



HAL
open science

FORMALISATION ET EVALUATION DES EXIGENCES NON FONCTIONNELLES POUR L'INGENIERIE DES SYSTEMES COLLABORATIFS

Nicolas Daclin

► **To cite this version:**

Nicolas Daclin. FORMALISATION ET EVALUATION DES EXIGENCES NON FONCTIONNELLES POUR L'INGENIERIE DES SYSTEMES COLLABORATIFS. Autre. Université de Montpellier II, 2017. tel-01928433v2

HAL Id: tel-01928433

<https://hal.science/tel-01928433v2>

Submitted on 7 Nov 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.

HABILITATION A DIRIGER DES RECHERCHES DE L'UNIVERSITÉ DE MONTPELLIER

École doctorale Informatique, Structure et Systèmes (I2S)

FORMALISATION ET EVALUATION DES EXIGENCES NON FONCTIONNELLES POUR L'INGENIERIE DES SYSTEMES COLLABORATIFS

Présentée par Nicolas Daclin

Docteur de l'Université de Bordeaux

Maître Assistant au Laboratoire de Génie Informatique et d'Ingénierie de la Production de
l'Ecole des Mines d'Alès

Le 1^{er} décembre 2017

Devant le jury composé de

M. Bernard Archimède, Professeur, Ecole Nationale d'Ingénieurs de Tarbes

Rapporteur

M. Alain Bernard, Professeur, Ecole Centrale de Nantes

Rapporteur

M. Yves Ducq, Professeur, Université de Bordeaux

Rapporteur

M. Frédéric Benaben, Professeur, IMT - Mines Albi-Carmaux

Examineur

M. Didier Crestani, Professeur, Université de Montpellier

Examineur

M. Hervé Pingaud, Professeur, Institut National Universitaire Champollion

Examineur

M. Vincent Chapurlat, Professeur, IMT - Mines Alès

Président



UNIVERSITÉ
DE MONTPELLIER

Contenu

Avant-Propos.....	7
I – PARTIE ADMINISTRATIVE.....	9
1. Curriculum vitae	10
1.1. Etat Civil.....	10
1.2. Position.....	10
1.3. Parcours.....	10
1.4. Formation	10
2. Activités de recherche.....	11
2.1. Publications/contributions.....	11
2.2. Projets	11
2.3. Encadrements.....	12
2.4. Autres activités de valorisation	13
3. Activités d’enseignements et encadrements.....	14
3.1. Résumé.....	14
3.2. Détails des enseignements et encadrements pour une année type.....	14
3.3. Responsabilités pédagogiques	15
3.4. Autres	16
II – PUBLICATIONS	17
1. Revues	18
2. Chapitres d’ouvrage collectif.....	19
3. Conférences.....	19
4. Workshops.....	22
5. Rapports de recherche	24
6. Groupes de travail & autres	25
III – RECHERCHE.....	27
1. Introduction.....	28
2. Problématique.....	30
2.1. Des réseaux collaboratifs.....	30
2.2. ...Vers l’Ingénierie des Systèmes de Systèmes	33
2.3. De l’interopérabilité des systèmes... ..	37
2.4. ... A l’exigence d’interopérabilité.....	40
2.5. Conclusion	49

3. Formalisation et évaluation de l'exigence d'interopérabilité dans les systèmes collaboratifs	51
3.1. Introduction.....	51
3.2. Dimension évaluation de l'exigence d'interopérabilité	52
3.3. Dimension expression et gestion des exigences d'interopérabilité.....	67
3.4. Dimension analyse de l'impact de l'interopérabilité	78
3.5. Conclusion	87
4. Programme de recherche.....	88
4.1. Vers une amélioration des techniques de formalisation et d'évaluation des exigences.....	88
4.2. Vers un écosystème des exigences non fonctionnelles « -ilities ».....	89
Bibliographie	95
Annexes	103
1. Détails des collaborations dans le cadre de projets.....	105
2. Détails des travaux de thèse et post doctoraux encadrés	107
3. Détails des enseignements et encadrements	109
4. Liste des publications jointes au mémoire	112

Liste des Figures

<i>Figure 1. Niveau et implication d'entités collaboratives selon [BEN 12] (gauche) et [CAM 09] (droite)</i>	32
<i>Figure 2. Système Complexe (ex. avion) et Système de Systèmes (ex. système de transport aérien, système de transport multimodal) [BLA 11]</i>	34
<i>Figure 3. Cadre d'interopérabilité d'entreprise (EIF – Enterprise Interoperability Framework, [ISO 09])</i>	39
<i>Figure 4. Modèle de maturité pour l'interopérabilité des entreprises MMEI (description des niveaux de maturité et définition du niveau 3) [GUE 12]</i>	46
<i>Figure 5. Exemple d'exigence en langage naturel traduite en graphe conceptuel [ALO 07]</i>	47
<i>Figure 6. Travaux pour le traitement de la problématique de formalisation et d'évaluation de l'exigence d'interopérabilité</i>	51
<i>Figure 7. Vers l'enrichissement conceptuel pour l'intégration des concepts de l'interopérabilité au langage de modélisation de processus et la transformation de modèle pour de l'évaluation d'une exigence</i>	53
<i>Figure 8. Extensions conceptuelles du langage BPMN (ajout des classes Ressource, Aptitude, Capacité... et des attributs sur l'objet Tâche)</i>	54
<i>Figure 9. Principe transformation d'un modèle de processus BPMN vers un graphe conceptuel</i>	55
<i>Figure 10. Vue générale du déploiement des graphes conceptuels pour l'évaluation de l'exigence d'interopérabilité sur des modèles de processus collaboratifs</i>	56

Figure 11. Vers un modèle comportemental du modèle de processus collaboratif pour l'utilisation de la technique du « model checking » et la structuration des exigences au sein d'un référentiel	58
Figure 12. Modèles à états pour la représentation d'une tâche (a – Rodde, b – Trinquet, c – OMG)...	59
Figure 13. Construction de l'objet branchement inclusif avec les branchements parallèles et exclusifs	62
Figure 14. Le modèle GRADEI	63
Figure 15. Vue générale du déploiement du model checking pour l'évaluation de l'exigence d'interopérabilité sur des modèles de processus collaboratifs et structuration au sein du modèle GRADEI	65
Figure 16. Vers un langage dédié d'expression des exigences d'interopérabilité et la définition d'un référentiel pour la gestion des exigences	67
Figure 17. Arbre de syntaxe d'une exigence conforme à la grammaire du langage dédié	70
Figure 18. Etats et variables du modèle à états de la tâche pour l'expression d'une proposition en logique temporelle.....	71
Figure 19. Arbre de syntaxe d'une exigence conforme à la grammaire TCTL	72
Figure 20. Vers la modélisation multi-points de vue pour l'ingénierie des systèmes de systèmes et l'analyse de l'impact de l'interopérabilité	79
Figure 21. Points de vue pour la modélisation des Systèmes de Systèmes	80
Figure 22. Concepts pour l'analyse de l'impact de l'interopérabilité sur l'adaptabilité du SdS (performance, cycle de vie de l'interopérabilité et perturbations).....	82
Figure 23. Vers un langage dédié d'expression d'exigences d'interopérabilité atemporelle	88
Figure 24. Poursuite des travaux sur les « -ilities »	90
Figure 25. Interdépendances entre les « -ilities » [DEW 12].....	91
Figure 26. Exploration des relations entre les « -ilities »	92
Figure 27. Dépendance et interdépendance entre les « -ilities »	92
Figure 28. Intensité des relations entre les « -ilities »	93
Figure 29. Degré d'impact entre les « -ilities »	93
Figure 30. Effet direct et indirect entre des « -ilities ».....	94

Liste des Tableaux

Tableau 1. Domaines d'intérêt pour l'Ingénierie des Systèmes et l'ingénierie des Systèmes de Systèmes [VAL 07]	35
Tableau 2. Langages d'expression des exigences et caractéristiques	43
Tableau 3. Evaluation quantitative et qualitative de l'interopérabilité	44
Tableau 4. Composantes du traitement de l'exigence d'interopérabilité	50
Tableau 5. Synthèse des travaux de recherche sur l'évaluation de l'exigence d'interopérabilité par les graphes conceptuels	57
Tableau 6. Modèle à états d'une tâche BPMN	60
Tableau 7. Modèle à états d'une ressource	60
Tableau 8. Modèle à états d'une activité BPMN utilisant une ressource	60
Tableau 9. Modèles à états existants pour les branchements et les évènements	61
Tableau 10. Synthèse des travaux de recherche sur l'évaluation de l'exigence d'interopérabilité par le model checking	66
Tableau 11. Grammaire du langage pour l'expression d'exigence	70
Tableau 12. Mise en œuvre de la grammaire TCTL	72
Tableau 13. Correspondance entre les modalités du DSL et les quantificateurs TCTL	73
Tableau 14. Correspondance entre une proposition du DSL et une proposition TCTL (correspondance partielle)	74
Tableau 15. Correspondances entre le DSL et TCTL	74
Tableau 16. Exemples d'exigences d'interopérabilité (C-compatibilité, I-interopération, A-autonomie, R-réversibilité)	76
Tableau 17. Synthèse des travaux de recherche sur l'écriture des exigences d'interopérabilité et leur structuration	78
Tableau 18. Matrice d'impact de l'interopérabilité sur les perspectives d'analyse	82
Tableau 19. Matrice d'impact de la compatibilité sur le SdS	83
Tableau 20. Impact de l'interopération sur le SdS	84
Tableau 21. Impact de l'autonomie sur le SdS	85
Tableau 22. Impact de la réversibilité sur le SdS	86
Tableau 23. Synthèse des travaux de recherche sur l'ingénierie des SdS et l'impact de l'interopérabilité	86

Avant-Propos

L'objectif de ce mémoire est de présenter une synthèse de mes activités d'enseignements et de recherche pour l'obtention d'une Habilitation à Diriger les Recherches. Il s'attache à démontrer la cohérence des recherches menées sur le domaine de l'interopérabilité des systèmes collaboratifs.

Depuis leur début, mes activités de recherche se positionnent dans le domaine de l'interopérabilité des entreprises collaboratives. Les premières recherches ont abordé l'interopérabilité en termes de méthodologie pour assurer le développement de l'interopérabilité des entreprises. Elles ont abouti, notamment, à la définition d'une démarche guidant la sélection et l'implantation de solutions d'interopérabilité, à un cadre d'interopérabilité pour structurer les concepts de l'interopérabilité et à la définition d'indicateurs pour l'évaluation de la compatibilité et l'évaluation des performances d'interopérabilité.

Les travaux présentés dans ce mémoire s'inscrivent toujours dans le cadre du développement de l'interopérabilité des organisations. Ils s'intéressent particulièrement à l'évaluation de l'interopérabilité pour guider les entreprises dans la réalisation de collaborations efficaces et efficientes du point de vue de l'interopérabilité. L'originalité de ces travaux, par rapport aux travaux antérieurs, est l'adoption du point de vue des « exigences » tel qu'il est appréhendé dans le domaine de l'ingénierie des systèmes et l'utilisation de techniques de vérification formelle pour assurer l'évaluation de cette exigence d'interopérabilité. Les résultats de cette recherche aussi bien du point de vue conceptuel, méthodologique ou encore technique ont été obtenus en traitant progressivement les différentes problématiques liées à cette thématique. Ils se sont concrétisés au travers de mon implication dans des projets de recherche et dans des encadrements de jeunes chercheurs doctorants et post-doctorants.

Ce mémoire se décompose en trois parties :

La première partie présente mon curriculum vitae, mes activités de recherche en termes de projets et d'encadrements de jeunes chercheurs ainsi que mes activités en termes d'enseignements et d'encadrements à l'école nationale supérieure des mines d'Alès.

La deuxième partie présente mon activité de dissémination de mes travaux de recherche en termes de publications scientifiques et de rapports réalisés dans le cadre de projets de recherche.

La troisième partie présente l'activité de recherche conduite ainsi que les résultats obtenus depuis l'obtention de mon doctorat, au travers de l'encadrement de thèses et de la participation à des projets de recherche.

La quatrième partie identifie un programme de recherche lié aux résultats des travaux sur la formalisation et l'évaluation de l'exigence d'interopérabilité ainsi qu'aux questions qui émergent sur le domaine de l'évaluation des exigences non fonctionnelles.

I – PARTIE ADMINISTRATIVE





1. Curriculum vitae

1.1. Etat Civil

Nicolas Daclin

Né le 19 juin 1979 à Vénissieux (Rhône)

Marié, 1 enfant

Coordonnées personnelles	Coordonnées professionnelles
173 rue Adolphe Blanchard 30000 Nîmes  +33 (0) 626 874 854  nicolas.daclin@gmail.com	LGI2P Parc Scientifique G. Besse - Site EERIE de l'EMA, 30035 Nîmes cedex 1  +33 (0) 466 387 039  nicolas.daclin@mines-ales.fr

1.2. Position

Maître-assistant à l'école nationale supérieure des mines d'Alès, Laboratoire de Génie Informatique et d'Ingénierie de la Production (LGI2P) au sein de l'équipe Interoperable System and Organization Engineering (ISOE). Les domaines étudiés sont les organisations collaboratives et l'interopérabilité des systèmes. Les travaux sont développés selon les paradigmes de l'Ingénierie des Systèmes et de la modélisation d'entreprise. Les domaines d'applications sont la santé, la gestion de crise ou encore, les systèmes industriels.

1.3. Parcours

- 2012-...** : Maître-assistant à l'école de mines d'Alès
Enseignant chercheur au LGI2P, équipe Interoperable Systems and Organizations Engineering (ISOE)
- 2009-2012** : Ingénieur de recherche et développement à l'école des mines d'Alès
Enseignant chercheur au LGI2P, équipe Interoperable Systems and Organizations Engineering (ISOE)
- 2008-2009** : Post-doctorat à l'école de mines d'Alès
Chercheur au LGI2P, équipe Interoperable Systems and Organizations Engineering (ISOE)
- 2003-2007**: Doctorant à l'Université de Bordeaux
Laboratoire de l'Intégration du Matériau au Système (IMS), département Automatique, Productique et Signal

Mots-clés

Interopérabilité, ingénierie système, modélisation d'entreprise, performance industrielle, processus collaboratif, ingénierie des exigences

1.4. Formation

- 2007** : Doctorat, spécialité Productique, Laboratoire de l'Intégration du Matériau au Système (IMS), département Automatique, Productique et Signal – Université Bordeaux 1. Mention *Très Honorable*
- 2003** : DEA Automatique, Productique Signal et Image (APSI), spécialité Productique, Laboratoire d'Automatique, Productique et Signal (LAPS) – Université Bordeaux 1. Mention *Bien*

- 2002 :** Maîtrise Sciences de la Production Industrielle (SPI) - Université Bordeaux 1
2001 : Licence Sciences de la Production Industrielle (SPI) - Université Bordeaux 1
2000 : DUT Génie Industriel et Maintenance (GIM) - Université Clermont 1

2. Activités de recherche

2.1. Publications/contributions

	Internationales	Nationales
Revue	7	1
Chapitres d'ouvrage	3	--
Conférences	28	3
Workshops	4	3
Forums	-	4
Groupe de travail	2	5
Doctoral workshop	2	-
Rapports de recherche	10	3

Communications référencées dans Web of Science

	Id
Revue	[RI2], [RI3], [RI4], [RI6]
Conférences	[CI23], [CI24], [CI25], [CI26], [CI22], [W13], [CI17], [CI14], [CI11], [CI7], [CI3], [CI8], [CI5]

2.2. Projets

Nom	Nature	Durée	Rôle			Montant ¹
			Coordination	Montage	Participation	
RE(H)STRAIN	ANR Franco- Allemand	24 mois (2015-2017)	✓	✓	✓	120K€
HOPI CAMP	FUI	36 mois (2015 – 2018)		✓	✓	66K€
CARIONER II	Carnot- Mines	12 mois (2012)			✓	NA
COREGI	ANR	36 mois (2007 – 2010)			✓	73K€
ISYCRI	ANR	36 mois (2006 - 2009)			✓	108K€
CARIONER	Carnot- mines	12 mois (2008-2009)			✓	54k€
INTEROP NoE	Européen	36 mois (2003 – 2006)			✓	NA
ATHENA IP	Européen	36 mois (2003 – 2006)			✓	NA

¹ Montant pour le centre de recherche

2.3. Encadrements

Encadrements de thèses de doctorat

Nom	Sihem Mallek
Durée	3 ans (2008-2011)
Titre	Contribution au développement de l'interopérabilité en entreprise : vers une approche de détection de problèmes d'interopérabilité dans des processus collaboratifs
Taux d'encadrement	50%
Communications	[RI2], [RI3], [CI13], [CI14], [CI15], [DI2], [RN1], [CN2], [CN3], [GTN3]

Nom	Stéphane Billaud
Durée	3 ans (2012-2015)
Titre	Contribution à l'Ingénierie de Système de Systèmes : modélisation multi points de vue et analyse de l'impact de l'exigence d'interopérabilité
Taux d'encadrement	80%
Communications	[WN2], [GTN4], [CI22], [CI24], [CI25]

Nom	Behrang Moradi
Durée	En cours (2016-2019)
Titre	Formalisation et évaluation de propriétés non-fonctionnelles pour l'ingénierie de systèmes de systèmes : application à la résilience
Taux d'encadrement	80% (cible)
Communications	[DW3]

Encadrements de post-docs

Nom	Sihem Mallek
Durée	12 mois (2012)
Titre	Spécification d'exigences d'interopérabilité à l'aide du langage SBVR
Projet	CARIONER II
Communications	[RI2], [RI3], [RI5], [RN1], [CI13], [CI14], [CI15], [CN2], [CN3], [WI3]

Nom	Nawel Amokrane
Durée	12 mois (2016-2017)
Titre	Etude de la résilience du réseau ferré grande vitesse Franco-Allemand
Projet	RE(H)STRAIN
Communications	[CI28]

2.4. Autres activités de valorisation

Activités	Détails
Relecture d'articles	<p>Revues :</p> <ul style="list-style-type: none"> • International Journal of Emergency Management (IJEM) • Enterprise Information Systems (EIS) • International Journal Of Design and Innovation Research (IJODIR) • International Journal of Computer Integrated Manufacturing (IJCIM) • International Journal of Engineering and Technological Sciences (JETS) <p>Conférences :</p> <ul style="list-style-type: none"> • Complex Systems Design & Management (CSDM) • IFAC World Congress • IFIP Working Conference on Virtual Enterprises (PRO'VE) • IFIP Working Conference on Enterprise Interoperability (IWEI) • EMEA Systems Engineering Conference • ACS/IEEE International Conference on Computer Systems and Applications (AICCSA) • Management des Technologies Organisationnelles (MTO) • Symposium Information Control Problems in Manufacturing (INCOM) • International Conference on Information Systems for Crisis Response and Management (ISCRAM)
Organisation d'évènements	<ul style="list-style-type: none"> • IWEI 2015 • JIAE 2010 (Journées de l'Interopérabilité des Applications d'Entreprise du pôle GSO) • I-ESA 2006 (Interoperability for Enterprise Software and Applications) • RoBAFIS 2013 • 2nd Workshop RE(H)STRAIN
Participation à la Communauté	<ul style="list-style-type: none"> • Représentant de l'école des mines d'Alès au Pôle Grand Sud-Ouest (PGSO) du laboratoire Virtuel Européen de Recherche sur l'interopérabilité (InterOp V-Lab) • Membre du bureau CA PGSO • Membre du GDR-MACS (GT EASY-DIM: Ingénierie d'Entreprise et de Système d'Information Dirigée par les Modèles et GT IS3C: Ingénierie des systèmes de conception et conduite du cycle de vie produit) • Membre de l'AFIS (Association Française d'Ingénierie Système)
Conseil/expertise	<ul style="list-style-type: none"> • Modélisation des processus métiers en vue de l'obtention de la certification HAS – société HAD 3G Santé

3. Activités d'enseignements et encadrements

3.1. Résumé

Ecole des Mines d'Alès (2009 - ...)

Ingénierie des besoins et des exigences – 2^{ème} année Option EMACS² (M1)

Modélisation d'entreprise - 2^{ème} année - option EMACS

Ingénierie Système - 1^{ère} année de formation initiale (L3)

Modélisation processus - 1^{ère} année de formation initiale

Productique - 1^{ère} année de formation initiale

Ingénierie des besoins et des exigences - apprentis mécatronique

Université de Bordeaux (2003 - 2007)

Bureautique – 1^{ère} année IUT MP

Programmation C – 1^{ère} année IUT MP

Mécanique des fluides – 2^{ème} année IUT MP

Modélisation fonctionnelle – 2^{ème} année IUP GEII

Gestion de projet – Licence SPI

Technologie mécanique – Licence SPI

Système à évènements discrets – Licence SPI, 2^{ème} année IUP GEII, Master I EEA

Organisation et gestion de la production – IUP Génie mécanique

Université Politehnica de Bucarest (2013 - ...)

Ingénierie des besoins et des exigences – Etudiants du département Automatic Control and Systems Engineering (Ecole d'été CANTI)

3.2. Détails des enseignements et encadrements pour une année type³

Enseignement	Ingénierie des besoins et des exigences	Cours	TD	TP	Projet
Public	2 ^{ème} année EMA - EMACS (~40 étudiants)	4	--	8	2 mois
	1 ^{ère} année EMA apprenti mécatronique (~20 étudiants)	4	7	9	3 mois
	Université Polytechnique de Bucarest (~15 étudiants)	4	--	--	--

Enseignement	Modélisation et maîtrise des processus	Cours	TD	TP	Projet
Public	1 ^{ère} année EMA formation initiale (180- 200 étudiants)	4	8	--	--

Enseignement	Modélisation d'entreprise	Cours	TD	TP	Projet
Public	2 ^{ème} année EMA - EMACS (~40 étudiants)	8	3	4	1 mois

² Engineering and Management of Complex Systems

³ La totalité des enseignements et encadrements réalisés depuis le début de mes activités pédagogiques est donnée en annexe.

Enseignement	Ingénierie Systèmes	Cours	TD	TP	Projet
Public	1 ^{ère} année EMA formation initiale (~40 étudiants)	--	8	9	1 mois

Encadrement	Projet de fin d'études	Entreprises	Durée
Public	3 ^{ème} année EMA - EMACS (3 étudiants)	SNCF	6 mois
		Air France Cargo	1 an (contrat pro)
		Air France Cargo	1 an (contrat pro)

Encadrement	Apprentis	Entreprises	Durée
Public	3 ^{ème} année EMA apprenti INFRES (1 étudiant)	Conseil Général de l'Hérault	3 ans
	2 ^{ème} année EMA apprenti INFRES (1 étudiant)	IBM	3 ans
	1 ^{ère} année EMA apprenti INFRES (1 étudiant)	SADA Assurances	3 ans

Encadrement	Mission R&D	Sujet	Durée
Public	2 ^{ème} année EMA – EMACS (2 étudiants)	Développement d'un éditeur de « boilerplates » pour l'expression d'exigences	5 semaines
	2 ^{ème} année EMA – EMACS (3 étudiants)	Modélisation et Evaluation du processus de gestion du cycle de l'eau et des déchets dans les hôpitaux	5 semaines

3.3. Responsabilités pédagogiques

Année	Module	Formation	Nature	Nombre d'intervenants
2012- ...	Ingénierie Système	EMA- EMACS M1	Responsable de l'enseignement Ingénierie des exigences techniques	3
2012- ...	Modélisation pour la productique	EMA- EMACS M1	Responsable de l'enseignement modélisation d'entreprise	1
2012- ...	Excellence opérationnelle	EMA- L3	Responsable de l'enseignement modélisation et maîtrise des processus	2
2013- ...	Evaluation et pilotage de la performance en entreprise	EMA- EMACS M2	Responsable du module	3
2013- ...	Méthode et stratégie pour l'entreprise	EMA- EMACS M2	Responsable du module	2
2015- ...	Ingénierie Système	EMA- mécatronique	Responsable de l'enseignement Ingénierie des exigences techniques	3

3.4. Autres

Activités	Détails
Expertise	Support au montage du master « Génie Industriel » de l'Université Polytechnique de Tirana, Albanie

II – PUBLICATIONS

1. Revues

Id	Revues internationales avec comité de rédaction
[R11]	N. Daclin, D. Chen, B. Vallespir Enterprise interoperability measurement: the potentiality <i>INCOSE INSIGHT, Special feature – The best of France – Forum académique et robAFIS</i> , vol. 11 - issue 3, p 18, juillet 2008
[R12]	S. Mallek, N. Daclin, V. Chapurlat The application of interoperability requirement specification and verification to collaborative processes in industry <i>Computers in Industry</i> , vol. 63, issue 7, pp 643-658, septembre 2012
[R13]	S. Mallek, N. Daclin, V. Chapurlat, B. Vallespir. Enabling model checking for collaborative process analysis: from BPMN to Network of Timed Automata. <i>Enterprise Information Systems</i> vol. 9 - issue 3, pp 279-299, avril 2015
[R14]	N. Daclin, D. Chen, B. Vallespir. Developing enterprise collaboration: a methodology to implement and improve interoperability. <i>Enterprise Information Systems</i> , vol. 10 – issue 5, pp 467-504, 2016
[R15]	N. Daclin, S. Mallek-Daclin, V. Chapurlat. Interoperability analysis in critical collaborative processes: an application to the healthcare sector for complex patient assistance. <i>International Journal of Emergency Management</i> , vol. 12 – issue 4, pp 435-456, 2016
[R16]	N. Daclin, S. Mallek-Daclin, V. Chapurlat, B. Vallespir Writing and verifying interoperability requirements: Application to collaborative processes <i>Computers in Industry</i> , volume 82, pp. 1-18, October 2016
[R17]	N. Daclin, G. Dusserre, L. Mailhac, B. Lhéritier, J. Blanchard, A. Picard Towards a maturity model to assess field hospitals' rollout <i>International Journal of Emergency Management</i> , (accepted, in press), 2017
Id	Revues nationales avec comité de rédaction
[RN1]	S. Mallek, N. Daclin, V. Chapurlat Catégorisation et formalisation des exigences d'interopérabilité dans les processus collaboratifs <i>Ingénierie des Systèmes d'Information</i> , vol. 15, pp 37-61, décembre 2010

2. Chapitres d'ouvrage collectif

id	Chapitres
[CO1]	V. Chapurlat, N. Daclin Designing natively interoperable complex systems: an interface design pattern proposal <i>Enterprise Interoperability, INTEROP_PGSO Vision</i> Edited by B. Archimède & B. Vallespir, chap. 3, Wiley, 244 pp., avril 2017
[CO2]	N. Daclin, D. Chen, B. Vallespir Decisional interoperability <i>Enterprise Interoperability, INTEROP_PGSO Vision</i> Edited by B. Archimède & B. Vallespir, chap. 5, Wiley, 244 pp., avril 2017
[CO3]	N. Daclin, D. Chen, B. Vallespir The interoperability measurement <i>Enterprise Interoperability, INTEROP_PGSO Vision</i> Edited by B. Archimède & B. Vallespir, chap. 6, Wiley, 244 pp., avril 2017

3. Conférences

id	Conférences internationales avec comité de rédaction
[CI1]	N. Daclin, D. Chen, B. Vallespir Développement de l'interopérabilité des applications de gestion industrielles : concepts de base et définitions. <i>6^{ème} Congrès International de Génie Industriel (CIGI)</i> , Besançon, France, 7-10 juin 2005
[CI2]	N. Daclin, D. Chen, B. Vallespir Design principles and pattern for decisional interoperability <i>IFIP 5.7 Advances in Production Management Systems (APMS), Modelling and Implementing the Integrated Enterprise</i> , Washington DC, USA, 18-21 septembre 2005
[CI3]	N. Daclin, D. Chen, B. Vallespir Towards a conceptualisation of decisional interoperability <i>Interoperability for Enterprise Software and Applications (I-ESA'06)</i> , Bordeaux, France, 20-24 mars 2006
[CI4]	N. Daclin, D. Chen, B. Vallespir A methodology to develop interoperability of enterprise applications <i>12th IFAC Symposium on Information Controls Problems in Manufacturing (INCOM 2006)</i> , Saint-Etienne, France, 17-19 mai 2006
[CI5]	N. Daclin, D. Chen, B. Vallespir Decisional interoperability : concepts and formalization <i>7th IFIP Working Conference on Virtual Enterprises (PRO-VE'06)</i> , Helsinki, Finlande, 25-27 septembre 2006 (taux d'acceptation 44%)
[CI6]	I. Santos, T. Kahl, N. Daclin, D. Chen SME interoperability establishing methodology: a guided roadmap for the journey to interoperability <i>eChallenges e-2006</i> , Barcelone, Espagne, 25-27 octobre 2006

- [CI7]** R. Grangel, R. Ben Salem, J-P. Bourey, N. Daclin, Y. Ducq
Transforming GRAI extended actigrams into UML Activity diagrams : a first step to Model Driven Interoperability
Interoperability for Enterprise Software and Applications (I-ESA'07), Madère, Portugal, 27-30 mars 2007
- [CI8]** D. Chen, N. Daclin
Barriers driven methodology for enterprise interoperability
8th IFIP Working Conference on Virtual Enterprises (PRO-VE'07), Guimaraes, Portugal, 10-12 septembre 2007 (taux d'acceptation 44%)
- [CI9]** N. Daclin, D. Chen, B. Vallespir
Methodology for enterprise interoperability
17th IFAC World Congress (IFAC'08), Séoul, Corée, 6-11 juillet 2008 (taux d'acceptation 72%)
- [CI10]** D. Chen, B. Vallespir, N. Daclin
An approach for enterprise interoperability measurement. *Model Driven Information Systems Engineering: Enterprise, User and Systems Models (Modise-EUS'08)*, Montpellier, France, 16-17 juin 2008
- [CI11]** N. Daclin, V. Chapurlat, F. Benaben
A verification approach applied for analyzing collaborative processes: the Anticipative Effect-Driven Approach
13th IFAC Symposium on Information Controls Problems in Manufacturing (INCOM 2009), Moscou, Russie, 3-5 juin 2009 (taux d'acceptation 62%)
- [CI12]** N. Daclin, V. Chapurlat
An anticipative effects-driven approach for analysing interoperability problems in collaborative process
10th IFIP Working Conference on Virtual Enterprises (PRO-VE'09), Thessalonique, Grèce, 7-9 octobre 2009 (taux d'acceptation 42%)
- [CI13]** S. Mallek, N. Daclin, V. Chapurlat
Towards a conceptualization of interoperability requirements
Interoperability of Enterprises Software and Applications (I-ESA'10), Coventry, Angleterre, 14-15 avril 2010
- [CI14]** S. Mallek, N. Daclin, V. Chapurlat
An Approach for Interoperability Requirements Specification and Verification
International IFIP Working Conference on Enterprise Interoperability, Stockholm, Suède, 23-24 mars 2011 (taux d'acceptation 31.91%)
- [CI15]** S. Mallek, N. Daclin, V. Chapurlat
Formalisation and Verification of Interoperation Requirements on Collaborative Processes
18th IFAC World Congress (IFAC'11), Milano, Italy, 28 aout-2 septembre 2011 (taux d'acceptation 68%)
- [CI16]** V. Chapurlat, N. Daclin
System interoperability ; definition and proposition of interface model in MBSE context
14th Symposium on Information Control in Manufacturing (INCOM 2012), Bucarest, Roumanie, 23-24 mai 2012 (taux d'acceptation 61%)

- [CI17]** N. Daclin
Embedding interoperability in System of Systems : definition and characterization of fundamental requirements
13th IFIP Working Conference on Virtual Enterprises (PRO-VE'12), Bournemouth, Angleterre, 1-3 octobre 2012 (taux d'acceptation 26%)
- [CI18]** V. Chapurlat, N. Daclin, P. Chucho, S. Marzin, P. Vien
Proposition d'un Guide d'Etude des Modes Opérationnels des Systèmes (GEMOS) pour l'ingénierie système
10^{ème} congrès International de Génie Industriel (CIGI2013), La Rochelle, France, 12-14 juin 2013
- [CI19]** V. Chapurlat, N. Daclin
Proposition of a guide for investigating, modeling and analyzing system operating modes: OMAG
Complex Systems Design & management (CSDM 2013), poster, Paris, France, 4-6 décembre 2013
- [CI20]** N. Daclin, S. Mallek
Capturing and structuring interoperability requirements: a framework for interoperability requirements
Interoperability for Enterprise Software and Applications (I-ESA'14), Albi, France, 24-28 mars 2014
- [CI21]** M. Bilal, N. Daclin, V. Chapurlat
System of Systems design verification: problematic, trends and opportunities
Interoperability for Enterprise Software and Applications (I-ESA'14), Albi, France, 24-28 mars 2014
- [CI22]** M. Bilal, N. Daclin, V. Chapurlat
Collaborative Networked Organizations as System of Systems: a model-based engineering approach
15th IFIP Working Conference on Virtual Enterprises (PRO-VE'14), Amsterdam, Pays-Bas, 6-8 octobre 2014 (taux d'acceptation 26%)
- [CI23]** N. Daclin, S. Mallek-Daclin
Towards a sustainable implementation of interoperability solutions: bridging the gap between interoperability requirements and solutions
6th International IFIP Working Conference on Enterprise Interoperability (IWEI 2015), Nîmes, France, 28-29 mai 2015
- [CI24]** S. Billaud, N. Daclin, V. Chapurlat
Interoperability as a key concept for the control and evolution of the System of Systems (SoS)
6th International IFIP Working Conference on Enterprise Interoperability (IWEI 2015), Nîmes, France, 28-29 mai 2015
- [CI25]** V. Chapurlat, N. Daclin, S. Billaud
An interface pattern model for supporting design of natively interoperable systems
6th International IFIP Working Conference on Enterprise Interoperability (IWEI 2015), Nîmes, France, 28-29 mai 2015

[CI26]	V. Chapurlat, N. Daclin System of Systems Architecting: A behavioural and Properties Based Approach for SoS “-ities” Modelling and Analysis <i>16th IFIP Working Conference on Virtual Enterprises (PRO-VE’15)</i> , Albi, France, 5-7 octobre 2015
[CI27]	S. Mallek-Daclin, N. Daclin, G. Dusserre, B. Lhéritier, J-B, I. Arnaud Maturity model-driven assessment of Field Hospitals <i>20th IFAC World Congress (IFAC 20)</i> , Toulouse, France, 9-14 juillet 2017
[CI28]	N. Amokrane, N. Daclin, V. Chapurlat Deducing complex scenarios for resilience analysis : application to the Franco-German high speed train network <i>14th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2017)</i> , Albi, France, 2-24 mai 2017

id Conférences nationales avec comité de rédaction

[CN1]	N. Daclin, V. Chapurlat Evaluation de l’interopérabilité organisationnelle et managériale : le projet CARIONER <i>1^{ère} journée du Management des Technologies Organisationnelles (MTO’08)</i> , Montpellier, France, 25 septembre 2008
[CN2]	S. Mallek, N. Daclin, V. Chapurlat Pilotage des processus collaboratifs : amélioration de l’interopérabilité <i>2^{èmes} Journées d’Etudes sur le Management des Technologies Organisationnelles (MTO’09)</i> , Montpellier, France, 15-16 octobre 2009
[CN3]	S. Mallek, N. Daclin, V. Chapurlat Exigences d’interopérabilité : caractérisation et formalisation en propriétés <i>Ingénierie d’Entreprise et des Systèmes d’Information, en association avec INFORSID 2010</i> , Marseille, France, 25 mai 2010

4. Workshops

id Workshops internationaux avec comité de sélection

[WI1]	D.Chen, N. Daclin Framework for enterprise interoperability <i>2nd IFAC Workshop on Enterprise Integration, Interoperability and Networking (EI2N)</i> , Bordeaux, France, 20-24 mars 2006
[WI2]	N. Daclin, D. Chen, B. Vallespir Enterprise interoperability measurement – Basic concepts <i>3rd Workshop on Enterprise Modelling and Ontologies for interoperability (EMOI)</i> , Luxembourg, Luxembourg, 5-6 juin 2006
[WI3]	S. Mallek, N. Daclin, V. Chapurlat, B. Vallespir Requirements formalization for systems engineering: an approach for interoperability analysis in collaborative process model <i>Workshop on selected applications and methods for enterprise interoperability, co-located with the 5th International IFIP Working Conference on Enterprise Interoperability (IWEI 2013)</i> , Enschede, Pays-Bas, 26 mars, 2013 (taux d’acceptation 50%)

[WI4] V. Chapurlat, N. Daclin, A. Bony-Dandrieux, J. Tixier, F. Benaben, D. Kamissoko, N. Salatge, E. Guigou
A Method for Critical Infrastructures Resilience Engineering: The MAIEUTIC project
Workshop on Interoperability in crisis management and homeland security, co-located with the 8th International Conference on Interoperability for Enterprise Systems and Applications (I-ESA2016), Guimaraes, Portugal, 29 mars – 1^{er} avril 2016

id Workshops nationaux avec comité de rédaction

[WN1] N. Daclin, B. Vallespir, D. Chen
Contribution au développement de l'interopérabilité décisionnelle des systèmes industriels
4^{ème} Workshop Ingénierie et Gestion des Processus d'Entreprise, Paris, France, 15 mai 2007

[WN2] M. Bilal, N. Daclin, V. Chapurlat
Multi-Agents approach for Systems of Systems (SoS) behavior validation
Workshop GT EASY-DIM 2013, Paris, France, 22 mai 2013

[WN3] L. Mailhac, N. Daclin, V. Chapurlat, G. Dusserre, F. Guarnieri, J. Blanchard, B. Lhéritier, A. Picard, K. Streff, I. Arnaud
Evaluation de la maturité des hôpitaux de campagne –Application au cas de l'ESCRIM.
Workshop GT EASY-DIM 2015, Bron, France, 29 juin 2015

id Workshops et forums nationaux

[FN1] N. Daclin, B. Vallespir, D. Chen
Mesure d'interopérabilité : la potentialité (poster)
Forum académique AFIS'07, Nantes, France, 28-29 novembre 2007

[FN2] V. Chapurlat, N. Daclin
Plateforme de services pour la coordination d'unités d'Hospitalisation à Domicile (HAD) (poster)
1^{er} Colloque de l'Institut Mines-Télécom pour l'Ingénierie et STIC pour la santé, Paris, France, 6-7 mars 2012

[FN3] N. Daclin
RE(H)STRAIN – Resilience of the Franco-German High Speed Train Network (poster)
Workshop Interdisciplinaire sur la sécurité globale WISG15 Troyes, France, 3-4 février 2015

[FN4] N. Daclin, M. Drissi-Habti
Résilience du réseau ferré grande vitesse franco-allemand (présentation orale du projet RE(H)STRAIN)
Workshop Interdisciplinaire sur la sécurité globale WISG16, Troyes, France, 10-11 février 2016

5. Rapports de recherche

id	Groupes de travail internationaux
[RA1]	Scientific integration conceptual model and its application in INTEROP <i>INTEROP WP4 Method for scientific integration and assessment</i> , Deliverable D4.1, 2004 (<u>contributeur</u>)
[RA2]	Practices, principles and patterns for interoperability <i>INTEROP WP6 Design principles for interoperability</i> , Deliverable D6.1, 2005 (<u>contributeur</u>)
[RA3]	Report on model establishment <i>INTEROP Task Group 2 Model Driven Interoperability</i> , Deliverable DTG2.1, 2005 (<u>co-auteur</u>)
[RA4]	Interoperability knowledge corpus – intermediate report <i>INTEROP Task Group Domain Interoperability</i> , Deliverable DI.1b , 2006 (<u>contributeur</u>)
[RA5]	Enterprise interoperability – framework and knowledge corpus – advanced report <i>INTEROP Task Group Domain Interoperability</i> , Deliverable DI.2, 2006 (<u>contributeur</u>)
[RA6]	Report on Model Interoperability <i>INTEROP Task Group 2 Model Driven Interoperability</i> , Deliverable DTG2.2, 2006 (<u>co-auteur</u>)
[RA7]	Guidelines and best practices for applying the ATHENA interoperability framework to support SME participation in digital ecosystems <i>ATHENA WP A8 SME interoperability in practice</i> , Deliverable D.A8.2, 2007 (<u>contributeur</u>)
[RA8]	Spécification détaillée des composants middleware, de l’outil de traduction et développement d’un démonstrateur <i>ISYCRI lot 2</i> , livrable T2.3, 2009 (<u>contributeur</u>)
[RA9]	Modeleur systémique et Analyse dans les cas des systèmes GRT-gaz et SNCF <i>COREGI</i> , livrable final, 2010 (<u>co-auteur</u>)
[RA10]	Representation of the Franco-German high-speed train system <i>Joint research project REHSTRAIN</i> , report 1.1, 2016 (<u>co-auteur</u>)
[RA11]	RE(H)STRAIN - Resilience of the Franco-German High Speed train Network <i>Joint project REHSTRAIN</i> , rapport intermédiaire ANR (<u>auteur</u>)
[RA12]	Automatic deduction of complex scenarios <i>Joint research project REHSTRAIN</i> , report 3.3, 2017 (<u>co-auteur</u>)
[RA13]	Conceptual Framework for HSS Resilience Analysis <i>Joint research project REHSTRAIN</i> , report 5.3, 2017 (<u>co-auteur</u>)

6. Groupes de travail & autres

id	Groupes de travail internationaux
[GTI1]	N. Daclin, D. Chen Contribution to task 2.1 <i>Document interne du WP1 du Réseau d'Excellence INTEROP – Knowledge map for Enterprise Modeling, Ontology, Architecture and Platform for interoperability</i> , 20 mai 2004
[GTI2]	D.Chen, N. Daclin Design principles and pattern for decisional interoperability (présentation). <i>INTEROP NoE WP6</i> <i>Design Principles for interoperability Workshop</i> , Luxembourg, Luxembourg, 22 novembre 2004

id	Communications dans des groupes de travail nationaux
[GTN1]	N. Daclin Interopérabilité décisionnelle : concepts de base et formalisation (présentation) <i>3^{ème} école de modélisation d'entreprise, Modélisation d'entreprise et décision</i> , Arcachon, France, 2-4 octobre 2006
[GTN2]	N. Daclin, V. Chapurlat Laboratoire de Génie Informatique et d'Ingénierie de Production – Ecole des Mines d'Alès (présentation des travaux en cours) <i>1^{er} Colloque sur l'interopérabilité des applications d'entreprise</i> , Ecole des Mines d'Albi-Carmaux, Albi, France, 4-5 juin 2008
[GTN3]	S. Mallek, N. Daclin, V. Chapurlat Approche anticipative dirigée par les effets : exigences d'interopérabilité (présentation) <i>2^{èmes} Journées sur l'interopérabilité des applications d'entreprises</i> , Ecole Nationale d'Ingénieurs de Tarbes, Tarbes, France, 22-23 octobre 2009
[GTN4]	M. Bilal, N. Daclin, V. Chapurlat Systems of Systems' (SoS) design verification by formal methods and simulation (poster) <i>21^{ème} journées Doctiss</i> , Montpellier, France, 28 mai 2013
[GTN5]	N. Daclin, S. Mallek, V. Chapurlat Interoperability requirements for collaborative process analysis (poster et démonstration). <i>Evaluation AERES</i> . Nîmes, France, 10-11 décembre 2013

id	Doctoral workshops internationaux avec comité de lecture
[DW1]	N. Daclin Contribution to a methodology to develop interoperability of enterprise application. <i>Interoperability of Enterprise Software and Applications Doctoral Symposium (I-ESA'05)</i> , Genève, Suisse, 21-22 février 2005
[DW2]	S. Mallek, N. Daclin, V. Chapurlat Developing interoperability in collaborative process: an anticipative effects-driven approach <i>Doctoral Workshop of APMS, Advances in Production Management Systems (APMS'10)</i> , Bordeaux, France, 19-20 septembre 2009
[DW3]	B. Moradi, N. Daclin, V. Chapurlat Formalisation and assessment of non-functional requirements for Systems of Systems Engineering: Application to resilience (poster) <i>14th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2017)</i> , Albi, France, 2-24 mai 2007
id	Autres
[A1]	N. Daclin Contribution au développement d'une méthodologie pour l'interopérabilité des entreprises <i>Thèse de doctorat</i> , Université Bordeaux 1, Bordeaux, France, 4 décembre 2007
[A2]	N. Daclin Analyse de la cohérence entre les objectifs des processus et les objectifs fonctionnels d'un système de production <i>Mémoire de DEA</i> , Université Bordeaux 1, Bordeaux, France, 2003

III – RECHERCHE

1. Introduction

La nécessité pour des organisations de mettre en œuvre des collaborations conduit naturellement à considérer un ensemble des propriétés qui participent au cycle de vie de ces collaborations depuis leur mise en œuvre jusqu'à leur démantèlement en passant par leurs réalisations. Ces propriétés (locales, *i.e.*, impactant une organisation ou globales, *i.e.*, impactant la collaboration - telles que l'agilité, la flexibilité et la réactivité...) participent pleinement à la réalisation des collaborations et à leurs performances et requièrent une attention particulière.

Parmi ces propriétés qui participent et facilitent le travail collaboratif d'organisations hétérogènes - aussi bien au niveau humain, technique ou encore comportemental - l'interopérabilité revêt un enjeu majeur, impacte et conditionne fortement la réussite de ces collaborations. Elle est donc naturellement étudiée selon différentes facettes aussi bien méthodologique, conceptuelle ou encore technique. L'interopérabilité est une attente des organisations qui collaborent. Au-delà des approches développées pour sa réalisation, il est intéressant pour des entreprises de s'assurer que cette attente est effectivement satisfaite et qu'elle contribue pleinement à la collaboration. A défaut, il est également intéressant d'identifier précisément et dans quelle mesure un problème impacte la collaboration pour permettre une correction rapide, précise et efficace. De ce fait la problématique générale traitée dans ces travaux de recherche peut se formuler de la manière suivante :

Est-ce que le système collaboratif satisfait l'exigence d'interopérabilité ?

En apparence simple, cette question se trouve pourtant au carrefour de différents domaines qui entrent en jeu pour apporter une réponse et qu'il est nécessaire de lier de manière cohérente. Tout d'abord, cette question fait appel à la notion de système et donc aux principes de la systémique [LEM 90] dans le but d'obtenir une représentation du système observé et selon l'analyse que l'on souhaite en faire. Dans le cas présent il s'agit de s'assurer de la satisfaction ou non de l'exigence d'interopérabilité. En effet, il est illusoire de vouloir appréhender un système dans toute sa complexité sans adopter ces notions de points de vue et de modélisation propres à la systémique. Ce premier point concerne donc la **modélisation du système** et sa faculté à permettre de représenter le concept d'interopérabilité afin de l'évaluer. Ensuite, cela fait appel à la notion d'exigence qu'il faut manipuler et exprimer de manière à s'assurer de sa satisfaction par le modèle représentant le système à l'étude. Là encore, cet aspect est régi par un ensemble de règles et de bonnes pratiques. Il s'agit alors de s'appuyer sur des principes éprouvés pour exprimer et manipuler une exigence relative à un domaine particulier. En ce sens, les principes de l'ingénierie des systèmes [INC 11], et plus précisément de l'ingénierie des exigences [MEI 12], permettent d'appréhender les exigences d'un domaine dans toutes leurs dimensions. Ce deuxième point concerne l'expression de l'**exigence d'interopérabilité**. Enfin, cela fait appel aux moyens d'évaluation de la satisfaction de l'exigence d'interopérabilité sur le modèle du système. Ces moyens doivent permettre une **évaluation rigoureuse** c'est-à-dire une évaluation qui aboutit à un résultat précis et fiable en vue de pouvoir analyser ce dernier et ainsi sélectionner une solution ou réaliser un ensemble d'actions correctives adéquates au besoin.

Ainsi, pour répondre à la question énoncée précédemment, il est nécessaire de considérer les aspects de modélisation des systèmes, de spécification de l'interopérabilité et de moyens rigoureux d'évaluation et, en fonction des spécificités et de l'hétérogénéité de ces différentes composantes, de les adapter les unes aux autres.

En effet, l'utilisation de moyens d'évaluation basés sur des techniques spécifiques peut imposer d'adapter le modèle du système lui-même basé sur un langage de modélisation particulier ainsi que l'expression de l'exigence basée également sur son propre langage. Ainsi, afin de répondre à la question initiale sans se focaliser directement sur un moyen d'évaluation ou un langage de modélisation ou de spécification particulier, il est possible de l'envisager de manière générique et reformulée ainsi :

$$\mathcal{M}_s \models Io?$$

Où \mathcal{M}_s est le modèle du système sur le quel on évalue l'interopérabilité, \models est la satisfaction (ou non $\not\models$) évaluée par un moyen m et Io , la spécification d'interopérabilité exprimée de telle manière à évaluer sa satisfaction \models par m sur le modèle \mathcal{M}_s .

La partie suivante s'attache à montrer la manière dont la question générale de l'évaluation de l'interopérabilité est abordée au travers de différents travaux de recherche. Cette partie s'articule autour de deux points :

- Le premier définit de manière précise la problématique en présentant, d'une part, les systèmes concernés par le développement de l'interopérabilité. Les contraintes inhérentes à ces systèmes ainsi que les caractéristiques qui nécessitent de maîtriser l'interopérabilité sont présentées. Cette partie, montre également l'enjeu de l'interopérabilité dans une perspective plus large, c'est-à-dire par sa considération et son intégration au sein de démarches telles que l'ingénierie des systèmes. Enfin, cette partie présente le concept de l'interopérabilité tel qu'il est défini et appréhendé dans les organisations. La problématique d'évaluation de l'interopérabilité au sein des organisations collaboratives et la manière dont elle est abordée permet de dégager un ensemble d'aspects à considérer pour **traiter adéquatement le problème de l'interopérabilité en tant qu'exigence**.
- Le second point présente le traitement des différents aspects de l'interopérabilité en tant qu'exigence au travers des thèses et projets réalisés. La problématique détaillée de chaque aspect est identifiée et les moyens et développements mis en œuvre pour y répondre sont exposés. De même, les relations de dépendances de l'ensemble des composantes participant à la résolution de la problématique sont présentées afin de montrer la cohérence des travaux réalisés.

Enfin, la dernière partie ouvre des perspectives de recherche sur la problématique générale initiale. Ces perspectives sont, d'une part, issues des travaux réalisés sur l'interopérabilité. En effet, ils soulèvent un ensemble de questionnement en termes d'améliorations ou encore sur la possibilité d'explorer des points nouveaux qui présentent un intérêt pour le domaine de la formalisation et de l'évaluation des exigences non fonctionnelles. D'autre part, ces perspectives sont également positionnées dans une vision plus large qui s'intègrent aux questions de recherche et émergent actuellement dans le domaine des exigences non fonctionnelles pour la conception de systèmes.

2. Problématique

2.1. Des réseaux collaboratifs...

L'évolution du contexte industriel s'est accompagnée d'une mutation des entreprises. Cette mutation s'est réalisée au niveau technique leur facilitant la maîtrise de leurs décisions, de leurs activités et de leurs informations. Elle s'est également réalisée au niveau organisationnel *via*, par exemple, l'adoption d'une vision processus leur offrant la possibilité de découpler leurs fonctions et de s'assurer que chaque unité locale participe de manière cohérente au fonctionnement global. Enfin, cette mutation s'est réalisée au niveau conceptuel *via* l'utilisation de modèles d'entreprise permettant l'amélioration continue *via* par exemple la mise en place de systèmes d'indicateurs sur ces modèles pour l'amélioration des performances et le pilotage. De la même manière que la mise en œuvre des moyens techniques, organisationnels et conceptuels a facilité le travail intra-organisation, ce contexte industriel a également vu les entreprises favoriser le travail collaboratif avec des partenaires externes. Un des objectifs est alors de saisir des opportunités auxquelles elles ne peuvent prétendre répondre de manière isolée, par manque de compétences, de moyens financiers ou encore de ressources.

D'un point de vue scientifique, ces structures de travail collaboratif sont clairement identifiées et définies dans la littérature ce qui permet d'obtenir une vision macroscopique des caractéristiques attendues de ces types d'organisations [CAM 05]. Il est possible de citer en exemple le réseau collaboratif qui se définit au travers des caractéristiques bien précises, à savoir [CAM 09] :

- *« Un réseau composé d'une variété d'entités – organisations et personnes – qui sont autonomes, géographiquement distribuées et hétérogènes en termes d'environnement opérationnel, de culture et d'objectifs.*
- *Des participants qui collaborent pour (mieux) atteindre des objectifs communs ou compatibles.*
- *Des interactions entre les participants supportées par des réseaux informatiques »*

La première lecture de cette définition fait émerger les primo-difficultés dans l'implémentation de ces réseaux collaboratifs. Tout d'abord, le fait d'être en présence d'entités autonomes impose de garantir au maximum le maintien de leur autonomie. En effet, une dépendance trop forte entre deux entités au niveau opérationnel (*ex.* en termes de ressources humaines, techniques ou encore de procédures) ou bien décisionnel risque de corrompre la capacité d'une entité à pleinement s'associer à un autre réseau collaboratif. La difficulté réside donc dans la recherche d'un point d'autonomie optimale c'est-à-dire acceptable par l'ensemble des entités. Ce point doit permettre à une entité, d'une part, de participer pleinement au(x) réseau(x) collaboratif(s) dans le(s)quel(s) elle évolue et, d'autre part, lorsque que la collaboration est terminée, de pouvoir en sortir et de participer tout aussi aisément à d'autres réseaux dans lesquels elle aurait un intérêt. Ensuite, il est nécessaire, selon la nature de la collaboration, de couvrir l'ensemble des niveaux d'une entité qui vont interagir avec les autres entités du réseau collaboratif. L'hétérogénéité nécessite de connaître l'ensemble de ces éléments entrant en jeu dans les interactions. En effet, selon la force de la relation, le nombre de ces éléments peut être important et entraîner également un effort (*ex.* financier, technique) important pour être mis en relation et pouvoir interagir efficacement.

Sans une connaissance parfaite de ces éléments, le risque est de permettre des interactions non nécessaires au bon déroulement de la collaboration avec pour conséquence un engagement de moyens sans aucune valeur ajoutée et des risques supplémentaires non nécessairement maîtrisés (ex. partage d'informations non souhaité). Cette notion est mise en avant dans les travaux de [BEN 12] sous le nom « *d'intensité de la collaboration* » ou encore dans les travaux de [CAM 09] comme « *maturité des niveaux d'interaction* » (Figure 1). Ces niveaux identifient strictement ce qui doit être partagé par l'ensemble des parties prenantes⁴ du réseau. Ainsi, on retrouve les niveaux (ici, « maturité des niveaux d'interactions » seulement):

- **Réseau.** Ce niveau concerne la **communication et l'échange basique** d'information pour un bénéfice mutuel. Il n'existe pas d'objectifs communs ou de structures communes qui peuvent influencer la temporalité des activités et il n'y a pas génération de valeur commune.
- **Réseau coordonné.** Au-delà du simple échange d'information, les **activités** de différentes entités participantes concernées par la relation peuvent être **ajustées** de manière à obtenir un meilleur résultat. Les objectifs sont toujours parfaitement distincts.
- **Coopération.** A ce niveau les **ressources** sont **partagées** pour atteindre des objectifs compatibles. La valeur ajoutée de la relation est obtenue par addition des différentes participations des entités. Une **planification commune** peut être envisagée de même qu'un **processus** mettant en commun des activités peut être réalisé même si celui-ci reste à un niveau macroscopique.
- **Collaboration.** Ce dernier niveau implique le **partage de responsabilités** et la construction de **processus collaboratifs** élaborés pour l'atteinte d'un **objectif commun**. Les risques, les pertes ou encore les profits sont également partagés. En ce sens, cela implique un engagement fort de la part des entités. A ce degré, le résultat de la mise en relation entre les différentes entités peut être vu comme une entité unique. A titre d'exemple, pour ce dernier niveau, [Ben 12] utilise le mot « *fusion* » tout à fait pertinent tant les entités sont liées.

⁴ « *Partie ayant un droit, une part ou une prérogative qui fait que le système ou certaines de ses propriétés doivent satisfaire les attentes de cette partie* » [ISO 15].

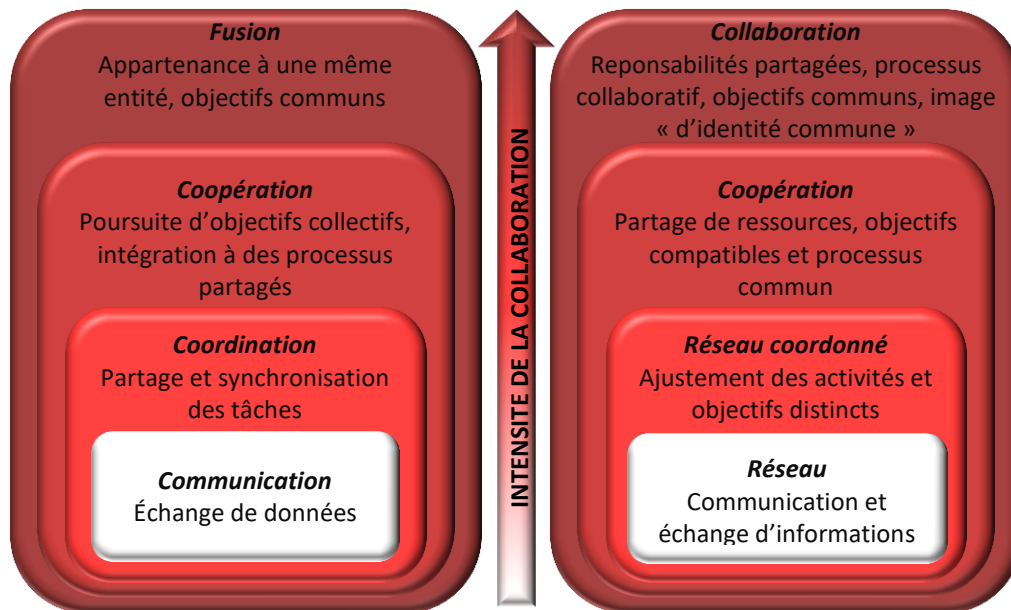


Figure 1. Niveau et implication d'entités collaboratives selon [BEN 12] (gauche) et [CAM 09] (droite)

Bien que l'alignement des niveaux de ces deux visions ne soit pas total (un niveau d'intensité peut par exemple englober plus ou moins de caractéristiques de maturité et *vice versa*), il est toutefois aisé de constater la complexité des interactions à établir entre différentes entités. En effet, la nature et le nombre de ces interactions sont bien différents depuis le plus bas niveau où seul l'échange d'information est considéré jusqu'au partage des objectifs en passant par les ressources, responsabilités ou encore la construction de processus collaboratifs engageant les activités des participants.

Au-delà de la définition de ces « *niveaux relationnels* » ainsi que de l'identification et de la définition des éléments pour lesquels il faut permettre la réalisation d'interactions, un autre aspect à considérer pour ces systèmes collaboratifs est la temporalité. Sans rentrer dans toutes les variantes des réseaux collaboratifs, il est possible de citer l'Entreprise Virtuelle [CAM 09] qui se caractérise par une alliance temporaire entre diverses organisations afin de répondre efficacement à une opportunité d'affaires. Ce type de réseau collaboratif permet de prendre conscience de deux aspects à considérer lors de la mise en œuvre d'un système collaboratif.

D'une part, le caractère limité dans le temps de la relation impose d'établir et de permettre des interactions dans des délais limités. Plus ce délai est limité et plus le temps de collaboration consacré à la réalisation effective des missions et l'atteinte des objectifs est important. D'autre part, cette limitation doit permettre à une entité de créer et de multiplier les collaborations. Ainsi, en plus de faciliter la mise en relation, il faut également tenir compte de l'achèvement de la relation. En ce sens, la collaboration et la mise en relation ne doivent pas impacter une entité au point de modifier en profondeur ses propres processus, méthodes, ressources ou encore outils et la pénaliser en termes de temps pour revenir à son état initial (ex. performance) ou pour établir une autre collaboration. Ces deux aspects rejoignent la notion d'autonomie exprimée initialement dans les caractéristiques des réseaux collaboratifs.

Ainsi, la mutation des entreprises vers des structures d'organisations collaboratives complexes à appréhender et à mettre en œuvre montre que la réussite de ces collaborations ne dépend pas seulement de l'atteinte des objectifs classiques liés à leur mission. Elle dépend également de la capacité à concevoir, mettre en œuvre et réaliser « justement » la collaboration en fonction du degré relationnel attendu ainsi qu'à établir et maintenir des interactions cohérentes et efficaces sans toutefois perturber l'autonomie des organisations participantes. En ce sens, il est nécessaire de s'assurer – en sus de la réalisation des missions - que le niveau de performance attendu des **interactions**, la **nature hétérogène** des entités ou encore leur **autonomie** sont respectés.

En conséquence, la réalisation de systèmes collaboratifs doit s'appuyer sur des méthodes et des outils rigoureux et éprouvés afin de s'assurer que les contraintes comme celles qui ont été définies précédemment soient pleinement considérées.

2.2. ...Vers l'Ingénierie des Systèmes de Systèmes⁵

Un système de systèmes se définit dans sa forme la plus simple comme « *un ensemble de systèmes intégrés collaboratifs* » [MAI 98]. Si lorsqu'on parle de SdS [GOR 04], nous sommes bien en présence de systèmes collaboratifs, d'autres auteurs [DEL 04] [SHEN 9] [HOL 95] [DEL 04] [KOT 97] donnent une définition plus précise de ces systèmes. Ces définitions concernent divers domaines mais le concept sous-jacent reste toujours le même : « *un système de systèmes est l'intégration d'un nombre fini de constituants qui sont indépendants et opérationnels et mis en réseau pour une période de temps définie pour atteindre des objectifs de plus haut niveau⁶* » [JAS 09]. Un SdS se démarque des autres systèmes par un ensemble de caractéristiques telles que :

- **L'indépendance opérationnelle** des constituants. Cela représente la capacité d'un constituant donné à opérer indépendamment et efficacement lorsque le SdS est désassemblé ou que le constituant n'est plus impliqué dans le SdS. Ainsi, lorsque le SdS est dissolu ou que le constituant a terminé sa mission au sein de ce dernier, il doit retrouver le niveau de performance opérationnelle avant sa participation au SdS et ce, quelles que soient les modifications opérées pour assurer sa participation.
- **L'indépendance managériale** des constituants. Cela représente la capacité d'un constituant à garder et à continuer à poursuivre ses objectifs opérationnels tout en participant au SdS. Ainsi, un constituant peut être amené à mettre en jeu des ressources ou des activités au sein d'un SdS tout en poursuivant ses missions. Dans le cas où les objectifs de la mission du constituant sont partiellement atteints du fait de sa participation au SdS il doit toutefois être en mesure « d'opérer à nouveau efficacement » à la dissolution du SdS (*cf.* indépendance opérationnelle).
- **La distribution géographique** des constituants. L'ensemble des constituants sont répartis géographiquement à une échelle large (*ex.* différents continents).
- **L'évolution**. Le développement et l'existence du SdS sont évolutifs. Des constituants, des fonctions de ses constituants, des missions ou encore des objectifs peuvent être ajoutés, enlevés ou bien modifiés.

⁵ Par la suite, l'acronyme SdS est utilisé.

⁶ Par « objectifs de plus haut niveau », il est entendu des objectifs qu'un constituant isolé n'est pas en capacité d'atteindre. Pour illustrer ce point, [JAS 09] prend en exemple un instrument qui ne peut réaliser seul une symphonie qui sera obtenue par le jeu de l'ensemble des instruments d'un orchestre.

- **L'émergence.** Des comportements émergents –prévus ou non – peuvent apparaître suite aux interactions entre les constituants. L'exemple le plus parlant est, la réalisation de la mission du SdS qui émerge de l'interaction de ses constituants.

Les trois premières caractéristiques sont classiques et se retrouvent également dans la définition des systèmes complexes [SIM 81]. En revanche, les deux premières sont discriminantes et permettent d'utiliser le terme de « Système de Systèmes ». La Figure 2 présente la notion de système complexe et de système de systèmes.

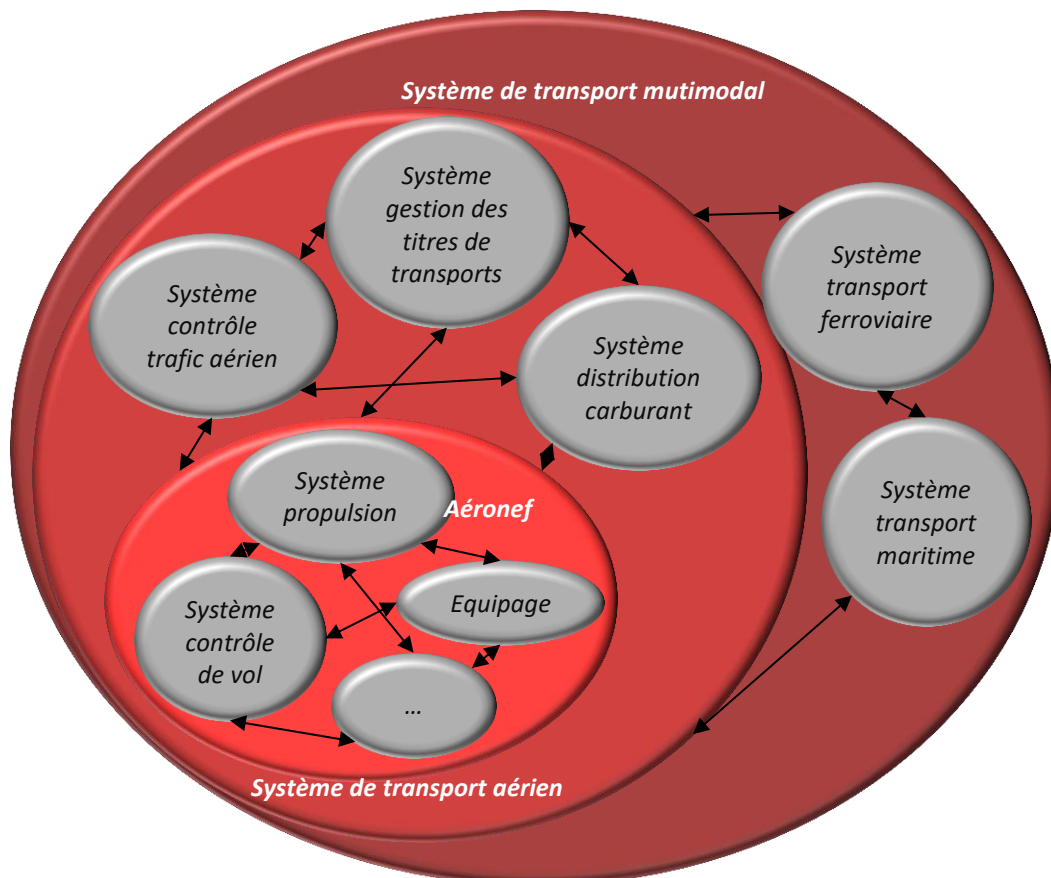


Figure 2. Système Complexe (ex. aéronef) et Système de Systèmes (ex. système de transport aérien, système de transport multimodal) [BLA 11]

Notons également que ces premières caractéristiques ont été étendues avec [DEL 04]:

- **L'hétérogénéité.** Les constituants du SdS sont de différentes natures et opèrent à des échelles de temps différentes.
- **L'interdisciplinarité.** L'étude des SdS nécessite la mise en commun de différentes connaissances comme l'économie, l'ingénierie et la gestion des opérations.
- **Les réseaux.** La connectivité entre les constituants du SdS est définie au travers de règles d'interactions qui sont nécessaires au SdS pour poursuivre sa mission.

On retrouve naturellement dans le concept de SdS les caractéristiques qui définissent également un réseau collaboratif telles que l'indépendance (aussi appelée autonomie par certains auteurs [BOA 06]), l'hétérogénéité ou encore la réalisation d'interactions dans le but de réaliser une mission.

Cependant, la vision SdS va bien au-delà de la simple définition des attendus comme cela peut être fait dans la définition des réseaux collaboratifs et de ses variantes. En effet, le développement des SdS est supporté par une approche appelée Ingénierie des Systèmes de Systèmes (SOSE). Cette approche n'est en soi pas nouvelle puisque celle-ci prend son essence dans l'approche d'Ingénierie Système classique qui permet de « *mettre en œuvre les moyens et les disciplines nécessaires pour permettre la réalisation d'un système qui satisfait les exigences des utilisateurs. Pour cela elle se concentre sur la définition des exigences des utilisateurs et les fonctionnalités requises, la documentation des exigences puis la conception et la validation du système tout en considérant le problème dans sa globalité* » [INC 11].

Ainsi, en s'appuyant sur l'ingénierie système, l'objectif est le développement d'un SdS qui, à terme, doit remplir la mission pour laquelle il est développé. Cependant le challenge n'est pas essentiellement la mise en œuvre d'activités d'ingénierie qui ont fait leurs preuves mais de pouvoir adapter ces activités aux SdS et intégrer leurs caractéristiques. Dès lors, cela ne consiste plus uniquement en une simple ingénierie des systèmes (ex. prise en compte de constituants qui sont eux-mêmes des systèmes complexes) car il est également nécessaire d'identifier tous les aspects à considérer pour obtenir un SdS cohérent. En ce sens, [VAL 07] propose une distinction entre l'ingénierie système et l'ingénierie des systèmes de systèmes selon différents domaines d'intérêt. Notons que les domaines d'intérêt présentés ci-dessous (*Tableau 1*) constituent la base de différenciation principale. En effet, d'autres sont identifiés en plus dans la littérature [GOR 08] [DAH 08] [HU 15].

	Ingénierie Système	Ingénierie des Systèmes de Systèmes
Objectif	Système unique qui satisfait les exigences des utilisateurs et les performances définies	Nouvelles capacités du système de systèmes en tirant partie des synergies entre les systèmes constituants
Architecture	Définie tôt dans les phases amont du cycle de vie, les exigences restent stables	Adaptation dynamique face à la modification ou l'apparition de nouvelles exigences
Interopérabilité	Exigences d'interface définies pour l'intégration des composants au système	Les constituants peuvent opérer indépendamment du SdS de manière efficace. Les protocoles et les standards sont indispensables pour permettre l'interopérabilité
« -ilities »	Classiques (ex. disponibilité, maintenabilité)	Attention particulière sur les « -ilities » comme la flexibilité ou l'adaptabilité
Management	Management centralisé	Les constituants sont gérés et opèrent individuellement
Anticipation des besoins	Phase de l'activité de conception pour déterminer les besoins	Phase intense d'analyse suivie par une anticipation continue et supportée par l'expérimentation
Coût	Partie prenante unique ou homogène avec un profil de coût et des critères de mesures du succès similaires	Parties prenantes hétérogènes et multiples avec des profils de coûts et des critères de mesure du succès instable

Tableau 1. Domaines d'intérêt pour l'Ingénierie des Systèmes et l'ingénierie des Systèmes de Systèmes [VAL 07]

Parmi ces différents points du SoSE, il en est un qui se concentre particulièrement sur la notion d'autonomie tout en permettant la participation d'un constituant au SdS à savoir, l'interopérabilité. L'interopérabilité est identifiée comme une « -ility⁷ » majeure à prendre en compte au cours de l'ingénierie d'un SdS. Elle « *caractérise que des systèmes peuvent fonctionner de manière indépendante mais également ensemble et ce bien qu'ils n'aient pas été conçus pour le faire* » [DEW 12]. Elle est parfaitement identifiée dans la littérature comme partie intégrante de l'ingénierie des SdS [MOR 04] et même considérée dans [KAN 14] comme un facteur clé permettant aux constituants du SdS de travailler ensemble de manière cohérente. En d'autres termes, sans la prise en compte de l'interopérabilité lorsqu'on met en œuvre un SdS, ce dernier peut ne pas être en mesure d'atteindre les objectifs et, dans le cas le plus défavorable, dans l'impossibilité de réaliser sa mission.

Ainsi, le SoSE doit tenir compte de ces spécificités par rapport à un système traditionnel. Pour ce faire, elle tient compte et s'appuie sur l'ingénierie systèmes qui définit l'ensemble de processus à mettre en œuvre pour obtenir un système satisfaisant au regard des exigences des parties prenantes, des entités avec leurs exigences propres et leurs contraintes dans le cas d'un SdS.

Elle doit également parfaitement identifier et définir les exigences qui participent pleinement au fonctionnement et, par extension, à la réussite d'un SdS. C'est le cas notamment de l'interopérabilité, « -ility » qui a émergé naturellement avec l'apparition des réseaux collaboratifs par exemple. Bien que ce ne soit pas la seule « -ility » d'importance dans la réalisation d'un SdS, elle permet de s'assurer que les caractéristiques d'indépendance, d'hétérogénéité, de réalisation des interactions, sont atteintes.

Mes travaux de recherche portent sur le développement de l'interopérabilité dans les systèmes collaboratifs vus comme des SdS. Ils s'inscrivent donc dans le cadre de l'ingénierie des SdS (ou de l'amélioration de SdS existants) en s'appuyant sur les principes de la modélisation d'entreprise, de l'interopérabilité des systèmes et de l'ingénierie système.

⁷ Pluriel « -ilities » : « *Exigences de développement, opérationnelles et de support auxquelles un programme doit répondre* ». [INC 11]. « *Ce sont des propriétés comme la maintenabilité ou la flexibilité qui se manifestent souvent après la mise en condition opérationnelle d'un système* » [DEW 12]. Le terme « -ilities » est anglophone et caractérise un ensemble d'exigences non fonctionnelles se terminant par ce suffixe (ex. flexibilité). Toutefois, certaines « -ilities » ne respectent pas cette construction (ex. résilience).

2.3. De l'interopérabilité des systèmes...

Selon [PIN 09] l'interopérabilité peut être définie comme « *une capacité de systèmes, nativement étrangers les uns par rapport aux autres, à interagir afin d'établir des comportements collectifs harmonieux et finalisés, sans avoir à modifier en profondeur leur structure ou leur comportement individuel* ». Sur la base de cette définition, l'interopérabilité se caractérise par :

- La présence de **systèmes hétérogènes** (nativement étrangers) non nécessairement développés pour travailler ensemble. Dans ce cas, il s'agit de s'assurer que ces systèmes sont **compatibles** et peuvent s'interfacer sans efforts particuliers afin de réaliser leur collaboration.
- Un ensemble d'**interactions** pour réaliser un travail collectif et cohérent dans le cadre d'une mission et d'objectifs identifiés (interagir pour établir un comportement collectif harmonieux et finalisé). Les systèmes exécutent, dans ce cas, un ensemble d'**interopérations** c'est-à-dire un ensemble d'échanges, de partages de données, de ressources, ou encore d'activités en s'assurant de leur synchronisation.
- L'**indépendance et l'autonomie** par la garantie que les systèmes impliqués garderont leurs identités propres (non modification en profondeur de structures et comportements individuels). Les systèmes peuvent participer à la collaboration tout en continuant leur propre mission (**autonomie**). Dans le cas où la mise en œuvre de l'interopérabilité nécessite des modifications (ex. diminution des performances de la mission), les systèmes peuvent revenir à leur état initial d'avant la collaboration (**réversibilité**).

Cette définition est cohérente avec l'ensemble des définitions de l'interopérabilité [WEG 96] [VER 96] [ISO 09] et couvre bien le spectre des caractéristiques attendues de l'interopérabilité des systèmes de systèmes en termes d'indépendance, d'interactions ou de préservation de l'identité des constituants. Pour traiter de la question de l'interopérabilité, de nombreuses initiatives en termes de recherche – particulièrement l'interopérabilité des entreprises - ont vu le jour ces 20 dernières années. Bien qu'identifiée depuis plus longtemps comme un frein potentiel au travail entre des organisations principalement au niveau technique et au sein des systèmes d'information, l'interopérabilité n'est plus aujourd'hui limitée à ces domaines. Elle concerne également les aspects organisationnels et conceptuels. Le corollaire direct est que l'interopérabilité ne se focalise plus exclusivement sur l'échange et le partage de données entre des entités mais inclut également, par exemple, les aspects d'échange et de partage de ressources ou, encore, est étudiée au sein des processus collaboratifs. L'interopérabilité est une notion dont la compréhension et la définition ont évolué rendant sa mise en œuvre, son suivi ou encore son amélioration plus complexe. Cela a nécessité *de facto* le développement de solutions conceptuelles, organisationnelles et techniques pour répondre efficacement aux diverses problématiques qu'elle peut représenter.

L'interopérabilité est ainsi devenue un enjeu majeur et un facteur clé de réussite de projets impliquant diverses organisations qui doivent collaborer et ce, dans de nombreux domaines [ISO 02] [IEE 90] [DoD 01a] [BEN 08]. Elle est également étudiée sous différents points de vue en termes de caractérisation [CHE 08], d'implantation [DAC 16] [BOU 07] [CHA 15] ou encore d'évaluation [C4I 98] [FOR 08].

Dans le domaine de l'interopérabilité d'entreprise, nous retenons que [ISO 09] définit cette dernière comme « *la capacité des entreprises et des entités au sein de ces entreprises à communiquer et à interagir* ». Pour aller plus loin dans la compréhension de l'interopérabilité, l'ISO, sur la base des travaux de [CHE 06], propose également de structurer l'interopérabilité selon diverses dimensions à savoir :

- Les **niveaux d'interopérabilité**. Il s'agit ici des points de vue à adopter au sein d'une organisation pour envisager et développer l'interopérabilité. Les niveaux d'interopérabilité sont au nombre de quatre :
 - Le niveau « métiers » traite de l'organisation globale des organisations qui collaborent en termes de prise de décision, de méthodes de travail ou encore de fonctionnement.
 - Le niveau « processus » vise à mettre en relation des processus propres aux organisations qui collaborent (processus internes). Il s'agit à ce niveau de faire émerger un processus commun de travail entre les organisations (processus collaboratifs) et impliquant différentes activités propres à chaque partie.
 - Le niveau « services » identifie et met en relation les différentes applications ou fonctions des organisations déployées au sein des processus.
 - Le niveau « données » assure l'échange et le partage de modèle de données hétérogènes présents au sein de bases de données différentes et déployés au travers d'outils différents au cours de la collaboration
- Les **approches d'interopérabilité**. Ces approches sont décrites initialement dans la norme ISO 14258 [ISO 99]. Elles définissent le moyen (de manière générale) d'appréhender l'interopérabilité pour assurer son développement et sont de trois types à savoir :
 - L'intégration. La mise en œuvre de cette approche s'appuie sur la présence d'un format commun et unique à toutes les entités qui collaborent (ex. format de données unique)
 - L'unification. Cette approche n'utilise pas de format unique mais s'appuie, cependant, sur la mise à disposition d'un méta modèle qui permet d'assurer une équivalence entre les divers formats utilisés par les organisations (ex. équivalence sémantique).
 - La fédération. Pour cette approche, il n'existe ni format unique ni de méta modèle prédéterminé. Il suppose donc que sans avoir de connaissances l'une de l'autre, les organisations vont devoir s'adapter dynamiquement pour assurer l'interopérabilité.
- Les **barrières d'interopérabilité** (aussi appelées catégories d'interopérabilité). Elles représentent les obstacles (problèmes) à l'interopérabilité entre les organisations et sont au nombre de trois :
 - Conceptuelle. Cette barrière est liée aux aspects sémantiques (sens) et syntaxiques (structure) des informations qui sont partagées entre les organisations. Cette catégorie intègre également l'expressivité de l'information à échanger, c'est-à-dire la capacité d'expression (ex. symbole graphique) de cette dernière.
 - Technologique. Cette barrière est liée aux problèmes d'incompatibilités entre des applications et des Systèmes d'Information des organisations.
 - Organisationnelle. Cette dernière catégorie concerne la définition des responsabilités et d'autorisation des acteurs des organisations qui collaborent ainsi que les aspects d'autorités, processus et réglementaires.

La réunion de ces trois dimensions a donné naissance au cadre d'interopérabilité d'entreprise (Figure 3).

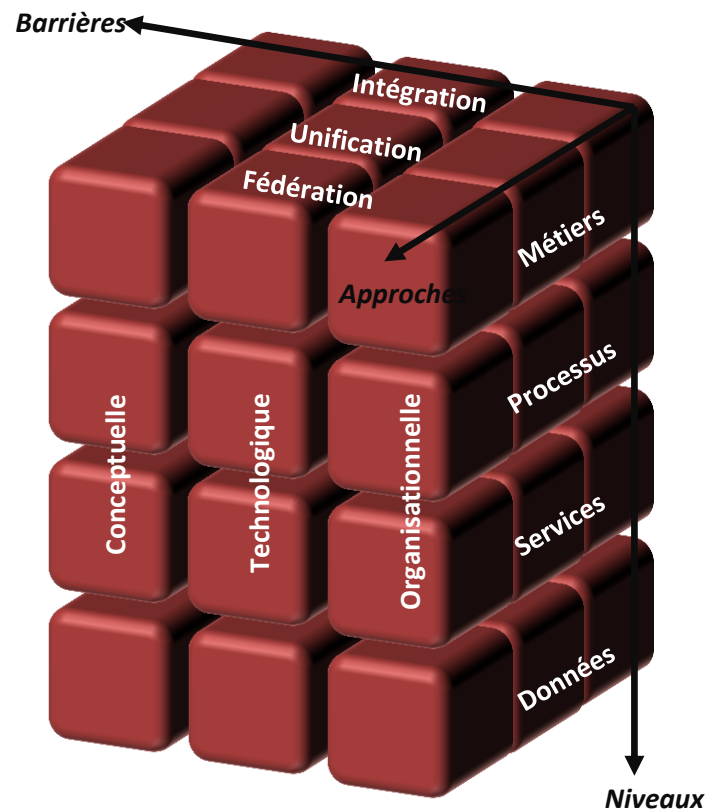


Figure 3. Cadre d'interopérabilité d'entreprise (EIF – Enterprise Interoperability Framework, [ISO 09])

Au-delà de la simple structuration des différents aspects de l'interopérabilité, le cadre permet de collecter les solutions d'interopérabilité pour résoudre les problèmes qui peuvent apparaître pour un niveau et selon une approche d'interopérabilité. Ainsi, ce ne sont pas moins d'une soixantaine de solutions qui sont identifiées [INT 07a] dans le cadre. Une solution peut couvrir plusieurs niveaux, catégories. De plus, plusieurs solutions peuvent satisfaire le même problème.

Il peut être alors difficile pour un utilisateur d'identifier la solution la mieux adaptée à son problème sur la simple expression de ses besoins en termes de niveaux d'interopérabilité, d'approches ou encore de barrières d'interopérabilité. La question de mettre à disposition un ensemble de solutions pour l'interopérabilité est tout à fait pertinente. Cependant, il apparaît nécessaire de permettre également à des organisations de définir, en amont du choix des solutions, leurs besoins en termes d'interopérabilité. Cela permet notamment d'éviter de sélectionner une solution qui, au final, ne résout que partiellement le problème voire, dans le cas le plus défavorable, n'apporte aucune solution.

Cet aspect des « besoins » semble pourtant avoir été considéré par le cadre d'interopérabilité car on retrouve une dimension « ingénierie » avec une phase de définition des exigences. Cependant, elle n'est pas développée et considérée plus en détail comme l'ingénierie des exigences est considérée en ingénierie des systèmes. Elle se limite à la simple expression du problème selon les trois dimensions du cadre.

Le cadre d'interopérabilité d'entreprise n'est pas le seul à structurer l'interopérabilité selon différentes dimensions [EC 08] [NEH 07] [ATH 07]. Dans la majorité des cas, ils tiennent compte des dimensions concernant les problèmes d'interopérabilité, les niveaux des systèmes auxquels l'interopérabilité doit être établie et enfin les solutions mais l'interopérabilité n'est toujours pas abordée de manière explicite comme une exigence à satisfaire.

Ainsi, l'interopérabilité est bien identifiée comme un problème à traiter (et donc en soit comme une exigence) à différents niveaux et notamment des processus collaboratifs. Ce choix de traiter l'interopérabilité dans les processus collaboratifs est pertinent et c'est sur ce niveau que se concentrent les travaux présentés. Cela permet selon la classification [ISO 09] de considérer le niveau supérieur (métier) mais également les niveaux inférieurs (services et données) sachant que ces niveaux sont liés pour la réalisation des objectifs du système. En effet, la réalisation des « métiers s'appuie sur la mise en œuvre de processus. Les processus sont réalisés par des services qui utilisent eux-mêmes des données. » [ISO 09]. De même, dans certains cas, la notion d'ingénierie et d'exigence est brièvement mentionnée. Cependant, l'interopérabilité n'est pas considérée comme une exigence au sens stricte du terme c'est-à-dire telle qu'une exigence peut être vue par exemple en ingénierie des systèmes, *i.e.*, depuis son élicitation, son expression, son évaluation et jusqu'à la sélection d'une solution qui la satisfait.

Les travaux de recherche réalisés depuis l'obtention de mon doctorat considèrent l'interopérabilité du point de vue des exigences tel qu'il est appréhendé en ingénierie systèmes. Cela implique d'étudier « l'exigence d'interopérabilité » dans ses dimensions de gestion, d'expression et d'évaluation dans les processus collaboratifs pour guider les acteurs des systèmes collaboratifs dans le développement de l'interopérabilité.

2.4. ... A l'exigence d'interopérabilité

Première dimension : une exigence à gérer et à exprimer

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 » [SUC 08]. Traiter l'interopérabilité comme une exigence, c'est dans un premier temps traiter la problématique de l'ingénierie des exigences⁸ pour un énoncé qui définit ce qui est prescrit en termes d'interopérabilité. Plus précisément, cet énoncé est lié à l'interopérabilité « lorsque des organisations hétérogènes travaillent ensemble, que cela exige le partage, l'échange, l'utilisation, la permission d'accès... à des fonctionnalités, ressources, données... entre ces mêmes organisations et, lorsque ces éléments de différentes organisations peuvent impacter leur relation ». Dès lors, on saisit aisément la difficulté à exprimer précisément cette exigence et la nécessité de l'appréhender de manière rigoureuse et dans toutes ses dimensions.

⁸ « Activité (ou discipline) méthodique pluridisciplinaire permettant de recueillir, de spécifier, d'affiner et de qualifier des exigences relatives à un système à positionner dans son environnement, et de maîtriser les évolutions de ces exigences dans le temps. » [MEI 12]

Une exigence fixe (ou contraint) sans ambiguïté et de manière cohérente ce qui doit être respecté et réalisé lorsqu'on développe une solution. Le processus d'ingénierie des exigences repose sur deux points essentiels [FAV 12] à savoir :

- La **gestion** des exigences qui se concentre sur les aspects de structuration des exigences, d'accès aux exigences, de gestion des versions, d'historique ou encore de traçabilité des exigences.
- L'**écriture** des exigences c'est-à-dire l'expression de l'énoncé qui prescrit ce qui est attendu.

De natures différentes, ces deux points n'en restent pas moins complémentaires. En effet, ils permettent à des parties prenantes de développer ou de sélectionner une solution parmi un ensemble de solutions candidates qui permettra de satisfaire un ensemble d'exigences. Ce processus revêt une importance capitale dans la recherche d'une solution satisfaisante et donc dans la réussite (et inversement dans l'échec) d'un projet [STA 16].

Cependant, bien que l'intérêt de l'ingénierie des exigences soit reconnu, cette activité reste encore trop souvent négligée et ce pour différentes raisons. Tout d'abord, cela peut venir d'un manque de connaissances sur la rédaction d'une exigence ou encore le manque de vision sur les exigences elles-mêmes (quelles sont mes attentes au regard du système que je souhaite ?). Ensuite, l'intérêt de ce processus peut être mal compris par les parties prenantes. En effet, les exigences sont souvent perçues comme des évidences (quel est l'intérêt de décrire de manière explicite qu'un smartphone doit émettre et recevoir des appels ?) et le processus d'ingénierie des exigences comme générateur de nombreux documents et ne présentant pas de réelle valeur ajoutée à court terme. Evidemment, le processus d'ingénierie des exigences est chronophage et consommateur de ressources. Cependant, réalisé de manière rigoureuse, il permet d'éviter des coûts de correction importants et des dérives de temps pour un projet [INC 11]. A cette fin, ce processus suit un ensemble de règles strictes et parfaitement identifiées.

Le premier point du traitement de l'interopérabilité comme une exigence est d'assurer correctement la gestion et l'utilisation des exigences. A cette fin, celles-ci doivent être positionnées au sein d'un référentiel d'exigences. Ce référentiel doit pour permettre la traçabilité, l'ajout, le retrait ou la modification d'exigences et enfin servir de référence dans le choix de solutions qui satisfont les exigences énoncées. Dans le cadre du développement de l'interopérabilité comme une exigence, il revêt une importance particulière. En effet, ce référentiel est partagé entre les différents participants du système collaboratif et contient des exigences d'interopérabilité qui concerne le système collaboratif (*ex.* exigence concernant les interactions) et les sous-systèmes (*ex.* autonomie). De même, certaines de ces exigences peuvent être dépendantes et s'impacter mutuellement, *i.e.*, une exigence d'un sous-système peut être incohérente avec une exigence du système collaboratif (*ex.* une ressource partagée n'est pas disponible pour réaliser une fonction de la collaboration).

Ainsi, un référentiel d'exigences d'interopérabilité doit donc respecter les caractéristiques de qualité d'un référentiel classique défini dans l'ingénierie des systèmes à savoir [INC 12]:

- La complétude. Le référentiel doit inclure la définition complète des exigences en termes d'interopérabilité.
- La cohérence. Il n'y a pas de conflits entre les exigences d'interopérabilité.
- La faisabilité. Les exigences d'interopérabilité sont couvertes par une ou plusieurs solutions pour les satisfaire.
- La limite. Le référentiel est uniquement limité aux concepts de l'interopérabilité.
- La structuration. Le référentiel est aisément manipulable sans effort cognitif important pour l'utilisateur.

Dans l'objectif de développement de l'interopérabilité dans les systèmes collaboratifs, la proposition d'un référentiel d'exigences pour l'interopérabilité doit s'appuyer sur les différents travaux qui structurent l'interopérabilité et notamment les cadres d'interopérabilité. Les différents points de vues, problèmes ou encore approches doivent être considérés, car à termes, l'objectif est bel et bien de guider le choix d'une solution au regard de l'exigence à satisfaire. Si le cadre d'interopérabilité des entreprises présenté précédemment offre une base solide pour la structuration de l'interopérabilité, il propose uniquement un ensemble de solutions sans liens directs avec un ensemble d'exigences d'interopérabilité. Il est donc indispensable d'assurer la mise en relation des exigences qui sont énoncées avec les solutions identifiées au sein du cadre. [DAC 15]

Le deuxième point du traitement de l'interopérabilité comme une exigence concerne la problématique d'expression des exigences. En ingénierie des exigences, des règles de bonnes pratiques guident une partie prenante dans la rédaction d'exigences qui respectent un ensemble de qualités propres.

A cette fin, il est possible de mentionner les acronymes tels que MUST⁹, SMART¹⁰ ou encore les recommandations de l'ISO [ISO 11] pour s'assurer qu'une exigence est correctement exprimée, diffusable et exploitable par l'ensemble des destinataires qui la manipulent. Plus précisément, les règles de bases suivantes permettent d'obtenir des exigences bien formées :

- L'unicité. Une exigence ne traite que d'un seul problème/sujet (une fonction, une contrainte, une performance).
- La non ambiguïté. Tous les destinataires ont la même compréhension de l'exigence.
- La faisabilité. Une exigence est réalisable (ex. la performance attendue du système peut effectivement être obtenue par une ou un ensemble de solutions).
- La simplicité. L'expression d'une exigence est simple, précise et concise.
- La vérification. Une exigence est vérifiable (expertise, analyse, test, simulation, vérification formelle) pour savoir si elle est satisfaite ou non.

⁹ MUST : Mesurable, Utile, Simple, Traçable

¹⁰ SMART : Spécifique, Mesurable, Atteignable, Réaliste, Traçable (ou Temporellement limitée)

Ces caractéristiques représentent la qualité d'une exigence. Il est à noter que d'autres caractéristiques sont identifiées et définies dans la littérature. Pour exprimer une exigence qui respecte les caractéristiques mentionnées précédemment, une partie prenante dispose de divers langages partant du langage naturel jusqu'aux langages formels *via* les langages semi-formels, chacun présentant des avantages et des inconvénients. Les langages naturels offrent une bonne lisibilité (l'exigence peut être comprise par n'importe quelle partie prenante à supposer qu'on parle le langage utilisé), de la richesse (la même exigence peut être exprimée avec différents termes) ou encore la possibilité d'être étendu (création de nouveaux termes, concepts). En revanche, il existe un risque fort d'ambiguïté entre les termes et un manque de précision (*ex.* termes vagues). De plus, l'évaluation de la satisfaction des exigences exprimées en langage naturel s'appuie habituellement sur l'expertise humaine. Les langages semi-formels s'appuient sur des formalismes graphiques qui sont aisés à appréhender, à utiliser et plus facilement partageables que des exigences en langage naturel. Le risque d'ambiguïté est toujours présent et l'évaluation des exigences s'appuient encore sur l'expertise humaine, cependant, certains langages peuvent s'appuyer sur des langages et outils qui possèdent des bases mathématiques qui peuvent autoriser l'utilisation d'outils de vérification formelle [CHAP 07]. Enfin, les langages formels s'appuyant sur des symboles mathématiques sont difficilement utilisables pour des utilisateurs non avertis et demandent un effort d'apprentissage important. En revanche, dans ce dernier cas, l'évaluation des exigences s'appuie sur des outils puissants et éprouvés (Tableau 2).

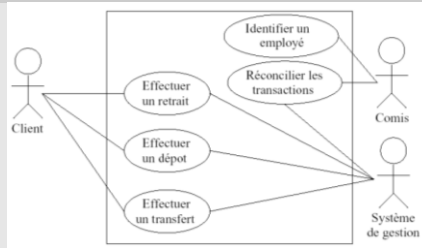
Langage	Avantages	Inconvénients	Tech. évaluation	Exemples
Naturel	Lisibilité, richesse, extensibilité	Ambiguïté, manque de précision	Expertise, test	« En mode dégradé, le système enregistre les paramètres de température »
Semi-formel	Facilité d'apprentissage et d'utilisation	Ambiguïté	Expertise, vérification formelle	
Formel	Grande précision, vocabulaire restreint	Faible lisibilité, difficulté d'apprentissage	Vérification formelle	« $AG(failure \Rightarrow AF_{<5} alarm)$ » ¹¹

Tableau 2. Langages d'expression des exigences et caractéristiques

Une première contribution concerne la dimension de gestion et d'expression de l'exigence d'interopérabilité selon le paradigme de l'ingénierie des exigences.

Enfin, dans le cadre du développement d'un langage pour l'expression d'exigence d'interopérabilité, ce dernier doit être suffisamment précis, simple et permettre d'exprimer des exigences dédiées à l'interopérabilité qui soit compréhensible et manipulable par les experts de l'interopérabilité.

¹¹ « L'alarme se déclenche en moins de 5 secondes après un défaut ».

Comme cela est abordé très brièvement, une exigence se doit d'être évaluée. En effet, s'arrêter à l'expression des exigences d'interopérabilité présente un intérêt limité si on ne peut s'assurer de leur satisfaction (ou non) par le système qui est étudié. Ainsi, au-delà de la simple expression d'une exigence d'interopérabilité et de sa gestion, il est nécessaire dans un second temps de permettre à des parties prenantes d'évaluer cette exigence.

Deuxième dimension : une exigence à évaluer

L'évaluation des exigences n'est pas, à proprement parler, une activité de l'ingénierie des exigences. Cependant, c'est une étape importante du cycle de vie d'une exigence du fait que : (1) une exigence doit pouvoir être évaluée (cf. qualité attendue d'une exigence) et (2), que le résultat de cette évaluation (satisfaction ou non de l'exigence) doit être porté à la connaissance des parties prenantes.

Ainsi, établir l'interopérabilité, atteindre un niveau d'interopérabilité souhaité ou encore implanter des solutions d'interopérabilité nécessite la connaissance de l'aptitude effective de différents partenaires à inter-opérer.

Divers moyens et méthodes pour évaluer l'interopérabilité des systèmes sont développés [REZ 14] pour avoir une vision du niveau d'interopérabilité de manière qualitative ou quantitative [BOT 16]. Sans se prétendre exhaustif, le *Tableau 3* présente diverses méthodes d'évaluation de l'interopérabilité (organisationnelle, sémantique, capacité, compatibilité...).

Quantitative	Qualitative
<ul style="list-style-type: none"> • Modes d'interopérabilité [FOR 08] • Attributs de qualité [KAS 04] • Interopérabilité sémantique [YAH 12] • Performance d'interopérabilité [DAC 15] • Ratio of interoperability [ELM 11] 	<ul style="list-style-type: none"> • LISI (Levels of Information Systems Interoperability) [C4I 98] • OIM (Organizational Interoperability Maturity Model) [FEW 03] • LCIM (Levels of Conceptual Interoperability Model) [TOL 07] • EIMM (Enterprise Interoperability Maturity Model) [ATH 05] • MMEI (Maturity Model for Enterprise Interoperability) [GUE 12] • Matrice de compatibilité [DAC 15] • Interview [MOR 04]

Tableau 3. Evaluation quantitative et qualitative de l'interopérabilité

L'évaluation quantitative de l'interopérabilité s'appuie largement sur une formalisation mathématique de l'interopérabilité et permet d'obtenir un résultat précis (ex. obtention d'un temps, d'un taux de qualité, d'une capacité...). C'est le cas, par exemple, de la mesure de la latence de données, mesure de temps qui est décomposée en différentes périodes de temps pour permettre une analyse fine et ainsi identifier parfaitement le segment incriminé pour mettre en œuvre des corrections adaptées. La latence des données est exprimée formellement ainsi [KAS 04] :

$$\overline{\Delta t} = t_r - t_e \tag{1}$$

$$\overline{\Delta t_o} = t_o - t_e \tag{2}$$

$$\overline{\Delta t_m} = t_{mr} - t_o \tag{3}$$

$$\overline{\Delta t_r} = t_r - t_m \tag{4}$$

Où,

- Δt : temps de latence, temps écoulé entre la date de l'évènement et la date de réception par l'utilisateur (latence de données).
- Δt_o : latence d'observation, temps écoulé entre la date de l'évènement et la date d'observation.
- Δt_m : latence de traitement, temps écoulé entre la date d'observation et la date d'achèvement du traitement.
- Δt_r : latence de réception, temps écoulé entre la date de fin de traitement et la date de réception par l'utilisateur.
- t_e : date de l'évènement.
- t_o : date d'observation.
- t_m : date de fin de traitement.
- t_r : date de réception.

Les approches d'évaluation de l'interopérabilité quantitatives et *via* des expressions mathématiques comme celles présentées ci-dessus permettent par des mesures formelles et quantitatives de réaliser une première analyse de l'interopérabilité vue comme une exigence. Premièrement ces outils ne sont pas directement accessibles à des utilisateurs non avertis et nécessitent une phase d'apprentissage qui peut être longue. Bien que l'exemple ci-dessous soit rapidement compréhensible, d'autres approches d'évaluation de l'interopérabilité le sont beaucoup moins [YAH 12]. Ensuite, les expressions mathématiques proposées ne sont pas des exigences au sens strict du terme. En reprenant l'exemple ci-dessus, l'exigence de latence de données serait plutôt la performance attendue en termes de temps de latence (*ex.* la latence maximum est de n unité de temps) et non la mesure directe à l'aide des différentes expressions. Cependant, il est possible de considérer le temps de latence comme une exigence et de permettre à un utilisateur de l'exprimer. Les calculs sont alors réalisés pour s'assurer que l'exigence est satisfaite (aucun écart entre la latence maximum admissible et le résultat de son calcul) ou non. Enfin, un avantage certain de ces approches est qu'elles s'appuient sur des données mesurables et quantifiables qui ne souffrent pas d'ambiguïté ou d'erreurs d'interprétation, le biais pouvant venir tout de même d'une erreur au cours de la mesure.

L'évaluation qualitative de l'interopérabilité de son côté peut être beaucoup moins précise que l'évaluation quantitative (*ex.* résultat binaire – 0 ou 1, oui ou non, niveau « organisé »). Parmi l'ensemble des outils pour la mesure qualitative de l'interopérabilité, le modèle de maturité représente certainement le moyen le plus puissant et le plus précis. Les modèles de maturité permettent d'évaluer l'interopérabilité selon un point particulier (appelé point d'intérêt) sur plusieurs niveaux de maturité¹² (niveau de progression). Chaque intersection d'un modèle de maturité pour l'interopérabilité décrit de manière précise un ensemble d'attendus pour valider un niveau d'interopérabilité. De la même manière, un ensemble de recommandations et de bonnes pratiques est prescrit pour évoluer à travers les niveaux et ainsi atteindre la pleine interopérabilité.

¹² Un niveau de maturité est évalué par l'atteinte d'un ensemble de buts spécifiques ou génériques appliqué à un point d'intérêt. [PAU 93]

A titre d'exemple, le modèle de maturité pour l'interopérabilité des entreprises (MMEI – [GUE 12]) propose 4 niveaux de maturité (non préparé, défini, aligné, organisé et adaptatif) et prend pour zones d'intérêts les dimensions du cadre d'interopérabilité d'entreprise (barrières d'interopérabilité, niveaux d'interopérabilité - [ISO 09], Figure 4). Enfin, ces modèles de maturité s'accompagnent de démarches permettant, étape par étape, de réaliser une évaluation et une progression cohérente et efficace.

Maturity level	Description
Level 4 – Adaptive	Capability of negotiating and dynamically accommodating with any heterogeneous partner
Level 3 – Organized	Capability of performing needed mappings with multiple heterogeneous partners
Level 2 – Aligned	Capability of making necessary changes to align to common formats or standards
Level 1 – Defined	Capability of properly modelling and describing systems to prepare interoperability
Level 0 – Unprepared	Ad-hoc interoperability capabilities or no will to interoperate

	Conceptual	Technical	Organizational	
Level 3 - Organized	Business	Adaptive business model	Adaptive IT infrastructure	Agile organization for on-demand business
	Process	Modeling for dynamic process re-engineering	Dynamic and adaptive tools and engines for processes.	Real-time monitoring of processes, adaptive procedures
	Service	Adaptive service modeling	Dynamically composable services, networked applications	Dynamic service and application management rules and methods
	Data	Adaptive data models (both syntax and semantics)	Direct database exchanges capability and full data conversion tool	Adaptive data management rules and methods

Figure 4. Modèle de maturité pour l'interopérabilité des entreprises MMEI (description des niveaux de maturité et définition du niveau 3) [GUE 12]

L'avantage de ce type d'approches est la simplicité d'utilisation (un modèle de maturité ou un questionnaire ne requiert pas de connaissances particulières). Elles permettent également une auto-évaluation (pas d'expression formelle, recommandations pour évoluer dans le cas des modèles de maturité, guide et méthodes d'utilisation). En revanche, ces moyens d'évaluation s'appuient principalement sur l'expertise humaine avec la possibilité de mauvaise interprétation, de mauvaise identification du problème à traiter et de temps conséquent pour réaliser l'évaluation. Cela peut donc conduire à la sélection d'un ensemble de solutions non adaptées dans le cas d'un résultat qui ne reflète pas la réalité (ex. mauvaise évaluation du niveau de maturité). Du point de vue des exigences, les approches qualitatives définissent un ensemble d'attendus auquel les systèmes doivent répondre en termes d'interopérabilité. Cependant ces attendus ne sont pas exprimés ou encore structurés comme peuvent l'être des exigences et comme cela est recommandé par les bonnes pratiques.

Ainsi, les travaux sur l'évaluation (quantitative et qualitative) de l'interopérabilité n'expriment pas d'exigences d'interopérabilité explicitement ou non formées comme l'ingénierie des exigences le recommande. Ils permettent néanmoins d'élucider un ensemble d'attendus en termes d'interopérabilité qui peut être exprimé sous la forme d'exigences. C'est donc dans un premier temps sur la base de ces travaux que sont exprimées les exigences d'interopérabilité. De la même manière, d'autres travaux (liés ou non directement à l'interopérabilité) peuvent être utilisés pour dégager des exigences d'interopérabilité.

C'est le cas notamment de la dernière version du TOGAF [TOG 11] qui met en avant la nécessité de définir des exigences d'interopérabilité et fournit des recommandations pour supporter leur définition. C'est également le cas des SLA¹³ qui définissent contractuellement les qualités de services attendus entre différentes parties et qui peuvent être vus comme des exigences [DES 10].

Ainsi, qu'elles soient quantitatives ou qualitatives, les approches d'évaluation de l'interopérabilité présentent des avantages et des inconvénients lorsque le point de vue de l'ingénierie des exigences est adopté. Il est intéressant dans ce cas de posséder les moyens pour structurer, exprimer les exigences et, enfin, pour qu'elles puissent être évaluées de manière à obtenir un résultat fiable, précis et indépendamment de toute interprétation humaine.

Dans ce sens, des premiers travaux ont été initiés dans le domaine de la modélisation d'entreprise afin de déployer des approches formelles de vérification afin « d'offrir la possibilité d'une preuve rigoureuse » de la satisfaction d'une exigence [CHA 07]. En effet, la vérification est définie « comme la confirmation par examen et apport de preuves tangibles que les exigences spécifiées ont été satisfaites ». Elle offre différentes techniques depuis celles non formelles (ex. inspection, revue de modèle) jusqu'aux plus formelles (ex. model checking) en passant par les techniques semi-formelles (ex. graphes de causes-effets) [ISO 94]. Bien que les approches de vérification représentent des avantages pour la vérification de modèles (réseau de Pétri, réseau d'automates...), l'application des approches les plus formelles à des modèles d'entreprises reste cependant peu connue, peu développée et marginale du fait d'une faible applicabilité sur des modèles d'entreprises « faiblement formalisables » [VAL 03]. Néanmoins, les premiers travaux de [KAM 04] ont démontré l'applicabilité de ce type de méthodes en s'appuyant sur les graphes conceptuels [SOW 84] appliqués à la vérification sur des processus (ex. processus de circuit du médicament) [ALO 07]. Ainsi, les graphes conceptuels permettent la représentation d'exigences aisément manipulable et compréhensible avec un minimum d'apprentissage tout en profitant de la puissance de la vérification formelle pour obtenir une évaluation rigoureuse et précise (Figure 5).

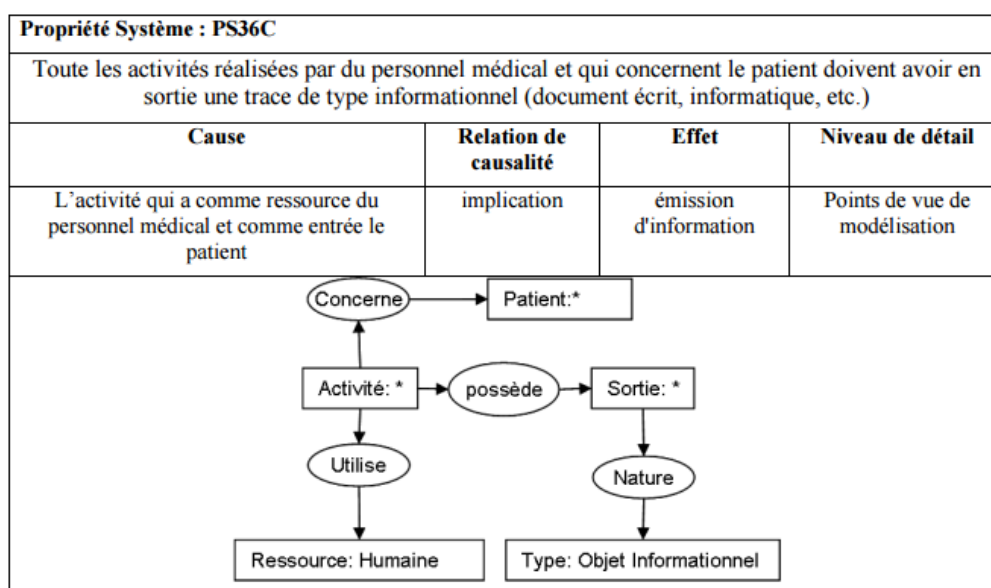


Figure 5. Exemple d'exigence en langage naturel traduite en graphe conceptuel [ALO 07]

¹³ Service Level Agreement – Accord de Niveau de Service

Bien que l'utilisation de ce type d'approche (ex. graphes conceptuels) se limite à un résultat de type qualitatif (valeur de vérité vrai/faux), l'utilisation d'approches formelles pour évaluer des exigences d'interopérabilité peut apporter une réelle valeur ajoutée (fiabilité du résultat, rigueur de l'évaluation) à supposer que l'on puisse allier la rigueur des moyens de vérification formelle (base mathématique) avec des exigences exprimées sur la base des autres méthodes d'évaluation de l'interopérabilité et sur des modèles (ex. processus collaboratifs) peu formels.

La deuxième contribution porte sur l'évaluation de l'exigence d'interopérabilité en s'appuyant sur les méthodes d'évaluation de l'interopérabilité existantes ainsi que les méthodes d'évaluation d'exigences – notamment formelles - sur les modèles de processus collaboratifs.

Troisième dimension : une évaluation à analyser

La dernière composante de l'étude l'interopérabilité comme une exigence est liée à l'analyse du résultat de l'évaluation de l'exigence d'interopérabilité pour connaître son impact sur le système collaboratif. En effet, si les différents moyens déjà développés renseignent sur une performance d'interopérabilité ou un niveau d'interopérabilité, ils offrent rarement les moyens d'analyser le résultat obtenu. En ce sens, ils mettent à disposition une valeur courante qui est reliée à un objectif permettant de connaître l'écart par rapport à ce dernier et précisant ainsi l'effort qui devra être fourni (ex. effort important si l'écart est important) sans nécessairement donner des renseignements concernant l'impact réel sur le système collaboratif. Ils offrent encore des niveaux indiquant, sur la base de critères « *remplis* », les moyens d'évoluer à travers ceux-ci mais toujours sans réellement fournir l'impact du niveau d'interopérabilité considéré sur le système collaboratif. A titre d'exemple, concernant l'évaluation du temps de latence, un résultat très inférieur à celui attendu ne signifie pas nécessairement que l'impact sur le système collaboratif est réellement positif. On peut s'autoriser à répondre par l'affirmative en première intention puisque la performance est meilleure que l'objectif visé. Cependant, sans une vision précise de l'impact sur les systèmes qui constituent le système collaboratif, cette réponse est moins évidente. L'ensemble des critères impacté doit donc être étudié pour s'assurer que cet impact est positif. Il s'agit alors dans ce cas de s'intéresser à l'impact sur les performances des sous-systèmes, au maintien de leur autonomie et enfin aux perturbations engendrées sur le système collaboratif sur la réalisation de ses missions et ses performances. Grâce à la connaissance de l'impact de l'interopérabilité au niveau local (sous-système) et global (système collaboratif), les acteurs du système collaboratif sont alors à même de décider du niveau d'interopérabilité à viser et des solutions à déployer pour atteindre ce niveau tout en garantissant aux systèmes les propriétés de performances, d'autonomie.

Ainsi, il ne s'agit plus simplement de connaître un niveau d'interopérabilité local (ex. entre deux systèmes du réseau collaboratif) mais de repositionner ce résultat dans une perspective d'analyse d'impact de l'interopérabilité sur son environnement afin de permettre aux acteurs de faire des choix pertinents et adaptés.

La dernière contribution concerne l'analyse des résultats de l'évaluation de l'exigence d'interopérabilité dans le but de connaître précisément son impact sur les systèmes collaboratifs et leurs constituants.

2.5. Conclusion

Cette première partie a montré que l'interopérabilité est une attente forte des systèmes collaboratifs - notamment pour les entreprises collaboratives - au regard des contraintes de temps, d'autonomie et d'hétérogénéité des systèmes qui doivent collaborer. D'un point de vue plus conceptuel, la notion de SdS montre également et valide que l'interopérabilité est une exigence clé (« -ilities ») dans le cadre de la conception de systèmes qui doivent réaliser une mission ensemble, toujours avec des contraintes d'indépendance (autonomie) d'hétérogénéité ou de temps de mise en œuvre et de démantèlement.

Les premiers travaux sur l'interopérabilité se sont donc attachés à définir précisément ce concept et ses différents aspects ainsi qu'à développer des outils (conceptuels, techniques ou organisationnels) qui répondent à cette exigence des systèmes collaboratifs. Cependant sans une identification claire des problèmes d'interopérabilité, il devient difficile de sélectionner ou de combiner un ensemble de solutions qui satisfait précisément le problème.

Pour cela il est nécessaire d'évaluer l'interopérabilité de manière quantitative ou qualitative dans le but d'identifier précisément le problème. Cependant, les deux types d'approches présentent chacune des avantages et des inconvénients en termes d'utilisation ou de résultat de l'évaluation pour guider la sélection d'une solution optimale par rapport au problème considéré. Ces questions de définition et d'identification des problèmes et de sélection de solution sont abordées dans des approches telles que l'ingénierie de systèmes et même appliquées au SdS *via* le développement de la recherche sur l'ingénierie des SdS au sein desquels l'interopérabilité est identifiée comme une exigence primordiale. Dans ce sens l'interopérabilité peut être - au-delà des approches d'évaluation de l'interopérabilité, des cadres qui structurent l'interopérabilité, des solutions d'interopérabilité ou encore des approches de développement de l'interopérabilité – considérée comme une exigence à part entière.

Aborder l'interopérabilité comme une exigence au sens de l'ingénierie des exigences présente une nouvelle vision de la question de l'interopérabilité et peut permettre de faciliter la conception de systèmes collaboratifs. Cela nécessite alors de répondre à la problématique suivante :

Comment permettre à des parties prenantes d'un système collaboratif d'exprimer leurs exigences d'interopérabilité, de gérer ces exigences, d'évaluer ces exigences et d'analyser l'impact de cette évaluation de manière rigoureuse pour développer et implanter des solutions qui satisfont ces exigences ?

La réponse à cette problématique concerne donc quatre aspects, introduits précédemment à savoir, l'écriture des exigences d'interopérabilité, la structuration des exigences d'interopérabilité au sein d'un référentiel, l'évaluation des exigences de manière rigoureuse et, enfin, l'analyse du résultat de cette évaluation en termes d'impacts de l'interopérabilité sur le système collaboratif (*Tableau 4*).

	Exigence d'interopérabilité			
	Gestion	Expression	Evaluation	Analyse
Besoins	<ul style="list-style-type: none"> • Accessibilité • Modification • Lien avec solutions 	<ul style="list-style-type: none"> • Simple • Précise • Evaluable 	<ul style="list-style-type: none"> • Rigoureuse • Rapide • Fiable 	<ul style="list-style-type: none"> • Impact sur système collaboratif • Impact sur sous-systèmes
Objectifs	<ul style="list-style-type: none"> • Référentiel d'exigences 	<ul style="list-style-type: none"> • Langage dédié 	<ul style="list-style-type: none"> • Vérification formelle • Expertise 	<ul style="list-style-type: none"> • Matrice d'impacts
Bases	<ul style="list-style-type: none"> • Cadre d'interopérabilité • Ingénierie des exigences 	<ul style="list-style-type: none"> • Ingénierie des exigences • Outils d'évaluation de l'interopérabilité 	<ul style="list-style-type: none"> • Graphes conceptuels • Model checking 	<ul style="list-style-type: none"> • Cycle de vie de l'interopérabilité • Perspectives d'analyse

Tableau 4. Composantes du traitement de l'exigence d'interopérabilité

3. Formalisation et évaluation de l'exigence d'interopérabilité dans les systèmes collaboratifs

3.1. Introduction

Le traitement de la problématique de formalisation et d'évaluation de l'exigence d'interopérabilité s'appuie sur la réalisation de différents travaux réalisés aux travers de projets de recherche et de travaux de thèses (Figure 6).

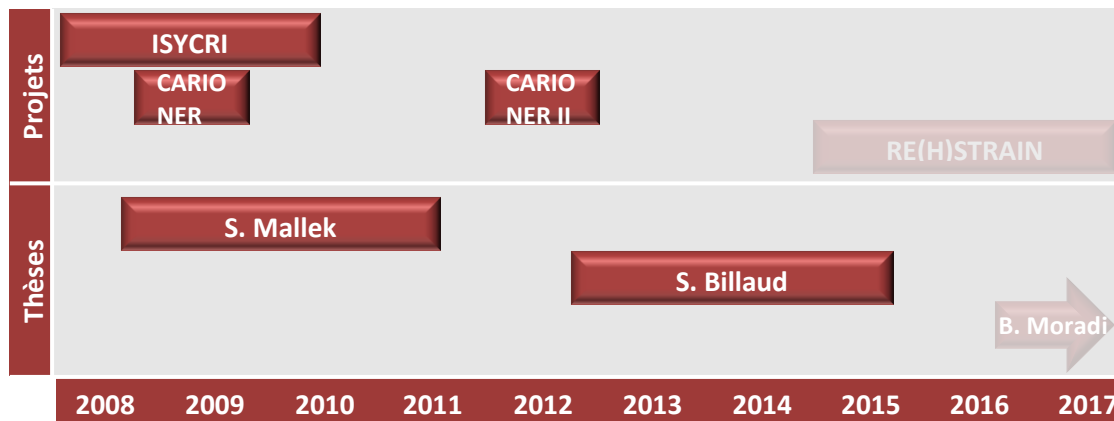


Figure 6. Travaux pour le traitement de la problématique de formalisation et d'évaluation de l'exigence d'interopérabilité

En termes de projets de recherche, le projet **CARIONER** (Caractérisation, analyse et amélioration de l'interopérabilité des organisations impliquées dans des processus collaboratifs) et le projet **ISYCRI** (Interopérabilité des Systèmes en Situation de Crise) se sont intéressés à l'application des graphes conceptuels pour la vérification d'exigences d'interopérabilité dans des processus collaboratifs. Le projet **CARIONER II** s'est intéressé à la spécification des exigences d'interopérabilité en vue de leur vérification dans des processus collaboratifs.

En termes de thèses, les travaux de S. Mallek se sont intéressés, d'une part, à la prise en compte de l'approche formelle du *model checking* pour la vérification d'exigences d'interopérabilité temporelle dans des processus collaboratifs et, d'autre part, à la structuration des exigences d'interopérabilité. Les travaux de S. Billaud se sont concentrés sur la prise en compte de l'exigence d'interopérabilité dans l'ingénierie des systèmes de systèmes en vue de l'analyse de son impact sur ces derniers.

Les sections suivantes montrent les moyens conceptuels, méthodologiques et techniques mis en œuvre pour répondre de manière structurée à la problématique de formalisation et d'évaluation de l'exigence d'interopérabilité dans les systèmes collaboratifs. Cette problématique est traitée selon trois dimensions à savoir (1) l'**évaluation** de l'exigence d'interopérabilité *via* des techniques de vérification formelle, (2) l'**expression** des exigences d'interopérabilité *via* un langage dédié ainsi que leur **structuration** et, enfin (3), l'**analyse** de l'impact de l'interopérabilité suite à l'évaluation des exigences. Les exigences d'interopérabilité exprimées dans la suite pour la présentation des travaux réalisés sont disponibles dans les travaux de [MAL 11]¹⁴.

¹⁴ Les exigences d'interopérabilité sont disponibles au lien suivant: <https://tel.archives-ouvertes.fr/tel-00666099/document>

3.2. Dimension évaluation de l'exigence d'interopérabilité

Problématique d'enrichissements et de transformation de modèles

Les premiers travaux conduits ont abordés l'évaluation de l'interopérabilité dans les processus collaboratifs (CARIONER) appliqués à la gestion de crise (projet ISYCRI). Le premier point s'est intéressé au développement et à l'utilisation des concepts développés pour l'application des graphes conceptuels sur des modèles de processus. L'utilisation des moyens de vérification formelle, notamment les graphes conceptuels, sur des processus collaboratifs de gestion de crise est tout à fait pertinente puisqu'ils offrent une analyse rapide et fiable et particulièrement dans le cas où celle-ci supporte une prise de décision ou la mise en œuvre d'actions correctives (perte d'une ressource partagée, incapacité ou inaptitude dans la réalisation d'une activité, reconfiguration de la réponse...). C'est le cas dans les processus collaboratifs de réponses à une crise qui nécessitent réactivité et efficacité. La vérification formelle représente alors un avantage non négligeable par rapport à une analyse par expertise humaine qui demande du temps (ex. modèles de processus collaboratifs avec de nombreuses activités) avec la possibilité d'une mauvaise interprétation ou d'une omission conduisant à une analyse erronée avec les conséquences qui en découlent.

L'application des graphes conceptuels demande, d'une part, d'exprimer les exigences à l'aide des graphes conceptuels et, d'autre part, d'utiliser les moyens de vérification propres aux graphes conceptuels (appelés projections) pour vérifier la valeur de vérité d'une exigence sur un modèle de processus collaboratif. Il ressort alors deux points essentiels qui représentent la base de l'évaluation de l'exigence d'interopérabilité dans les processus collaboratifs *via* les graphes conceptuels.

La première est liée à la présence de deux langages différents devant cohabiter. D'un côté, le langage de modélisation utilisé dans le cadre de ce projet et plus largement dans le cadre de cette recherche pour décrire les processus collaboratifs c'est-à-dire BPMN¹⁵ [OMG 11]. Le langage BPMN, permet la représentation graphique de processus à l'aide de différents objets tels que les activités, les événements, les branchements ou encore les flux. De l'autre côté, les graphes conceptuels, initialement développés pour représenter la sémantique d'une phrase et s'appuyant sur des objets tels que les concepts et les relations conceptuels [SOW 84]. Les graphes conceptuels permettent de réaliser des opérations sur les graphes appelées également projections et à des fins de vérification. Ainsi, le premier problème est de pouvoir appliquer les mécanismes de vérification offerts par un langage (les graphes conceptuels) sur un autre langage (BPMN) et concerne l'aspect **transformation de modèles**.

La seconde est liée à l'expression d'un ensemble d'exigences d'interopérabilité, en vue de son évaluation sur un modèle de processus collaboratif. Dans ce cas, les concepts de l'interopérabilité doivent être décrits dans une exigence en graphes conceptuels mais surtout être vérifiés sur le modèle du processus. Le second problème est donc de permettre au langage de modélisation d'embarquer les concepts de l'interopérabilité en vue de sa vérification et concerne l'aspect **enrichissement conceptuel**.

La *Figure 7* synthétise les deux aspects mentionnés précédemment. A titre d'exemple, si l'on considère l'étude de l'interopérabilité selon l'EIF au niveau processus pour la catégorie organisationnelle, il est nécessaire de s'intéresser à la définition des responsabilités [ISO 09].

¹⁵ Business Process Model and Notation

Un objectif de la définition des responsabilités est de s'assurer que chaque tâche du processus collaboratif dispose d'un responsable identifié. Ainsi, une exigence d'interopérabilité exprimée en langage naturel et traitant cet aspect de responsabilité peut s'exprimer tel que « *Toutes les tâches du processus ont un responsable défini* ». L'expression de cette exigence est modélisée en graphe conceptuel (partie haute) mais ne peut s'appliquer au modèle de processus exprimé en BPMN. Il est donc nécessaire d'opérer une transformation de modèle pour pouvoir appliquer les mécanismes de vérification des graphes conceptuels sur le modèle du processus collaboratif. Ensuite, le langage BPMN n'intègre pas les concepts et relations liés à la définition des responsabilités. Dans ce cas, la vérification de l'exigence est impossible puisque la projection des relations et concepts comme « *taskHasResponsible* » ou encore « *PersonnResource* » n'existent pas dans le langage BPMN. Il est donc également nécessaire de procéder à l'enrichissement conceptuel du langage BPMN afin qu'il autorise la prise en compte de ces concepts et relations.

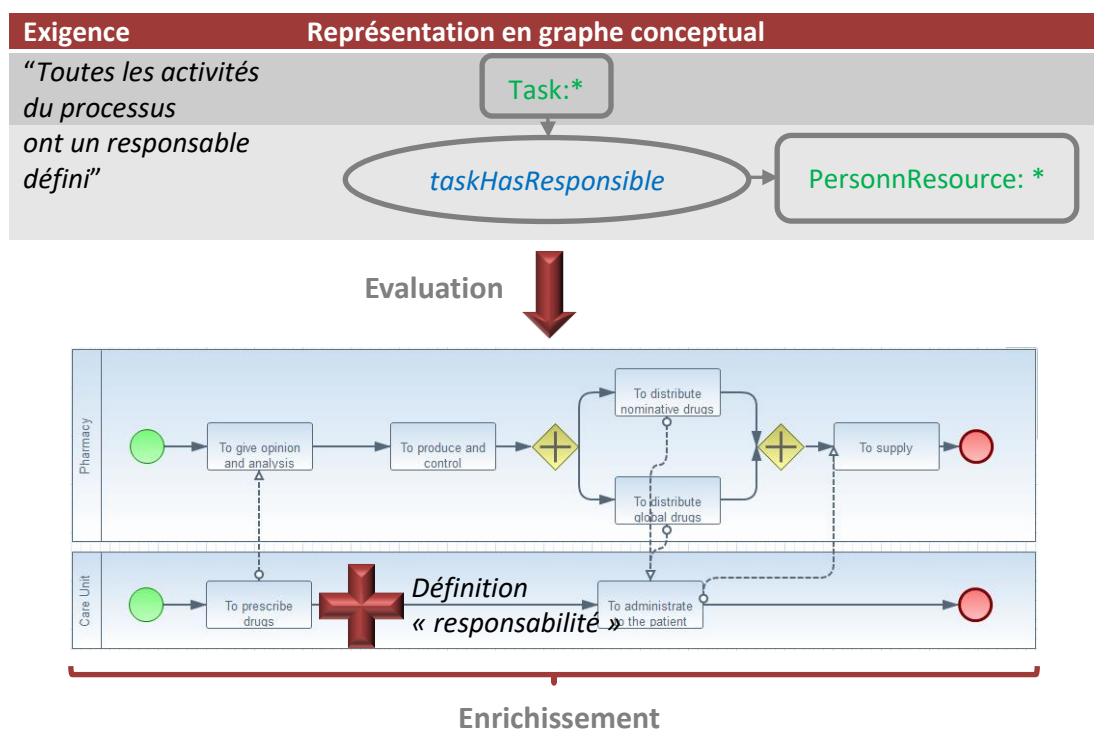


Figure 7. Vers l'enrichissement conceptuel pour l'intégration des concepts de l'interopérabilité au langage de modélisation de processus et la transformation de modèle pour de l'évaluation d'une exigence

Le premier point traité concerne la prise en compte des concepts de l'interopérabilité au sein du langage utilisé pour la modélisation du processus collaboratif [ROQ 09] [MAL 11]. Ainsi, à partir du cadre d'interopérabilité EIF qui considère les différents concepts d'interopérabilité selon les différentes catégories d'interopérabilité et les niveaux d'interopérabilité, il est possible d'extraire ceux qui sont utiles à des fins de vérification. Dans le cadre de cette recherche, les concepts visés concernent plus particulièrement la catégorie organisationnelle au niveau processus.

Dans ce sens, l'aspect organisationnel pour un processus concerne la notion de durées, de missions, de responsabilités, d'autorisations et de ressources (humaines, matériels, logiciels) à partager entre les différentes tâches d'un processus collaboratif. Le langage BPMN ne considérant pas la notion de ressource, par exemple, une extension de ce concept est donc réalisée.

De la même manière, certains concepts doivent être caractérisés à l'aide d'attributs qui sont également liés à la notion d'interopérabilité. A titre d'exemple le concept de tâche BPMN peut être étendu avec les attributs tels que « *authorization* » ou encore « *authorizationValidity* » dans le cas où une tâche demande une autorisation à une ressource partagée pour pouvoir accéder à ces applications, données, matériels... (Figure 8).

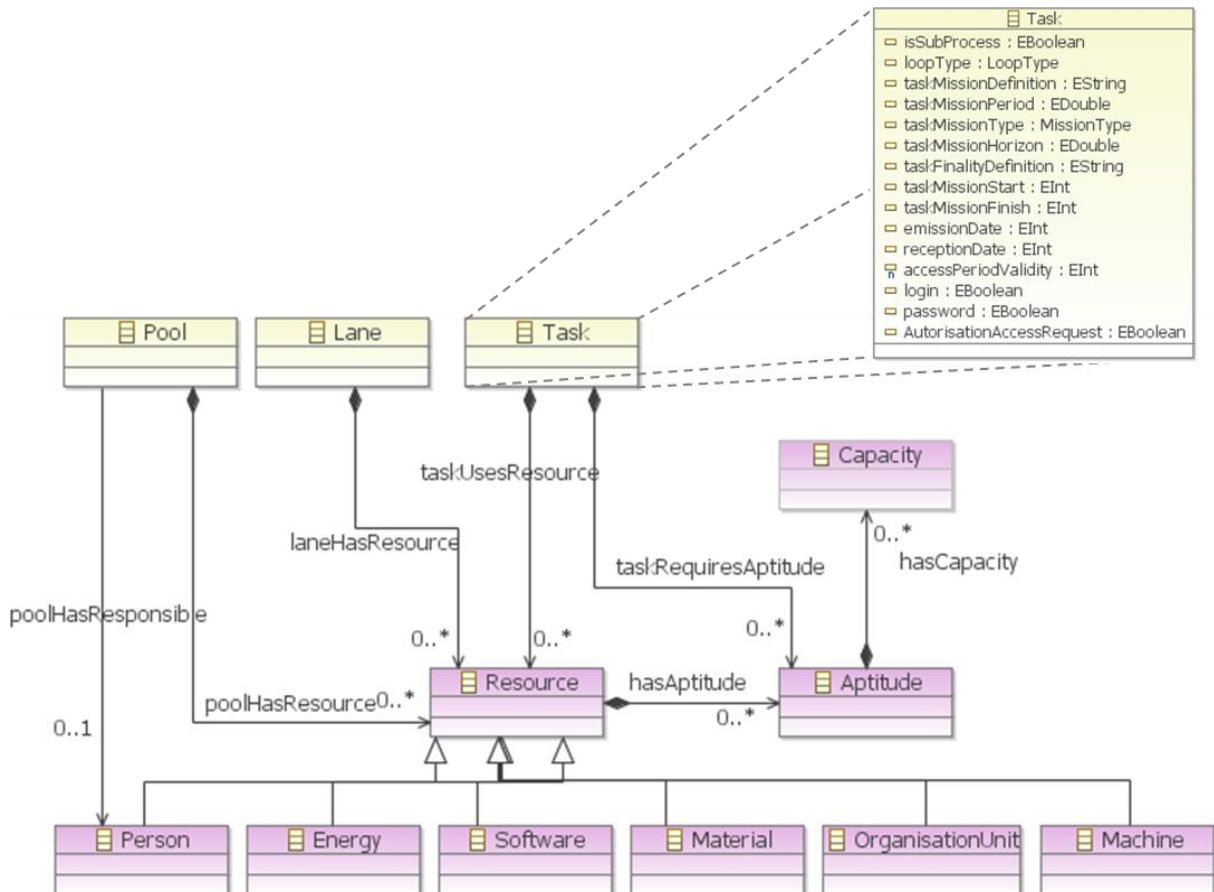


Figure 8. Extensions conceptuelles du langage BPMN (ajout des classes Ressource, Aptitude, Capacité... et des attributs sur l'objet Tâche)

Le deuxième point concerne l'utilisation des mécanismes de vérification des graphes conceptuels sur des modèles de processus collaboratifs en langage BPMN. Il est dans ce cas nécessaire de procéder à une transformation de modèles du processus vers les graphes conceptuels. Cette transformation s'appuie sur les principes de transformation de modèles tels qu'ils sont appréhendés dans l'ingénierie dirigée par les modèles (IDM – [KEN 02] [MET 05]), à savoir dans ce cas les transformations exogènes (méta modèle source ≠ méta modèle cible).

Ainsi, un processus collaboratif décrit à l'aide de concepts et de relations du langage BPMN (modèle source) est transformé en concepts et relations pour décrire un graphe conceptuel (modèle cible) via des règles de transformations. Dans le cas de cette étude, l'outil permettant l'application des graphes conceptuels, COGITANT, permet d'appliquer les mécanismes de projection pour la vérification.

C'est donc vers le méta modèle déployé dans COGITANT (cible) [GEN 14] que le méta modèle BPMN étendu (source) est transformé (Figure 9). Les transformations sont réalisées à l'aide du langage de transformation ATL¹⁶.

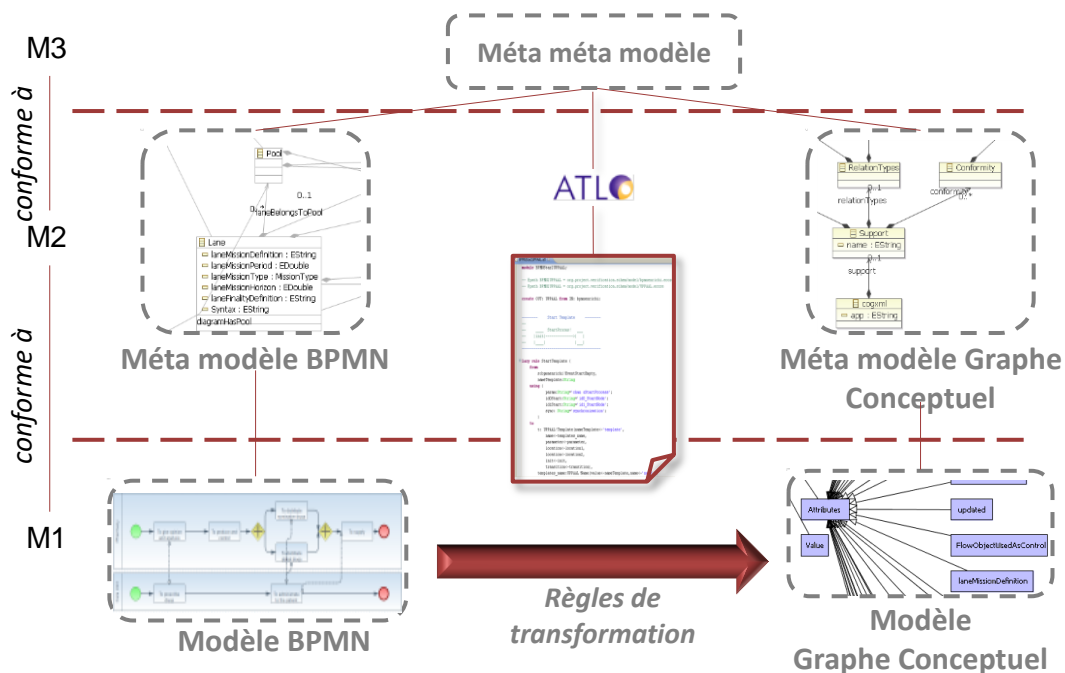


Figure 9. Principe transformation d'un modèle de processus BPMN vers un graphe conceptuel

A partir du modèle de processus transformé en un graphe conceptuel, il est possible d'appliquer les mécanismes de projection pour un graphe conceptuel représentant une exigence. Les principes de projections sont les suivants :

- Positive. Le graphe conceptuel qui représente l'exigence est composé d'une cause et d'une conclusion (cf. Figure 9). Toute projection de la cause sur le graphe conceptuel représentant le processus doit permettre également de projeter la conclusion sur le même graphe conceptuel. Dans ce cas, l'exigence est considérée comme satisfaite.
- Négative. Le graphe conceptuel qui représente l'exigence ne possède ni cause ni conclusion. Dans ce cas, toute projection sur le graphe conceptuel représentant le processus signifie que l'exigence n'est pas satisfaite.

Ainsi, via des mécanismes de transformations, permettant la mise en correspondance (« mapping ») d'un modèle de processus collaboratif dans un langage source (BPMN) avec un modèle de représentation de la connaissance dans un langage cible (graphe conceptuel), via des enrichissements permettant de manipuler les concepts d'interopérabilité dans les modèles de processus collaboratifs, il est possible d'appliquer les moyens de raisonnements offerts par les graphes conceptuels à des fins d'évaluation de l'interopérabilité (Figure 10).

¹⁶ Atlas Transformation Language

Dès lors, les concepts développés sur l'utilisation des techniques de vérification formelle sur des modèles peu formalisés s'avèrent pertinents. Ils offrent la possibilité d'évaluer un ensemble d'exigences dans un temps limité, sur des modèles de processus collaboratifs complexes (nombreuses activités, nombreuses informations sur les objets du processus) avec l'obtention d'une valeur de vérité qui ne peut être remise en cause.

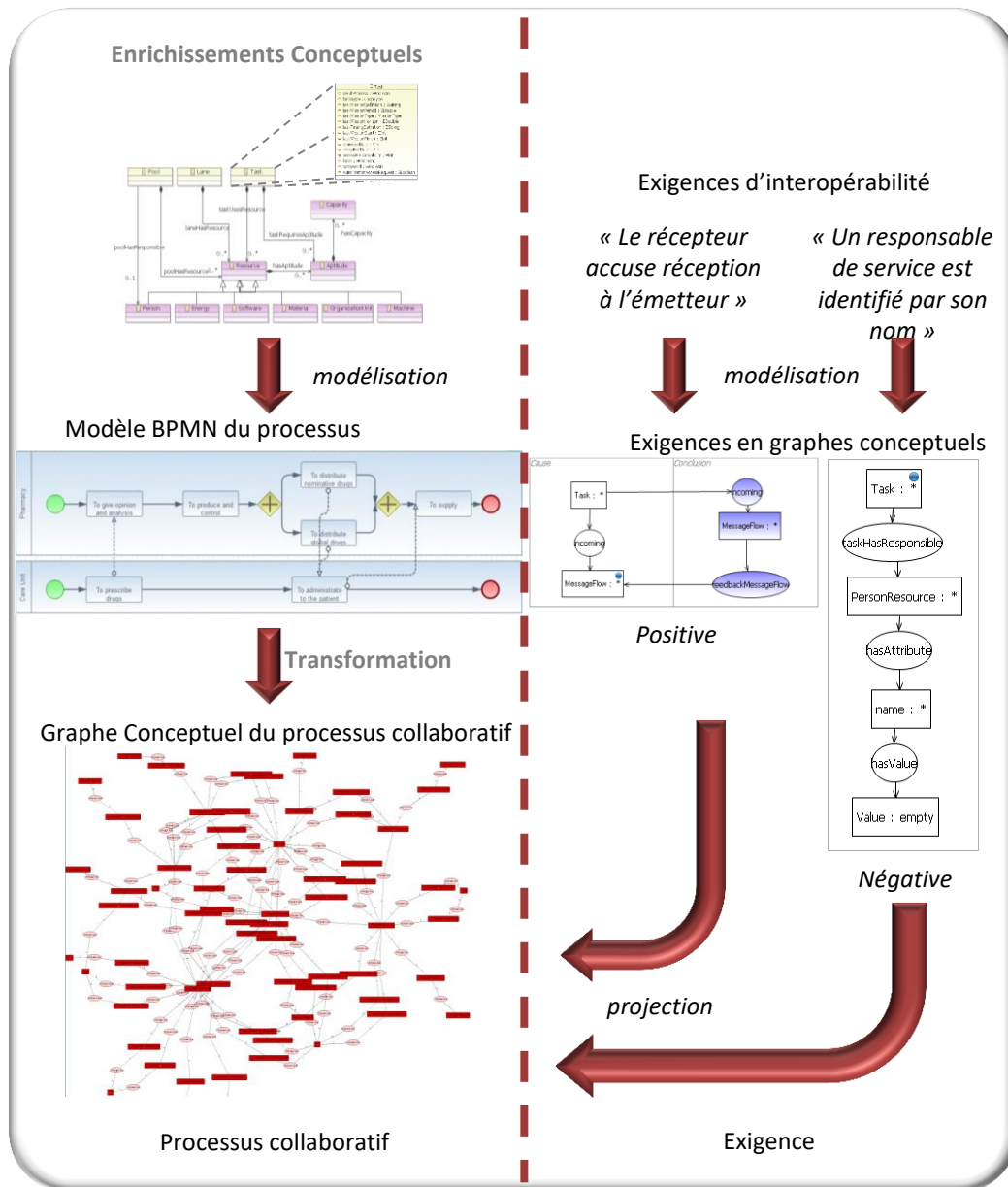


Figure 10. Vue générale du déploiement des graphes conceptuels pour l'évaluation de l'exigence d'interopérabilité sur des modèles de processus collaboratifs

Si l'utilisation des graphes conceptuels s'est montrée efficace, elle est limitée à l'analyse d'exigences d'interopérabilité lorsqu'elle est indépendante du temps (*i.e.*, l'interopérabilité ne dépend pas du temps comme par exemple pour l'évaluation de la compatibilité de deux outils).

Cependant, un des aspects de l'interopérabilité est lié à la notion de temps comme le montre, par exemple, l'évaluation de la latence de données. Il est donc indispensable de permettre également l'évaluation d'exigences d'interopérabilité temporelle et ce, toujours en s'appuyant sur des techniques de vérification formelle afin de bénéficier des avantages qu'elles procurent.

La table suivante synthétise les travaux effectués dans le cadre de l'évaluation de l'exigence d'interopérabilité *via* les graphes conceptuels.

Communications	[CI11], [CI12], [CN1], [GTN2]
Projet	ISYCRI, CARIONER

Tableau 5. Synthèse des travaux de recherche sur l'évaluation de l'exigence d'interopérabilité par les graphes conceptuels

Problématique de structuration des exigences et sémantique opérationnelle

La problématique d'évaluation des exigences d'interopérabilité temporelle a été traitée dans la thèse de S. Mallek, dans le cadre du développement d'une approche anticipative de défauts d'interopérabilité dans des processus collaboratifs. Le principe est de mettre à disposition les moyens d'évaluer l'interopérabilité avant qu'un processus ne soit réalisé afin de mettre en œuvre les solutions d'interopérabilité adaptées aux problèmes identifiés. En considérant les exigences d'interopérabilité temporelle en plus d'atemporelle issues des travaux précédents, cette recherche a fait face au problème de **structuration des exigences** en établissant le lien entre la temporalité des exigences d'interopérabilité avec les concepts issus de l'EIF. Dans ces travaux, l'évaluation est effectuée à l'aide de la technique de vérification formelle du « *model checking* » pour la vérification d'exigences d'interopérabilité temporelle (ex. durée des interactions) [MAL 12]. L'objectif est de vérifier un ensemble d'exigences sur des modèles comportementaux (dans ce cas des automates à états temporisés) qui décrivent le système (décrit initialement par un modèle de processus collaboratif). A partir d'un mécanisme d'exploration exhaustif de tous les états atteignables du système, les exigences sont satisfaites (ou non) dans certains de ces états. Dans cette étude, l'outil (UPPAAL [BEH 04]) est utilisé. Il permet de traiter un **modèle de comportement** décrit sous la forme d'un réseau d'automates temporisés et l'écriture d'exigences dans la logique temporelle TCTL¹⁷ [SCH 02].

Dans le cadre de ces travaux, il a donc été nécessaire de proposer une structuration des exigences d'interopérabilité en fonction des objectifs d'évaluation (ex. exigences d'interopérabilité temporelle/atemporelle) et également de permettre l'application de la technique du « *model checking* » pour l'évaluation des modèles de processus collaboratifs (*Figure 11*).

¹⁷ Timed Computation Tree Logic.

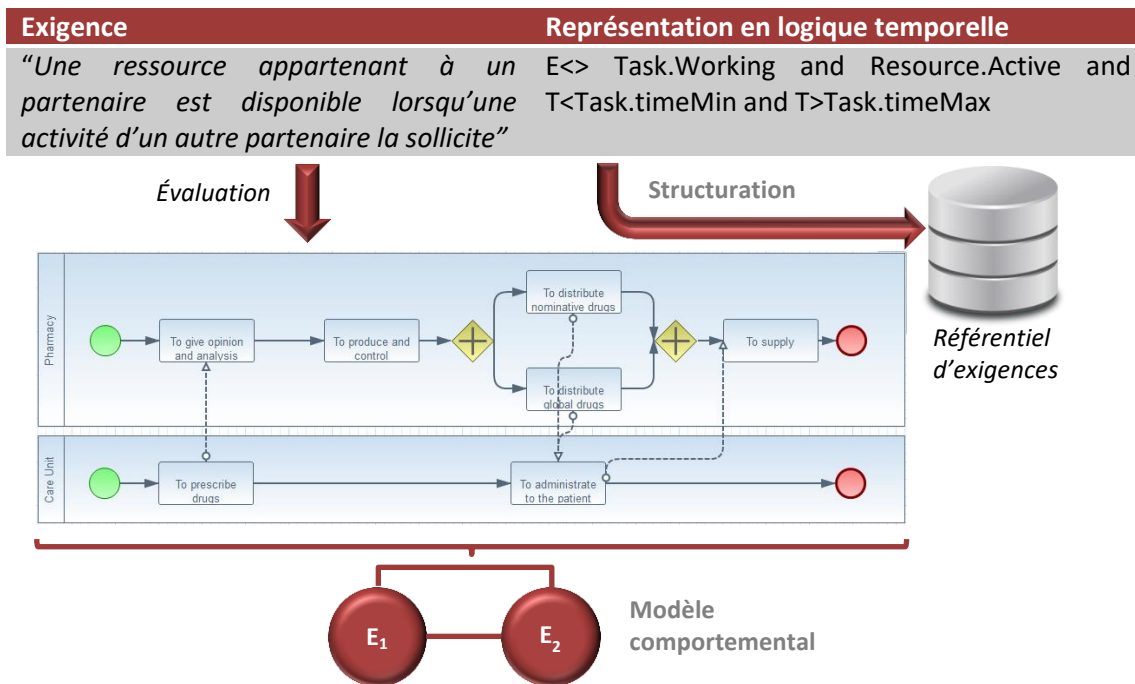


Figure 11. Vers un modèle comportemental du modèle de processus collaboratif pour l'utilisation de la technique du « model checking » et la structuration des exigences au sein d'un référentiel

Le principe de transformation de modèles vu précédemment s'applique également dans le cas de cette étude et reste le même que pour les graphes conceptuels. Cependant, il est nécessaire dans un premier temps - pour, à terme, opérer les transformations requises - de fournir un modèle comportemental du modèle du processus autorisant la mise en œuvre des principes du « model checking ». On parle alors de sémantique opérationnelle et celle-ci doit être définie pour les objets du modèle du processus (ex. tâche, flux, ressource, branchement...). Dans le cas présent, les éléments considérés sont la tâche, la ressource, les branchements (exclusif, inclusif et parallèle), l'événement de début et de fin [MAL 15]. Les modèles à états proposés dans la suite s'appuient sur les travaux de [GRU 05] qui proposent un ensemble de modèles à états pour les objets BPMN (tâche, événements début/fin, branchements parallèle/exclusif). Pour les autres objets (ex. ressource), des modèles à états sont développés et, dans certains cas, il est nécessaire d'étendre les modèles existants. Enfin, la définition de ces modèles s'appuie également, quand cela est possible, sur la description comportementale des objets BPMN fournie par l'OMG [OMG 11].

Le modèle à état de la tâche s'appuie sur les modèles à états proposés dans la littérature tels que le modèle de Rodde [ROD 89], de [TRI 97] ou encore de BPMN [OMG 11] qui proposent de modéliser le comportement d'une tâche selon différents états (Figure 12).

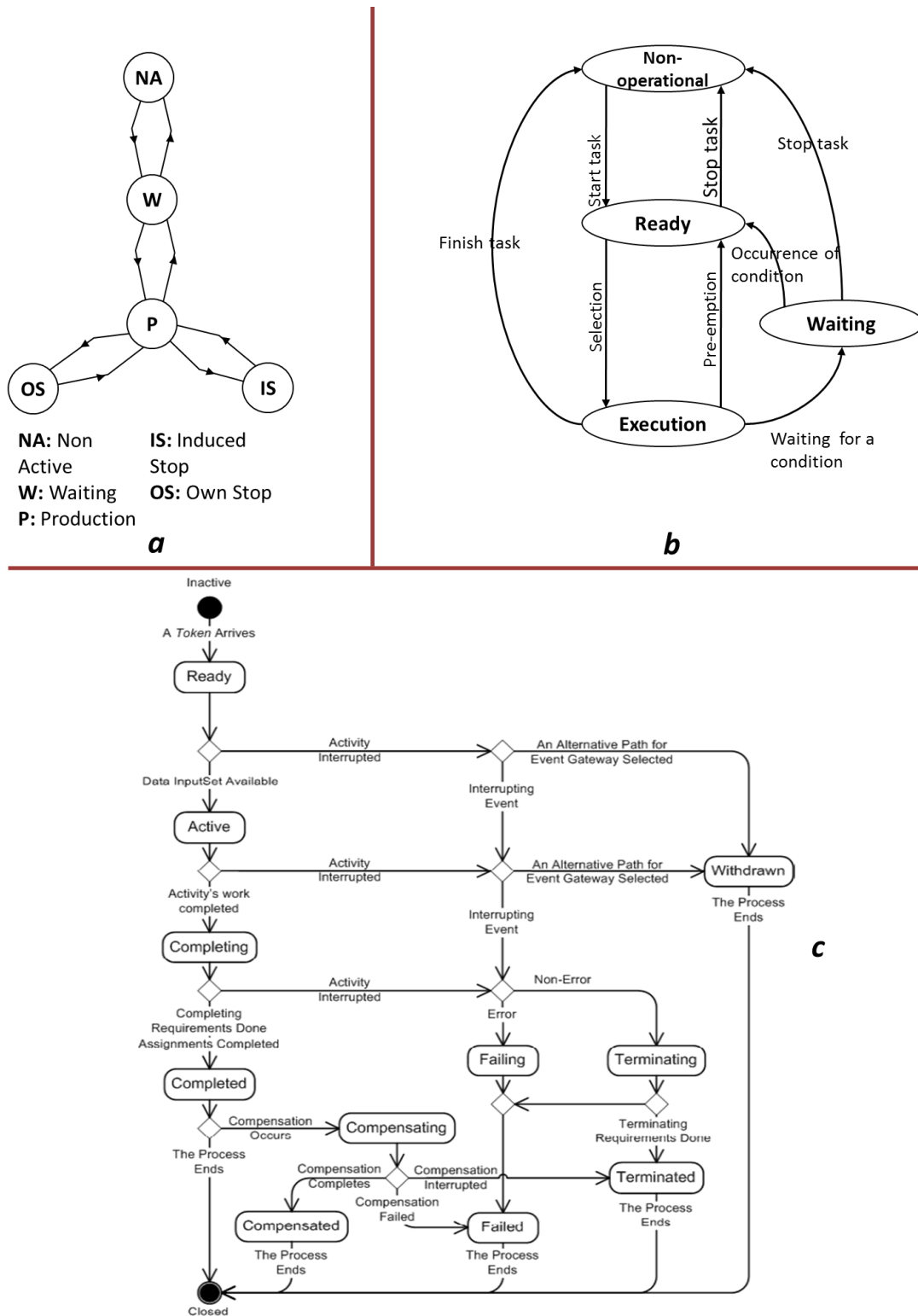


Figure 12. Modèles à états pour la représentation d'une tâche (a – Rodde, b – Trinquet, c – OMG)

Il existe ainsi différents modèles pour caractériser le comportement d'une tâche. Malgré différents états, ces modèles possèdent tous des états fondamentaux qui permettent d'obtenir un modèle « atomique » de comportement d'une tâche. Ce sont ces états qui sont considérés pour obtenir le modèle à état de base pour une tâche à savoir : « attente », « lancement », « production » et « arrêt » (Tableau 6).

Modèle BPMN Modèle à états



Tableau 6. Modèle à états d’une tâche BPMN¹⁸

Le modèle à états proposé pour la ressource (Tableau 7) comporte deux états : « disponible » et « active ». Dans l’état « disponible », la ressource est en attente d’une sollicitation par une tâche. Dans l’état « active », la ressource est utilisée par une tâche. Lorsque la ressource n’est plus utilisée, elle retourne à l’état disponible.

Modèle BPMN Modèle à états

**Pas d’objet graphique,
création d’une instance
de l’objet et affectation
directe sur la tâche**

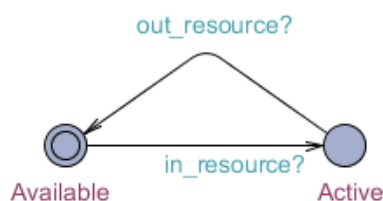


Tableau 7. Modèle à états d’une ressource

Une ressource pouvant être allouée à une activité sur le modèle du processus collaboratif, il est nécessaire de développer un modèle à état d’une tâche qui utilise une ressource. Ce modèle est une extension du modèle initial d’une tâche (Tableau 8).

Modèle BPMN Modèle à états



Tableau 8. Modèle à états d’une activité BPMN utilisant une ressource

Enfin, les modèles à états pour les objets « événement de début », « événement de fin », « branchement parallèle » et « branchement exclusif » existants déjà dans la littérature ont été implantés directement (Tableau 9).

¹⁸ *in_channel ?* et *in_channel !* représentent des synchronisations récepteur et émetteur qui permettent aux différents modèles à états de communiquer. [BEH 04]

Modèle BPMN Modèle à états

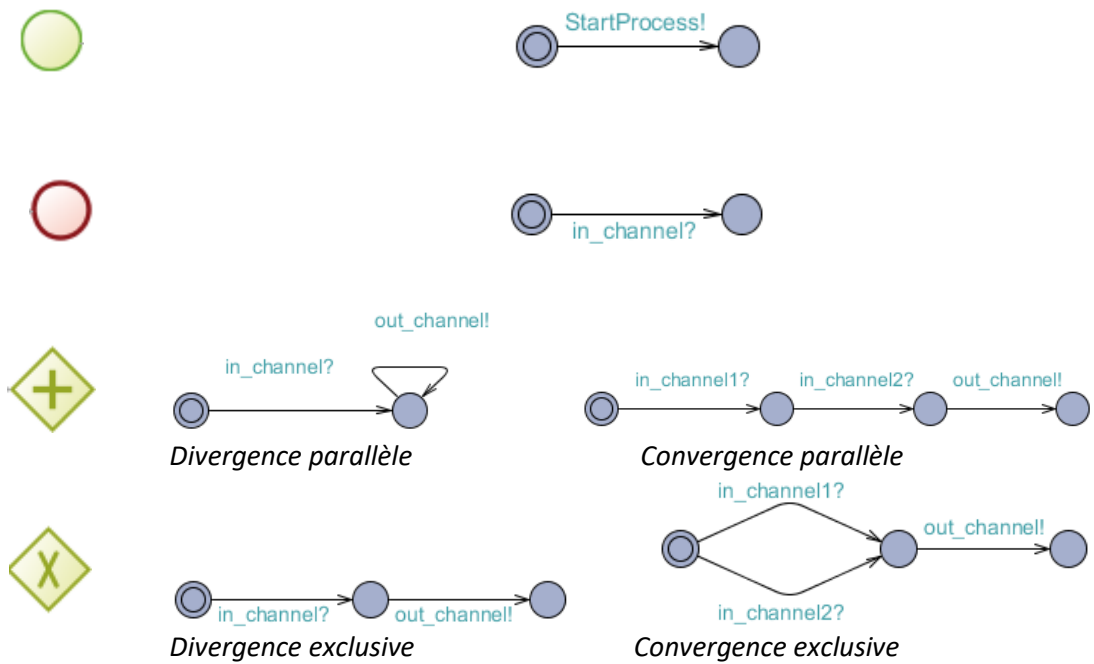


Tableau 9. Modèles à états existants pour les branchements et les évènements

Le dernier modèle à état spécifiquement défini est le « *branchement inclusif* » en divergence et convergence. Ce modèle est développé sur la base du comportement de l'objet au sein d'un modèle de processus et grâce à la combinaison des objets « *branchement parallèle* » et « *branchement exclusif* » et de leur comportement. Ainsi, la construction de ce modèle s'appuie dans un premier temps sur la réalisation d'un modèle de comportement équivalent à la divergence et la convergence sur deux flux d'un processus (Figure 13).

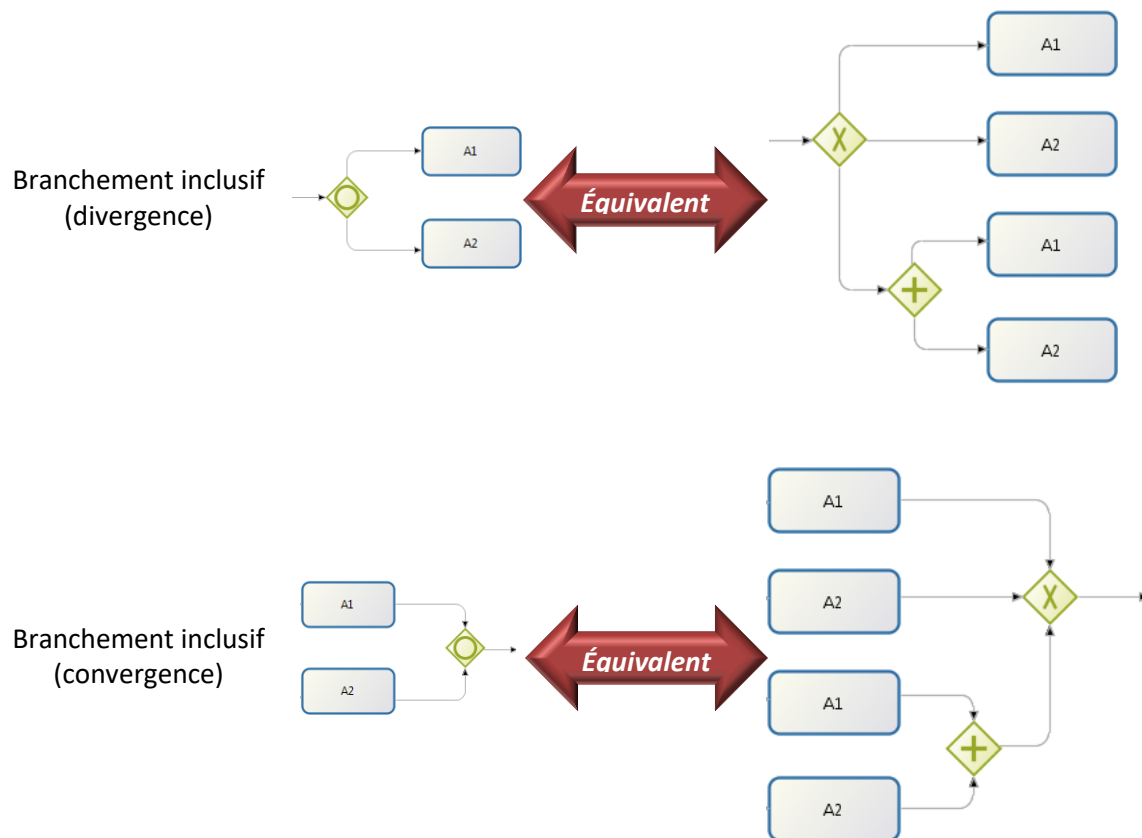


Figure 13. Construction de l'objet branchement inclusif avec les branchements parallèles et exclusifs

Le modèle à état du « *branchement inclusif* » est donc obtenu par combinaison des modèles à états des « *branchements parallèles* » et « *exclusifs* » appliquée à deux chemins (deux tâches au sein du branchement). Ce comportement est valide pour deux branches mais nécessite cependant d'être généralisé à N branches. Dans ce cas, les opérateurs suivants sont utilisés pour connaître le nombre de « *branchements parallèles* » et « *exclusifs* » à combiner (N , nombre de branches de sorties du « *branchement inclusif* ») :

- $2^N - 1$, le nombre de branches de sorties à considérer pour le « *branchement exclusif* ».
- C_N^p , le nombre de « *branchements parallèles* » avec p le nombre de branches de sorties du « *branchement parallèle* », $p \in [2, N]$.

En conséquence, c'est sur la base des modèles à états des objets de modélisation BPMN qui composent le modèle du processus collaboratif - et de leurs synchronisations pour communiquer - qu'il est possible de construire un Réseau d'Automates Temporisés représentant le comportement du processus sur lequel la vérification d'exigences d'interopérabilité peut être effectuée.

Le deuxième aspect de ces travaux de recherche est la structuration des exigences d'interopérabilité au sein d'un référentiel. Si le référentiel développé n'a pas pour objectif principal la gestion des exigences mais la localisation précise de problèmes d'interopérabilité, il permet néanmoins d'obtenir une classification de l'exigence d'interopérabilité selon les différents concepts de l'interopérabilité. Les deux premiers concernent les concepts considérés par l'EIF à savoir les catégories d'interopérabilité (conceptuelle, technologique et organisationnelle) et les niveaux d'interopérabilité (métiers, processus, services et données).

Ces deux premiers aspects sont cohérents pour établir une classification des exigences d'interopérabilité. En effet, le premier concerne les problèmes d'interopérabilité et, comme cela est défini précédemment, une exigence d'interopérabilité permet d'exprimer une attente qui doit être satisfaite par une solution adaptée. Le second concerne les niveaux des systèmes collaboratifs auxquels l'interopérabilité doit être établie. L'aspect approches d'interopérabilité (intégration, unification et fédération) n'est en revanche pas considéré dans cette classification. En effet, cette dernière dimension est liée au choix des solutions et ne permet pas d'apporter de la précision dans l'expression d'une exigence, contrairement aux deux autres qui sont décomposées (la valeur ajoutée dans l'expression de l'exigence se limite au souhait de développer une approche particulière).

Toutefois, un aspect important de l'interopérabilité doit être pris en compte et concerne son « cycle de vie ». L'interopérabilité est liée au cycle de vie d'un système collaboratif et les besoins d'interopérabilité ne sont pas les mêmes au début (création), en cours (opération entre les systèmes) ou à la fin (dissolution) d'une collaboration. Lorsque la collaboration est initiée les exigences sont liées à l'interfaçage des systèmes en présence pour assurer une collaboration effective. Pendant la collaboration, les exigences concernent les interactions (échange, partage entre les systèmes). Enfin, les exigences liées à la fin de vie du système collaboratif permettent de s'assurer que sa dissolution n'affecte pas ses sous-systèmes (ex. perte de performance). Ce cycle de vie de l'interopérabilité est lié à des caractéristiques identifiées et attendues des systèmes interopérables à savoir, la **compatibilité** (création du système collaboratif), l'**autonomie** (indépendance des sous-systèmes) et l'**interopération** (opération entre les sous-systèmes) et, enfin, la **réversibilité** (dissolution du système collaboratif). Cet aspect de l'interopérabilité permet une classification plus précise que les « approches d'interopérabilité » notamment dans la recherche d'un problème d'interopérabilité et, par extension, dans la recherche d'une solution. C'est sur la base des catégories, des niveaux et du cycle d'interopérabilité (classes) qu'est proposé le modèle GRADEI (GRAPhe de Décomposition d'Exigences d'Interopérabilité) pour regrouper et manipuler les exigences d'interopérabilité (Figure 14).

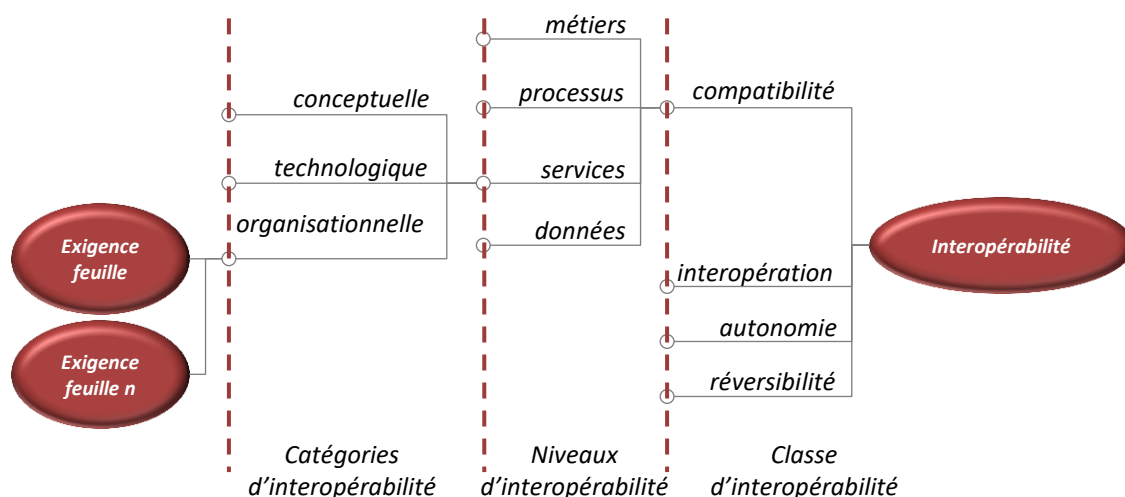


Figure 14. Le modèle GRADEI

Le modèle GRADEI est un graphe composé de nœuds qui représentent un niveau de détail d'une exigence et d'arcs qui lient les différents nœuds et représentent la relation de décomposition. Il permet de décomposer les exigences d'interopérabilité depuis les plus abstraites (nœud racine) jusqu'aux exigences concrètes (nœud terminal – exigence feuille) qui sont vérifiées en passant par les classes, niveaux et catégories d'interopérabilité. Une relation qui lie un nœud cible (exigence abstraite d'un niveau) à ses nœuds sources (exigences plus précises d'un niveau) est caractérisée par une fonction logique qui permet de connaître l'influence des nœuds sources sur le nœud cible. L'objectif est de trouver la valeur de vérité du nœud racine en définissant les fonctions logiques des relations qui lient les différents nœuds afin d'obtenir la valeur de véracité de chacun puis en propageant ces valeurs depuis les exigences concrètes vers l'exigence la plus abstraite.

A titre d'exemple il est possible d'évaluer la valeur de vérité de l'exigence abstraite d'interopérabilité selon les relations suivantes :

$$\textit{Interopérabilité} = \textit{compatibilité} \wedge \textit{interopération} \wedge \textit{autonomie} \wedge \textit{réversibilité} \quad [5]$$

$$\textit{compatibilité} = \textit{métiers} \wedge \textit{processus} \wedge \textit{services} \wedge \textit{données} \quad [6]$$

$$\textit{services} = \textit{conceptuelle} \wedge \textit{technologique} \wedge \textit{organisationnelle} \quad [7]$$

$$\textit{organisationnelle} = \textit{exigence feuille} \vee \textit{exigence feuille } n \quad [8]$$

Les exigences feuilles sont vérifiées à l'aide des techniques de vérification formelle (graphes conceptuels, *model checking*) ou encore par expertise quand les techniques précédentes ne peuvent être déployées. Dans le cas où l'interopérabilité est non considérée au sein du système collaboratif, la valeur de vérité sur le chemin du graphe est directement définie (vrai/faux). Ainsi dans l'exemple précédent pour satisfaire l'exigence abstraite d'interopérabilité, les exigences de compatibilité, interopération, autonomie et réversibilité doivent être satisfaites. Le nœud de l'exigence de compatibilité est satisfait si les exigences métiers, processus, services et données sont satisfaites. De la même manière, l'exigence services est satisfaite si les exigences de type conceptuel, technologique et organisationnel sont satisfaites. Enfin, la valeur de vérité de l'exigence organisationnelle est obtenue par vérification de ses exigences feuilles. Cette exigence est satisfaite si une des deux exigences feuilles est satisfaite.

Ainsi, le modèle GRADEI met à disposition un ensemble d'exigences d'interopérabilité (évalué *via* la technique du *model checking* et les graphes conceptuels) structurées de manière précise et cohérente avec les concepts de l'interopérabilité. De plus, la force de ce modèle est de propager les résultats de l'évaluation locale des exigences d'interopérabilité pour obtenir une évaluation globale de l'interopérabilité selon les besoins des systèmes collaboratifs.

La poursuite des travaux sur l'utilisation des techniques de vérification formelle appliquées à des modèles de processus a permis la mise en œuvre du *model checking* pour l'évaluation d'exigences d'interopérabilité temporelle. L'utilisation de ces outils a nécessité le développement de modèles comportementaux des objets du langage de modélisation de processus BPMN et la vérification d'exigences décrites dans la logique temporelle TCTL. De plus, ces travaux ont également traité le problème de structuration des exigences d'interopérabilité en proposant un modèle développé selon les concepts fondamentaux de l'interopérabilité.

L'ensemble des exigences d'interopérabilité élicité est regroupé au sein du modèle GRADEI et mis à la disposition des utilisateurs en vue de l'évaluation de l'interopérabilité. Au terme de cette recherche ce sont 89 (cf. note de bas de page 14) exigences d'interopérabilité décrites en langage naturel et qui peuvent être évaluées, soit par *model checking*, soit par graphes conceptuels ou encore par expertise. Lorsque les exigences sont évaluées par *model checking* et par graphes conceptuels, elles sont écrites directement en logique temporelle et en graphe conceptuel. Lorsqu'une exigence n'est pas évaluable par une de ces deux techniques, elle est destinée à l'expertise (Figure 15).

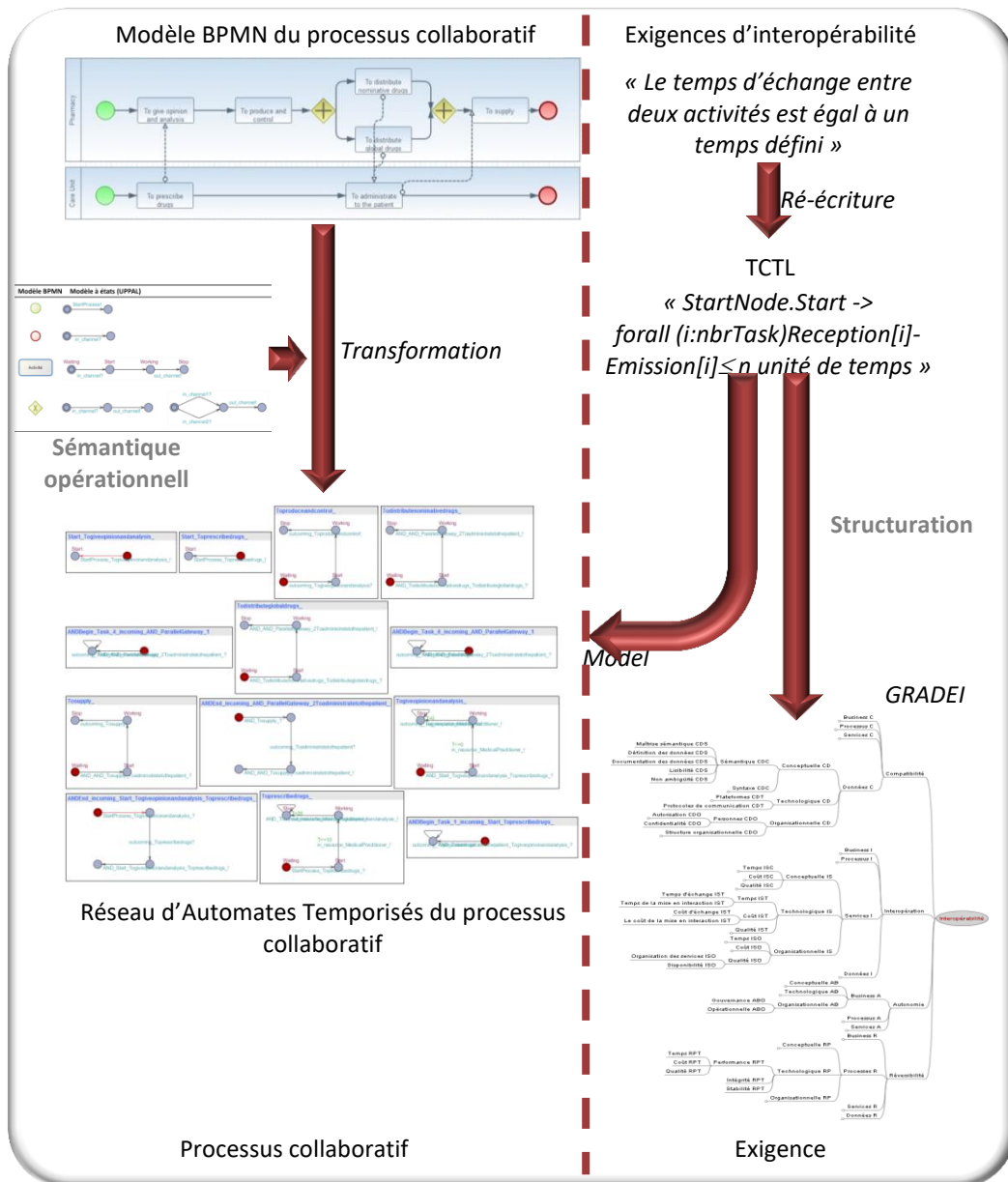


Figure 15. Vue générale du déploiement du model checking pour l'évaluation de l'exigence d'interopérabilité sur des modèles de processus collaboratifs et structuration au sein du modèle GRADEI

Une fois encore, l'évaluation de l'interopérabilité par des techniques de vérification formelle présente un avantage en comparaison de techniques comme l'expertise. D'une part, le temps de vérification est réduit et, d'autre part, le résultat de l'évaluation est irréfutable.

En effet, il est particulièrement difficile (voir impossible) pour un expert de parcourir l'ensemble des chemins d'un modèle de processus en un temps raisonnable (quelques secondes pour le *model checking*) et, dans le cas où il pourrait garantir un parcours exhaustif, il devrait encore garantir le résultat de son évaluation. En conséquence, proposer un ensemble d'exigences identifiées précisément et préalablement au sein d'un référentiel et dont une partie de l'évaluation est assurée par des moyens rigoureux et éprouvés représente un support essentiel dans les systèmes collaboratifs souhaitant développer l'interopérabilité ou encore identifier des problèmes d'interopérabilité. Un support en termes de temps puisque l'évaluation est rapide mais également en termes de maîtrise des coûts puisque une évaluation efficace et précise évite d'implanter des solutions d'interopérabilité inadaptées.

L'utilisation des techniques de vérification formelles sur un langage de modélisation non formel tel que BPMN permet de montrer l'efficacité et l'utilité de ce type d'approches encore peu développées. Cette utilisation nécessite la mise en œuvre de transformations lourdes ainsi que la réalisation d'enrichissements conceptuels et la définition de modèles comportementaux. Cependant, ces aspects sont traités de manière automatique et ne nécessitent pas d'intervention de la part d'un acteur qui souhaite opérer une vérification. Enfin, un point particulier lié à l'exigence d'interopérabilité n'est pas traité et implique cette fois-ci pleinement les acteurs. Ce point concerne d'une part l'expression des exigences d'interopérabilité et, d'autre part, la gestion de ces dernières. Concernant l'aspect « expression », les exemples d'exigences décrites en graphes conceptuels ou encore en logique temporelle montrent la difficulté de maîtrise, de compréhension et de lecture de ces langages pour des utilisateurs non avertis. Dans ce cas, l'utilisateur ne maîtrise pas l'expression de ses propres exigences puisque, dans ces travaux, celles-ci sont préalablement exprimées en langage naturel et liées au langage formel pour pouvoir être vérifiées. Il doit alors choisir l'expression qui correspond à son exigence d'interopérabilité et réaliser sa vérification. Néanmoins, les langages (graphes conceptuels et logique temporelle) sont laissés à l'utilisation libre sans processus qui guide l'écriture, *i.e.*, un utilisateur peut utiliser la logique temporelle pour décrire son exigence s'il possède une bonne connaissance de ce langage. Concernant la gestion des exigences, là encore le modèle GRADEI offre peu de maîtrise et se concentre plutôt sur l'évaluation globale de l'interopérabilité. Aussi, il ne met pas clairement les exigences en relation avec des solutions d'interopérabilité dans le cas où une exigence n'est pas satisfaite.

Le *Tableau 10* synthétise les travaux effectués dans le cadre de la structuration des exigences d'interopérabilité et leur évaluation *via* le *model checking*.

Communications	[RI2], [RI3], [RI5], [RN1], [CI13], [CI14], [CI15], [CN2], [CN3]
Thèse	S. Mallek
Démonstrations	https://goo.gl/s0dY9r <ul style="list-style-type: none">• Conference.avi• HealthCareProcess.avi

Tableau 10. Synthèse des travaux de recherche sur l'évaluation de l'exigence d'interopérabilité par le *model checking*

3.3. Dimension expression et gestion des exigences d'interopérabilité

Les travaux sur l'expression des exigences d'interopérabilité ont été initiés dans le projet CARIONER II [MAL 13] avec pour objectifs (1) de permettre à des parties prenantes d'exprimer eux-mêmes leurs exigences d'interopérabilité afin de les évaluer et (2), de permettre leur gestion (Figure 16). Ces travaux se sont concentrés sur l'expression des exigences d'interopérabilité temporelle et ont abouti à la proposition d'un langage dédié¹⁹ à l'expression des exigences²⁰ [DAC 16]. Ces travaux se sont également intéressés à la définition d'un référentiel d'exigences d'interopérabilité basé sur GRADEI pour, à terme, le mettre en relation avec les solutions d'interopérabilité existantes dans l'EIF.

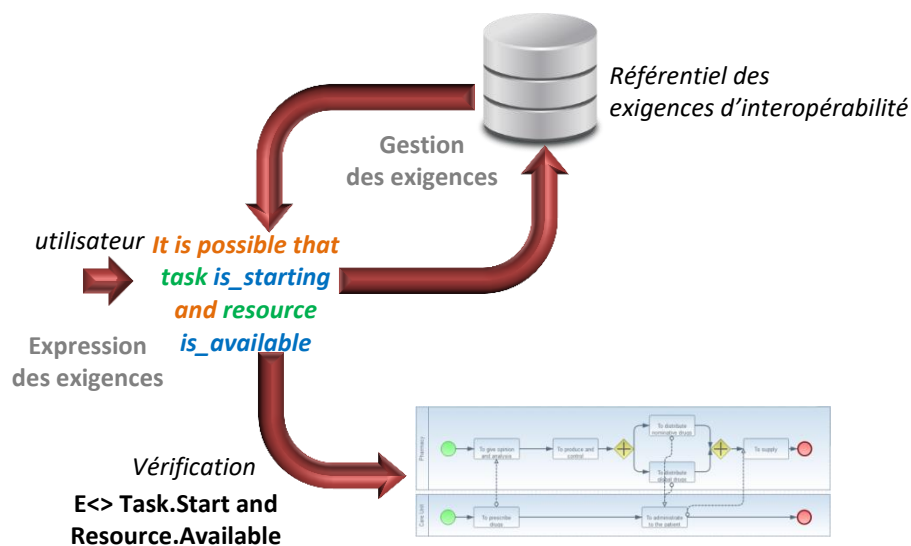


Figure 16. Vers un langage dédié d'expression des exigences d'interopérabilité et la définition d'un référentiel pour la gestion des exigences

Il existe plusieurs langages avec leurs propres vocabulaires et sémantiques pour exprimer des exigences et éviter l'emploi du langage naturel avec toutes ses faiblesses (pour mémoire : ambiguïté, manque de précision). Cependant, il reste difficile de choisir ou de développer un langage permettant (1) d'exprimer des exigences correctement (respect des critères de qualités) (2), de spécifier un domaine particulier (ici, l'interopérabilité) (3), d'autoriser l'emploi d'outils de vérification et (4), suffisamment simple pour être compris aisément et largement.

Parmi les différentes approches et langages dédiés à l'expression des exigences, on retrouve par exemple les cartes mentales ou encore les interviews dans les techniques les moins formelles. D'un point de vue un peu plus formel, on retrouve la méthode KAOS et ses graphes de buts [VLA 91], les checklists pour des exigences standardisées [ISO 03] ou encore User Requirements Notation qui combine deux notations : GRL et UCM [ITU 12].

¹⁹ Un langage dédié (ou Domain Specific Language – DSL) est défini comme « *un langage de programmation ou un langage de spécification exécutable qui offre par des notations et des abstractions appropriées, un pouvoir expressif axé sur, et habituellement limité à, un domaine particulier.* » [VAN 00]

²⁰ Ce langage est développé en anglais, les exemples donnés dans la suite de cette partie présentent donc des développements et des exigences exprimés en anglais.

Toujours dans les aspects non formels, mais proches du langage naturel, la technique des boilerplates [FAN 11] ou SBVR (Semantic Business Vocabulary Rules) [OMG 08] offrent des moyens d'expression basés sur le langage naturel mais limités et contraints. Par exemple, SBVR propose un vocabulaire pour exprimer des règles métiers qui représentent des exigences, basé sur le langage naturel et donc aisément manipulable en comparaison avec des langages tels que la logique temporelle. Enfin, les langages formels (CTL, LTL, TCTL, etc.) permettent d'exprimer des exigences avec une grande précision. En revanche, ceux-ci sont difficiles à appréhender, à lire et à comprendre sans un niveau d'expertise accru.

En conséquence, le langage développé doit (1) s'appuyer sur un ensemble de règles strictes, (2) être aisément manipulable et, enfin (3), suffisamment formel pour permettre l'utilisation de moyens de vérification (*ex. model checking*). Sur la base de ces contraintes, le développement d'un Langage Naturel Contrôlé (CNL) [WYN 09] (à l'exemple des « boilerplates ») représente une solution pertinente qui permet de garder l'avantage de l'évaluation par des méthodes formelles tout en gardant le confort et la puissance d'expression du langage naturel. En d'autres termes un CNL permet de « combler le fossé entre un langage naturel et un langage formel par la médiation entre ces langages » [SCH 10].

Ainsi, le langage dédié à l'expression des exigences d'interopérabilité doit, d'une part, respecter les attendus énoncés précédemment (facilité d'utilisation, pas de connaissances avancées, utilisation de la vérification formelle). D'autre part, il doit être en accord avec le domaine étudié à savoir l'interopérabilité et son évaluation sur des modèles de processus collaboratifs. Les concepts considérés par le langage sont donc :

- Le langage utilisé pour modéliser le processus collaboratif (concept du langage BPMN : tâches, ressources, évènements...).
- La technique de vérification utilisée (ici, le *model checking* via l'outil UPPAL) pour modéliser le comportement du processus et exprimer les exigences [BEH 04]. Tous les modèles à états des objets BPMN définis précédemment sont donc considérés (*ex. une tâche dans l'état attente*). De la même manière, l'ensemble des objets du langage de spécification et leurs sémantiques doivent être considérés (*ex. quantificateur « E » : il existe un chemin*).
- Les concepts d'interopérabilité utilisés pour exprimer des exigences dédiées au domaine de l'interopérabilité (aptitude, responsabilité, autorisation, date d'émission, date de réception...).

Le langage proposé pour exprimer les exigences d'interopérabilité est ainsi restreint uniquement à ces concepts :

$$DSL = \{BPMN_concepts, VerificationMean_concepts, Interoperability_concepts\} \quad [1]$$

Enfin, le langage doit également guider l'utilisateur au cours de l'écriture des exigences, l'informer d'une erreur d'écriture et dans ce cas interdire la vérification et, enfin, être lisible par toute partie prenante destinataire de l'exigence.

A titre d'exemple, l'exigence « *it is possible that a task is working and a resource is active* » est construite à partir des concepts BPMN (tâche et ressource), des états des modèles à états d'une tâche et d'une ressource (working et active). De plus, à la place d'utiliser la modalité « *E<>* » propre au langage de spécification formel (« il est possible d'atteindre un état »), ce dernier est exprimé par « *it is possible that* » qui est plus aisément compréhensible.

En conséquence, le langage met à disposition les objets fondamentaux pour l'expression des exigences d'interopérabilité et leur vérification dans un modèle de processus collaboratif. Ainsi, les modalités (ex. « *it is inevitable* »), les éléments impliqués dans le processus collaboratif (ex. les ressources) et l'état de ces éléments (ex. disponibilité des ressources) et les opérateurs (ex. booléen, comparaison) sont proposés. C'est avec ces objets fondamentaux qu'il est possible de créer une exigence d'interopérabilité. D'un point de vue formel, une exigence d'interopérabilité peut donc être exprimée en respectant la syntaxe suivante :

$$IoReq ::= \textit{modality} p \mid p \textit{leadTo} q \quad [9]$$
$$\textit{modality} ::= \{ \textit{reachable}, \textit{invariantly}, \textit{inevitable}, \textit{potentially} \} \quad [10]$$
$$\textit{leadTo} ::= \{ \textit{lead to} \} \quad [11]$$
$$p, q ::= \textit{proposition} \quad [12]$$

Une proposition peut contenir divers opérateurs (ex. booléen) pour exprimer des exigences plus complexes. Une proposition intègre également les concepts BPMN (ex. tâche), les concepts du langage de spécification des exigences (ex. itération) et les concepts d'interopérabilité (ex. émission message, ressource). Enfin, le langage peut évoluer en fonction des besoins d'expression des exigences d'interopérabilité (ex. un nouveau concept d'interopérabilité peut être ajouté). La grammaire des exigences est présentée dans le *Tableau 11*.

Requirement: (mod=Modality p+=Proposition+) | (p+=Proposition+ lead=LeadTo q+=Proposition+);
 Proposition: obp+=OpenBracket* (iter=Iter? | neg='not?') fact=Fact cbp+=CloseBracket* bool=Bool?;
 Fact: sT=StartFact (cbf+=CloseBracket* | eF+=EndFact*);
 StartFact: (stateTerm=StateTerm (verb=Verb | timeTerm=TimeTerm)) | tab=Tab | val=Value | c=Clock;
 EndFact: (comp=Comparator | op=Operator) ((stateTerm=StateTerm timeTerm=TimeTerm) | val=Value | tab=Tab);
 StateTerm: st=("task" | "resource");
 Clock: c="clock";
 Value: val="value";
 TimeTerm: tt=("timeMax" | "timeMin");
 Verb: v=('is_waiting' | 'is_working' | 'is_starting' | 'is_finished' | 'is_stopped' | 'is_available' | 'is_active');
 Comparator: comp=('is_less_than' | 'is_greater_than' | 'is_equal_to' | 'is_not_equal_to');
 Operator: op=('plus' | 'minus');
 Bool: b=('and' | 'or');
 Iter: it=('for all' | 'it exists') itType=IterType;
 IterType: id_it=('tasks,' | 'sequence_flows,' | 'message_flows,');
 Tab: tab=('task_start [index]' | 'task_end [index]' | 'emission_message_start [index]' | 'emission_message_end [index]' | 'reception_message_start [index]' | 'reception_message_end [index]' | 'reception_sequence_start [index]' | 'emission_sequence_end [index]');
 Modality: reachable="It is possible that" | invariantly="Invariantly," | inevitable="It is inevitable that" | potentially="There is potentially always";
 LeadTo: lt='lead to';
 CloseBracket: cb=')';
 OpenBracket: ob='(';

Tableau 11. Grammaire du langage pour l'expression d'exigence

L'exemple suivant (Figure 17) montre une exigence exprimée et conforme à la grammaire définie.

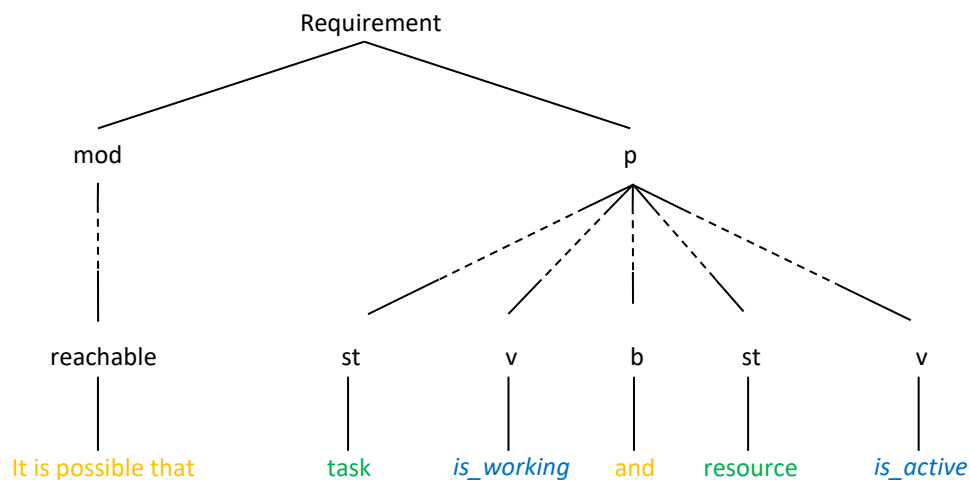


Figure 17. Arbre de syntaxe d'une exigence conforme à la grammaire du langage dédié

Un des objectifs de ces travaux étant la vérification formelle des exigences, il est nécessaire également de disposer de la grammaire qui permet de spécifier les exigences dans le langage qui permet à l'outil de procéder à la vérification. En effet, c'est dans ce langage que sont transformées par la suite les exigences exprimées initialement avec le langage dédié. Dans le cas présent, l'outil UPPAL permet de spécifier des exigences basées sur la logique TCTL. TCTL permet de considérer plusieurs futurs possibles basés sur l'état d'un système. Pour le langage de spécification, UPPAAL met à disposition 4 quantificateurs (2 quantificateurs de chemin : « A », pour tout chemin et « E », il existe un chemin – 2 quantificateurs temporels : « [] », tous les états dans un chemin et « <> », certains états dans un chemin) pour introduire une proposition ou encore l'objet « mène à » pour lier deux propositions. D'un point de vue formel, une exigence en TCTL peut être exprimée en respectant la syntaxe suivante :

$$\text{Query} ::= \text{quantifier } p \mid p \text{ leadTo } q \quad [13]$$

$$\text{quantifier} ::= (\text{pathQuantifier}, \text{temporalQuantifier}) \quad [14]$$

$$\text{pathQuantifier} ::= \{[], \langle \rangle\} \quad [15]$$

$$\text{temporalQuantifier} ::= \{E, A\} \quad [16]$$

$$\text{leadTo} ::= \{\rightarrow\} \quad [17]$$

$$p, q ::= \text{proposition} \quad [18]$$

Une expression (p) est écrite en accord avec les modèles à états existants (états et variables) comme montré dans la Figure 18 pour une tâche décrite par 4 états (Waiting, Start, Working et Stop) et deux variables (timeMin et timeMax) utilisées par une horloge T.

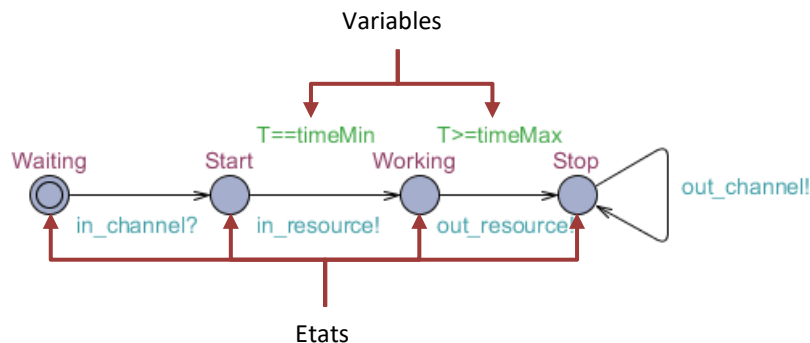


Figure 18. Etats et variables du modèle à états de la tâche pour l'expression d'une proposition en logique temporelle

Une proposition peut contenir divers opérateurs (ex. booléen) pour exprimer des exigences plus complexes. La grammaire d'expression des exigences avec la grammaire TCTL est présentée dans le Tableau 12 (adaptée de [BEH 04]).


```

Query: ((quantifier = QuantifierPath) p = AbstractExpression) |
      (p = AbstractExpression lead = QuantifierLead q = AbstractExpression);
AbstractExpression: atom = Atom (binary = Binary | atom1 += Atom | point += Point)*;
Point: pID = '.' ident = Ident;
Word: w = wValue;
enum wValue: deadlock = 'deadlock' | false = 'false' | true = 'true';
Atom: (unary = Unary? | iter = Iteration) (ident = Ident | word = Word);
Unary: unaryValue = uValue;
enum uValue: plus = '+' | minus = '-' | no='!' | not = 'not';
Binary: binaryValue = bValue;
enum bValue: inf = '<' | infE = '<=' | strictE = '==' | dif = '!=' | supE = '>=' | sup='>' | plus = '+' | minus
= '-' | mult = '*' | div = '/' | percent = '%' | sAnd = '&' | sOr = '|' | exp = '^' | hInf = '<<' | hSup = '>>' |
and = '&&' | or = '||' | infD = '<?' | supD = '>?' | orL = 'or' | andL = 'and' | imp = 'imply';
Iteration: iterOp = ('forall' | 'exists') '(' iterBody1 = Ident ':' iterBody2 = Type ')';
Ident: iValue = ID;
Type: prefix=Prefix typeld = Typeld;
Typeld: identType = Ident | typeWord = TypeWord | scalar = Scalar | int = Inte | struct = Struct;
Struct: s = 'struct' '{' field1 = FieldDecl field2 += (FieldDecl) '}';
FieldDecl: Type ID array1 += ArrayDecl (' ID array2 += ArrayDecl)* ';
ArrayDecl: '[' a = Atom ']' | '[' t = Type ']';
Inte: i = 'int' '[' exprei = Atom ',' exprei1 = Atom ']';
Scalar: scal = 'scalar' '[' expres = Atom ',' expres1 = Atom ']';
TypeWord: typeWordValue = twValue;
enum twValue: int = 'int' | clock = 'clock' | chan = 'chan' | bool = 'bool';
Prefix: prefVaLue = pValue;
enum pValue: urgent = 'urgent' | broadcast = 'broadcast' | meta = 'meta' | const = 'const';
QuantifierLead: qL = '-->';
QuantifierPath: quant = QuantValue;
enum QuantValue: invariantly = "A[]" | inevitable = "A<>" | potentially = "E[]" | reachable = "E<>";

```

Tableau 12. Mise en œuvre de la grammaire TCTL

L'exemple suivant montre une exigence exprimée conforme à la grammaire TCTL (Figure 19).

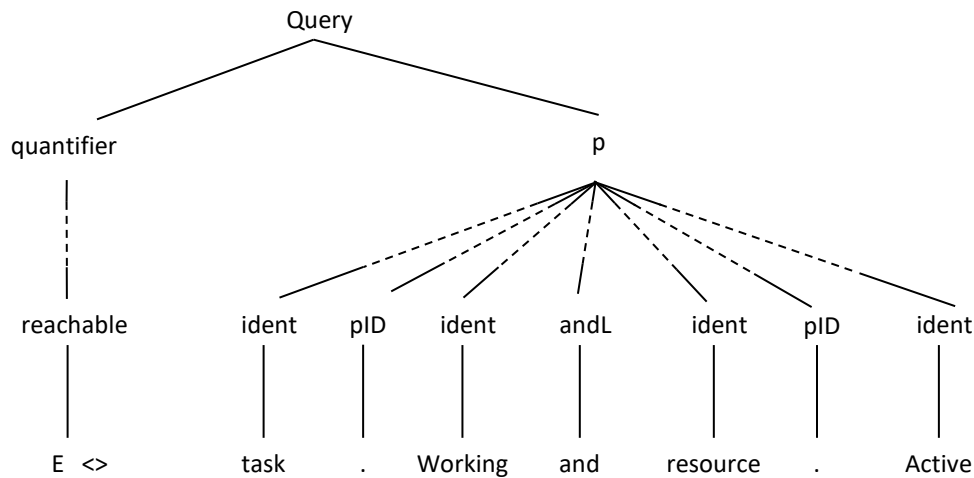


Figure 19. Arbre de syntaxe d'une exigence conforme à la grammaire TCTL

Lorsque les exigences sont écrites avec le langage dédié (DSL), elles doivent être réécrites dans le langage utilisé pour appliquer la technique de vérification (TCTL). Les deux langages permettant d'écrire une exigence sur la même base, la réécriture concerne naturellement la mise en correspondance des deux parties d'une exigence (*modalité proposition* et *quantifier proposition*).

La première correspondance porte sur les modalités utilisées dans le langage dédié et les quantificateurs utilisés en TCTL (Tableau 13). Ainsi, quatre modalités sont définies dans le DSL et gardent le sens des quantificateurs de TCTL. Ces modalités sont lisibles, compréhensibles par un utilisateur non averti et expriment les quantificateurs de chemin (E, A) et temporels ([], <>).

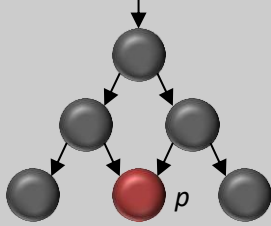
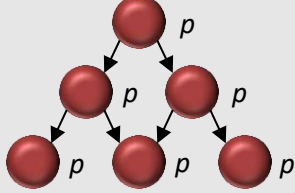
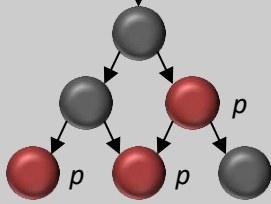
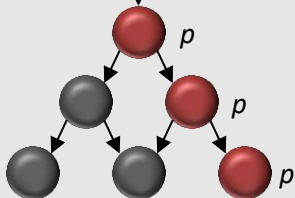
Modalités DSL	Quantificateur TCTL	Lecture
It is possible that (p est satisfait dans certains mondes possibles)	E <> (p est vrai dans un état atteignable)	
Invariantly (p est satisfait dans tous les mondes)	A[] (p est vrai dans tous les états atteignables)	
It is inevitable that (p sera inévitablement satisfait)	A <> (p vrai dans certains états de tous les chemins)	
There is potentially always (p est potentiellement toujours satisfait)	E[] (il existe un chemin dans lequel p est vrai dans tous les états)	

Tableau 13. Correspondance entre les modalités du DSL et les quantificateurs TCTL

La seconde correspondance concerne la proposition exprimée dans le langage dédié avec la proposition exprimée avec TCTL. Par exemple un « *stateTerm* » (ex. « *task* ») est mis en correspondance avec « *iValue* » (chaîne de caractère). De la même manière, un « *Verb* » (ex. « *is_waiting* ») est également mis en correspondance avec « *iValue* ». Cette correspondance est simple en apparence cependant toute la structure d'une exigence dans le langage source doit être mise en correspondance avec la structure d'une exigence dans le langage cible (TCTL). A titre d'exemple, le « *StateTerm* » « *task* » présent dans une exigence signifie que la correspondance est liée avec le modèle à état « *task* ». En outre, la correspondance du « *Verb* » « *is_waiting* » est liée avec l'état « *Waiting* » du modèle de la tâche.

Ainsi, la structure d'une proposition comme « *task is_waiting* » dans le langage dédié doit respecter la structure d'une proposition dans le langage TCTL basé sur les modèles à états et leurs états, dans ce cas « *task.Waiting* ». Cela signifie donc que le symbole « . » doit être inséré entre le nom du modèle à état et le nom de l'état. Cette seconde correspondance tient donc compte de l'aspect « correction » de la proposition en langage TCTL en considérant la règle de grammaire « *pID* ».

En conséquence, la mise en correspondance ne dépend pas uniquement de la simple relation entre des éléments d'une exigence dans le DSL avec des éléments d'une exigence dans TCTL (correspondance 1 à 1). Elle dépend également de la manière dont ces éléments sont arrangés dans le langage source et la correspondance avec l'arrangement qui doit être obtenu dans le langage cible. Le *Tableau 14* présente des exemples de correspondances entre les deux langages.

Proposition DSL	Proposition TCTL
StateTerm (st)	Ident (iValue)
Verb (v)	Ident (iValue)
StateTerm Verb (st v)	Point (pId)
Operator (op)	Binary (binaryValue)
Bool (b)	Binary (binaryValue)
Iter (it)	Iteration (iterop)

Tableau 14. Correspondance entre une proposition du DSL et une proposition TCTL (correspondance partielle)

C'est donc sur la base des correspondances définies que tous les éléments d'une exigence dans le DSL sont transformés en une exigence en TCTL. A cette fin, un ensemble de règles de réécriture est défini. Par exemple, l'exigence « *It is possible that task is_working and resource is_active and clock is_less_than value* » est transformée en « *E<> Task.Working and Resource.Active and T<value* » en respectant les règles de correspondances pour les modalités et les propositions (*Tableau 15*).

Exigence (entrée utilisateur)	DSL		TCTL	Exigence (sortie vérification)
<i>It is possible that</i>	Modality		QuantifierPath	<i>E<></i>
<i>task</i>	StateTerm		Ident	<i>Task</i>
<i>is_starting</i>	Verb		Point (. Ident)	<i>. Starting</i>
<i>and</i>	Bool		Binary	<i>and</i>
<i>resource</i>	StateTerm	>>Ré-écriture>>	Ident	<i>Resource</i>
<i>is_active</i>	Verb		Point (. Ident)	<i>. Active</i>
<i>and</i>	Bool		Binary	<i>and</i>
<i>clock</i>	Clock		Ident	<i>T</i>
<i>is_less_than</i>	Operator		Binary	<i><</i>
<i>value</i>	Value		Ident	<i>value</i>

Tableau 15. Correspondances entre le DSL et TCTL

Enfin, pour réaliser l'évaluation de l'exigence exprimée, il faut l'instancier avec les éléments du processus collaboratif qui est à l'étude. Cela implique de proposer, juste avant de lancer la vérification, les éléments du processus collaboratif qui composent l'exigence.

Ainsi, lorsque l'exigence concerne les tâches et les ressources, l'ensemble des tâches et des ressources du processus collaboratif est proposé pour instancier l'exigence avec les objets sur lesquels l'évaluation porte.

Dans le cas de l'exemple "*It is possible that task is_working and resource is_active and clock is less than value*", tous les noms des tâches et des ressources impliquées dans le processus doivent être proposés pour instancier l'exigence et l'évaluer. De même, il doit être possible de définir la valeur de l'élément « value ». Ce point n'est pas étudié plus en détail car il ne concerne pas directement la rédaction d'une exigence mais sa vérification comme cela est montré dans les parties précédentes.

Le second point de ces travaux s'est concentré sur la gestion des exigences d'interopérabilité. Ce travail initié dans les travaux de [MAL 11] s'est poursuivi pour mettre à disposition un référentiel respectant les critères de qualité d'un référentiel d'exigence et guider les acteurs dans le choix de solutions d'interopérabilité [DAC 14]. Ainsi ce référentiel permet :

- Modifier, retirer, créer des exigences d'interopérabilité et de les positionner dans le référentiel selon les aspects de l'interopérabilité.
- Disposer d'un ensemble d'exigences d'interopérabilité préalablement défini et générique, c'est-à-dire qui peut s'appliquer à tout processus collaboratif (indépendant du contexte), et évaluable par des techniques formelles (graphes conceptuels, *model checking*) ou par expertise. Ces exigences sont celles définies dans les travaux précédents.
- S'assurer par une vision structurée et classée des exigences qu'il n'y a pas de redondance ou de conflits entre des exigences du référentiel et que ces dernières sont aisément identifiables et accessibles.
- Sélectionner un ensemble de solutions dans le cas où une exigence évaluée n'est pas satisfaite.

Le référentiel reprend les dimensions définies dans le modèle GRADEI à savoir les niveaux d'interopérabilité, les catégories d'interopérabilité et les classes d'interopérabilité (cycle de vie de l'interopérabilité) (*Tableau 16*). Cette première structuration cohérente avec l'EIF permet d'établir le lien entre les exigences d'interopérabilité élicitées dans les différents travaux et les solutions d'interopérabilité embarquées dans le cadre. Ainsi, l'ensemble des solutions identifiées ne reste plus isolé ou simplement lié à des exigences abstraites comme les catégories, les niveaux et les approches mais bien à des exigences concrètes, bien formées, précises, vérifiables et cohérentes avec les aspects fondamentaux de l'interopérabilité.

	Conceptuelle	Organisationnelle	Technologique
Métiers	--	<ul style="list-style-type: none"> • Les structures décisionnelles sont définies (C) • Une stratégie commune est définie (I) • La stratégie d'un partenaire n'est pas altérée par la stratégie commune (A) 	--
Processus	<ul style="list-style-type: none"> • Les processus internes sont modélisés 	<ul style="list-style-type: none"> • Le processus collaboratif est modélisé (I) 	--
Services	<ul style="list-style-type: none"> • Les services sont décrits (C) 	<ul style="list-style-type: none"> • Les responsabilités sont identifiées (C) • Un récepteur accuse réception auprès de l'émetteur (I) • Les services internes sont réalisés après la collaboration (R) • Un service impliqué dans un système collaboratif peut toujours être effectué pour des objectifs propres (A) 	<ul style="list-style-type: none"> • Un protocole de communication est défini (C) • La durée d'échange est inférieure à une valeur définie (I) • La durée nécessaire pour utiliser les informations échangées est inférieure à une valeur définie (I) • La durée d'exécution d'une tâche après la collaboration est identique à la durée d'exécution pré-collaboration (R)
Données	<ul style="list-style-type: none"> • Les données sont non ambiguës (C) • Les données reçues sont conformes aux données requises (I) • Les données ne sont pas dégradées après la collaboration (R) 	<ul style="list-style-type: none"> • Les données internes sont sécurisées (C) • Les données à échanger sont disponibles et accessibles (I) 	<ul style="list-style-type: none"> • La quantité de données requise est égale à la quantité reçue (I)

Tableau 16. Exemples d'exigences d'interopérabilité (C-compatibilité, I-interopération, A-autonomie, R-réversibilité)

De plus, le référentiel proposé va plus loin en identifiant un ensemble de critères qui peut orienter le choix de solutions en considérant l'impact de l'exigence sur les objets du système collaboratif (collaboration, participants), sur les performances, stabilité et intégrité (collaboration, participants) et enfin sur les autres exigences. L'ensemble, permet de sélectionner une ou plusieurs solutions parfaitement adaptées aux problèmes identifiés par la non satisfaction d'exigences.

En ce qui concerne **l'objet affecté** par l'interopérabilité. La recherche et la mise en œuvre d'une solution d'interopérabilité sont liées à l'identification de l'objet impacté. En effet, une solution peut influencer, d'une part, le système collaboratif (ex. mission, objectif) et, d'autre part, un acteur de ce système en particulier (ex. mission, objectifs, ressource, service...). A titre d'exemple, l'exigence « *un partenaire dispose des autorisations nécessaires pour accéder aux données partagées* » affecte le système collaboratif dans l'exécution d'un service (ex. perte de temps pour accéder à l'information nécessaire à la réalisation d'un service). L'exigence « *un service « s » réalisé par une ressource « r » impliquée dans le système collaboratif est toujours réalisé au sein du partenaire* » traduit un impact sur un partenaire du système collaboratif (exécution d'une fonction propre à un partenaire).

A partir de cette connaissance, les acteurs peuvent sélectionner/ adapter/ construire une solution d'interopérabilité ou encore « relâcher l'exigence ».

Ensuite, le choix d'une solution peut être guidé par la connaissance du **critère impacté** par l'implantation d'une solution d'interopérabilité en termes de performance, d'intégrité ou de stabilité. Les exigences liées au critère de **performance** concernent les attendus en termes de performance d'interopérabilité ainsi que l'impact sur un partenaire ou le système collaboratif lui-même. Par exemple, l'exigence exprimée comme « *la durée pour connecter des applications est inférieure à x unités de temps* » impacte les performances du système collaboratif notamment en termes de temps d'interfaçage et par extension de temps requis pour réaliser un service. Les exigences liées à la **stabilité** concernent la capacité du système collaboratif (et de ses sous-systèmes) à s'adapter à des changements venant de son environnement. L'exigence « *une ressource « r » s'adapte à un changement de mission du système collaboratif* » illustre la notion de stabilité. En effet, cette exigence indique un changement dans la mission du système collaboratif qui doit être réalisée par la ressource initialement allouée. Dans ce cas, la stabilité du système collaboratif est affectée puisque la ressource doit être capable de participer à la nouvelle mission (ex. nouveau service ou produit attendu) dans des conditions satisfaisantes et en respectant le niveau de performance attendue. Enfin, l'**intégrité** est liée à la capacité d'un système collaboratif (et de ses sous-système) à réaliser ses fonctions suite à une modification interne (ex. perte d'une ressource, d'un sous-système). A titre d'illustration, l'exigence vue précédemment et qui exprime « *qu'un service « s » réalisé par une ressource « r » impliquée dans le système collaboratif est toujours réalisé au sein du partenaire* » altère l'intégrité d'un sous-système puisque sa propre ressource est allouée au système collaboratif et un service interne doit tout de même être réalisé. Dans ce cas, le partenaire doit mettre en œuvre les actions nécessaires pour maintenir son intégrité (ex. surcharge d'une autre ressource interne capable de réaliser le service, heures supplémentaires, sous-traitance...).

Enfin, le dernier critère concerne l'**interdépendance** entre les exigences, c'est-à-dire l'impact de la satisfaction d'une exigence une fois une solution mise en œuvre sur les autres exigences. Par exemple, la non satisfaction de l'exigence « *si une tâche nécessite une aptitude, la ressource allouée possède l'aptitude* » peut mener à une réalisation partielle de la tâche voir sa non réalisation dans le cas le plus défavorable. Une solution peut être de remplacer cette ressource par une autre qui possède l'aptitude ou encore de former la ressource initialement allouée. Le choix de la solution qui consiste à remplacer la ressource initiale peut avoir un impact sur la satisfaction d'autres exigences et plus précisément celles liées à la réalisation des tâches et au partage de ressources (ex. « *Il est possible qu'une tâche démarre et la ressource est disponible* »).

Les travaux réalisés sur l'expression et la gestion des exigences se sont concentrés sur deux points. Le premier point concerne l'expression des exigences d'interopérabilité sur la base d'un langage naturel contrôlé. Ce langage est développé de manière à exprimer des exigences pour des processus collaboratifs, lié au domaine de l'interopérabilité et à des fins de vérification par *model checking*. Ainsi, ce langage offre la possibilité d'exploiter la puissance des techniques de vérification formelle tout en gardant la souplesse et la facilité d'utilisation du langage naturel même si ce dernier est contraint. Enfin, il est suffisamment générique pour exprimer des exigences qui ne relèvent pas de l'interopérabilité

Le deuxième point, partiellement abordé dans les travaux précédents, permet de structurer des exigences déjà identifiées ou nouvellement exprimées et en accord avec les concepts de l'interopérabilité définis notamment dans l'EIF [ISO 09]. Il autorise la gestion d'exigence selon les principes de l'ingénierie des exigences et permet de mettre en relation cet ensemble d'exigences avec les solutions initialement capitalisées dans le cadre.

Ainsi, la sélection de solutions se fait en accord avec le résultat de la vérification et également avec l'ensemble des informations mises à disposition par le cadre pour limiter le risque de choix d'une solution inadaptée. Cependant, si les moyens de vérification formelle offrent des avantages non négligeables (ex. rapidité, fiabilité) le résultat obtenu (vérifié/non vérifié) souffre d'un manque de précision au regard de l'impact réel sur le système collaboratif et ses sous-systèmes (une non vérification n'implique pas nécessairement l'absence totale d'interopérabilité). Bien que cette connaissance de l'impact soit introduite à travers les informations supplémentaires apportées par le cadre, il reste à un niveau de précision trop faible pour permettre une exploitation effective.

Le *Tableau 17* synthétise les travaux effectués dans le cadre de l'écriture des exigences d'interopérabilité et de leur structuration.

Communications	[RI6], [CI20], [CI23], [WI3]
Projet	CARIONER II
Démonstrations	https://goo.gl/s0dY9r <ul style="list-style-type: none">• drugCircuit.avi• AdvancedRequirementWriting.avi

Tableau 17. Synthèse des travaux de recherche sur l'écriture des exigences d'interopérabilité et leur structuration

3.4. Dimension analyse de l'impact de l'interopérabilité

L'analyse de l'impact de l'interopérabilité a pour objectif d'obtenir une vision plus précise de l'impact que peut avoir une exigence d'interopérabilité dans le cas de la satisfaction (ou non) de cette dernière. Bien que les approches formelles déployées dans les travaux précédents présentent des avantages (rapidité, véracité), le résultat obtenu (qualitatif : satisfaite/non satisfaite) n'est pas suffisamment précis pour connaître l'impact réel d'un défaut d'interopérabilité. De la même manière, si des métriques d'interopérabilité sont évaluées, le résultat ne renseigne aucunement sur l'impact produit sur le système étudié. Les travaux sur l'analyse ont été réalisés dans le cadre de la thèse de M. Billaud pour l'ingénierie de systèmes de systèmes [BIL 15]. Dans ce sens, ils ne se concentrent pas uniquement sur un point de vue particulier d'un système collaboratif comme cela est le cas avec les travaux précédents avec le point de vue fonctionnel (modélisation processus avec le langage BPMN). Ils considèrent plusieurs points de vue qui permettent d'analyser plus précisément l'interopérabilité et guider ainsi la conception d'un SdS.

Pour répondre à cette problématique, il est nécessaire d'identifier les points de vue pertinents et de mettre à disposition des langages de modélisation permettant de prendre en compte les concepts des SdS et les concepts nécessaires à l'analyse de l'interopérabilité comme cela a été le précédemment.

Ensuite, il est également nécessaire de fournir les moyens d'analyse de l'impact de l'interopérabilité suite à son évaluation selon des perspectives d'analyse qui sont fondamentales pour un SdS (Figure 20).

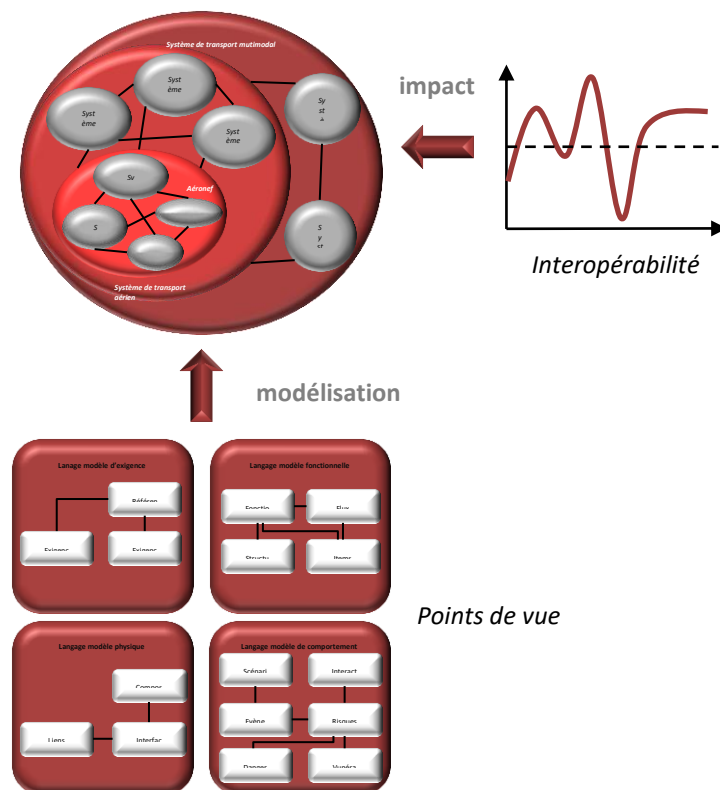


Figure 20. Vers la modélisation multi-points de vue pour l'ingénierie des systèmes de systèmes et l'analyse de l'impact de l'interopérabilité

Les langages de modélisation proposés permettent de faciliter la compréhension, la description et l'analyse des SdS selon différents points de vue. Au-delà des besoins de modélisation classiques pour les systèmes complexes (ex. représentation fonctionnelle), ces langages considèrent également les besoins de modélisation spécifiques au SdS. Par exemple, les notions de missions propres au SdS ainsi que les interactions entre les différents systèmes composants le SdS doivent être caractérisées. De la même manière, les exigences doivent être décrites au niveau du SdS (ex. interopérabilité, évolution) ainsi qu'au niveau des sous-systèmes (ex. indépendance). Ainsi, les langages de modélisation déployés tiennent compte des concepts et des relations qui permettent une description précise des SdS en vue de leur ingénierie et surtout de l'analyse de l'exigence d'interopérabilité.

Pour ce faire, les langages s'appuient sur les langages classiquement utilisés en modélisation des systèmes et enrichis afin de couvrir les besoins d'analyse de l'impact de l'interopérabilité (ex. définition des interfaces, des interactions...). Ces langages de modélisation permettent la couverture de quatre points de vue à savoir : exigences, comportemental, fonctionnel et physique. Ces différents points de vue ainsi que leurs concepts fondamentaux sont présentés dans la Figure 21.

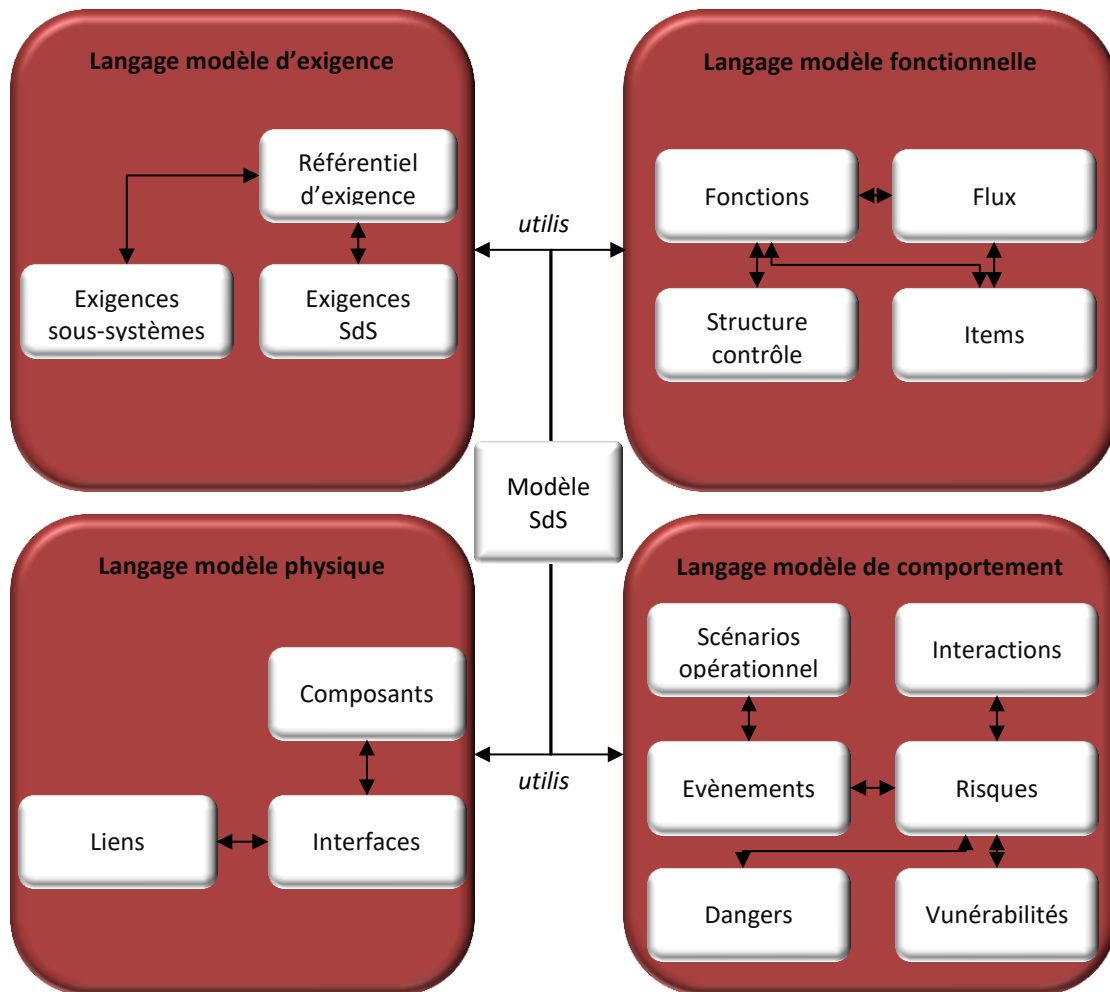


Figure 21. Points de vue pour la modélisation des Systèmes de Systèmes

Le **point de vue des exigences** permet la description et la structuration des exigences (1) des parties prenantes du système de systèmes en termes de mission, de fonctions ou encore de performance, (2) des sous-systèmes qui composent le SdS en termes de mission, d'indépendance, de performance et (3) du SdS lui-même en termes d'évolution, de connectivité. Le langage de modélisation proposé est une extension du diagramme des exigences SYSML [OMG 15]

Le **point de vue comportemental** permet la description de la dynamique du SdS. En plus de la description des scénarios opérationnels, ce point de vue intègre également les phénomènes liés aux interactions et leurs effets aussi bien internes, *i.e.*, entre des sous-systèmes, qu'externes, *i.e.*, entre le SdS et son environnement. Les effets sont ceux définis dans [MAN 02] : efficace, inefficace, absent, excessif et néfaste.

Le **point de vue fonctionnel** est utilisé pour la description des fonctions réalisées par l'ensemble des sous-systèmes ainsi que leur enchaînement pour exécuter la mission du SdS. Le langage de modélisation défini pour la modélisation de ce point de vue est une extension des eFFBD²¹ [DOD 01].

²¹ enhanced Functional Flow Block Diagram

Enfin, le **point de vue physique** permet la description de l'organisation des sous-systèmes au sens humain, matériel et logiciel ainsi que de leurs interfaces, *i.e.*, techniques, organisationnelles, interface Homme-Machine, nécessaires pour réaliser les interactions. La modélisation de ce point de vue est supportée par le langage PBD²² étendu [INC 11].

C'est sur la base de cette modélisation intégrant les divers concepts liés à l'interopérabilité (interfaces, interactions, exigences d'interopérabilité, processus collaboratif) que l'impact de l'interopérabilité sur les SdS est analysé. Celui-ci est alors défini sur la base des travaux initiés sur l'impact de l'exigence d'interopérabilité dans un SdS au cours de son cycle de vie et selon les aspects de performance, de stabilité et d'intégrité du SdS [DAC 12].

Ces travaux proposent donc d'étudier l'impact selon les perspectives d'analyse de **performance** et d'**adaptabilité**. La perspective de performance concerne l'analyse de l'impact de l'interopérabilité sur la performance du SdS qui peut résulter d'un défaut d'interopérabilité entre des sous-systèmes. La performance est analysée en termes de :

- Délai de réalisation de la mission du SdS. Ce délai dépend du délai de réalisation des activités des sous-systèmes mais également du délai d'interaction entre les sous-systèmes.
- Qualité du service rendu par le SdS. Cette qualité dépend de la qualité de réalisation des activités mais également de la qualité des interactions entre les sous-systèmes.
- Coût engagé pour la réalisation de la mission du SdS. De la même manière que pour le délai et la qualité, ce coût dépend du coût des activités des sous-systèmes mais également du coût des interactions entre les sous-systèmes.

Pour évaluer ces critères de performance, ces travaux s'appuient sur les métriques développés dans la littérature pour évaluer la performance d'interopérabilité selon son cycle de vie (*ex.* latence de données).

La perspective d'adaptabilité regroupe les aspects d'intégrité et de stabilité du SdS et, est définie comme la capacité d'un SdS à revenir à un niveau de performance défini suite à une perturbation (interne ou externe) liée à l'interopérabilité de ses sous-systèmes. Deux types de perturbations sont caractérisés pour définir l'impact de l'interopérabilité à savoir :

- Les perturbations externes, *i.e.*, l'ajout d'un sous-système au SdS, la création d'une interaction, la destruction, la modification d'une interaction entre un sous-système du SdS et un système dans l'environnement et, enfin, la modification d'un système dans l'environnement du SdS (*ex.* participation à une nouvelle collaboration avec un autre SdS).
- Les perturbations internes, *i.e.*, la modification d'un sous-système, le retrait d'un sous-système du SdS, la modification, la création et, enfin, la suppression d'une interaction entre sous-systèmes.

Ainsi l'impact de l'interopérabilité est analysé pour toutes les phases de son cycle de vie (compatibilité, interopération, autonomie et réversibilité) sur les performances et l'adaptabilité d'un SdS.

²² Physical Block Diagram

L'impact sur les performances est caractérisé selon la typologie classique des variations de performance [DUC 99] à savoir l'augmentation, la stabilité et la diminution. Cette première analyse donne donc une **vision statique** de l'impact de l'interopérabilité. L'impact sur l'adaptabilité s'appuie sur la courbe établissant la relation entre les performances du SdS et le cycle de vie de l'interopérabilité ainsi que sur les différentes perturbations définies (Figure 22). Cette deuxième analyse donne une **vision dynamique** de l'impact de l'interopérabilité.

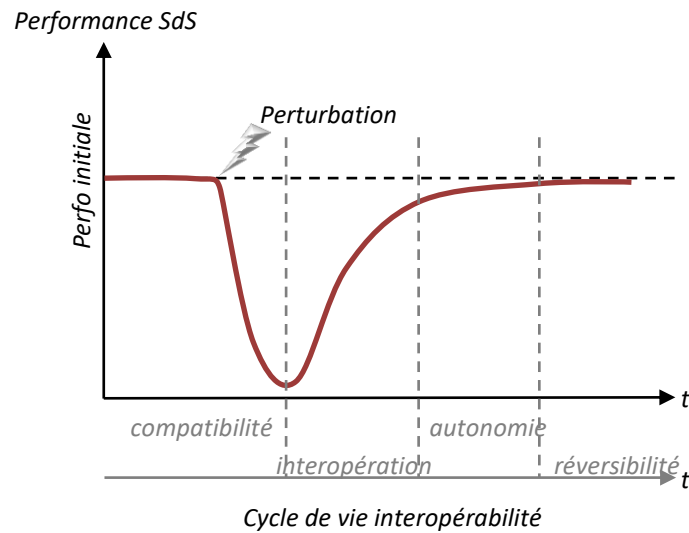


Figure 22. Concepts pour l'analyse de l'impact de l'interopérabilité sur l'adaptabilité du SdS (performance, cycle de vie de l'interopérabilité et perturbations)

Il en résulte une **matrice d'impact** à deux dimensions : interopérabilité et perspectives d'analyse (Tableau 18).

	Interopérabilité			
	Compatibilité	Interopération	Autonomie	Reversibilité
Performance	→			
Adaptabilité				

Tableau 18. Matrice d'impact de l'interopérabilité sur les perspectives d'analyse

Pour étudier l'impact de la compatibilité sur les performances, trois niveaux de compatibilité sont distingués : l'incompatibilité, la compatibilité partielle et la compatibilité totale. L'incompatibilité traduit l'utilisation de moyens techniques, de ressources incapables d'interagir et nécessitant un effort d'interfaçage important. Le SdS n'est pas en mesure d'atteindre les performances souhaitées. La compatibilité partielle se traduit par la possibilité de réaliser des interactions mais à des niveaux de performances inférieurs à ceux souhaités. La compatibilité totale indique l'utilisation d'outils, méthodes identiques ou encore l'utilisation d'interfaces standards autorisant l'interfaçage direct et l'atteinte des niveaux de performances souhaités.

En ce qui concerne l'impact de la compatibilité sur l'adaptabilité, elle est étudiée selon les perturbations telles que l'interfaçage de sous-systèmes et la modification d'un sous-système nécessitant la définition ou la modification d'interface existantes. L'adaptabilité est alors caractérisée selon l'écart de performance Δp_{comp} (performance initiale du SdS et performance atteinte au cours de la connexion des sous-systèmes) et l'écart de temps Δt_{comp} (date d'introduction de la perturbation et date de possibilité d'interaction). L'étude de l'impact de la compatibilité sur le SdS est présentée dans le *Tableau 19*.

		Compatibilité	
Performance	Incompatibilité :	diminution	
	Compatibilité partielle :	diminution	
	Compatibilité totale :	stabilité	
Adaptabilité	Perturbation 1/3 :	Ajout sous-système	
	2 :	Modification sous-système	
	Profil		
	Définition	$\Delta p_{comp} \rightarrow 0$ $\Delta p_{comp} \text{ observé} \leq \Delta p_{comp} \text{ attendu}$	$\Delta t_{comp} \rightarrow 0$ $\Delta t_{comp} \text{ observé} \leq \Delta t_{comp} \text{ attendu}$

Tableau 19. Matrice d'impact de la compatibilité sur le SdS

L'étude de l'impact de l'interopération sur les performances du SdS suit le même principe que pour la compatibilité par la définition de trois niveaux d'interopération. Ainsi, la non-interopération traduit l'absence d'interactions entre des sous-systèmes et donc l'absence de performances. L'interopération partielle indique la présence d'interactions mais à un niveau de performance différent du niveau attendu. L'interopération totale indique que les interactions sont réalisées au niveau de performance souhaité.

L'impact de l'interopération est caractérisé directement après que les systèmes soient compatibles et puissent interopérer selon l'écart de performance Δp_{int1} (performance attendue du SdS et performance obtenue du SdS) et l'écart de temps Δt_{int1} (date d'obtention des performances et date de début d'interopération). Il est également caractérisé selon la perturbation de modification d'un sous-système pendant l'interopération avec l'écart de performance Δp_{int2} (niveau de performance pré-perturbation et niveau de performance post-perturbation) et l'écart de temps Δt_{int2} (date d'apparition de la perturbation et date d'obtention d'un niveau de performance de référence).

Enfin, cet impact est également étudié pour la suppression d'une interaction. Le *Tableau 20* présente l'étude de l'impact de l'interopération sur le SdS en termes de performance et d'adaptabilité.

		Interopération	
Performance	Non interopération :	diminution	
	Interopération partielle :	diminution	
Adaptabilité	Interopération partielle :	augmentation	
	Interopération totale :	stabilité	
Perturbation		1/2/3 : sans perturbations 4/5/6 : modification sous-système 7 : suppression interaction	
Profil			
Définition		$\Delta p_{int1} \rightarrow 0$ $\Delta p_{int1} > 0$ (tolérance acceptée) $\Delta p_{int1} < 0$ (tolérance acceptée) $\Delta p_{int2} \rightarrow 0$ $\Delta p_{int2} > 0$ (tolérance acceptée) $\Delta p_{int2} < 0$ (tolérance acceptée)	$\Delta t_{int1} \rightarrow 0$ $\Delta t_{int1} \text{ observé} \leq \Delta t_{int1} \text{ attendu}$ $\Delta t_{int2} \rightarrow 0$ $\Delta t_{int2} \text{ observé} \leq \Delta t_{int2} \text{ attendu}$

Tableau 20. Impact de l'interopération sur le SdS

L'impact de l'autonomie sur le SdS est étudié selon les performances du SdS au regard des performances de ses sous-systèmes. Trois cas sont alors étudiés. Premièrement, le sous-système privilégie ses propres performances au dépens de celles du SdS. Ensuite, le sous-système privilégie sa participation aux performances du SdS au détriment des siennes. Enfin, le sous-système atteint ses performances et permet au SdS d'atteindre les siennes comme cela est initialement prévu.

L'étude de l'impact de l'autonomie sur l'adaptabilité du SdS est réalisée sur la considération de la modification d'un sous-système au regard de sa participation au SdS. Ainsi, l'adaptabilité est caractérisée par la variation de performance du SdS dans des limites minimales et maximales qui peuvent être atteintes en même temps que les sous-systèmes conservent des niveaux de performances acceptables. L'étude de l'impact de l'autonomie sur le SdS est présentée dans le *Tableau 21*.

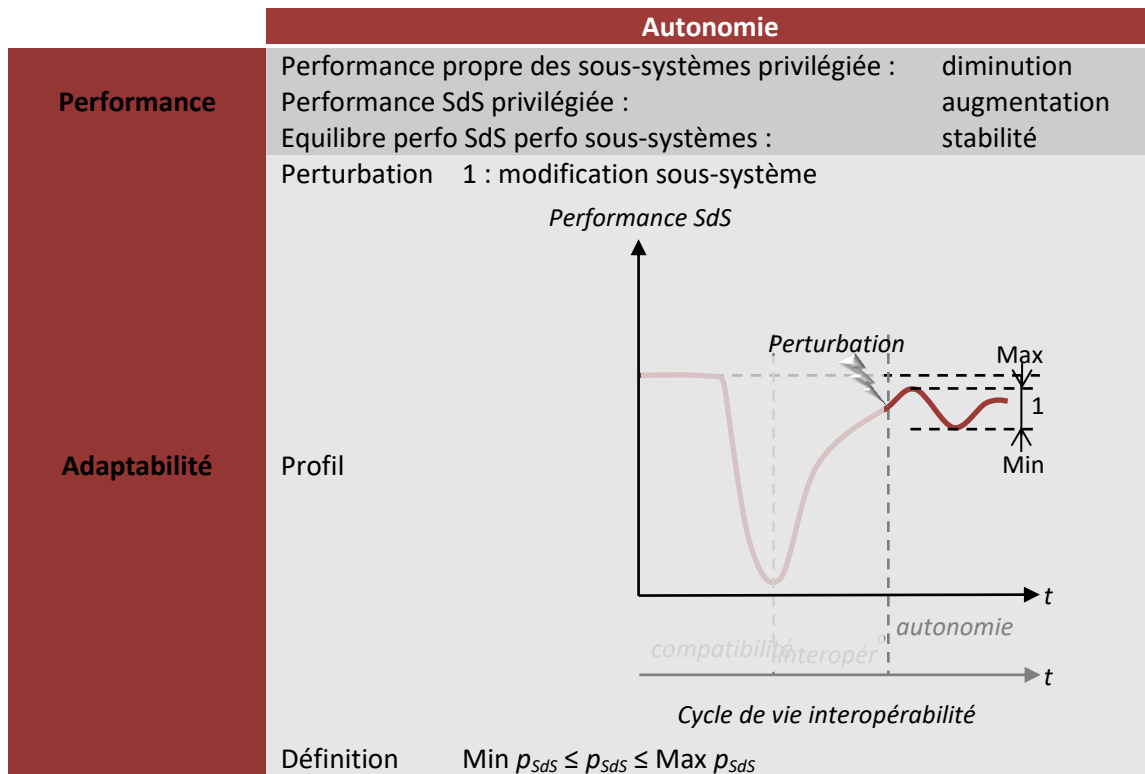


Tableau 21. Impact de l'autonomie sur le SdS

L'impact de la réversibilité sur les performances du SdS est caractérisé, premièrement, par le fait que le SdS exige la séparation d'un sous-système car celui-ci ne participe pas de manière adéquate au SdS et ne lui permet pas d'atteindre ses performances. Il est caractérisé, ensuite, par la séparation prévue d'un sous-système et la poursuite normale de la mission du SdS. Il est caractérisé, enfin, par un sous-système qui se sépare volontairement du SdS alors que ce dernier requiert toujours ses services pour réaliser sa mission.

L'impact de la réversibilité sur le SdS concerne le retrait d'un sous-système et se caractérise par deux écarts à savoir sur les performances après la perte d'un sous-système Δp_{rev} (performance courante du SdS et performance post perte d'un sous-système) et le temps Δt_{rev} (date de l'arrêt de la participation du sous-système et date de retour à un niveau de référence). Le *Tableau 22* présente l'impact de la réversibilité sur le SdS.

Réversibilité	
Performance	Séparation et performance du SdS acceptable : stabilité
	Séparation d'un sous-système perturbateur : augmentation
Réversibilité	Séparation sous-système mais service toujours requis : diminution
	Perturbation 1 : retrait sous-système 2 : retrait sous-système
Profil	<p style="text-align: center;">Cycle de vie interopérabilité</p>
Définition	$\Delta p_{rev} \rightarrow 0$ $\Delta p_{rev, observé} \leq \Delta p_{rev, attendu}$
	$\Delta t_{rev} \rightarrow 0$ $\Delta t_{rev, observé} \leq \Delta t_{rev, attendu}$

Tableau 22. Impact de la réversibilité sur le SdS

Les travaux sur la prise en compte de l'interopérabilité dans l'ingénierie des SdS ont initié deux aspects importants pour le développement des systèmes collaboratifs. Tout d'abord ils ne se sont pas concentrés uniquement sur un aspect comme les précédents sur les processus collaboratifs, mais ont adopté une vision multi point de vue : exigence, fonctionnelle, physique et comportementale. Cela a conduit à la proposition de langages de modélisation pour faciliter la conception des SdS pour développer et analyser l'exigence d'interopérabilité. Ensuite, une analyse de l'impact de l'interopérabilité sur les perspectives d'analyse de performance et d'adaptabilité pour chaque phase du cycle de vie de l'interopérabilité est proposée et permet d'obtenir une matrice d'impact offrant une vision précise de l'impact de l'interopérabilité sur un SdS. A partir des modèles du SdS qui permettent de caractériser les aspects de l'interopérabilité tels que les interfaces ou encore les interactions et des outils pour évaluer l'interopérabilité (métrique, vérification) et de leurs résultats, l'impact peut être déterminé sur la base de la matrice d'impact. Ces travaux marquent ainsi la fin du cycle de recherche sur les composantes de l'interopérabilité en tant qu'exigence à satisfaire dans les systèmes collaboratifs.

Le *Tableau 23* synthétise les travaux effectués dans la modélisation multi points de vue pour l'ingénierie des SdS et l'analyse de l'impact de l'interopérabilité.

Communications	[CI21], [CI22], [CI24], [CI25]
Thèse	Stéphane Billaud

Tableau 23. Synthèse des travaux de recherche sur l'ingénierie des SdS et l'impact de l'interopérabilité

3.5. Conclusion

Les travaux sur la formalisation et l'évaluation de l'exigence d'interopérabilité ont permis d'explorer l'interopérabilité sous une vision formelle et selon le paradigme de l'ingénierie des systèmes. Plus précisément, ces travaux ont permis :

- D'appliquer des techniques de vérification formelle pour analyser l'interopérabilité dans des processus collaboratifs. Comme cela est mentionné dès le début, ces deux domaines travaillent à des niveaux de formalisation radicalement différents. En conséquence, leur mise en relation demande la mise en œuvre de moyens tels que les enrichissements d'un langage au niveau conceptuel et opérationnel. Cela demande également de mettre en œuvre des mécanismes de transformation afin de permettre l'applicabilité des concepts d'un langage dans un autre langage. Cette mise en œuvre paraît évidemment délicate mais montre bien que les techniques de vérification peuvent supporter une partie de l'évaluation de l'interopérabilité et ainsi offrir à des parties prenantes de se concentrer sur les exigences qui peuvent être uniquement évaluées par expertise.
- D'exprimer leur propres exigences d'interopérabilité ainsi que de disposer d'exigence d'interopérabilité générique (*i.e.*, applicable à n'importe quel processus collaboratif). Cette expression, basée sur des principes strictes issus de l'ingénierie des exigences, permet à chaque destinataire (utilisateurs finaux, développeur de solutions) de disposer d'exigences partageables, compréhensibles et évaluables pour les uns et permettant de développer des solutions adéquates pour les autres. Bien que ce langage dédié est utilisé pour exprimer des exigences liées à l'interopérabilité, il est suffisamment générique pour pouvoir décrire et vérifier des exigences qui sont liées à d'autres caractéristiques (*ex.* gestion des ressources). Il en est de même pour les graphes conceptuels qui permettent la description d'exigences autres que celles liées à l'interopérabilité
- D'établir le lien entre le domaine du problème (exigence et analyse) et le domaine de la solution (cadre EIF) pour autoriser un choix de solution précis.

Ces travaux ont également montré qu'une réponse précise à la question initiale « *Est-ce que le système collaboratif satisfait l'exigence d'interopérabilité ?* » demande d'appréhender le système à l'étude dans toute sa complexité et nécessite de faire interagir des domaines et moyens techniques issus :

- De la **modélisation d'entreprise** avec la modélisation de processus collaboratifs pour obtenir une représentation de la collaboration et la définition d'attributs pour obtenir, à travers ce modèle, une vision de l'interopérabilité. C'est sur ce modèle que la satisfaction (ou non) de l'exigence d'interopérabilité est évaluée.
- De l'**ingénierie système** avec ses principes rigoureux d'ingénierie des exigences pour définir précisément le domaine du problème qui doit être satisfait à terme.
- De la **vérification formelle** avec le *model checking* s'appuyant sur les modèles à états et la logique temporelle ou encore les graphes conceptuels et ses mécanismes de projection.
- De l'**ingénierie dirigée par les modèles** avec ses principes de transformations dans le but d'établir un pont entre un modèle d'entreprise et des graphes conceptuels/modèles à états et des exigences en langage naturel contrôlé et les graphes conceptuels/logique temporelle et ce afin de pouvoir exploiter les outils de vérification et ainsi vérifier l'exigence d'interopérabilité.

4. Programme de recherche

4.1. Vers une amélioration des techniques de formalisation et d'évaluation des exigences.

Vérification formelle

L'utilisation de techniques telles que le *model checking* est rapidement limitée lorsque le processus devient plus complexe (nombre d'activités, d'attributs). Cette limite de type explosion combinatoire est parfaitement connue au sein de la communauté « vérification ». Dans ce cas, plus le nombre d'objets (activités, branchements, ressources...) du modèle du processus est important et plus le nombre d'états pour représenter le processus par un réseau d'automates temporisés est important (taille de l'espace d'état). L'évaluation peut alors demander du temps et la technique perd cet avantage par rapport à l'expertise. Dans le cas le plus défavorable, l'évaluation peut ne pas aboutir. Dans ce cas, les techniques (symbolique, ordre partiel, symétrie) qui permettent de vérifier des modèles de grande taille doivent être explorées pour que cette technique conserve ses avantages (rapidité d'analyse, fiabilité du résultat) [BOU 15].

Si ces techniques sont propres à la vérification formelle, elles doivent être également explorées sur des modèles moins formels tels que les modèles de processus collaboratif. D'une part, cela peut préparer, en amont, l'utilisation d'outils de vérification formels et éviter ces problèmes de vérification de grands modèles. D'autre part, cette étude peut être utile pour l'expertise, moins marginale dans le domaine de la modélisation d'entreprise. En effet, cette technique est également fortement limitée face à des modèles de grande taille et comportant énormément d'informations. En ce sens, le développement de techniques pour cibler des points ou des chemins d'analyse pertinents, fournir un volume d'information manipulable ou encore obtenir un modèle « compact » et « fidèle » au modèle initial tout en garantissant une évaluation complète, précise et fiable, doit être réalisé.

DSL pour l'expression des exigences

Dans les travaux précédents, le langage dédié concerne seulement les exigences d'interopérabilité lorsque celle-ci est liée au temps. Actuellement les exigences d'interopérabilité atemporelle sont proposées à la vérification uniquement sur la base d'un primo-CNL directement lié aux graphes conceptuels. En d'autres termes, une partie prenante ne peut exprimer ses exigences dédiées à la vérification par les graphes conceptuels. Celles-ci sont déjà exprimées en langage naturel sur la base des exigences existantes et directement proposées.

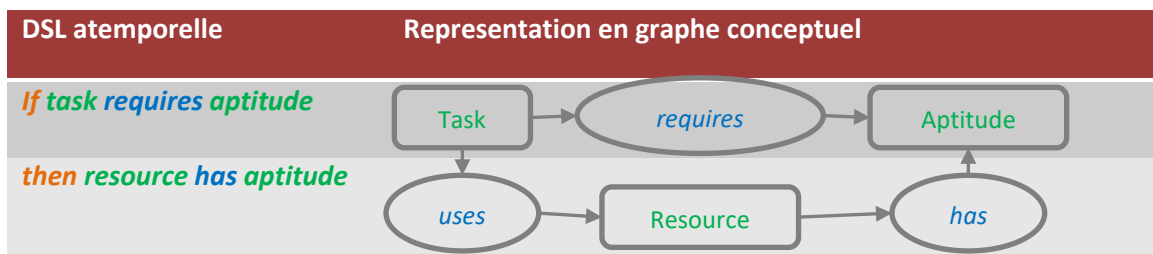


Figure 23. Vers un langage dédié d'expression d'exigences d'interopérabilité atemporelle

Il s'agit alors de développer un langage dédié – sur les mêmes principes que pour le premier langage - autorisant l'expression et ensuite la vérification d'exigence atemporelle avec les graphes conceptuels.

Enfin, pour exprimer des exigences indépendamment d'un langage dédié selon la technique utilisée, il est nécessaire d'unifier ceux définis précédemment pour obtenir un langage unique d'expression des exigences pour l'évaluation par *model checking*, par graphes conceptuels ou encore par expertise. Cet aspect de langage unifié a été exploré, par exemple, avec LUSP [CHAP 13]. Cependant, il reste d'un niveau formel et d'expertise trop élevé pour envisager une utilisation par des non experts. Il peut néanmoins, de par son niveau de formalisation et son lien possible avec des techniques de vérification formelle être utilisé comme pivot entre un langage naturel contrôlé et les méthodes formelles d'évaluation.

4.2. Vers un écosystème des exigences non fonctionnelles « -ilities »

Les travaux sur l'évaluation de l'interopérabilité ont montré l'importance du traitement de ce type d'exigences non fonctionnelles pour supporter la réalisation des missions et l'atteinte des performances d'un système collaboratif d'une part, et le maintien de l'indépendance et de l'autonomie pour ses sous-systèmes d'autre part. Les études sur les exigences non fonctionnelles se concentrent souvent sur une caractéristique propre de celles-ci (objectif d'évaluation, d'expression/modélisation ou encore de définition selon le domaine d'application, de développement de solutions pour obtenir sa satisfaction) sans nécessairement les repositionner dans leur environnement, *i.e.*, sans tenir compte des autres exigences avec lesquelles elles peuvent être liées. C'est le cas par exemple avec l'interopérabilité, pour laquelle des solutions sont développées sans forcément s'intéresser aux problèmes que peut générer celle-ci sur d'autres aspects de la collaboration. En ce sens, les travaux initiés sur l'étude de l'impact de l'interopérabilité ont montré l'importance de la prise en compte de cet aspect notamment pour les exigences de type performances d'un système collaboratif ou encore les propriétés comme la stabilité et l'intégrité (regroupées sous la notion d'adaptabilité). Ces primo travaux ont considéré la relation « un à un » (interopérabilité *impacte* performance), unidirectionnelle (interopérabilité → performance) sans tenir compte de l'intensité de l'impact de l'interopérabilité sur ces performances soit de manière générique (applicable à tous les domaines) soit par rapport au besoin spécifique de performances du système collaboratif. Enfin, bien que la notion d'effet soit abordée, son étude n'a pas été poussée suffisamment loin pour une exploitation effective.

En outre, rappelons que l'interopérabilité appartient à un ensemble d'exigences non fonctionnelles appelées « -ilities » qui peuvent également impacter le système collaboratif en termes de performances mais qui peuvent aussi être liées entre elles et s'impacter mutuellement. Sans une définition, une caractérisation et une analyse des relations complexes qui peuvent exister entre ces différentes « -ilities », les méthodes et outils développés spécifiquement pour l'étude d'une « -ility » en particulier se trouvent limités. En effet, ils ne considèrent pas le point de vue essentiel de la prise en compte des interactions inter-« -ilities » et de leurs possibles effets sur leur environnement. De même, ces outils et méthodes se focalisent localement sur le problème sans étudier les impacts possibles sur l'environnement et, donc, le système collaboratif à terme. C'est le cas notamment de l'étude de l'impact de l'interopérabilité sur l'adaptabilité qui ne tient pas compte, par exemple, des possibles autres relations dans le même temps (1) de l'interopérabilité avec des « -ilities » et (2) de l'adaptabilité, impactée, avec d'autres « -ilities ». Ainsi, la connaissance de ces relations et de leurs effets peut permettre des choix d'implantation de solutions autre que ceux qui peuvent être réalisés pour satisfaire une exigence particulière. A titre d'exemple, l'augmentation de l'interopérabilité peut potentiellement engendrer des problèmes de sécurité des informations et donc de la satisfaction de

cette exigence. La mise à disposition d'informations pertinentes sur la relation entre ces deux exigences doit faciliter, à terme, la recherche d'une solution satisfaisant ces dernières. Plusieurs études démontrent l'importance des « -ilities » et de leur impact notamment dans l'ingénierie des systèmes [ROS 15] de systèmes et des recherches sont initiées pour permettre une prise en compte de ces dernières lors de l'ingénierie [RIC 14].

L'étude de ces aspects a été initiée au travers du projet re(h)strain qui s'intéresse à la résilience du réseau ferré grande vitesse franco-allemand et des travaux de thèse de B. Moradi sur la formalisation et l'évaluation de propriétés non-fonctionnelles appliquées à la résilience (Figure 24).

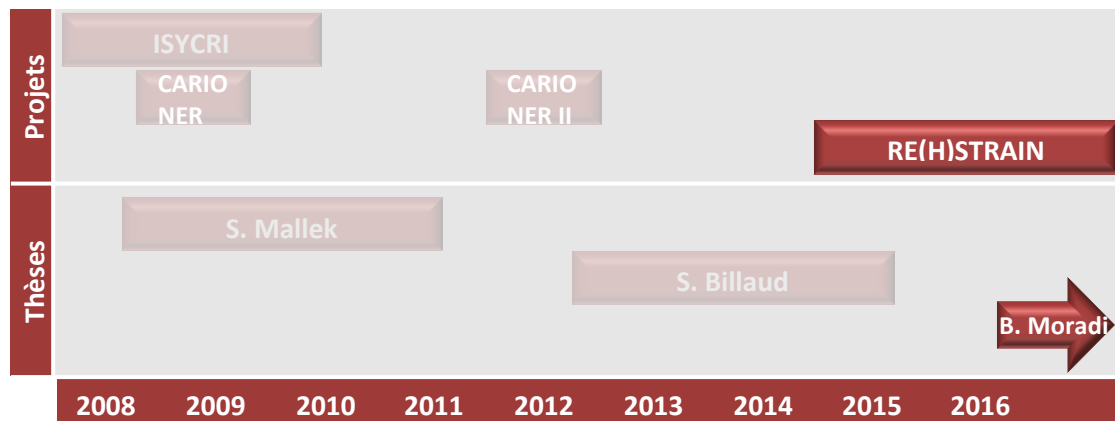


Figure 24. Poursuite des travaux sur les « -ilities »

La dépendance entre les « -ilities » est présente dans la littérature notamment dans les travaux de [DEW 12]. A partir de requêtes sur le web concernant les « -ilities » qui jouent un rôle majeur dans les systèmes, il établit alors un graphe de relation entre toutes les « -ilities » comme montré dans la Figure 25. A titre d'exemple, il permet de constater que l'exigence de qualité est liée à l'exigence de fiabilité. En d'autres termes, l'étude de la qualité est liée également à l'étude la fiabilité. Ce graphe montre également l'intensité du lien (épaisseur des flèches sur le graphe) entre les « -ilities » selon les recherches (au sens scientifique) effectuées sur ces relations. A titre d'exemple l'étude de l'interopérabilité avec la sécurité est génératrice de nombreux travaux. Ce graphe revêt une importance particulière car il met en évidence la notion d'impact entre les « -ilities » et permet de dégager un ensemble de questions à traiter pour étudier précisément cet impact.

En conséquence, c'est sur la base de ce graphe que la suite développe une réflexion sur les travaux de recherche concernant les exigences non fonctionnelles de type « -ilities » considérées comme un écosystème, c'est-à-dire « un ensemble d'« -ilities » qui interagissent entre elles et avec les autres exigences (fonctionnelles, non fonctionnelles) du système dans lequel elles sont développées ». L'objectif n'est donc plus d'aborder une « -ility » de manière isolée mais de la considérer selon différents aspects qui permettent de faire émerger les relations et leur nature avec son environnement à savoir : l'identification de la **relation**, la **direction** de la relation, l'**intensité** de la relation, le **degré** de relation et les **effets** de ces relations.

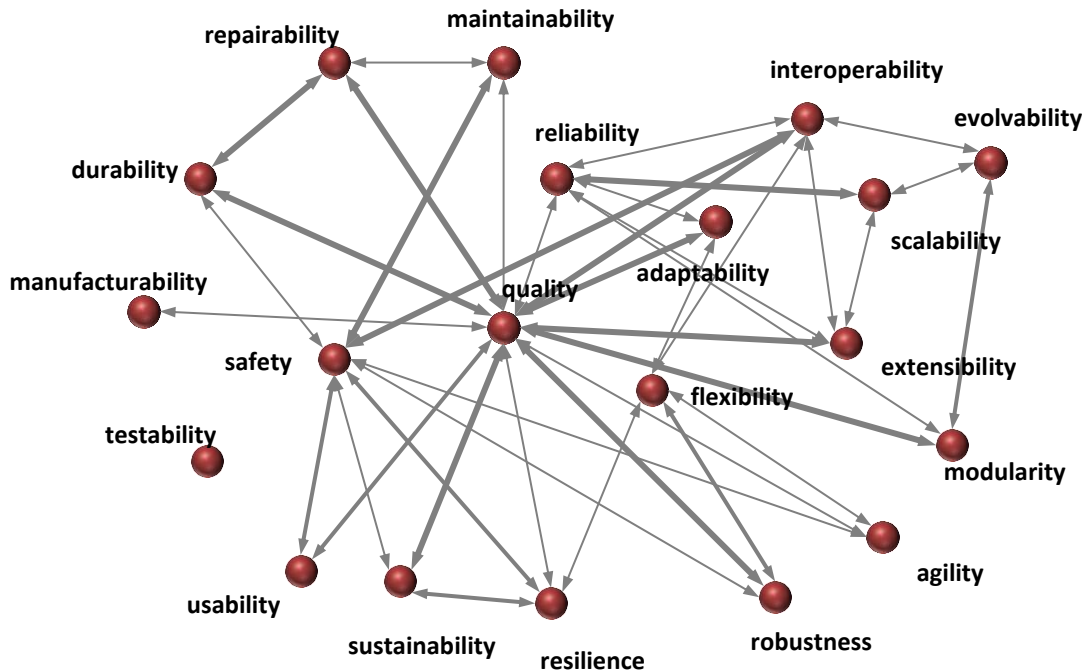


Figure 25. Interdépendances entre les « -ilities » [DEW 12]

Relations entre exigences non fonctionnelles

Ce premier point consiste à définir l'ensemble des relations entre une exigence sur laquelle porte l'étude avec les autres exigences. Sans cette connaissance, l'étude de l'impact est biaisée puisqu'il manque alors une relation qui n'est pas prise en compte dans l'étude de l'impact. Le graphe présenté précédemment est actuellement incomplet en termes d'exploration de relations entre des exigences (Figure 26). Des relations ne sont pas encore établies soit parce que le lien n'est pas encore formellement identifié ou encore parce que l'état des recherches sur une exigence est encore trop récent. C'est le cas notamment de l'interopérabilité dont on peut supposer qu'elle est liée à d'autres exigences mais sans que ce lien soit encore découvert et étudié du fait de la nouveauté du domaine [DEW 12]. A titre d'exemple, on peut supposer que l'interopérabilité est liée à la résilience car elle peut, dans un système collaboratif, participer à sa « capacité à absorber les perturbations et à se réorganiser tout en subissant des changements afin de conserver encore essentiellement la même fonction, structure, identité » [WAL 04]. De la même manière, les premiers travaux sur l'impact de l'interopérabilité tendent à montrer ce lien dans le cas de perturbations sur les systèmes sans que celui-ci ne soit encore recensé. Dans la même veine, certains travaux établissent le lien entre agilité et interopérabilité [BEN 12] sans que ce lien soit visible.

Ainsi les travaux naissants sur la résilience s'attachent par exemple à démontrer l'existence des liens entre elle-même et les exigences dans son environnement et ce, dans le but de son évaluation. C'est le cas par exemple de l'interopérabilité ou de la robustesse qui sont liées au cycle de résilience mais dont le lien n'est pas encore clairement identifié.

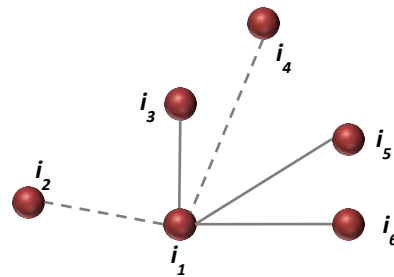


Figure 26. Exploration des relations entre les « -ilites »

Direction de la relation

La direction de la relation permet de connaître l'existence de l'influence (ou non) d'une exigence sur l'autre. On distingue alors deux types de relations. La dépendance unique et l'interdépendance (Figure 27). La dépendance unique représente une relation unidirectionnelle entre deux exigences. L'interdépendance montre que deux exigences s'influencent mutuellement et que la variation de l'une entraîne une variation de l'autre. La connaissance de la direction de la relation permet également, lorsqu'on étudie une exigence, de connaître l'ensemble des « entrées » qui vont l'impacter, *i.e.*, les « -ilites » qui influent sur elle, mais également de connaître l'ensemble des « sorties », *i.e.*, l'ensemble des « -ilites » qui sont impactées à leur tour.

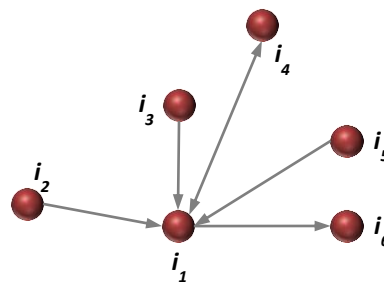


Figure 27. Dépendance et interdépendance entre les « -ilites »

Intensité de la relation

Ce point concerne la définition de la force du lien entre les « -ilites » et permet de définir plus précisément la notion d'influence. Cette intensité permet de connaître « l'importance » d'une « -ility » sur l'autre dans sa mise en œuvre et sa réalisation (Figure 28). De plus, cette intensité peut-être variable au cours du cycle de vie de l'exigence à l'étude. A titre d'exemple, au cours du cycle de vie de la résilience, certaines exigences impactent celle-ci dans certaines phases de son cycle de vie avec plus ou moins d'intensité. C'est le cas par exemple pour la robustesse qui intervient dans les phases de prévention et de mitigation mais pas nécessairement (ou de manière moindre) dans la phase de rétablissement. Il est donc indispensable, au-delà de la connaissance de la relation et de sa direction, d'identifier l'intensité de la relation entre des exigences mais également de préciser l'intensité de ce lien selon le cycle de vie des « -ilites ».

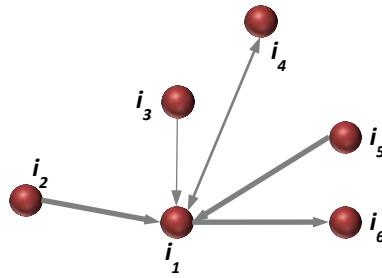


Figure 28. Intensité des relations entre les « -ilities »

Degré d'impact

Cet aspect tient compte des relations qui peuvent exister entre des « -ilities » dans l'environnement de celle à l'étude. En d'autres termes, et de la même manière que pour l'exploration des relations entre les « -ilities », il s'agit ici d'identifier les chaînes fermées (ex. i_1 - i_4 - i_3 , Figure 29) c'est-à-dire les cas où deux « -ilities » sont en relation avec celle étudiée tout en s'impactant elles-mêmes mutuellement. Cette aspect doit être pris en compte pour aboutir à une analyse précise puisque la variation d'une « -ility » (ex. i_4) peut impacter celle dont on souhaite l'amélioration (ex. i_1) mais peut également entraîner la variation d'une « -ility » avec laquelle elle est liée (ex. i_3). Dans le cas d'une boucle fermée, cette dernière variation reviendrait à impacter l'« -ility » étudiée (ex. i_1). Cette analyse doit permettre de s'assurer que toute variation, sur les exigences, aboutit bien aux attentes initiales. Il faut noter également que l'étude du degré d'impact doit être considérée pour les deux types de relations présentées précédemment (dépendance unique et interdépendance).

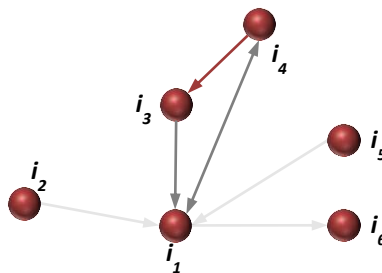


Figure 29. Degré d'impact entre les « -ilities »

Notion d'effets

La définition des effets s'appuie sur les aspects présentés précédemment (relation, direction, intensité, et degré) et qui ne peut être réalisée sans ces derniers. Il s'agit ici de caractériser l'effet résultant des relations entre les exigences et dépendant de leur orientation, intensité et degré. Cette définition des effets est réalisée en accord avec les études sur les notions d'effets et introduites dans les travaux de [BIL 15] à savoir en termes de nature (efficace, inefficace, absent, excessif et néfaste [MAN 02]) et de phénomènes (effet boule de neige, effet cascade). Il s'agit alors de donner la valeur d'un effet direct (entre deux « -ilities » $i_2 \rightarrow i_1$) et indirect (résultant de l'effet direct $i_2 \rightarrow i_1 \rightarrow i_6$) s'il existe (Figure 30). De la même manière ces effets devront être agrégés pour obtenir une valeur permettant de constater la satisfaction (ou non) de l'« -ility » impactée.

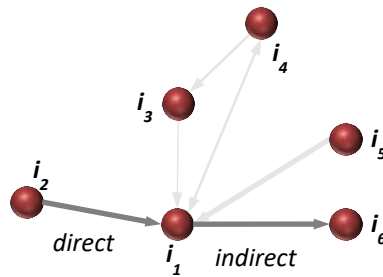


Figure 30. Effet direct et indirect entre des « -ilities »

Cette notion d'effets est primordiale puisque c'est elle, au final, qui renseigne sur l'impact des exigences les unes sur les autres et qui concrétise la caractérisation des composantes de relation, intensité et degré d'impact par la détermination d'une fonction avec pour résultat la nature et le phénomène de l'effet.

L'étude des exigences non fonctionnelles de type « - ilities » comme un écosystème est actuellement en phase d'exploration au travers d'un projet et d'une thèse appliquée à une exigence particulière : la résilience. Cette étude est complexe de par (1) les différentes interactions existantes entre les « - ilities », (2) les interactions qui nécessitent d'être explorées ou encore, (3) l'apparition de nouvelles « - ilities » au sein de cet écosystème. Cette étude est d'autant plus complexe que l'on se place dans le contexte de l'ingénierie des systèmes collaboratifs avec pour chacun des systèmes participants, ses priorités au regard de ses exigences.

Cette première recherche doit aboutir à deux travaux de recherche supplémentaires. Le premier concerne la généralisation de ces primo travaux à l'écosystème complet et toujours selon les caractéristiques précédentes. Il s'agit d'obtenir une méthode qui soit applicable à chaque « - ility » quel que soit ses particularités (définition, évaluation, cycle de vie). Le second concerne le point particulier des impacts et effets. Ils se limitent pour l'instant à l'étude des impacts et effets de premier niveau ($i_2 \rightarrow i_1 \rightarrow i_6$). Là encore, l'objectif est d'étendre ces notions d'impacts et d'effets à l'ensemble des « - ilities » en relation dans l'écosystème.

Bibliographie

A

- [ALO 07] Aloui S.: Contribution à la modélisation et l'analyse du risque dans une organisation de santé au moyen d'une approche système, thèse de doctorat, Ecole des Mines de Paris, 2007.
- [ATH 05] ATHENA IP: Framework for the Establishment and Management Methodology, ATHENA Integrated Project, ATHENA Deliverable DA1.4, 2005.
- [ATH 07] ATHENA IP: Guidelines and best practices for applying the ATHENA interoperability framework to support SME participation in digital ecosystems, ATHENA Integrated Project, ATHENA deliverable A8.2, 2007.

B

- [BEH 04] Behrmann, G., David A., Larsen K. G.: A tutorial on Uppaal. Eds. Bernardo & Corradini, Formal Methods for the Design of Real-Time Systems, pp. 200-236, 2004.
- [BEN 08] Bénaben F., Hanachi C., Lauras M., Couget P., Chapurlat V: A metamodel and its ontology to guide crisis characterization and its collaborative management, Proc. 5th International ISCRAM Conference, 2008.
- [BEN 12] Bénaben F.: Conception de Système d'Information de Médiation pour la prise en charge de l'Interopérabilité dans les Collaborations d'Organisations, Habilitation à Diriger les Recherches, Institut National Polytechnique de Toulouse, October 2012.
- [BIL 15] Billaud S.: Contribution à l'ingénierie de système de systèmes : modélisation multi points de vue et analyse de l'impact de l'exigence d'interopérabilité, thèse de doctorat, Université Montpellier II, 2015.
- [BLA 11] Blanchard B.S., Fabricky W.J.: Systems Engineering and analysis, 5th Edition, Pub. Prentice Hall International Series in industrial and systems engineering, 800 pages, 2011.
- [BOA 06] Boardman J., Sauser B.: System of Systems – the meaning of, Proc. IEEE/SMC International Conference on Systems of System Engineering, 2006.
- [BOT 16] Botelho Espadinha Da Cruz P. E.: Business Interoperability: a methodology to analyse and re-design interoperable buyer-supplier dyads, PhD Thesis, Université de Lisbonne, 2016.
- [BOU 15] Bourdil P. A.: Contribution à la modélisation et à la vérification formelle par model-checking – Symétries pour les Réseaux de Petri Temporels, thèse de doctorat, Université de Toulouse, 2015.
- [BOU 07] Bourey J.P., Grangel R., Doumeingts G., Berre A. J.: Report on Model Driven Interoperability, INTEROP Network of Excellence IS 508011, deliverable DTG 2.3, 2007.

C

- [CAM 05] Camarinha-Matos L., Afsarmanesh H.: Collaborative networks: a new scientific discipline. Journal of Intelligent Manufacturing, vol. 16, issue 4, pp. 439-452, October 2005.

- [CAM 09] Camarinha-Matos L., Afsarmanesh H., Galaneo N., Molina A.: Collaborative networked organizations – Concepts in manufacturing enterprises, *Computers & Industrial Engineering*, vol. 57, issue 1, pp. 46-60, August 2009.
- [CHAL 16] Chalmeta R., Pazos V.: A step-by-step methodology for enterprise interoperability projects, *Enterprise Information Systems*, vol. 9, issue 4, pp. 436-464, 2016.
- [CHAP 07] Chapurlat V.: Vérification et validation de modèles de systèmes complexes: application à la Modélisation d'Entreprise, Habilitation à Diriger les Recherches, Université Montpellier II, 2007.
- [CHA 13] Chapurlat V.: UPSL-SE: A model verification framework for Systems Engineering, *Computers in Industry*, vol. 64, issue 5, pp. 581-597, 2013.
- [CHE 06] Chen D., Daclin N.: Framework for Enterprise Interoperability, Proc. 2nd IFAC Workshop on Enterprise Integration, Interoperability and Networking, 2006.
- [CHE 08] Chen D., Doumeingts G., Vernadat F. Architectures for enterprise integration and interoperability: Past, present and future. *Computers in industry*, vol. 59, issue 7, pp. 647-659, 2008.
- [C4I 98] C4ISR: Levels of Information Systems Interoperability (LISI), USA Department of Defense, 1998.

D

- [DAC 12] Daclin N.: Embedding interoperability in System of Systems: definition and characterization of fundamental requirements, Proc. 13th IFIP Working Conference on Virtual Enterprises (PRO-VE'12), 2012.
- [DAC 14] Daclin N., Mallek S.: Capturing and structuring interoperability requirements: a framework for interoperability requirements. *Interoperability for Enterprise Software and Applications (I-ESA'14)*, 2014.
- [DAC 15] Daclin N., S. Mallek S.: Towards a sustainable implementation of interoperability solutions: bridging the gap between interoperability requirements and solutions, Proc. 6th International IFIP Working Conference on Enterprise Interoperability, 2015.
- [DAC 16] Daclin N., Chen D., Vallespir B.: Developing enterprise collaboration: a methodology to implement and improve interoperability, *Enterprise Information Systems*, vol. 10, issue 5, pp. 467-504, 2016.
- [DAC 16] Daclin N., Mallek-Daclin S., Chapurlat V., Vallespir B.: Writing and verifying interoperability requirements: Application to collaborative processes, *Computers in Industry*, vol. 82, pp. 1-18, October 2016.
- [DAH 08] Dahmann J., Lane J.A., Rebovich G. Jr., Baldwin K. J.: A Model of Systems Engineering in a System of Systems Context, Proc. Conference on Systems Engineering Research, 2008.
- [DEL 04] DeLaurentis D., Callaway R.K.: A System-of-Systems Perspective for Public Policy Decisions. *Review of Policy Research*, vol. 21, issue 6, pp. 829–837, November 2004.
- [DES 10] Desai, J.: *Service Level Agreements – A Legal and Practical Guide*, Pub. IT Governance Publishing, 120 pages, 2010.

- [DEW 12] De Weck O., Roos D., Magee C.: Life-Cycle Properties of Engineering Systems: The Illities, Engineering Systems: Meeting Human Needs in a Complex Technological World, chapter 4, Ed. The MIT Press, 232 pages, 2012.
- [DOD 01a] DoD: Department of Defense Dictionary of Military and Associated Terms, US Department of Defense, 2001.
- [DOD 01b] DoD: Systems Engineering Fundamentals, US Department of Defense, 2001.
- [DUC 99] Ducq Y.: Contribution à une méthodologie d'analyse de la cohérence des systèmes de production dans le cadre du modèle GRAI, thèse de doctorat, Université Bordeaux I, 1999.

E

- [EC 08] European Commission: European Interoperability Framework for European Public Services (EIF). Version 2.0 draft document as basis for EIF 2.0, 2008.
- [ELM 11] Elmir B., Bounabat B.: A Novel Approach for Periodic Assessment of Business Process Interoperability, International Journal of Computer Science Issues, vol. 8, issue 4, pp. 298-306, July 2011.

F

- [FAN 11] Fanmuy G., Llorens J., Fraga A.: Requirements verification in the industry, Proc. 2nd International Conference on Complex Systems Design & Management – CSDM, 2011.
- [FAV 12] Favaro J., De Koning H.S., Schreiner R., Olive X.: Next generation requirements engineering, Proc. 22th Annual INCOSE International Symposium, 2012.
- [FEW 03] Fewell S., Clark T.: Organisational Interoperability: Evaluation and Further Development of the OIM Mode, Proc. 8th International Command and Control Research and Technology Symposium, 2003.
- [FOR 08] Ford C.T.: Interoperability measurement, PhD Thesis, Department of the Air Force Air University, Air Force Institute of Technology, 2008.

G

- [GEN 14] Genest D.: Cogitant Reference Manual, version 5.3.0, 2014.
- [GOR 08] Gorod A., Sauser B., Boardman J.: System-of Systems Engineering Management: A Review of modern History and a Path Forward, IEEE Systems Journal, vol. 2, issue 4, pp. 484-499, 2008.
- [GRU 05] Gruhn V., Laue R.: Using Timed Model Checking for Verifying Workflows, Proc. Computer Supported Activity Coordination, pp.75-88, 2005.
- [GUE 12] Guédria, W., Y. Naudet, and D. Chen.: Maturity Model for Enterprise Interoperability, Enterprise Information Systems, vol. 9, issue 1, pp. 1-28, 2015.

H

- [HOL 95] Holland J.H.: Hidden order: How Adaptation Builds Complexity, Eds. Basic Books, 185 pages, 1995.
- [HU 15] Hu L., Yongliang T., Yuan G., Jinpeng B., Jiangan Z.: System of systems oriented vehicle conceptual design: Perspectives and progresses, Chinese Journal of Aeronautics, vol. 28, issue 3, pp. 617-635, 2015.

I

- [IEE 90] IEEE: IEEE standard computer dictionary: a compilation of IEEE standard computer glossaries, New York: Institute of Electrical and Electronics Engineers, 1990.
- [INC 11] INCOSE: Systems Engineering Handbook—A guide for System life cycle processes and activities, INCOSE Systems Engineering Handbook v.3.2.2 INCOSE TP 2003 002 03.2.2, 2011.
- [INC 12] INCOSE: Guide for writing requirements, v1.0, Requirements WG-INCOSE, INCOSE-TP-2010-006-01, 2012.
- [ISO 94] ISO: Quality management and quality assurance, ISO 8402, Vocabulary, Second edition 1994-04-01, 1994.
- [ISO 99] ISO: Concepts and Rules for Enterprise Models, ISO 14258– Industrial Automation Systems, ISO TC 184/SC 5/WG1, 1999.
- [ISO 02] ISO: ISO 16100, 1 - Industrial automation systems and integration – Manufacturing software capability profiling for interoperability – Part 1: Framework”, ISO/TC 184/SC 5/WG 4, 2002.
- [ISO 03] ISO: Systems and software engineering - System life cycle processes. SO/IEC/IEEE 15288:2015, ISO/IEC JTC 1/SC 7 Software and systems engineering, May 2015.
- [ISO 09] ISO: Advanced automation technologies and their applications. Part 1: Framework for enterprise interoperability, ISO/DIS 11354-1, ISO/TC 184/SC 5/WG, 2009.
- [ISO 11] ISO: Systems and software engineering – Life cycle processes – Requirements engineering, ISO/IEC/IEEE 29148:2011, 2011.
- [ITU 12] ITU: Series Z: Languages And General Software Aspects For Telecommunication Systems - Formal description techniques (FDT)—User Requirements Notation (URN), Language definition, ITU – T Z.151 – Telecommunication Standardization Sector Of ITU, 2012.

J

- [JAM 08] Jamshidi M.: Systems of Systems Engineering: Principles and Applications, Ed. CRC Press, 480 pages, 2008.
- [JAM 09] Jamshidi M.: System of Systems Engineering: Innovations for the Twenty-First Century, Pub. Wiley, 616 pages, 2009.

K

- [KAM 04] Kamsu Foguem B.: Modélisation et vérification des propriétés de systèmes complexes : Application aux processus d'entreprise, thèse de doctorat, Université Montpellier II, 2004.
- [KAN 14] Kang C.S., Pee E.Y., Sim K.W., Pang C.K.: Framework For Managing System-Of-Systems Ilities, DSTA Horizons, pp. 56-65, 2013/2014.
- [KAS 04] Kasunic M., Anderson W.: Measuring Systems Interoperability: Challenges and Opportunities, technical note, CMU/SEI-2004-TN-003, 2004.
- [KEN 02] Kent S.: Model Driven Engineering, Proc. International Conference on Integrated Formal Methods, pp. 286-298, 2002.
- [KOT 97] Kotov V.: Systems of Systems as Communicating Structures, Hewlett Packard Computer Systems Laboratory, HPL-97-124, October 1997.

L

- [LEM 06] Le Moigne J.L.: La théorie du système générale – théorie de la modélisation, Les Classiques du Réseau Intelligence de la Complexité, édition de 1994, nouvelle présentation 2006, 339 pages, 2006.

M

- [MAI 98] Maier M.W.: Architecting Principles for System-of-Systems. Systems Engineering, vol. 1, issue 4, pp. 267–284, 1998.
- [MAL 11] Mallek S.: Contribution au développement de l'interopérabilité en entreprise : vers une approche anticipative de détection de problèmes d'interopérabilité dans des processus collaboratifs, thèse de doctorat, Université Montpellier II, 2011.
- [MAL 12] Mallek S., Daclin N., Chapurlat V.: The application of interoperability requirement specification and verification to collaborative processes in industry, Computers in Industry, vol. 63, issue 7, pp 643-658, 2012.
- [MAL 13] Mallek S., Daclin N., Chapurlat V., Vallespir B.: Requirements formalization for systems engineering: an approach for interoperability analysis in collaborative process model, Proc. Workshop on selected applications and methods for enterprise interoperability, co-located with IWEI 2013, 2013.
- [MAL 15] Mallek S., Daclin N., Chapurlat V., Vallespir B.: Enabling model checking for collaborative process analysis: from BPMN to Network of Timed Automata, Enterprise Information Systems, vol. 9 - issue 3, pp 279-299, 2015.
- [MAN 02] Mann D.: Hands on systematic innovation, CREAX Press, 2002.
- [MEI 12] Meinadier J.-P.: Découvrir et comprendre l'ingénierie système, Ed. Cepadues, 392 pages, 2012.
- [MET 05] Metzger A: A Systematic Look at Model Transformations, Eds. Beydeda, Book & Grhn, Model- Driven Software Development – Par I, pp. 19-33, 2005.

[MOR 04] Morris E., Levine L., Meyers C., Place P., Plakosh D.: System of Systems Interoperability (SOSI): Final Report, Technical Report, CMU/SEI-2004-TR-004 ESC-TR-2004-004, 2004.

N

[NEH 07] NEHTA: Towards a Health Interop Framework, version 2.0, 2007.

O

[OMG 08] OMG: Semantics of Business Vocabulary and Business Rules (SBVR)—version 1.0, Open Management Group, 2008.

[OMG 11] OMG: Business Process Model and Notation (BPMN), Open Management Group, version 2.0, 2011.

[OMG 15] OMG: OMG Systems modelling Language (OMG SysML™), Open Management Group, version 1.4, 2015.

[OWE 96] Owens A. W.: The Emerging U.S. Systems of Systems, Eds. National Defense University, 4 pages, 1996.

P

[PAU 93] Paulk M. C., Curtis B., Chrissis M. B., Weber C. V.: Capability Maturity Model for Software, Version 1.1", CMU/SEI-93-TR-24, ESC-TR-93-177, Software Engineering Institute, 1993.

[PIN 09] Pingaud H. : Rationalité du développement de l'interopérabilité dans les organisations., Proc. 8ème congrès international de Génie Industriel, 2009.

R

[REZ 14] Rezaei R., Chiew T. K., Lee S. P., Aliee Z., S.: Interoperability evaluation models: A systematic review, Computer in Industry, vol. 65, issue 1, pp. 1-23, 2014.

[RIC 14] Ricci N., Fitzgerald E. M., Ross A. M., Rodhes D. H.: Architecting Systems of Systems with Ilities: an Overview of the SAI Method, Proc. Conference on Systems Engineering Research (CSER 14), Procedia Computer Science, vol. 28, pp. 322-321, 2014.

[ROD 89] Rodde G.: Les systèmes de production : modélisation et performances, Eds. HERMES, 330 pages, 1989.

[ROQ 09] Roque M., Chapurlat V.: Interoperability in collaborative processes: requirements characterization and proof approach, Proc. 10th IFIP Working Conference on Virtual Enterprises (PRO-VE'09), 2009.

[ROS 15] Ross A. M., Rodhes D. H.: Towards a Prescriptive Semantic Basis for Change-type Ilities, Proc. Conference on Systems Engineering Research (CSER 15), Procedia Computer Science, vol. 44, pp. 443-453, 2015.

S

[SCH 02] Schnoebelen Ph., « The Complexity of Temporal Logic Model Checking ». Advances in Modal Logic, vol. 4, pp. 1-44, 2002.

- [SCH 10] Schwitter R.: Controlled natural languages for knowledge representation, Proc. 23rd International Conference on Computational Linguistics: Posters, 2010.
- [SHE 95] Shenhar A.: A new systems engineering taxonomy, Proc. 4th INCOSE International Symposium, 1995.
- [SIM 81] Simon H.A.: The Sciences of the Artificial, 3rd Edition, Ed. The MIT press, 221 pages, 1996.
- [SOW 84] Sowa J.F.: Conceptual structures: information processing in mind and machine, Pub. Wesley Longman Publishing, 481 pages, 1984.
- [STA 16] The Standish Group: The Chaos Report, The Standish Group International, Inc, 1995, 2016.
- [SUC 08] Scucanec S.J., Van Gaasbeek J.R.: A day in the life of a verification requirement, U.S Air Force T&E Days, 2008.

T

- [TOG 11] TOGAF: The Open Group Architecture Framework: TOGAF version 9, chapter 29 Interoperability requirements, 2011.
- [TOL 07] Tolk A., Diallo S. Y., Turnitsa C. D.: Applying the Levels of Conceptual Interoperability Model in Support of Integrability, Interoperability, and Composability for System-Of-Systems engineering, Systemics Cybernetics and Informatics vol. 5, pp. 65-74, 2007.
- [TRI 97] Trinquet Y.: Un aperçu sur quelques exécutifs temps réel du marché, Ecole d'été Temps Réel'97 Applications, Réseaux et Systèmes, 1997.

V

- [VAL 03] Vallespir B. : Modélisation d'entreprise et architecture de conduite des systèmes de production, Habilitation à Diriger les Recherches, Université Bordeaux 1, 2003.
- [VAL 07] Valerdi R., Ross A.M., Rhodes D.H.: A Framework for Evolving System of Systems Engineering. Crosstalk, vol. 20 pp. 28–30, October 2007.
- [VAN 00] Van Deursen A., Klint P., Visser J.: Domain-Specific Languages: An Annotated Bibliography, ACM SIGPLAN NOTICES, vol. 35, pp. 26-36, 2000.
- [VER 96] Vernadat F.B.: Enterprise Integration Modeling and Integration : Principles and Applications, Eds. Chapman & Hall, 513 pages, 1996.
- [VLA 91] Van Lamsweerde A., Dardenne A., Delcourt B., Dubisy F.: The KAOS project, Knowledge Acquisition in Automated Specification of Software, AAAI Spring Symposium Series, American Association for Artificial Intelligence, 1991.

W

- [WAL 04] Walker B., Holling C.S., Carpenter S. R., Kinzig A.: Resilience, adaptability and transformability in social–ecological systems, Ecology and Society vol.9, issue 2, 2004.
- [WEG 96] Wegner P.: Interoperability, ACM Computing Survey, vol. 28, issue 1, pp. 285–287, 1996.

[WYN 09] Wyner A., Angelov K., Barzdins G., Damljanovic D., Davis B., Fuchs N., Hoefler S., Jones K., Kaljurand K., Kuhn T., Luts M., Pool J., Rosner M., Schwitter R., Sowa J.: On Controlled Natural languages: Properties and Prospects, Workshop on Controlled Natural Language CNL2009, 2009.

Y

[YAH 12] Yahia E., Aubry A., Panetto H.: Formal measures for semantic interoperability assessment in cooperative enterprise information systems, Computers in Industry, vol. 63, issue 5, pp.443-457, 2012.

Annexes

1. Détails des collaborations dans le cadre de projets

Projet	RE(H)STRAIN - Résilience du réseau ferré grande vitesse franco-allemand	
Type/durée	ANR - 24 mois	
Partenaires	Académiques <ul style="list-style-type: none"> • Université des Forces Armées de Munich (UniBW) • Laboratoire Central de la Préfecture de Police de Paris (LCP) • Université Bonn-Rhein-Sieg (HBRS) • Université de Cologne (THK) • Institut Fédéral Scientifique et Technique (BAM) • Institut français des sciences et technologies des transports, de l'aménagement et des réseaux (IFSTTAR) 	Industriels <ul style="list-style-type: none"> • Efectis • HBI
Sujet	Le projet RE(H)STRAIN s'intéresse à la sécurité préventive et son impact sur la résilience du réseau ferroviaire grande vitesse Franco-Allemand dans le cas d'attaques terroristes.	

Projet	HOPICAMP - Hôpitaux de Campagne Nouvelle Génération	
Type/durée	FUI - 36 mois	
Partenaires	Académiques <ul style="list-style-type: none"> • Université de Nîmes • SDIS 30 	Industriels <ul style="list-style-type: none"> • UTILIS • BEWEIS • H4D • Crisis Simulation Engineering
Sujet	Le projet HOPICAMP traite de l'amélioration de la rapidité du déploiement des hôpitaux de campagne, mais aussi la qualité des soins prodigués ainsi que la sécurité sanitaire.	

Projet	CARIONER II - Caractérisation, analyse et amélioration de l'interopérabilité des organisations impliquées dans des processus collaboratifs	
Type/durée	CARNOT-MINES - 12 mois	
Partenaires	Académiques <ul style="list-style-type: none"> • Université de Bordeaux 	Industriels --
Sujet	Le projet CARIONER II s'est intéressé à la spécification d'exigences d'interopérabilité à l'aide du langage SBVR (Semantics of Business Vocabulary and Rules) et la vérification de ces exigences sur des modèles de processus collaboratifs	

Projet	ISYCRI - Interopérabilité des Systèmes en Situation de Crise	
Type/durée	ANR - 36 mois	
Partenaires	Académiques <ul style="list-style-type: none"> • Ecole des mines d'Albi-Carmaux • IRIT 	Industriels <ul style="list-style-type: none"> • Thalès • EBM Websourcing
Sujet	Le projet ISYCRI s'est intéressé au développement de l'interopérabilité pour améliorer la collaboration entre des organisations impliquées dans la résolution d'une crise.	

Projet	COREGI - Conception de la Résilience des Systèmes en Génie Industriel	
Type/durée	ANR – 36 mois	
Partenaires	Académiques <ul style="list-style-type: none"> • Ecole des mines de Paris 	Industriels <ul style="list-style-type: none"> • GRT-GAZ • SNCF
Sujet	Le projet COREGI s'est intéressé à l'aide à la conception, à l'amélioration et au pilotage de la résilience d'organisations industrielles.	

Projet	CARIONER - Caractérisation, analyse et amélioration de l'interopérabilité des organisations impliquées dans des processus collaboratifs	
Type/durée	Carnot-Mines (12 mois)	
Partenaires	Académiques --	Industriels --
Sujet	Le projet CARIONER s'est intéressé au développement des graphes conceptuels pour la vérification d'exigence d'interopérabilité dans les processus collaboratif.	

Projet	INTEROP – Réseau d'Excellence	
Type/durée	Européen - 36 mois	
Partenaires	Académiques	Industriels >30 partenaires
Sujet	Le réseau d'excellence INTEROP a étudié l'interopérabilité des entreprises selon trois aspects : l'ontologie, la modélisation d'entreprise et les architectures. Il a également structuré la recherche Européenne sur l'interopérabilité.	

Projet	ATHENA IP - Advanced Technologies for Interoperability of Heterogeneous Enterprise Networks and their Application	
Type/durée	Européen - 36 mois	
Partenaires	Académiques	Industriels 25 partenaires
Sujet	Le projet ATHENA s'est intéressé au développement de l'interopérabilité des entreprises et au transfert des résultats aux secteurs industriels.	

2. Détails des travaux de thèse et post doctoraux encadrés

Thèse	S. Mallek	
Titre	Contribution au développement de l'interopérabilité en entreprise : vers une approche de détection de problèmes d'interopérabilité dans des processus collaboratifs	
Soutenance/Jury	14 octobre 2011	
	Didier Crestani, Pr., Université Montpellier 2	Président
	Jean-Pierre Bourey, Pr., Ecole Centrale de Lille	Rapporteur
	David Chen, Pr., Université Bordeaux 1	Rapporteur
	Michaël Petit, Pr., Facultés FUNDP	Examineur
	Christian Braesch, MdC, Université de Savoie	Examineur
	Vincent Chapurlat, Pr., Ecole des Mines d'Alès	Encadrant
	Nicolas Daclin, IR, Ecole des Mines d'Alès	Co-encadrant
Résumé	Ces travaux abordent la problématique de détection de défauts d'interopérabilité dans des processus collaboratifs via l'utilisation des techniques de vérification formelle et notamment les graphes conceptuels et le <i>model checking</i> . Ils se sont donc intéressés d'une part à la définition des besoins en interopérabilité devant être pris en compte dans un contexte collaboratif. D'autre part, ils ont traité de la formalisation des besoins en un ensemble d'exigences d'interopérabilité, de manière aussi formelle que possible dans un objectif de vérification formelle.	
Situation actuelle	Ingénieur de Recherche & Développement, Ecole des mines d'Alès, Alès	
Thèse	S. Billaud	
Titre	Contribution à l'Ingénierie de Système de Systèmes: modélisation multi points de vue et analyse de l'impact de l'exigence d'interopérabilité	
Soutenance/Jury	17 novembre 2015	
	Frédéric BENABEN, MA HdR, Ecole des Mines d'Albi	Rapporteur
	Bernard ARCHIMEDE, Professeur ENI Tarbes	Rapporteur
	Jean-Philippe AUZELLE, DRGI, U. Lorraine	Examineur
	Didier CRESTANI, Professeur, U. Montpellier	Président
	Jean-Paul PIGNON, Ing. Docteur, THALES C&S, Paris	Invité
	Nicolas DACLIN, MA, Ecole des Mines d'Alès	Encadrant
	Vincent CHAPURLAT, Pr., Ecole des Mines d'Alès	Co-encadrant
Résumé	Cette thèse a formalisé et développé une méthode pour accompagner la modélisation, la vérification de modèles et l'analyse de l'interopérabilité dans un SdS. Cette méthode repose sur 1) un ensemble de concepts et de relations entre ces concepts pour décrire un SdS et la propriété d'interopérabilité, 2) des langages spécifiques de modélisation (DSML) pour manipuler ces concepts et relations et donc créer des « modèles » de SdS, 3) d'un processus opératoire et 4) d'outils de modélisation, de vérification des modèles, de simulation du comportement et d'évaluation de l'interopérabilité et de son influence sur la performance, la stabilité et l'intégrité du SdS en cours de fonctionnement	
Situation actuelle	Project Manager - VP&White, Paris	

Thèse	B. Moradi
Titre	Formalisation et évaluation de propriétés non-fonctionnelles pour l'ingénierie de systèmes de systèmes : application à la résilience
Soutenance/Jury	En-cours
Résumé	<p>La problématique de ces travaux de recherche concerne l'étude et l'évaluation de propriétés non-fonctionnelles des systèmes de systèmes et, plus particulièrement, de la propriété de résilience.</p> <p>Il s'agit de définir, spécifier et développer les méthodes, concepts et moyens nécessaires, <i>in fine</i>, à l'évaluation d'une propriété non-fonctionnelle, et d'appliquer ces outils à l'évaluation de la résilience à un niveau global.</p>
Situation actuelle	--

Post-doc	S. Mallek
Projet	CARIONER II
Résumé	<p>Ces travaux ont fait suite à ceux développés dans la thèse de S. Mallek. Ils ont porté sur la spécification d'exigences d'interopérabilité à l'aide du langage SBVR (Semantics of Business Vocabulary and Rules) et la vérification de ces exigences - sur des modèles de processus collaboratifs - à l'aide de techniques de vérification formelle telles que les graphes conceptuels et le model checking. Une application a été développée avec l'IDE Eclipse.</p>

Post-doc	N. Amokrane
Projet	RE(H)STRAIN
Résumé	<p>Ces travaux abordent la problématique de la résilience des infrastructures critiques (réseau ferré grande vitesse franco-allemand). Ils portent sur le développement de mesure de sécurité pour la résilience organisationnelle en termes de prévention, de mitigation et de rétablissement. La définition de ces mesures s'appuie sur l'exploration de scénarios possibles sur le système et l'analyse des solutions existantes actuellement.</p>

3. Détails des enseignements et encadrements

Années	Enseignements	Public	CM	TD	TP
2016-2017	Ingénierie des besoins et des exigences	M1 EMA	4		8
	Modélisation processus	L3 EMA	4	8	
	Ingénierie des besoins et des exigences	Mécatronique	4	7	9
	Modélisation d'entreprise	M1 EMA	8	3	4
	Ingénierie Système	L3 EMA		8	9
	Ingénierie des besoins et des exigences	M1 UPB	4		
2015-2016	Ingénierie des besoins et des exigences	M1 EMA	4		8
	Modélisation processus	L3 EMA	4	8	
	Ingénierie des besoins et des exigences	Mécatronique	4	7	9
	Modélisation d'entreprise	M1 EMA	8	3	4
	Ingénierie des besoins et des exigences	M1 UPB	4		
2014-2015	Ingénierie des besoins et des exigences	M1 EMA	3		9
	Modélisation processus	L3 EMA	4	8	
	Ingénierie Système	L3 EMA		12	9
	Modélisation d'entreprise	M1 EMA	7	3,5	4
2013-2014	Ingénierie des besoins et des exigences	M1 EMA ²³²⁴	3	8	9
	Modélisation processus	L3 EMA	4	12	
	Ingénierie Système	L3 EMA	12	3	
	Modélisation d'entreprise	M1 EMA	9		4
	Ingénierie des besoins et des exigences	M1 UPB ²⁵	2		
2012-2013	Modélisation processus	L3 EMA	4	8	
	Ingénierie Système	L3 EMA	12	12	
	Modélisation d'entreprise	M1 EMA	9	6	7
	Ingénierie des besoins et des exigences	M1 EMA	3		9
	Ingénierie des besoins et des exigences	M1 UPB	2		
2011-2012	Productique	2A EMA	8	16	
	Ingénierie Système	2A EMA	8	8	
2010-2011	Productique	2A EMA	8	16	
	Ingénierie Système	2A EMA	8	8	
2009-2010	Productique	2A EMA	8	16	
	Ingénierie Système	2A EMA	8	8	
2006-2007	Bureautique	1A IUT ²⁶			32
	Programation C	1A IUT			62
½ ATER	Mécanique des fluides	2A IUT			52

²³ Ecole des Mines d'Alès

²⁴ Département Engineering and Management of Complex Systems (EMACS)

²⁵ Université Politehnica de Bucarest

²⁶ IUT Mesures Physique

2005-2006	Modélisation fonctionnelle	IUP GEII ²⁷ 2	8	8	
Moniteur	Gestion de projet	IUP GEII 2			18
	Technologie mécanique	Licence SPI ²⁸			4
	Gestion de projet	Licence SPI			
	Systèmes à événements discrets	IUP GEII 3			32
2004-2005	Techniques industrielles	IUP GEII 2		3	
Moniteur	Modélisation fonctionnelle	IUP GEII 2		8	
	Systèmes à événements discrets	IUP GEII 2	6.7		32
	Technologie mécanique	Licence SPI		10.7	
	Organisation et gestion de production	IUP GM ²⁹		4	
	Systèmes à événements discrets	Master 1 EEA ³⁰			32
2003-2004	Systèmes à événements discrets	IUP GEII 3			32
Moniteur	Systèmes à événements discrets	Licence SPI			6
	Systèmes à événements discrets	Master 1 EEA			54

Années	Encadrements	Public	Durée
2016-2017	Mission R&D (2 étudiants) <i>Développement d'un éditeur de « boilerplates » pour l'expression d'exigence</i>	M1 EMA	5 sem.
2016-2017	Mission R&D (3 étudiants) <i>Modélisation et Evaluation du processus de gestion du cycle de l'eau et des déchets dans les hôpitaux</i>	M1 EMA	5 sem.
2016-2019	Apprenti ingénieur – SADA Assurances	Mr. Tommy Kaindoh	en cours
2015-2016	PFE (contrat pro) – Air France Cargo <i>Mise en place d'un plan d'action d'amélioration de la performance, et fiabilisation d'un outil d'analyse de la non-Qualité</i>	M2 EMA	1 an
2015-2016	PFE (contrat pro) Air France Cargo <i>Implémentation d'un nouvel indicateur et analyse Qualité de service opérationnelle appliquées aux standards de l'industrie du fret aérien – Cargo iQ Management</i>	M2 EMA	1 an
2015-2016	PFE – SNCF <i>Définition et intégration de plans de travaux des nouvelles rames REGIO 2N</i>	M2 EMA	6 mois
2015-2018	Apprenti ingénieur - IBM	Mr. Samuel Aubertin	en cours
2014-2017	Apprenti ingénieur – Conseil Général de l'Hérault	Mr. Chauzal Maxime	en cours
2013-2016	Apprenti ingénieur – Airbus Helicopter	Mr Girard Quentin	terminé
2012-2015	Apprenti ingénieur – IBM	Mr Behr Simon	terminé

²⁷ Génie Electrique et Informatique Industrielle

²⁸ Sciences de la Production Industrielle

²⁹ Génie Mécanique

³⁰ Electronique, Electrotechnique et Automatique

2014-2015	PFE - AGCO SAS <i>Chef de projet Implantation Logistique - domaine Machinisme Agricole</i>	M2 EMA	6 mois
2014-2015	PFE – MERSEN <i>Projet zéro retouche</i>	M2 EMA	6 mois
2014-2015	PFE (contrat pro) – Air France Cargo <i>Analyste coordinateur plans d'actions QDS</i>	M2 EMA	1 an
2013-2014	PFE (contrat pro) - Air France Cargo <i>Mise en place de la démarche qualité Cargo 2000</i>	M2 EMA	1 an
2013-2014	PFE - DCNS <i>Amélioration de la performance industrielle sur des entités opérationnelles</i>	M2 EMA	6 mois
2013-2014	Mission (3 étudiants) - ESAT Pierre Laporte <i>Création de produits ou services innovants : Etude de marché, conception et réalisation d'un chariot de courses électrique</i>	L3 EMA	5 sem.
2013-2014	Mission (3 étudiants) - PICHETA <i>Conseil en organisation</i>	L3 EMA	5 sem.
2012-2013	PFE - Lowendalmasaï <i>Conseil Achats et Besoins en fonds de roulement</i>	M2 EMA	6 mois
2012-2013	Mission (3 étudiants) - SOGEA Sud <i>Création d'activité nouvelle : génie civil industriel</i>	L3 EMA	5 sem.
2012-2013	Mission (3 étudiants) – Atouts faire <i>Conseil en organisation</i>	L3 EMA	5 sem.
2011-2012	PFE - ERDF GRDF UCF PACA ouest <i>Amélioration de la performance opérationnelle</i>	4A EMA	7 mois
2011-2012	PFE – SPARTOO SAS <i>Optimisation du processus de mise en ligne des produits</i>	4A EMA	5 mois
2011-2012	Mission R&D (3 étudiants) <i>Transformation du langage LUSP vers le langage TCTL</i>	3A EMA	5 sem.
2011-2012	Mission (4 étudiants) - SERCI <i>Conseil en organisation</i>	2A EMA	5 sem.
2011-2012	Etude de cas (1 étudiant) - société 3G Santé <i>Modélisation des processus métiers dans le cadre du développement d'une plateforme de soins et de services pour la prise en charge de patients complexes</i>	4A EMA	5 sem.
2010-2013	Apprenti ingénieur – CN-Tricastin	Mr Baniol Thibaud	terminé
2010-2011	PFE - SCSi (Supply Chain Service International) <i>Mise en place d'un système de management de la qualité ISO 9001 : 2008</i>	4A EMA	4 mois

2009-2010	Etude de cas (3 étudiants) <i>Utilisation de la démarche REGAL en ingénierie des exigences pour la spécification d'exigences d'interopérabilité de système</i>	4A EMA	5 sem.
2006-2007	Stage de fin d'étude (1 étudiant) - Société SAFT <i>Etude d'optimisation des machines (Maintenance)</i>	Master PII ³¹	5 mois
2005-2006	Stage de fin d'étude (1 étudiant) - AZCAR Distribution Division <i>Contrôle des marchandises et suivi d'expéditions à l'international</i>	Master PII	5 mois
2004-2005	Stage de fin d'étude (1 étudiant) – Pescados y Mariscos AHORRAMAS <i>Analyse organisationnelle et décisionnelle de l'entreprise</i>	Master PII	5 mois
2004-2005	TER ³² (1 étudiant) <i>Etude bibliographique et synthèse sur l'interopérabilité</i>	Master 1 EEA	1 mois
2004-2005	TER (1 étudiant) <i>Etude bibliographique et synthèse sur les différentes étapes des méthodes de modélisation d'entreprise</i>	Master 1 EEA	1 mois

4. Liste des publications jointes au mémoire

S. Mallek, N. Daclin, V. Chapurlat

The application of interoperability requirement specification and verification to collaborative processes in industry

Computers in Industry, vol. 63, issue 7, pp 643-658, septembre 2012

S. Mallek, N. Daclin, V. Chapurlat, B. Vallespir

Enabling model checking for collaborative process analysis: from BPMN to Network of Timed Automata
Enterprise Information Systems vol. 9 - issue 3, pp 279-299, avril 2015

N. Daclin, S. Mallek-Daclin, V. Chapurlat, B. Vallespir

Writing and verifying interoperability requirements: Application to collaborative processes

Computers in Industry, vol. 82, pp. 1-18, octobre 2016

S. Billaud, N. Daclin, V. Chapurlat

Interoperability as a key concept for the control and evolution of the System of Systems (SoS)

6th International IFIP Working Conference on Enterprise Interoperability, Nîmes, mai 2015

³¹ Productique et Informatique Industrielle

³² Travaux d'Etudes et de Recherches