proposent de voir les patrons comme des « modèles réutilisables » et proposent un paradigme d'ingénierie appelé *Pattern-Based Systems Engineering*. Avec l'avènement de l'ISBM, ce cadre de modélisation a conduit à la création d'un groupe de travail à l'INCOSE (*International Council On Systems Engineering*) appelé *MBSE Patterns (Model-Based Systems Engineering Patterns)*. Le cadre permet de configurer ou de spécialiser des patrons dans des lignes de produits ou des ensembles de systèmes. Dans ce contexte, les auteurs de [27], [28] développent leur propre approche car ils considèrent les « patrons comme des modèles réutilisables » et les appliquent aux exigences et à la conception. À un haut niveau d'abstraction, ils forment un modèle de patron générique de système qui peut être personnalisé pour les besoins, la configuration ou l'utilisation d'une entreprise, afin que les ingénieurs puissent tirer profit des concepts de l'ISBM sans être un expert des méthodes de modélisation. Dans [8], les auteurs appliquent cette approche durant le processus de V&V, tandis que dans [29] elle est appliquée sur le marché pharmaceutique.

III. NIVEAU D'ABSTRACTION

La formalisation de patrons étant possible sur plusieurs niveaux d'abstraction, nous proposons une classification en quatre niveaux (Figure 1), allant du modèle réalisé par l'ingénieur au plus bas niveau d'abstraction jusqu'aux patrons abstraits au plus haut niveau d'abstraction, et précisant le formalisme d'expression à chacun des niveaux :

- Patrons abstraits
- Patrons de conception (*Design Patterns*)
- Modèles génériques
- Modèles

Cela permet de décrire un flot remontant et un flot descendant, correspondant respectivement à un processus

d'abstraction / minage et à un processus de concrétisation / implémentation.

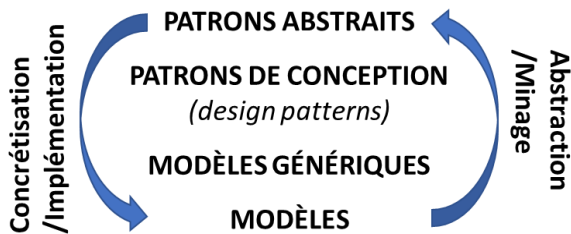


Figure 1. NIVEAUX D'ABSTRACTION MODELE - PATRON

A. Modèles

À chaque projet ses modèles, exprimés dans des langages métiers (*DSL – Domain Specific Language*). Ces ensembles de modèles, spécifiques à chaque projet, peuvent donc être isolés en « silos ». Ils ont comme caractéristiques d'avoir potentiellement des syntaxes et des sémantiques différentes s'il n'existe pas de règle de modélisation commune. De même, les règles de nommage peuvent être spécifiques à chaque projet, tout comme la structure du projet et les interactions avec l'environnement d'ingénierie.

Or, ces modèles sont le résultat de savoir-faire d'ingénierie. Ainsi, dans le cas où la capitalisation est mise en place et où l'on promeut la réutilisation, on doit être capable entre chaque projet de reconnaître des structures, syntaxes, sémantiques, ... communs. On facilite ainsi la lecture et la communication même si les modèles sont toujours spécifiques à chaque projet.

B. Modèles génériques

À ce niveau, les modèles sont également exprimés dans un DSL. Cependant, contrairement aux modèles de la sous-partie A, les modèles génériques ont pour vocation de mettre en valeur les similarités et les motifs identifiés dans le système. Ils peuvent être de différents types [1] : exigences, design, interfaces, fonctions ... L'idée est de proposer un modèle d'un niveau d'abstraction supérieur afin de pouvoir réfléchir à un plus haut niveau système, en vue de son instanciation dans les modèles. Cela permet notamment d'améliorer la lisibilité et la compréhension des modèles.

C. Patrons de conception (*design patterns*)

Le but de ce patron de conception est de représenter les relations entre les éléments d'un modèle générique indépendamment du langage de modélisation métier. L'objectif est donc la capture d'un savoir-faire, dans un formalisme indépendant du contexte de chaque équipe d'ingénierie, afin d'en faciliter sa promotion et sa réutilisation. Cependant, ces patrons restent décrits et adaptés à un domaine spécifique.

D. Patrons abstraits

À ce niveau, on s'affranchit du domaine d'utilisation finale, en présentant des principes généraux de structuration, en quelque sorte des invariants sur lesquels les projets peuvent se baser. Il s'agit d'un niveau qu'il n'est pas obligatoire d'atteindre mais qui permet une abstraction de très haut niveau par rapport au système étudié. On pourra y trouver de nombreux avantages pédagogiques dans la volonté d'expliquer un système à très haut niveau. Il est également possible d'y trouver un attrait lorsqu'on est dans une démarche d'innovation.

IV. APPROCHE MMI

Pour répondre aux problématiques classiquement identifiées en capitalisation des connaissances (repérer,

préservé, valoriser, actualiser) [30], l'objectif est de proposer une méthode permettant la formalisation et la réutilisation de patrons du Système d'Intérêt. Celle-ci consiste en la recherche (*repérer* et *préservé*) de patrons (processus de *minage*), la maturation (*actualiser*) de ces patrons en vue de leur réutilisation (processus de *maturation*), et enfin la réutilisation concrète (*valoriser*) de ces patrons capitalisés à différents niveaux d'abstraction pour la modélisation (processus d'*implémentation*). L'approche méthodologique proposée peut être synthétisée par la Figure 2. Les processus qu'elle met en œuvre sont détaillés ci-dessous.

A. Minage

La mise en place d'une démarche de capitalisation n'est pas anodine et demande un investissement important qu'il est parfois difficile de concilier avec la vie d'un projet et ses multiples contraintes (délais, coûts, effectifs, ...). C'est pourtant la condition sine qua non qui permet de lancer la démarche de minage de patrons. En effet, l'acte de capitalisation nécessite une prise de décision afin d'enclencher la démarche. Le processus de minage nécessite d'analyser les modèles des projets précédents et, si c'est possible, des projets en cours. L'analyse de ces modèles va permettre d'identifier, de repérer, d'isoler des motifs ou des similarités d'Ingénierie Système qui sont réutilisés à plusieurs endroits dans le même projet ou dans des projets différents, pour en proposer des modèles génériques. Elle permettra également de déterminer des patrons de conception (*design pattern*) indépendants du langage de modélisation utilisé dans les projets.

Ces deux types d'éléments seront des guides de modélisation système, ils seront donc utiles à la fois pour la conception et l'analyse de systèmes. Les intérêts principaux sont donc de pouvoir concevoir un modèle « générique » d'un projet ou d'une partie d'un projet, et de réfléchir à un niveau d'abstraction plus élevé.

B. Maturation

La maturation est un processus crucial de l'approche méthodologique car elle a un impact très fort sur le processus d'« implémentation ». En effet, il va falloir faire évaluer les modèles génériques et les patrons de conception identifiés afin que leur niveau de maturité, et donc le niveau de confiance avec lequel ils vont pouvoir être réutilisés, corresponde à un niveau permettant leur réutilisation sur des nouveaux projets. À cheval entre le minage et le début du processus de maturation, la vérification et la validation par des experts est donc nécessaire avant d'envisager la réutilisation de ces patrons. Il est de plus bénéfique de mettre en place des cycles courts proches de l'approche agile afin de pouvoir rapidement converger vers un patron de haute maturité.

Lors du premier cycle de capitalisation, une stratégie de concrétisation et d'implémentation doit être décidée. En effet, on peut vouloir laisser une grande liberté à l'utilisateur, ou au contraire guider la modélisation en fonction de la stratégie d'implémentation choisie. Cela demande la création d'une bibliothèque dont il va falloir déterminer les conditions d'accès, de recherche et d'insertion de patrons.

Ensuite, il faut implémenter cette démarche dans un outil, en intégrant des exigences relatives à son utilisation et notamment à l'interface homme-machine (IHM). L'objectif est de faciliter le travail des utilisateurs afin qu'ils puissent se focaliser sur l'ingénierie plutôt que sur l'esthétique des différentes vues du modèle.

C. Implémentation

Lors de ce processus, lorsqu'une opportunité de réutilisation est identifiée, la stratégie identifiée plus haut permet soit la concrétisation d'un patron abstrait ou de conception (*design pattern*), soit la réutilisation directe d'un modèle générique en vue de modéliser le système. Le processus d'implémentation laisse une part active à l'utilisateur qui va intégrer les différents éléments réutilisables à son modèle, en fonction des exigences à satisfaire. Cette intégration n'est pas automatique dans le sens où il faut que l'ingénieur puisse modifier son modèle en fonction

des regroupements fonctionnels, logiques, organiques qu'il souhaite faire.

V. EXEMPLE

Pour illustrer nos propos, nous adressons les deux premiers niveaux d'abstraction (modèles et modèles génériques), sur un exemple issu d'un cas d'étude provenant de Zodiac Aero Electric dans lequel, pour des raisons de confidentialité, les noms ont été modifiés ou floutés. Ce cas d'étude se focalise sur un modèle d'architecture fonctionnelle.

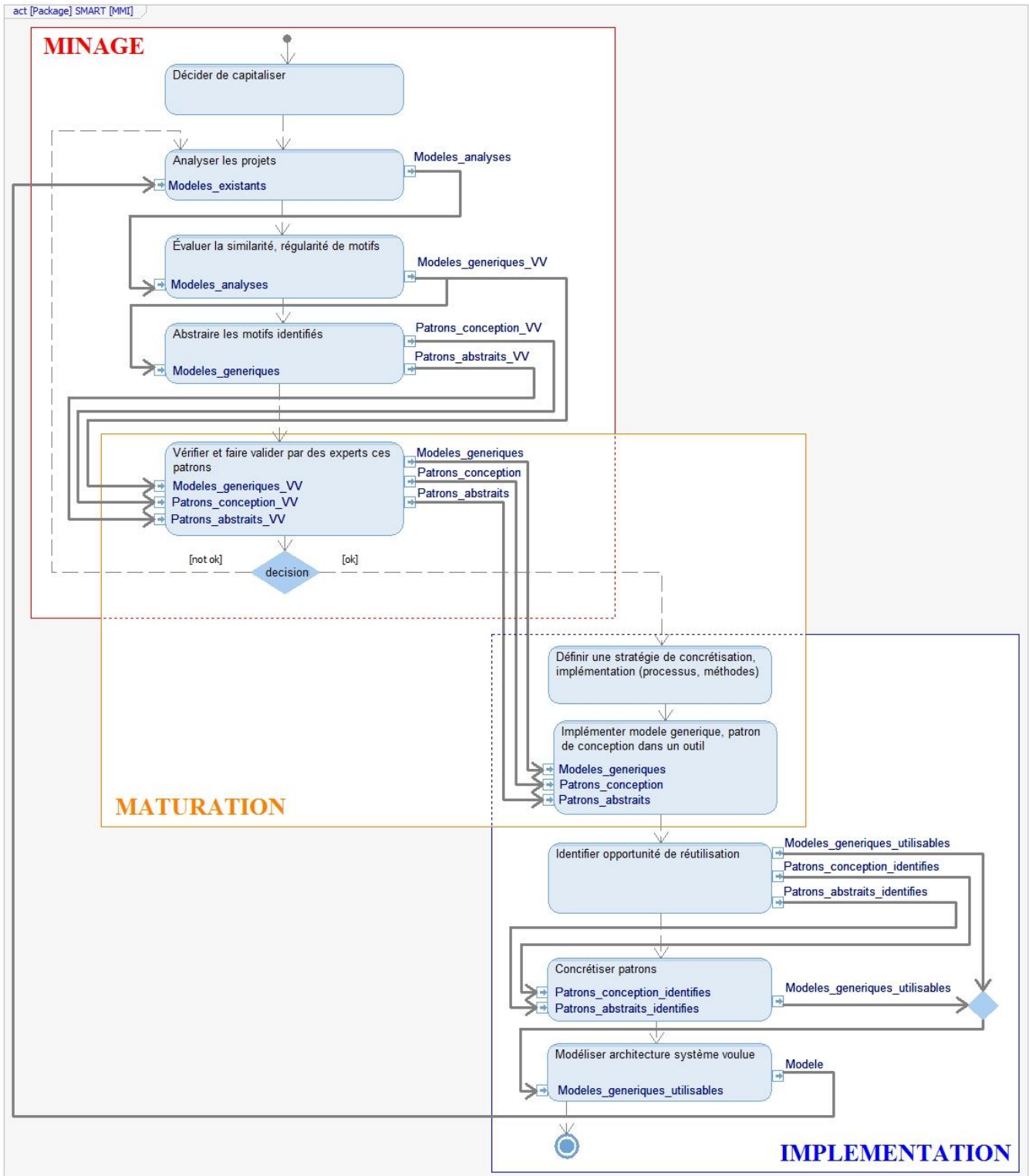


Figure 2. APPROCHE METHODOLOGIQUE MINAGE – MATURATION – IMPLEMENTATION

A. Données d'entrées

La base de travail est une décomposition fonctionnelle, dans laquelle interviennent 700 fonctions, organisées en 8 niveaux hiérarchiques. La taille du livrable de ce travail de décomposition fonctionnelle empêche une navigation efficace entre les différentes fonctions, et ce même pour les auteurs. De plus, le lien entre les fonctions est très difficile à visualiser et la gestion de la cohérence de l'ensemble du document est compliquée, d'autant qu'aucune métrique n'est disponible.

B. Application de l'approche MMI

1) Minage

À la suite de nombreuses itérations sur la décomposition, des motifs qui se répétaient, constitués d'un même ensemble de fonctions, ou des fonctions ayant un ensemble commun d'entrées/sorties, ont été identifiés (Figure 3). Ces itérations ont également permis d'identifier des représentations multiples de certaines fonctions.

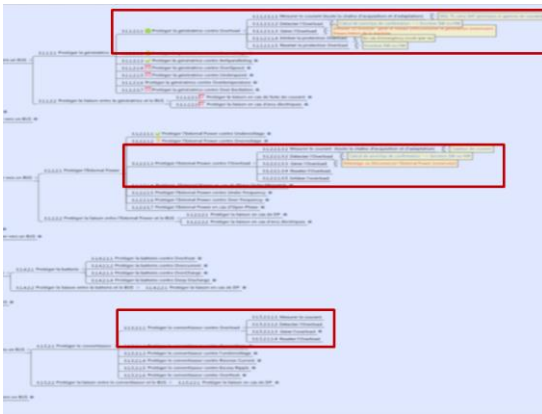


Figure 3. IDENTIFICATION DE MOTIFS SUR LA DECOMPOSITION FONCTIONNELLE BRUTE

L'utilisation de modèles génériques de fonctions a permis d'utiliser une forme de factorisation du modèle, et de réaliser une décomposition fonctionnelle générique, où il n'y a plus que 6 niveaux hiérarchiques et 119 fonctions. On retrouve l'ensemble des fonctions similaires à un seul endroit. Ainsi la facilité de lecture et la navigabilité sont grandement améliorées (Figure 4).

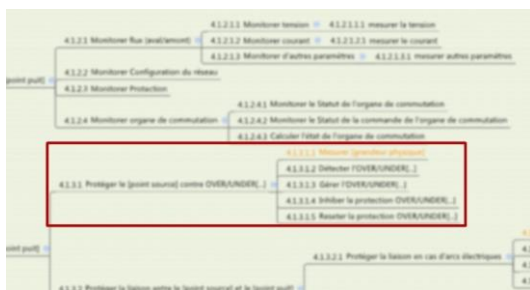


Figure 4. DECOMPOSITION FONCTIONNELLE FACTORISEE

Sur cette base de décomposition fonctionnelle, nous avons modélisé l'architecture fonctionnelle à l'aide du diagramme d'activité du langage SysML.

2) Maturation

À haut niveau, les fonctions qui se répétaient ont donc été modélisées afin de représenter les entrées et sorties nécessaires. Au lieu de le faire à chaque itération, la réflexion sur un modèle générique permet d'effectuer le travail à un niveau d'abstraction plus élevé, et de le réutiliser ensuite plusieurs fois. On peut ensuite faire vérifier et valider par des experts l'architecture

fonctionnelle générique, dans laquelle on retrouve des modèles génériques de fonctions (Figure 5).

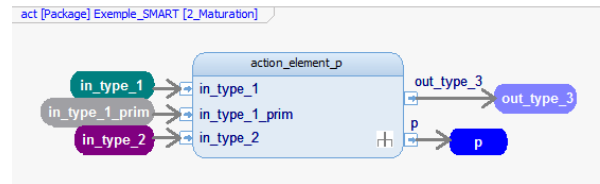


Figure 5. EXEMPLE DE MODELE GENERIQUE DE FONCTION

Dans notre exemple, 3 flux d'objets entrent et 2 flux sortent. L'un des flux de sorties (le flux *p* en bleu) correspond au flux principal identifié lors de l'analyse fonctionnelle et sera important à prendre en compte lors de l'implémentation.

3) Implémentation

Dans notre exemple, la fonction générique de la Figure 5 est appelée 4 fois car l'on souhaite 4 flux de sorties différents : *a*, *b*, *c*, *d*. Nous avons donc étudié comment implémenter ce modèle générique dans un outil, et quelle stratégie appliquer afin de l'instancier. Dans le premier cas (Figure 6), le modèle générique est instancié 4 fois (manuellement), chaque fonction conservant la structure du modèle générique.

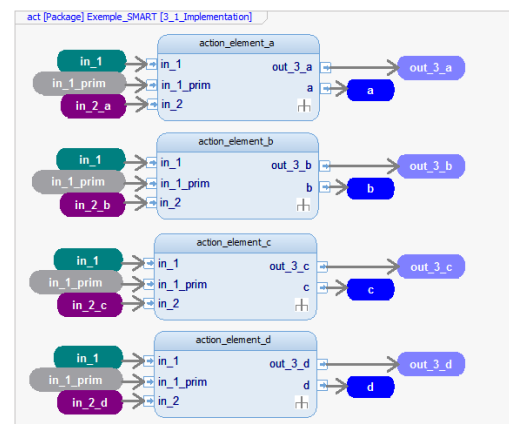


Figure 6. IMPLEMENTATION STRATEGIE 1

Dans le deuxième cas (Figure 7), le modèle générique est instancié 1 fois (manuellement), la fonction instanciée conservant la structure du modèle générique. Cependant, on duplique les ports d'entrées et de sorties afin d'obtenir les 4 flux de sorties. Encore une fois, la stratégie adoptée doit correspondre aux besoins de l'architecte.

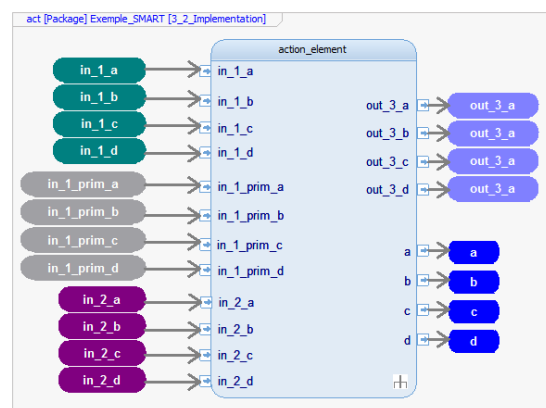


Figure 7. IMPLEMENTATION STRATEGIE 2

VI. CONCLUSION ET PERSPECTIVES

L'approche méthodologique MMI s'utilise de manière itérative et va permettre de venir enrichir une base de savoir-faire composée de patrons du système à faire. Cependant, la réutilisation de patrons nécessite la possibilité d'identifier, sélectionner, et appliquer des patrons de manière fluide afin que l'utilisateur puisse se concentrer sur les besoins lors du développement. Une approche agile doit être envisagée afin d'engager des cycles de minage, de maturation et d'implémentation qui soient courts.

Ainsi, dans nos futurs travaux concernant la réutilisation des patrons, nous allons développer une échelle qui permettra d'identifier des niveaux de maturité du processus de réutilisation d'une entreprise pour ainsi permettre de cibler les efforts à fournir. Cette échelle sera multiaxiale afin de couvrir les différents aspects du processus (modèles, capitalisation...). Cela permettra d'une part de quantifier des degrés de maturité qui seront spécifiques à certaines activités, et d'autre part un niveau de maturité global au niveau du processus qui dépendra du niveau sur chaque axe.

VII. REFERENCES

- [1] W. Schindel and T. Peterson, "An Overview of Pattern-Based Systems Engineering," in *INCOSE Enchantment Chapter Webinar*, 2014.
- [2] S. Aries, B. Le Blanc, and J. Ermine, "MASK : une méthode d'ingénierie des connaissances pour l'analyse et la structuration des connaissances," in *Management et ingénierie des connaissances, modèles et méthodes*, Hermes-Lavoisier, Ed. 2008, p. 208.
- [3] G. Schreiber *et al.*, *Knowledge Engineering and Management - The CommonKADS Methodology*, no. January. MIT Press, 2001.
- [4] L. Gzara, D. Rieu, and M. Tollenaere, "Product information systems engineering: An approach for building product models by reuse of patterns," *Robot. Comput. Integr. Manuf.*, vol. 19, no. 3, pp. 239–261, 2003.
- [5] L. A. Bollinger and R. Evins, "Facilitating model reuse and integration in an Urban Energy Simulation platform," *Procedia Comput. Sci.*, vol. 51, no. 1, pp. 2127–2136, 2015.
- [6] R. H. Barter, "A Systems Engineering Pattern Language," in *INCOSE International Symposium*, 1998, pp. 350–353.
- [7] R. J. Cloutier, "Model Driven Architecture for Systems Engineering," *INCOSE Int. Work.*, no. September, 2008.
- [8] D. Cook and W. Schindel, "Utilizing Mbse Patterns To Accelerate System Verification," *Insight*, vol. 20, no. 1, pp. 32–41, 2017.
- [9] N. Gautam, R. B. Chinnam, and N. Singh, "Design reuse framework : a perspective for lean development," *Int. J. Prod. Dev.*, vol. 4, no. 5, pp. 485–507, 2007.
- [10] C. Haskins, "Application of patterns and pattern languages to systems engineering," in *INCOSE International Symposium*, 2005, pp. 1619–1627.
- [11] A. Korff, "Re-using sysml system architectures," in *Proceedings of the 4th International Conference on Complex Systems Design and Management*, 2013, pp. 257–266.
- [12] G. Wang, R. Valerdi, and J. Fortune, "Reuse in systems engineering," *IEEE Syst. J.*, vol. 4, no. 3, pp. 376–384, 2010.
- [13] A. Majchrzak, L. P. Cooper, and O. E. Neece, "Knowledge Reuse for Innovation," *Manage. Sci.*, vol. 50, no. 2, pp. 174–188, 2004.
- [14] A. Rockley, P. Kostur, and S. Manning, *Managing Enterprise Content: A Unified Content Strategy*. New Riders, 2003.
- [15] D. C. Rine and R. M. Sonnemann, "Investments in reusable software. A study of software reuse investment success factors," *J. Syst. Softw.*, vol. 41, no. 1, pp. 17–32, 1998.
- [16] A. Le Put, *Systems Product Line Engineering Handbook*, AFIS. Éditions Cepaduès, 2016.
- [17] N. Niu, J. Savolainen, Z. Niu, M. Jin, and J.-R. C. Cheng, "A Systems Approach to Product Line Requirements Reuse," *IEEE Syst. J.*, vol. PP, no. 99, pp. 1–10, 2013.
- [18] F. Pfister, V. Chapurlat, M. Huchard, C. Nebut, and J.-L. Wippler, "A Proposed Meta-Model for Formalizing Systems Engineering Knowledge, Based on Functional Architecture Patterns," *Syst. Eng.*, vol. 15, no. 3, 2012.
- [19] Q. Wu, D. Gouyon, É. Levrat, and S. Boudau, "A Review of Know-How Reuse with Patterns in Model-Based Systems Engineering," in *Proceedings of the Ninth International Conference on Complex Systems Design & Management, CSD&M Paris 2018*, 2018, pp. 219–229.
- [20] C. Alexander, S. Ishikawa, and M. Silverstein, *A Pattern Language*. Oxford university press, 1977.
- [21] A. Gaffar and N. Moha, "Semantics of a Pattern System," *Proc. STEP Int. Work. Des. Pattern Theory Pract. IWDPTP05*, 2005.
- [22] A. Vogelsang, T. Amorim, F. Pudlitz, P. Gersing, and J. Philipps, "Should I Stay or Should I Go? On Forces that Drive and Prevent MBSE Adoption in the Embedded Systems Industry," 2017.
- [23] C. Oster, M. Kaiser, J. Kruse, J. Wade, and R. Cloutier, "Applying Composable Architectures to the Design and Development of a Product Line of Complex Systems," *Syst. Eng.*, vol. 19, no. 6, pp. 522–534, Nov. 2016.
- [24] G. Barbieri, K. Kernschmidt, C. Fantuzzi, and B. Vogel-Heuser, "A SysML based design pattern for the high-level development of mechatronic systems to enhance re-usability," in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 2014, vol. 19, no. 3, pp. 3431–3437.
- [25] L. Gasser, "Structuring activity diagrams," in *14th IFAC Symposium on Information Control Problems in Manufacturing, Bucharest, Romania*, 2012.
- [26] R. S. Kalawsky, D. Joannou, Y. Tian, and A. Fayoumi, "Using architecture patterns to architect and analyze systems of systems," *Procedia Comput. Sci.*, vol. 16, no. March 2015, pp. 283–292, 2013.
- [27] W. Schindel, "Requirements Statements Are Transfer Functions: An Insight from Model-Based Systems Engineering," in *INCOSE International Symposium*, 2005, pp. 1604–1618.
- [28] W. Schindel, S. A. Lewis, J. J. Sherey, and S. K. Sanyal, "Accelerating MBSE Impacts Across the Enterprise : Model-Based S * Patterns," in *INCOSE International Symposium*, 2015.
- [29] J. L. Bradley, M. T. Hughes, and W. Schindel, "Optimizing delivery of global pharmaceutical packaging solutions, using systems engineering patterns," *20th Annu. Int. Symp. Int. Counc. Syst. Eng. INCOSE 2010*, vol. 3, pp. 2441–2447, 2010.
- [30] M. Grundstein, "From Capitalizing on Company Knowledge to Knowledge Management," in *Knowledge management: Classic and Contemporary Works*, MIT, Ed. Cambridge (Mass), 2000, pp. 261–287.