



HAL
open science

Exigences d'interopérabilité: caractérisation et formalisation en propriétés

Sihem Mallek, Nicolas Daclin, Vincent Chapurlat

► **To cite this version:**

Sihem Mallek, Nicolas Daclin, Vincent Chapurlat. Exigences d'interopérabilité: caractérisation et formalisation en propriétés. Atelier INFORSID: Ingénierie des systèmes d'information, Apr 2010, Marseille, France. hal-00588826

HAL Id: hal-00588826

<https://hal.science/hal-00588826>

Submitted on 20 Jun 2023

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.

Exigences d'interopérabilité : caractérisation et formalisation en propriétés

Sihem Mallek, Nicolas Daclin, Vincent Chapurlat

*Laboratoire de Génie Informatique et d'Ingénierie de Production - LGI2P - site de
Nîmes de l'Ecole des Mines d'Alès - Parc Scientifique Georges Besse, 30035 Nîmes
cedex 1, France*

{Sihem.Mallek, Nicolas.Daclin, Vincent.Chapurlat}@mines-ales.fr

RÉSUMÉ. L'interopérabilité revêt un enjeu important pour l'industrie et peut être vue comme un des principaux freins à un travail collaboratif aussi bien inter entreprises que intra entreprise. Il paraît donc pertinent d'analyser et de détecter d'éventuels manques ou défauts d'interopérabilité dans l'organisation ou le comportement des entreprises impliquées dans un processus collaboratif. Ces travaux de recherche proposent, dans un contexte d'ingénierie de processus guidée par les modèles, d'utiliser des techniques de vérification formelle de modèles pour détecter différents types de problème liés à l'interopérabilité. Ceci induit, dans un premier temps, de pouvoir définir les besoins particuliers en interopérabilité devant être pris en compte dans un contexte collaboratif. Dans un second temps, il est nécessaire de formaliser ces besoins en un ensemble d'exigences d'interopérabilité, de manière aussi formelle que possible. Enfin, les exigences d'interopérabilité doivent être formalisées en propriétés afin de permettre leur vérification au moyen de techniques formelles. Cet article présente la définition et la formalisation des exigences d'interopérabilité sous forme de propriétés prouvables de manière formelle ou analysables selon d'autres techniques.

ABSTRACT. Interoperability is becoming a crucial issue for industry and can be seen as an important barrier to collaborative work. Indeed, interoperability characterises the ability of any enterprises to interact within a collaborative process. Prior to any effective collaboration, enterprises that aim to work together want to know if they are able to interoperate. This research work proposes to use formal verification techniques to detect interoperability problems. On the one hand, this induces to be able to define the particular needs to consider. On the other hand, it requires to formalise these needs as a set of unambiguous and, as formal as possible, requirements. Finally, interoperability requirements must be formalized into properties with formal language in order to make their verification possible using formal techniques. This paper focuses on the definition and the formalisation of interoperability requirements into interoperability properties.

MOTS-CLÉS : Exigences d'interopérabilité, compatibilité, interopération, réversibilité, vérification, propriétés, processus collaboratif.

KEYWORDS: Interoperability requirements, compatibility, interoperation, reversibility, verification, properties, collaborative process.

1. Introduction

Dans le contexte de mondialisation actuel, les entreprises se trouvent impliquées de plus en plus dans des projets communs, ce qui leur permet de se concentrer sur leurs savoir-faire, leur cœur de métier et d'améliorer notablement leur positionnement et leur compétitivité face à des marchés devenus instables et plus complexes à appréhender isolément. Pour évoluer sereinement dans ce contexte les entreprises doivent réfléchir, améliorer et optimiser leurs capacités à interagir avec d'autres partenaires. Ainsi, l'interopérabilité est devenue un enjeu important pour les entreprises dans ce contexte de collaboration (European Community report, 2008), (Cabrel *et al.*, 2006). L'interopérabilité est définie dans (IEEE, 1990) comme : « *l'habilité de deux (ou plus) systèmes ou composants à échanger des informations et à utiliser les informations échangées* ». Dans cet article, l'interopérabilité est envisagée comme une exigence que chaque entreprise doit être en mesure de satisfaire. Dans le cas où cette exigence n'est pas satisfaite, l'entreprise peut se poser des questions sur d'éventuelles actions d'amélioration qu'elle doit alors entreprendre. Cette exigence est formalisée classiquement à partir de l'expression de besoins relatifs à cette aptitude d'interopérabilité. Ces besoins, exprimés généralement par des responsables au sein de l'entreprise en langage naturel, peuvent être ambigus, non cohérents, incomplets et ne sont pas forcément vérifiables de manière tangible, d'où la nécessité de les formaliser en exigences. Il est donc nécessaire, d'une part, de s'appuyer sur des modèles de processus d'entreprise, et d'autre part, de procéder à un travail de formalisation et de construction de cette exigence afin d'analyser ces modèles. Cette analyse permet de détecter, de manière anticipative, d'éventuelles lacunes ou problèmes liés au non respect de cette exigence. Le but est donc de mettre en évidence les problèmes d'interopérabilité pour pouvoir les solutionner localement par la suite. Dans ce cadre, cet article présente une approche de formalisation et d'analyse de l'exigence d'interopérabilité. Les besoins en interopérabilité, présentés en section 2, sont structurés puis exprimés au moyen d'un modèle d'exigences dont les concepts sont présentés en section 3. Ces exigences sont d'abord exprimées à l'aide de termes et de concepts provenant du langage naturel. Elles sont ensuite formalisées dans la section 4 en propriétés au moyen d'un langage formel pour permettre leur vérification *via* des techniques de vérification formelle. Cette approche est enfin appliquée en section 5 sur un modèle de processus collaboratif (modélisé avec le langage BPMN enrichi (BPMN, 2009)).

2. Besoins d'interopérabilité

Plusieurs travaux concernant l'amélioration de l'aptitude et de la capacité d'une entreprise à être interopérable existent dans la littérature. Notons, par exemple, l'existence de modèles de maturité liés à l'interopérabilité (LCIM, OIM, LISI, EIMM...), (Tolk *et al.*, 2003), (Clark *et al.*, 1999), (C4ISR, 1998), (ATHENA, 2005) qui mettent en évidence différents niveaux de maturité caractérisant ces

capacités et donnant des recommandations pour pouvoir évoluer d'un niveau vers le suivant. Ces modèles permettent donc d'évaluer un niveau de maturité et de percevoir des voies d'amélioration possibles pour atteindre le niveau de pleine maturité. De même, une méthodologie a été développée dans (Daclin *et al.*, 2008) pour mesurer l'interopérabilité des entreprises. Cette mesure inclut une mesure de compatibilité et une mesure de performance de l'interopération c'est-à-dire la phase durant laquelle les acteurs sont impliqués dans une action commune pour remplir une mission identifiée. La mesure de la compatibilité s'effectue à l'aide d'une matrice de compatibilité pour informer les entreprises de la présence (ou pas) de problèmes d'interopérabilité. Pour la mesure de la performance d'interopération, des critères liés à l'interopération (coût, temps, et qualité) sont utilisés. Pour évaluer la performance de l'interopérabilité des entreprises, une caractérisation de l'interopérabilité est proposée dans (Blanc *et al.*, 2007). Cette caractérisation identifie cinq « règles d'interopérabilité ». Cependant, ces règles sont génériques et difficiles à exploiter (*e.g.* la première explique qu'il doit exister une boucle de retour entre deux systèmes). Afin de structurer ces travaux de recherche, ceux-ci ont été intégrés dans le cadre d'interopérabilité développé dans le Réseau d'Excellence INTEROP (INTEROP, 2007). Ce cadre tient compte des deux aspects fondamentaux pour traiter les besoins d'interopérabilité : les barrières d'interopérabilité et les niveaux d'interopérabilité. L'objectif est de parvenir à une collaboration efficace en levant les barrières d'interopérabilité (conceptuelle, technologique et organisationnelle) pour chaque niveau d'interopérabilité (business, processus, services, et données).

A partir de ces différents exemples de travaux de recherche, un premier référentiel des besoins en interopérabilité peut être identifié. Cependant, afin de consolider ces besoins, et pour enrichir ce premier référentiel, une enquête auprès d'industriels est en cours d'exploitation.

Ces besoins, exprimés en langage naturel, peuvent être difficiles à manipuler de par la nature et la complexité de l'expression des besoins eux mêmes. Il s'agit donc, dans un premier temps, d'exprimer ces besoins de manière claire et non ambiguë. Ceux-ci sont donc formalisés par des exigences exprimés également en langage naturel mais épurées de tout blanc, redite (objective ou subjective) et ambiguïté, pour obtenir un référentiel d'exigences cohérent et consistant.

3. Exigences d'interopérabilité : catégorisation et modèle

Comme mentionné précédemment, l'interopérabilité est vue comme une exigence globale, pouvant être raffinée en exigences plus simples, que des systèmes en interaction doivent satisfaire pour être les plus efficaces possible durant cette interaction. Une exigence est définie comme : « *un énoncé qui prescrit une fonction, une aptitude ou une caractéristique à laquelle doit satisfaire un produit ou un système dans un contexte donné* » (Scukanec *et al.*, 2008).

Cette première opération de raffinement fait apparaître trois classes d'exigence. Tout d'abord, l'interopérabilité est généralement liée à la notion de *compatibilité*. La compatibilité signifie harmoniser les systèmes (comportement, organisation, méthode de travail...) entre eux. Par exemple des données hétérogènes échangées doivent être comprises et exploitées sans effort d'interfaçage important. Cependant, le développement de l'interopérabilité ne se limite pas uniquement à la notion de compatibilité. En effet, pendant une collaboration des problèmes d'autres natures, liés essentiellement à la dynamique de la collaboration, peuvent apparaître. Par exemple, certains moyens ou ressources sont indisponibles ou s'avèrent incapables de mener à bien leur mission dans les conditions actuelles. De fait, « *deux systèmes interopérables sont compatibles mais deux systèmes compatibles ne sont pas nécessairement interopérables* » (Kasunic *et al.*, 2004). Cette notion dynamique de l'interopérabilité au cours de l'interaction, i.e. lors des échanges, est plus connue sous le nom d'interopération.

Enfin, à l'issue d'une collaboration, chaque partenaire souhaite en général reprendre son autonomie et retrouver ses performances originelles. Cependant, toute collaboration peut entraîner, consciemment ou non, de légères perturbations voire des modifications plus ou moins profondes dans le système (organisation, comportement, modes de décision et de gouvernance, ...). Ces modifications surgissent pendant l'interaction pour que les entreprises puissent s'adapter voire anticiper les événements. Ce souhait de retour à l'autonomie, et à une certaine idée de leur performance passée, peut donc être vu comme un besoin par le partenaire. Celui-ci s'intéressera donc à des exigences de *réversibilité* (INTEROP, 2007). La réversibilité représente la capacité et l'aptitude d'un système (entreprise, équipe, ...) à pouvoir retourner à son état d'origine de performance et d'autonomie décisionnelle à la fin d'une collaboration, ou tout au moins à parvenir à un état stable et acceptable en termes de performance et de gouvernance.

En résumé, les exigences d'interopérabilité peuvent être classées en :

– exigence de compatibilité est définie par : « *un énoncé qui prescrit une fonction, une aptitude ou une caractéristique, indépendant du temps et lié aux barrières d'interopérabilité pour chaque niveau d'interopérabilité, que les entreprises doivent satisfaire avant toute collaboration* ».

– exigence d'interopération est définie par : « *un énoncé qui prescrit une fonction, une aptitude ou une caractéristique, dépendant du temps et lié à la performance de l'interaction, que les entreprises doivent satisfaire lors de la collaboration* ».

– exigence de réversibilité est définie par : « *un énoncé qui prescrit une fonction, une aptitude ou une caractéristique, indépendant du temps et lié à la capacité qu'a l'entreprise à reprendre son autonomie et à retourner à son état d'origine (en terme de ses propres performance), que l'entreprise doit satisfaire à la fin de la collaboration* ».

A titre d'illustration, ces trois classes d'exigence sont cohérentes avec le cycle de vie de l'entreprise virtuelle (c.f. figure 1) qui met en avant les étapes de création, opération et dissolution (Camarinha-Matos *et al.*, 2003) pendant lesquelles le respect d'exigences d'interopérabilité joue un rôle important. En effet, deux entreprises qui désirent collaborer ensemble doivent, (1) créer la collaboration (prise en compte des exigences de compatibilité), (2) collaborer efficacement (prise en compte des exigences d'interopération), et (3) arrêter la collaboration (prise en compte des exigences de réversibilité).

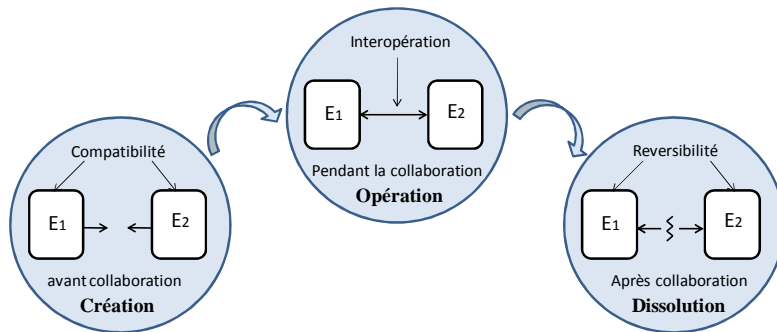


Figure 1. Les trois catégories d'exigences d'interopérabilité liés au cycle de vie de l'entreprise virtuelle [adaptée de (Camarinha-Matos *et al.*, 2003)]

Cependant, de par le nombre, le niveau d'abstraction requis ou la complexité inhérente de certaines exigences, il est nécessaire de disposer d'un modèle d'exigence structurant et guidant l'acteur humain dans sa spécification. L'objectif est de pouvoir raffiner des exigences, de les manipuler, et enfin de facilement les formaliser à des fins d'analyse sur un modèle du système visé (un processus collaboratif, des entreprises, des équipes, ...). Le modèle d'exigence repose sur le principe de causalité : une exigence se raffine à partir de la combinaison d'exigences considérées comme des causes primaires. Le résultat est un arbre causal orienté (c.f. figure 2) qui permet de structurer et de raffiner les exigences d'interopérabilité.

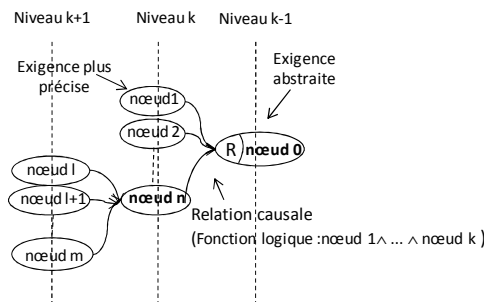


Figure 2. Raffinement des exigences à l'aide d'un arbre causal

Chaque nœud représente une exigence d'un niveau d'abstraction ou de causalité donné. Les arcs liants un nœud d'un niveau d'abstraction à d'autres nœuds de niveaux d'abstraction moins élevées représentent la relation de raffinement sous forme d'une relation causale entre ces nœuds. A titre d'exemple, le modèle (partiel) des exigences d'interopérabilité résultant est représenté en figure 3. Le nœud racine (ellipse grise) représente l'exigence d'interopérabilité globale la plus abstraite. Exprimer et prouver cette exigence revient à exprimer et à prouver la capacité et l'aptitude d'une entreprise ou de partenaires à pouvoir interopérer. Ce nœud racine est raffiné par les trois catégories d'exigences présentées précédemment. Chaque catégorie est raffinée, par la suite, à l'aide des concepts introduits par le cadre développé dans le NoE INTEROP (barrières d'interopérabilité et niveaux d'interopérabilité) (INTEROP, 2007). Par choix, chaque catégorie est raffinée par les quatre niveaux d'interopérabilité (business, processus, services et données) qui sont eux même raffinés par les trois barrières d'interopérabilité. L'arbre causal est ensuite complété avec des exigences extraites des travaux présentés en section 2 et des résultats de l'enquête effectuée auprès d'industriels.

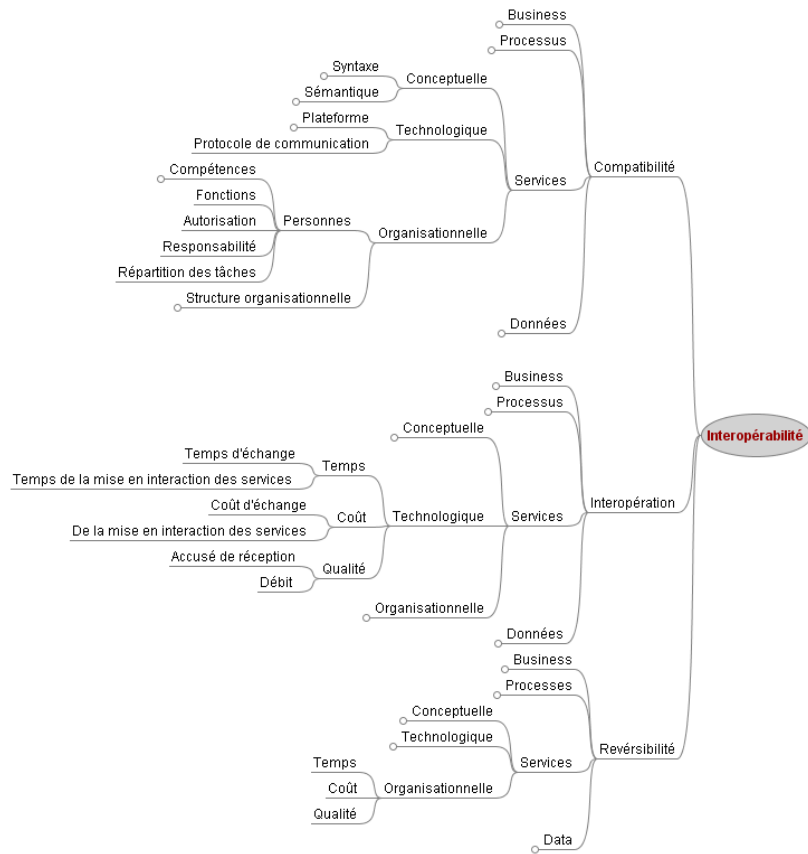


Figure 3. Vue partielle de l'arbre causal des exigences d'interopérabilité

Ce principe de raffinement permet donc de décomposer une exigence abstraite d'un niveau k sous forme d'un ensemble d'exigences de niveau d'abstraction moindre $k+n$. La relation qui lie un nœud cible (niveau k) à ses nœuds causes appelés aussi sources (niveau $k+1$) est de type implication, équivalence ou influence. Elle est ensuite contrainte d'une fonction logique permettant d'exprimer comment les nœuds causes sont associés et pris en compte dans le raffinement du nœud cible.

A titre d'exemple, pour l'arbre causal présenté en figure 3, les fonctions logiques utilisées, et qui sont à la base des principes d'interprétation de l'arbre, sont les suivantes :

$$\text{Interopérabilité} = \text{Compatibilité} \wedge \text{Interopération} \wedge \text{Réversibilité} \quad [1]$$

$$\text{Compatibilité} = \text{Business} \wedge \text{Processus} \wedge \text{Services} \wedge \text{Données} \quad [2]$$

$$\text{Services} = \text{Conceptuelle} \wedge \text{Technologique} \wedge \text{Organisationnelle} \quad [3]$$

$$\text{Conceptuelle} = \text{Syntaxe} \wedge \text{Sémantique} \quad [4]$$

En d'autres termes, pour atteindre la pleine interopérabilité, les exigences de compatibilité, d'interopération et de réversibilité doivent être respectées comme indiqué par l'équation [1]. Pour que l'exigence de compatibilité soit respectée, il faut que les exigences reliées aux niveaux d'interopérabilité soient à leur tour respectées comme représenté sur l'équation [2]. Les exigences de chaque niveau d'interopérabilité sont respectées si et seulement si chaque barrière d'interopérabilité est respectée à ce niveau comme le montre l'équation [3] avec le niveau services. Enfin, les barrières d'interopérabilité sont respectées comme le montre l'équation [4] qui se focalise ici seulement sur les concepts liés à la barrière conceptuelle. Les exigences de chaque niveau peuvent être analysées séparément. Cette analyse, indiquant la satisfaction ou non d'une exigence, repose sur l'usage de mécanismes de preuve présentés en figure 5 (statique ou dynamique) ou sur l'expertise.

Enfin, une exigence peut être caractérisée par une hypothèse temporelle. Elle peut être statique, c'est-à-dire considérée comme indépendante du temps ou vraie à chaque instant. Elle peut être dynamique, c'est-à-dire devant être vérifiée seulement à certaines phases du cycle de vie de la collaboration. Le choix des techniques de vérification alors employées sera différent comme proposé dans la section 4.

D'une manière plus formelle, l'arbre causal présenté en figure 3 peut être défini, comme représenté en figure 4, en adoptant les conventions d'écriture suivantes :

m = nombre de nœuds (N) de l'arbre

n = nombre d'arcs (E) de l'arbre

T représente l'ensemble des instants de l'échelle temporelle choisie. Par hypothèse, $T = \mathbb{N}$ dans le cas d'une échelle de temps discret ou $T = \mathbb{R}$ dans le cas d'une échelle de temps continu.

Un arbre causal est définie par: $G ::= (N, E, N_v)$

- $N = \{N_i / i \in [0, m] ; N_i ::= (\text{nom}_i, \text{description}_i, \text{modèle}_i, \text{relation}_i, \text{fait}_i, \text{niveau}_i)\}$
- $\exists ! N_0 = (\text{nom}_0, \text{description}_0, \text{modèle}_0, \text{relation}_0) \in N$ appelé racine. Il correspond à l'exigence globale d'interopérabilité la plus abstraite qui soit
- $E = \{E_j / j \in [0, n] ; E_j ::= (\text{SourceN}, \text{CibleN}) / (\text{SourceN}, \text{CibleN}) \in N \times N \text{ et } \text{SourceN} \neq \text{CibleN} \text{ et } \text{niveau}_{\text{SourceN}} > \text{niveau}_{\text{CibleN}}\}$

Où:

- $(\text{nom}, \text{description}) \in \text{String} \times \text{String}$
- $\text{modèle} ::= (\text{modèle du processus})$
- $\text{fait} = \{\text{fait est extrait du modèle et/ou du méta modèle: variables de modélisation}\}$
- $\text{niveau} \in \mathbb{N}$
- $\text{relation} ::= (\text{type}, \text{quantificateur}, T, \theta_e, \theta_c)$ avec :
 - $\text{Type} \in \{\text{"équivalence"}, \text{"influence"}, \text{"implication"}\}$
 - $\text{quantificateur} \in \{\text{"\forall"}, \text{"\exists"}\}$
 - $T \subseteq \mathcal{T} / \text{Set of possible moments}$
 - $\theta_e : \text{Fait} \times \mathcal{N}_{N_i} \rightarrow \{0, 1\}$
 - $\theta_c : \mathcal{N}_{N_i} \rightarrow \{0, 1\}$

avec \mathcal{N}_{N_i} l'ensemble de nœuds sources tel que:

$$\mathcal{N}_{N_i} = \{N_j \in N / j \in [0, m], j \neq i, \exists E_k (N_i, N_j) / k \in [0, n]\}$$

- $N_v = \{N_j / j \in [0, m] ; N_j \cdot \text{relation} \cdot \text{quantificateur} \cdot (\theta_c = 1 \Rightarrow \theta_e = 1)\}$

est l'ensemble de nœuds vérifiés

Figure 4. Représentation de la formalisation d'un arbre causal

Le principe d'interprétation de cet arbre en phase d'analyse est le suivant. Pour qu'une exigence soit respectée, toutes les exigences qui la raffinent doivent être respectées en tenant compte de la fonction logique qui lie une exigence abstraite à des exigences plus précises. Certaines de ces exigences peuvent être formalisées pour permettre d'en assurer la vérification par preuve formelle. Pour cela, un langage de propriétés est utilisé comme présenté dans la suite.

4. Propriétés d'interopérabilité : formalisation

Différents mécanismes sont mis en avant dans ces travaux pour vérifier les exigences d'interopérabilité sur un modèle de processus collaboratif (*c.f.* figure 5). Il s'agit de techniques de vérification formelle (model checking et theorem proving) (Edmund *et al.*, 1999), (Bérard *et al.*, 2001), (Duffy, 1991), d'une technique basée sur l'expertise par un acteur humain ou d'une technique basée sur la simulation. Les deux techniques de vérification formelle sont, d'une part, basée sur les graphes conceptuels pour la vérification d'exigences statiques (Roque *et al.*, 2009) et, d'autre part, une technique basée sur le model checking pour les exigences dynamiques (Behrmann *et al.*, 2004). Les techniques d'expertise et de simulation sont enfin

utilisées pour analyser des exigences difficilement formalisables ou non vérifiables sur le modèle.

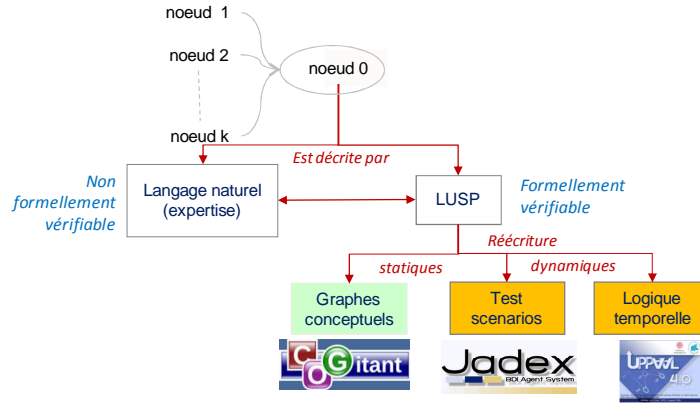


Figure 5. Mécanismes utilisés pour la vérification des exigences d'interopérabilité

Pour permettre la mise en œuvre de techniques de vérification formelle, les exigences d'interopérabilité sont formalisées en propriétés via un langage de spécification de propriétés. Pour cela, le langage LUSP a été choisi (Lamine, 2001), (Chapurlat, 2007). Il permet d'avoir un langage généraliste et ouvert. Ce langage décrit une propriété à savoir « une connaissance que l'on a du système ou d'un modèle qui traduit une exigence, une finalité ou une caractéristique à satisfaire » et propose un modèle de propriété sous forme d'une relation causale, typée et contrainte entre deux ensembles de faits extraits d'un modèle. A titre d'exemple, et afin d'analyser un modèle de processus décrit au moyen d'une extension de BPMN, la figure 6 montre trois exigences d'interopérabilité pour chaque catégorie formalisées avec le langage LUSP.

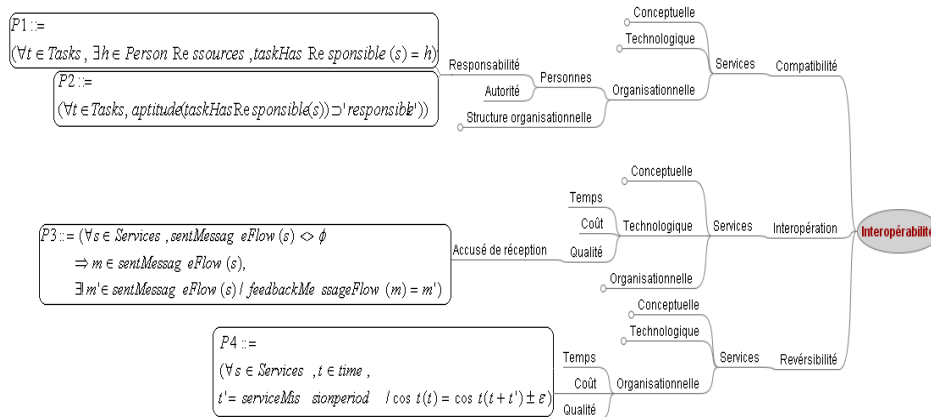


Figure 6. Propriétés d'interopérabilité structurées dans un arbre causal

L'exigence de compatibilité au niveau des services pour la barrière organisationnelle et nommée « *responsabilité* » est décrite par : « *un service doit avoir un responsable clairement défini* ». Cette exigence est formalisée par les deux propriétés P1 et P2. De ce fait, la relation liant l'exigence aux propriétés est définie telle que : $\text{responsabilité} = P1 \wedge P2$.

L'exigence d'interopération au niveau service pour la barrière technologique et nommée « *accusé de réception* » est décrite par : « *pour chaque donnée reçue, un accusé de réception doit être renvoyé* ». Cette exigence est formalisée par la propriété P3.

L'exigence de réversibilité au niveau service pour la barrière organisationnelle et nommée « *coût* » est décrite par : « *le coût d'un service donné après la collaboration doit correspondre au coût d'avant la collaboration en considérant une variation acceptable* ». Cette exigence est formalisée par la propriété P4.

Les propriétés P1, P2 et P3 peuvent être vérifiées sur le modèle fourni par le langage BPMN enrichi. Cependant, la propriété P4, ne peut pas être vérifiée sur le modèle, car les concepts liés à cette propriété ne sont pas pris en compte dans le modèle, elle sera donc destinée à l'expertise.

Pour illustrer l'utilisation du modèle d'exigence d'interopérabilité, un cas d'application est présenté dans la section suivante.

5. Cas d'application

Ce cas d'application s'appuie sur un processus collaboratif issu de (Singular Enterprise, 2003). Il représente la collaboration entre un détaillant et un fournisseur. Le département vente du fournisseur souhaite collaborer avec un détaillant externe et indépendant chargé de fournir des produits aux clients. Les différentes activités et flux entre les partenaires (détaillant et fournisseur) sont modélisés avec le langage BPMN enrichi comme représenté en figure 7 (où l'objet de données « *règles de distribution* » agit comme un contrôle sur la tâche « *contrôler quantité* »). L'objectif est de détecter, grâce au modèle d'exigences d'interopérabilité, si des problèmes de compatibilité, d'interopération ou de réversibilité peuvent apparaître. De ce fait, les exigences présentées précédemment et formalisées en propriétés peuvent être vérifiées sur ce processus collaboratif.

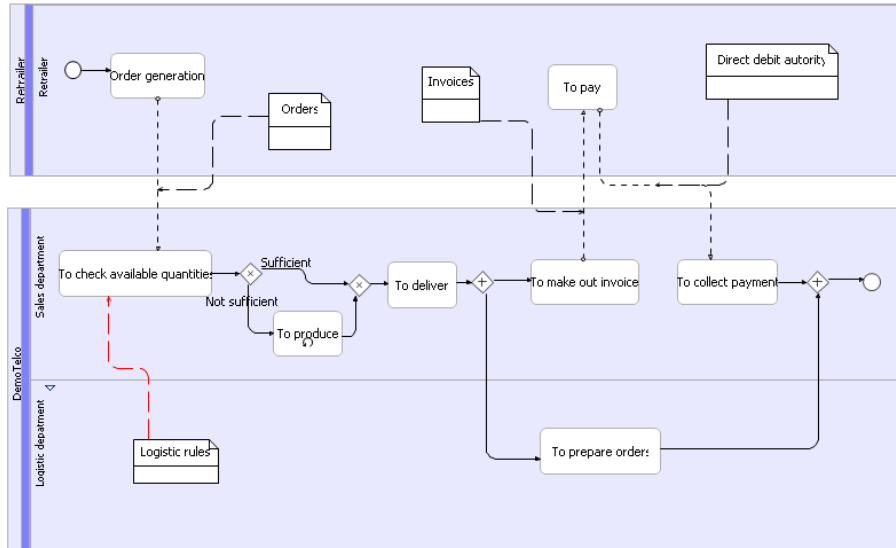


Figure 7. Processus collaboratif entre un fournisseur et un détaillant

Par exemple, avant que le détaillant n'envoie sa commande, l'exigence de compatibilité précédemment décrite par : « *un service doit avoir un responsable clairement défini* » et formalisée par les propriétés P1 et P2 doit être vérifiée. La non vérification de cette exigence entraîne la non vérification de l'exigence racine baptisée « interopérabilité ». Il s'agit pour le détaillant et DemoTelco d'être vigilant quant à une détérioration possible de la collaboration (perte de temps pour localiser le bon responsable).

Pendant la collaboration, il est possible de vérifier l'exigence d'interopération décrite par : « *pour chaque donnée reçue, un accusé de réception doit être renvoyé* » en procédant à la vérification de la propriété P3. Par exemple, la réception de factures induit l'envoi d'un accusé de réception par le détaillant, pour assurer le paiement de la facture par le détaillant.

Enfin, à la fin de la collaboration, l'exigence réversibilité décrite par : « *le coût d'un service donné après la collaboration doit correspondre au coût d'avant la collaboration en considérant une variation acceptable* » ne pourra pas être vérifiée sur le modèle présenté en figure 6, elle sera donc destinée à l'expertise.

Le choix de devoir vérifier à tous les coups chacune des exigences relève d'une stratégie propre à chaque utilisateur. En effet, une exigence non satisfaite ne signifie pas que le système n'est pas interopérable. Cela peut révéler une erreur de modélisation dans le cas où le modèle du système étudié n'a pas été au préalable vérifié et validé. De plus, un utilisateur peut choisir et imposer qu'une exigence soit

satisfaite par défaut en appliquant sa propre expertise, ou qu'une exigence non vérifiée n'entraîne pas nécessairement une détérioration de la collaboration

6. Conclusion

L'interopérabilité prend, de nos jours, une place importante pour les entreprises voulant évoluer de façon pérenne dans un environnement de globalisation. Les entreprises cherchent à trouver leurs problèmes en interopérabilité pour les résoudre et arriver à une collaboration efficace. Par conséquent, la formalisation et la vérification des exigences d'interopérabilité s'avère être une solution pour aider les entreprises à identifier leurs problèmes d'interopérabilité. Cet article s'articule autour des définitions des exigences d'interopérabilité et de leur formalisation en propriétés avec le langage LUSP. Pour enlever l'ambiguïté pouvant exister, les exigences sont structurées sur un arbre causal permettant le raffinement des exigences abstraites en exigences plus claires. Ces exigences sont par la suite formalisées en propriétés pour permettre leur vérification.

Les perspectives de ces travaux concernent d'abord l'outillage destiné à la vérification des exigences d'interopérabilité *via* des techniques formelles dépendant de la nature des exigences (statiques ou dynamiques). Ensuite, la stratégie de vérification de l'utilisateur qui peut alterner l'usage de techniques de preuves ou usage d'expertise, impose de s'assurer de la traçabilité (gestion des conflits, suivi et mémorisation, résultats de vérification) de toutes les exigences (compatibilité, interopération et réversibilité) tout au long de la collaboration. Cela représente une autre perspective d'ouverture de ce travail qui est de proposer des solutions pour remédier aux problèmes d'interopérabilité constatés grâce à l'approche proposée.

7. Bibliographie

- IEEE, « A compilation of IEEE standard computer glossaries », *Standard computer dictionary*, New York, 1990.
- BPMN, « Business Process Modeling Notation, V1.2 », 2009. <http://www.bpmn.org/>,
- Tolk A., Muguira J.A., « The Levels of Conceptual Interoperability Model », *Proceedings of Fall Simulation Interoperability Workshop (SIW)*, Orlando, USA, 2003.
- Clark T., Jones R., « Organisational Interoperability Maturity Model for C2. », *Proc of Command and Control Research & Techn. Symposium*, Newport, USA, 1999.
- C4ISR, « Levels of Information Systems Interoperability (LISI) », Architecture Working Group, United States of America, Department of Defence, 1998.
- ATHENA, « Framework for the establishment and management methodology », Integrated Project ATHENA, deliverable A1.4, 2005.

- Daclin N., Chen D., Vallespir B., « Methodology for enterprise interoperability », *17th IFAC World Congress (IFAC'08)*, Seoul, Corée, 6-11 juillet 2008.
- Blanc S., Ducq Y., Vallespir B., « A Graph based approach for interoperability evaluation », *Interoperability for Enterprise Software and applications – IESA 2007*, Madeira, Portugal, 2007.
- INTEROP, « Enterprise Interoperability-Framework and knowledge corpus - Final report, INTEROP NoE, FP6 – Contract n° 508011 », Deliverable DI.3, 21 mai 2007.
- Scucanec S. J., Van Gaasbeek J. R., « A day in the life of a verification requirement. », *U.S Air Force T&E Days*, Los Angeles, Californie, février 2008.
- Kasunic M., Anderson W., « Measuring systems interoperability: challenges and opportunities », *Software engineering measurement and analysis initiative*, Technical note CMU/SUE-2004-TN-003, 2004.
- V.Chapurlat, Vérification et validation de modèles de systèmes complexes: application à la Modélisation d'Entreprise, habilitation à diriger des recherches de l'Université des Sciences et Techniques du Languedoc (USTL) Montpellier II, 2007.
- Camarinha-Matos L. M., Afsarmanesh H., Rabelo R. J., « Infrastructure developments for agile virtual enterprises », *Journal of Computer Integrated Manufacturing*, ISSN 0951-192X, Vol. 16, N. 4-5, juin-Août 2003.
- Roque M., Chapurlat V., « Interoperability in collaborative processes: Requirements characterisation and proof approach », *PRO-VE'09, 10th IFIP Working Conference on VIRTUAL ENTERPRISES*, Thessaloniki, Grèce, 7-9 Octobre 2009.
- Behrmann G., David A., Larsen K. G., A tutorial on Uppaal. Department of Computer Science, Aalborg University, Denmark, 2004.
- Lamine E., Définition d'un modèle de propriété et proposition d'un langage de spécification associé : LUSP, PhD thesis, Université de Montpellier 2, Décembre 2001.
- Singular Enterprise, Implementation methodology – Blue Print – Demo TelCo, 2003.
- Edmund M. Clarke Jr., Orna Grumberg, Doron A. Peled, Model checking, The MIT Press, 1999.
- Bérard B., Bidoit M., Finkel A., Laroussinie F., Petit A., Petrucci L., Schnoebelen Ph., McKenzie P., Systems and Software verification: model checking techniques and tools, Springer, 2001.
- Duffy David A., Principles of Automated Theorem Proving, John Wiley & Sons, 1991.
- European Community report, « Unleashing the potential of the European Knowledge Economy, Value Proposition for Enterprise Interoperability », Final Version (Version 4.0), 21 janvier 2008. (http://cordis.europa.eu/fp7/ict/enet/ei/ei-issg_en.htm)
- Cabrel R., Doumeingts G., Li M. S., Popplewell K., « Enterprise Interoperability Research Roadmap », Version 4.0, 2006.