



HAL
open science

**Contexte augmenté basé sur les prédictions pour une
réutilisation efficace des connaissances métier en
conception : application à la conception proactive pour
l'assemblage**

Bertrand Marconnet

► **To cite this version:**

Bertrand Marconnet. Contexte augmenté basé sur les prédictions pour une réutilisation efficace des connaissances métier en conception : application à la conception proactive pour l'assemblage. Autre. Université Bourgogne Franche-Comté, 2017. Français. NNT : 2017UBFCA018 . tel-01869735

HAL Id: tel-01869735

<https://theses.hal.science/tel-01869735>

Submitted on 6 Sep 2018

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.

SPIM

Thèse de Doctorat



école doctorale sciences pour l'ingénieur et microtechniques
UNIVERSITÉ DE TECHNOLOGIE BELFORT-MONTBÉLIARD

Contexte augmenté basé sur les prédictions pour une réutilisation efficace des connaissances métier en conception

Application à la conception proactive pour l'assemblage

 BERTRAND MARCONNET



SPIM

Thèse de Doctorat



école doctorale sciences pour l'ingénieur et microtechniques
UNIVERSITÉ DE TECHNOLOGIE BELFORT-MONTBÉLIARD

N° X X X

THÈSE présentée par

BERTRAND MARCONNET

pour obtenir le

Grade de Docteur de

l'Université de Technologie de Belfort-Montbéliard

Spécialité : **Sciences pour l'ingénieur**

Contexte augmenté basé sur les prédictions pour une réutilisation efficace des connaissances métier en conception

Application à la conception proactive pour l'assemblage

Unité de Recherche :

Laboratoire ICB-COMM (UMR 6303 CNRS)

Soutenue publiquement le 24 Novembre 2017 devant le Jury composé de :

PROF. DIMITRIS KIRITSIS	Président	Ecole Polytechnique Fédérale de Lausanne
PROF. NICOLAS PERRY	Rapporteur	Ecole Nationale Supérieure d'Arts et Métiers - Bordeaux
PROF. LIONEL ROUCOULES	Rapporteur	Arts et Métiers ParisTech - Aix-en-Provence
PROF. SAMUEL GOMES	Directeur de thèse	Université de Technologie Belfort- Montbéliard
DR. FRÉDÉRIC DEMOLY	Co-encadrant	Université de Technologie Belfort- Montbéliard
DR. DAVY MONTICOLO	Co-encadrant	Université de Lorraine

REMERCIEMENTS

Tout d'abord, je tiens à remercier les membres du jury, notamment les rapporteurs de thèse, à savoir M. Nicolas PERRY, Professeur de l'Ecole Nationale Supérieure d'Arts et Métiers - Bordeaux, et M. Lionel ROUCOULES, Professeur des Arts et Métiers ParisTech - Aix-en-Provence, d'avoir accepté d'évaluer mes travaux de recherches.

En outre, je remercie le Professeur Dimitris KIRITSIS de m'avoir fait l'honneur de présider ma soutenance de thèse.

J'exprime également ma gratitude à mon directeur de thèse, Monsieur Samuel GOMES, Professeur de l'Université de Technologie Belfort-Montbéliard, pour m'avoir conseillé et orienté dans mes projets de recherches tout au long de ces années de thèse.

J'adresse toute ma reconnaissance à mes co-encadrants, Monsieur Frédéric DEMOLY, Maître de Conférences à l'UTBM, pour ses précieux conseils en mécanique, son investissement ainsi que sa patience et son soutien. Il en est de même, à l'égard de Monsieur Davy MONTICOLO, co-encadrant et Maître de Conférences HDR à l'Université de Lorraine, pour son expertise en informatique.

Je ne manque pas de remercier mes encadrants industriels de la société ACCELINN, pour m'avoir épaulé en m'apportant leur aide dans la répartition de mon temps de travail tout au long de ces années de thèse en entreprise.

Sans oublier, le Professeur Guoyi TANG de l'Université Tsinghua pour m'avoir accueilli dans son laboratoire, l'Institut Avancé de Matériaux à Shenzhen.

Merci à David BASSIR, Maître de Conférences à l'UTBM pour m'avoir accueilli en Chine, ainsi que le travail effectué ensemble sur les problématiques de drone.

Je remercie particulièrement les institutions qui ont financées mes travaux de recherches dans le cadre de la bourse **CIFRE** (**C**onventions **I**ndustrielles de **F**ormation par la **R**Echerche), à savoir l'**ANRT** (**A**ssociation **N**ationale de la **R**echerche et de la **T**echnologie), l'entreprise ACCELINN et le laboratoire ICB-COMM (anciennement IRTES-M3M).

Je tiens également à remercier toutes les personnes dans l'ombre qui m'ont soutenu et encouragé, durant ces années de recherche et qui m'ont permis d'avancer dans l'élaboration de ces travaux. Je dédicace cette thèse à ma famille, mes amis et mes collègues de travail.

SOMMAIRE

Remerciements	v
Sommaire	x
Liste des figures	xiv
Liste des tableaux	xvi
Liste des abréviations et des symboles	xvii
Introduction générale	xix
1 Contexte et problématique	1
1.1 Enjeux industriels	2
1.2 Contexte scientifique	2
1.3 Problématique et objectifs de nos travaux de recherche	3
1.3.1 Problématique générale	3
1.3.2 Objectifs à poursuivre	4
1.4 Conclusion	6
2 État de l’art	9
2.1 Introduction	10
2.2 Contexte augmenté et conception proactive pour l’assemblage	10
2.2.1 Définition et objectif du contexte de conception proactive	10
2.2.2 Sensibilité au contexte de « <i>conception pour X</i> » et intention de conception	12
2.2.2.1 Intentionnalité	12
2.2.2.2 Sensibilité au contexte	14
2.2.2.3 Conception pour X	15
2.2.3 Modélisation du contexte	17
2.2.3.1 Modèle de Zimmermann et al.	17
2.2.3.2 Modèle de Viera et al.	18
2.2.3.3 Modèle de Nadoveza et Kiritsis	19

2.2.3.4	Modèle de Dhuieb et al.	20
2.2.4	Description et génération des futurs contextes	21
2.2.4.1	Description de la prédiction	21
2.2.4.2	Description de l'anticipation	22
2.2.5	Modélisation du futur contexte	23
2.2.5.1	Approche de Mayrhofer	23
2.2.5.2	Approche de Zamenopoulos et Alexious	24
2.2.5.3	Approche de Demoly et al.	26
2.2.5.4	Approche d'Andolfatto	28
2.2.6	Bilan et positionnement	29
2.3	Réutilisation des connaissances	30
2.3.0.1	Définition et objectif de la réutilisation des connaissances	31
2.3.1	Méthodes de gestion et d'ingénierie des connaissances	32
2.3.2	Ontologie et règles de raisonnement sémantique	34
2.3.2.1	Objectif de l'ontologie	34
2.3.2.2	Représentation du système de raisonnement	35
2.3.2.3	Règles de raisonnement sémantique	37
2.3.3	Modélisation de la connaissance avec le raisonnement	38
2.3.3.1	Approche AsD	38
2.3.3.2	Approche de Zhu et al.	39
2.3.3.3	Approche PRONOA2	40
2.3.4	Bilan et positionnement	41
2.4	Gestion du Cycle de vie des Produits	42
2.4.1	Concept et objectifs du PLM	42
2.4.2	Gestion des informations et des connaissances produit-process	45
2.4.3	Modélisation du PLM	46
2.4.3.1	Gestion intelligente	47
2.4.3.2	Gestion des connaissances	48
2.4.3.3	Gestion collaborative	50
2.4.4	Bilan et positionnement	51
2.5	Conclusion : nécessité d'un cadre méthodologique et d'une approche	52
3	Cadre méthodologique du contexte de conception augmenté orienté assemblage - DAMON	55
3.1	Introduction	56
3.2	Cadre méthodologique - DAMON	58

3.2.1	Objectif du cadre méthodologique DAMON	58
3.2.2	Vue d'ensemble du cadre méthodologique DAMON	61
3.3	Description de DAMON	63
3.3.1	Définition des relations du produit via le graphe orienté (étape 1)	63
3.3.2	Génération de la planification des séquences d'assemblage avec les relations composants à composants (étape 2)	64
3.3.3	Actualisation du contexte de conception (étape 3)	66
3.3.4	Requête des connaissances possibles (étape 4)	71
3.3.5	Instanciation des connaissances appropriées dans l'activité de conception (étape 5)	77
3.4	Conclusion	82
4	Approche de modélisation d'un système sensible au contexte - MERCURY2	85
4.1	Introduction	86
4.2	Approche - MERCURY2	87
4.2.1	Vue d'ensemble et objectif de l'approche - MERCURY2	87
4.2.2	Étapes de modélisation	90
4.3	Architecture de l'application sémantique	91
4.3.1	Vue d'ensemble de l'architecture retenue pour l'application sémantique	91
4.3.2	Conception de la couche interface applicative (étapes 1 à 2)	94
4.3.3	Conception de la couche à base d'ontologie (étapes 3 jusqu'à 8)	99
4.3.4	Conception de la couche logique de raisonnement (étapes 9 jusqu'à 11)	109
4.4	Conclusion	116
5	Cas d'études illustrant le cadre méthodologique DAMON et l'approche MERCURY2	117
5.1	Introduction	118
5.2	Premier cas d'étude : Pince de désincarcération	118
5.2.1	Présentation	118
5.2.2	Description	119
5.3	Deuxième cas d'étude : Conception d'un drone montre	124
5.3.1	Présentation	124
5.3.2	Description	126
5.3.3	Démarche d'optimisation numérique pour améliorer l'efficacité de l'approche	135
5.3.4	Description du problème d'optimisation	136
5.4	Conclusion	142
6	Conclusion générale et perspectives	145

6.1	Conclusion générale	146
6.2	Perspectives de recherche	147
6.2.1	Perspectives à court terme	147
6.2.2	Perspectives à moyen terme	148
6.2.3	Perspectives à long terme	148
	Références bibliographiques	151
7.1	Valorisation des travaux de thèse	151
7.1.1	Revue internationale	151
7.1.2	Conférences internationales	151
7.1.3	Colloques nationaux	151
7.2	Références externes	151

LISTE DES FIGURES

1	Représentation synoptique de l'organisation de la thèse	xxi
1.1	Notre positionnement sur la réutilisation des connaissances et des prédictions	5
1.2	Domaines de recherche traités dans le cadre de nos travaux de recherche	6
2.1	De l'ingénierie séquentielle vers l'ingénierie proactive en début du cycle de vie [Demoly <i>et al.</i> , 2013]	11
2.2	Le triangle sémantique selon différentes théories de l'intention de conception [Zamenopoulos, 2012]	13
2.3	Modèle Need-Function-Principle-System - NFPS par [Chen <i>et al.</i> , 2015a] [Chen <i>et al.</i> , 2015b]	14
2.4	Les méthodologies employées en conception pour X, tout au long du cycle de vie du produit selon [Bralla, 1996] [Pahl & Beitz, 1996] [Pahl <i>et al.</i> , 2007]	16
2.5	Définition du contexte selon [Zimmermann <i>et al.</i> , 2007]	17
2.6	Meta-modèle du contexte par [Viera <i>et al.</i> , 2011]	19
2.7	Modèle conceptuel du contexte par [Nadoveza & Kiritsis, 2014]	19
2.8	Modèle de contexte proposé par [Dhuieb <i>et al.</i> , 2016]	20
2.9	Cadre méthodologique proposé par [Mayrhofer, 2004]	23
2.10	Modèle basé sur la prédiction proposée par [Rosen, 1985]	24
2.11	Modèle basé sur la prédiction par [Zamenopoulos & Alexious, 2007]	26
2.12	Approche proposée par [Demoly, 2010]	27
2.13	Approche proposée par [Andolfatto, 2013]	28
2.14	Cartographie des domaines relatifs aux contexte et futur contexte, représentant le « <i>contexte augmenté</i> »	30
2.15	La pyramide DIKW [Ackoff, 1989]	31
2.16	Typologie des connaissances selon [Kim & Kim, 2011]	32
2.17	Processus de capitalisation des connaissances basé sur la norme ISO 15489 et sur la gestion des enregistrements [Laroche, 2007]	33
2.18	Architecture du Web Sémantique [Antonioni & Van Harmelen, 2008]	35
2.19	Exemples d'une approche conventionnelle imbriquant connaissances déclaratives et procédurales (à gauche) et d'une convention de programmation (à droite), qui sépare mieux les affirmations déclaratives et les actions procédurales [Wray <i>et al.</i> , 2004]	37
2.20	Modèle ontologique AsD proposé par [Kim <i>et al.</i> , 2008]	38

2.21 Exemple de règles SWRL proposées par [Kim <i>et al.</i> , 2006] introduisant les concepts de : direction de référence, surface de contact, relation spatiale, degré de liberté, etc.	39
2.22 Exemple de raisonnement sémantique proposé par [Zhu <i>et al.</i> , 2011]	40
2.23 Modèle ontologique PRONOA2 proposé par [Gruhier <i>et al.</i> , 2015]	41
2.24 Règles logique de description (DL) dans PRONOA2 [Gruhier <i>et al.</i> , 2015]	41
2.25 Cartographie des approches exploitées pour réutiliser les connaissances	42
2.26 Elements fondamentaux du PLM [Terzi <i>et al.</i> , 2010]	44
2.27 Cartographie des systèmes PLM existants selon [Demoly <i>et al.</i> , 2013]	45
2.28 Elements de base pour modéliser l'approche de gestion du cycle de vie des produits [Abramovici, 2007]	47
2.29 Evolution du PDM et du PLM basés sur le PDD [Weber, 2003]	48
2.30 Approche PROMISE [Kiritsis, 2013]	49
2.31 Concept du HUB Incubateur [Petrazoller <i>et al.</i> , 2014]	50
2.32 Collaborative Product Innovation [Sharma, 2005]	51
2.33 Cartographie des systèmes PLM en début de cycle de vie et positionnement de l'outil PEGASUS	52
3.1 Introduction du cadre méthodologique DAMON	56
3.2 Comparatif entre la situation de conception actuelle et la situation de conception dotée d'un « <i>contexte augmenté</i> »	57
3.3 Les trois situations temporelles appliquées au cadre méthodologique proposé	58
3.4 « <i>Contexte augmenté</i> » pour la réutilisation des connaissances	59
3.5 Situation de conception en lien avec le contexte de conception augmentée	60
3.6 Vue d'ensemble du cadre méthodologique DAMON	62
3.7 Logigramme décrivant les différentes étapes du cadre méthodologique DAMON	62
3.8 Définition du contexte de conception - cas d'application choisi d'une boite	64
3.9 Génération du scénario d'assemblage	65
3.10 Les informations spatiale et temporelle se chevauchent afin de former le « <i>contexte augmenté</i> »	66
3.11 Génération des entités géométriques dans chaque situation	67
3.12 Notre proposition d'amélioration de la théorie JANUS	68
3.13 Description des définitions des espaces de conception pour le positionnement des liaisons cinématiques	69
3.14 Description des définitions des espaces de conception pour le positionnement des liaisons technologiques	70
3.15 Correspondance des relations candidates entre les connaissances et le graphe spatio-temporel	71
3.16 Composition d'une connaissance conditionnelle - Ck	72

3.17 Représentation des connaissances (1)	73
3.18 Représentation des connaissances (2)	74
3.19 Exemples de spécifications appliquées avec les connaissances	75
3.20 Chainage de règles	76
3.21 Allocation des connaissances sur le graphe spatio-temporel	78
3.22 Gestion du cycle de vie de la connaissance	79
3.23 Etapes d'instanciation des connaissances pendant l'activité de conception	80
3.24 Formalisation des templates de connaissances dans KrossRoads pour les instancier dans CATIA	81
3.25 Comparatif des différentes itérations de la boîte	82
4.1 Introduction de l'approche MERCURY2	86
4.2 Vue d'ensemble du HUB Incubateur - PEGASUS	88
4.3 Réconciliation des systèmes PDM, MPM, CAO et KB avec la Bill Of Relations (BOR)	88
4.4 Flux d'informations et des connaissances avec le HUB	89
4.5 Processus de conception du HUB illustrant le découpage entre la conception du profil utilisateur, la conception de l'ontologie et du volet raisonnement	90
4.6 Différentes couches composent le HUB	91
4.7 Processus pour instancier les éléments dans la conception CAO	93
4.8 Diagramme de cas d'utilisation du HUB PEGASUS	95
4.9 Diagramme UML de séquence décrivant le déroulement des actions dans le HUB PEGASUS	96
4.10 Interface de PEGASUS CAD Assistant : vue contextuelle	97
4.11 Interface de PEGASUS CAD Assistant : vue comportementale	98
4.12 Interface de PEGASUS CAD Assistant : vue géométrique	98
4.13 Modèle ontologique de PRONOA3	100
4.14 Résultat du test de consistance de PRONOA3 effectué avec l'éditeur d'ontologie Protégé et le moteur d'inférence Pellet	109
4.15 Introduction des unités de raisonnement RUN	109
4.16 But de la fonction de dimension du bord de rivet	111
4.17 Instances de départ de l'ontologie, représentant le « <i>contexte augmenté</i> », avant l'instanciation de la fonction	112
4.18 Résultat des instances pour l'unité de raisonnement 8	115
5.1 Schéma cinématique de la pince de désincarcération étudiée	118
5.2 Relations composant-à-composant de la pince de désincarcération	119
5.3 Génération et sélection de planification de séquence d'assemblage avec PEGASUS	120

5.4	Graphe spatio-temporel avec les connaissances allouées	121
5.5	Exemple de l'instanciation d'une connaissance représentant la fonction d'auto-alignement	122
5.6	Comparatif des différentes itérations de la pince de désincarcération	123
5.7	Comparatif des connaissances instanciées en cours de conception	124
5.8	Concept du prototype de drone-montre	125
5.9	Définition de la eBOM, dans le PDM de la plateforme ACSP	126
5.10	Génération des séquences admissibles d'assemblage par l'application PEGASUS	127
5.11	Définition des opérations d'assemblage dans NOTIXIA	128
5.12	Restructuration automatique de la eBOM dans l'outil CAO	129
5.13	Graphe spatio-temporel du bras du drone	130
5.14	Instances des éléments issus du « <i>contexte augmenté</i> » du bras du drone	131
5.15	Synthèse des connaissances capitalisées dans KrossRoads	132
5.16	Allocation de la fonction tapered end feature sur le bras du drone	133
5.17	Espace de conception du moteur et porte-moteur sur le bras du drone	134
5.18	Evolution des composants avec les connaissances	134
5.19	Première définition du produit	135
5.20	Diagramme simplifié des relations entre les principaux composants avec les paramètres pour l'optimisation du temps de vol	137
5.21	Résultats d'optimisation entre le temps de vol, le poids du drone, le prix et la capacité de la batterie	139
5.22	Optimisation basée sur un cas critique - chute de deux mètres absorbée par un bras	140
5.23	Visualisation des configurations A, B et C	142
5.24	Prototype du drone-montre DWARF	143

LISTE DES TABLEAUX

2.1	Synthèse de la littérature scientifique dans le domaine des contextes actuels et futurs	29
2.2	Vue d'ensemble des méthodes de réutilisation des connaissances	34
3.1	Parties prenantes identifiées	63
4.1	Extrait des concepts et des terminologies associés	101
4.2	Extrait du tableau de spécification des concepts	102
4.3	Extrait du tableau des propriétés des objets	103
4.4	Extrait du tableau des propriétés des données	104
4.5	Règles DL de PRONOA3 décrivant les liaisons cinématiques	105
4.6	Règles DL de PRONOA3 décrivant les liaisons technologiques	106
4.7	Règles DL de PRONOA3 décrivant les fonctions d'assemblage (1)	107
4.8	Règles DL de PRONOA3 décrivant les fonctions d'assemblage (2)	108
4.9	Sommaire des unités de raisonnement - RUN	110
5.1	Liste des composants et des matériaux associés pour le cas d'étude de la pince	118
5.2	Liste des composants pour le cas d'étude du drone-montre DWARF	126
5.3	Exigences du drone-montre DWARF	136
5.4	Exemple des paramètres techniques pour les variables continues	138
5.5	Exemple des paramètres techniques pour les variables discrètes	138
5.6	Exemple des fonctions objectifs (la description des variables est détaillée dans la liste des abréviations et des symboles au début du document)	138
5.7	Exemple des contraintes	138
5.8	Exemple de règles, pour vérifier les critères d'optimisation (la description des variables est détaillée dans la liste des abréviations et des symboles au début du document)	139
5.9	Les paramètres et leurs limites utilisés lors de l'optimisation dans CATIA	140
5.10	Résultat des cinq simulations dans les différentes configurations	141

LISTE DES ABRÉVIATIONS ET DES SYMBOLES

α	Décharge maximale de la batterie
Δt	Autonomie (temps de vol)
\forall	Quantificateur universel
ω	Révolution du moteur
ρ	Densité de l'air
\sqcap	Conjonction
\sqcup	Disjonction
\sqsubseteq	Hiérarchie des rôles
C	Capacité de la batterie
D	Diamètre de l'hélice
F	Poids du drone
h	Hélice géométrique des hélices
p	Pression d'air
S	Zone circulaire utilisée par les hélices
T_b	Tension de la batterie
V	Vitesse de l'air
v	Vitesse induite par l'hélice

ACSP Atelier Coopératif de Suivi de Projet

ASDA Assembly Sequences Definition Algorithm

BOL Beginning Of Lifecycle

BOM Bill Of Materials

BOP Bill Of Process

BOR Bill Of Relations

CADBOM Computer Aided-Design Bill Of Materials

CAO Conception Assisté par Ordinateur

DAMON Design context AugMentatiON

DFA Design For Assembly

- DFMA** Design For Manufacture and Assembly
- eBOM** engineering Bill Of Materials
- EOL** End Of Lifecycle
- KB** Knowledge Base
- mBOM** manufacturing Bill Of Materials
- MERCURY** Management appRoach of prodUct and process Relationships in assembly and design phases
- MOL** Middle Of Lifecycle
- MPM** Manufacturing Process Management
- OWL** Ontology Web Language
- PDM** Product Data Management
- PEGASUS** Product design Engineering based on Generative Assembly SeqUenceS planning
- PLM** Product Lifecycle Management
- PRONOIA** PROduct relatioNships description based On mereotopologicAl theory
- RUN** Reasoning UNit
- SQWRL** Semantic Query-enhanced Web Rule Language
- SWRL** Semantic Web Rule Language
- UML** Unified Modeling Language

INTRODUCTION GÉNÉRALE

Notre contexte socio-économique en constante évolution, implique de grandes difficultés pour les entreprises en termes d'anticipation de contraintes liées à la conception du couple produit-process, notamment dans l'exploitation des connaissances servant à réduire le temps de mise sur le marché du produit. Cette situation contraint les concepteurs à s'adapter aux exigences des parties prenantes, pour les intégrer dans la conception, entraînant parfois une exploitation inappropriée des informations et des connaissances. Pour y remédier et afin de soutenir les différentes phases du développement d'un produit, les acteurs métiers utilisent des systèmes informatisés pour gérer et anticiper les informations issues du cycle de vie du produit.

La stratégie d'entreprise, s'appuyant sur un système d'information, appelée gestion du cycle de vie des produits (**Product Lifecycle Management - PLM**) est une démarche globale permettant de gérer l'ensemble des informations parcourant le cycle de vie des produits. Le système PLM exploite des informations et des connaissances en lien avec le produit et son process, ce qui fait de leur réutilisation une partie des enjeux majeurs dans le domaine de la recherche, surtout, dès les premières phases de conception [Liu *et al.*, 2009], afin de prendre en compte les impacts des choix de conception, sur l'ensemble du cycle de vie du produit : (i) Début de cycle de vie (**Begin Of Lifecycle - BOL**), (ii) Milieu du cycle de vie (**Middle Of Lifecycle - MOL**) et (iii) Fin de cycle de vie (**End Of Lifecycle - EOL**). Pendant le processus de conception, une bonne compréhension de ces impacts sur le cycle de vie permettent au concepteur de pouvoir modifier, par exemple et en toute connaissance de cause, la structure et les informations relationnelles entre les composants.

Actuellement, la réutilisation des connaissances, est une démarche complexe et encore sous-exploitée, car elle n'est que partiellement diffusée dans les outils de **Conception Assistée par Ordinateur - CAO (Computer Aided Design - CAD)** [Chandrasegaran *et al.*, 2013]. Les connaissances utilisées par les concepteurs doivent elles-même faire l'objet d'une gestion de leur cycle de vie en raison de l'évolution de l'activité métier, pour être en phase avec l'organisation de l'entreprise. Pour cela, il est nécessaire de comprendre la chronologie des notions de passé, présent et futur, notions essentielles à la stratégie d'une entreprise. En effet, une projection dans le futur offre la possibilité de générer plusieurs scénarios de conception au cours de l'acte de conception, permettant d'anticiper les interactions entre le produit et son environnement tout au long de son cycle de vie [Demoly *et al.*, 2011]. En conséquence, cela réduirait le nombre d'erreurs en conception grâce à une prise de conscience du concepteur lors de sa phase d'exploitation des dites connaissances métier [Belkadi *et al.*, 2013] en lien avec le futur. Ainsi, plongé dans son contexte, le concepteur pourra anticiper les impacts de ses choix de conception dans le développement du couple produit-process. Tout ceci, favoriserait, en adéquation avec son activité, une compréhension complète des besoins du concepteur sur les possibilités d'assembler, de fabriquer, de transporter ou toute autre situation de vie en lien avec le produit.

L'orientation retenue pour aborder cette question de recherche consiste à générer des prédictions (le « *futur* ») pour augmenter le contexte (le « *présent* ») [Mayrhofer, 2004]. Le but est d'activer des connaissances (le « *passé* ») appropriées en lien avec le « *contexte augmenté* » [Sohn *et al.*, 2015]. Nous présenterons donc le concept appelé « *contexte augmenté* » qui consiste à améliorer l'exploitation et la réutilisation de connaissances. La sensibilisation du concepteur sur

les impacts de ses actions au sein de l'activité de conception fait l'objet de nos travaux de recherche, basés sur l'automatisation de la recherche des connaissances au travers de systèmes sensibles au contexte (**Context-Sensitive Systems**) [Viera *et al.*, 2011]. Dans le cas particulier de la conception orientée assemblage, le contexte de conception facilite la réutilisation des connaissances issues des règles de la conception pour l'assemblage (**Design For Assembly - DFA**) [Boothroyd *et al.*, 1994] [Boothroyd *et al.*, 2011]. Différents outils et systèmes existent pour gérer l'information produit au sein du PLM, notamment les systèmes de Gestion des Données Produit (**Product Data Management - PDM**), Gestion du Processus de Fabrication (**Manufacturing Process Management - MPM**) et Base de Connaissances (**Knowledge Base - KB**).

Nos travaux de recherche se sont déroulés au sein de la société **ACCELINN**, et du département **Conception Optimisation et Modélisation en Mécanique – COMM** du laboratoire **ICB UMR 6303 CNRS** de l'**Université de Technologie de Belfort-Montbéliard - UTBM**. **ACCELINN** est une start-up dont le cœur de métier est le transfert de l'innovation entre laboratoire et entreprise, tout en facilitant l'intégration des travaux de recherches appliqués au secteur industriel au sein de domaines variés (ferroviaire, automobile, nucléaire, etc.). Tandis que la partie recherche a quant à elle, été effectué au sein du département **COMM** de l'**ICB**, conduite et animée par Samuel GOMES. L'équipe concernée travaille depuis plusieurs années dans le domaine de la conception proactive de systèmes mécaniques, et de la gestion de cycle de vie des produits, en intégrant l'ingénierie à base de connaissances [Gomes, 2008] [Monticolo, 2008]. Les travaux de recherche se focalisent notamment sur l'amélioration des processus routiniers de conception dans le cycle de vie des produits. Une plateforme baptisée **ACSP (Atelier Coopératif de Suivi de Projet)** a été développée, dont la continuité a évoluée vers un système HUB pour réconcilier les domaines produit et process, appelé **PEGASUS (Product design Engineering based on Generative Assembly SequenceS planning)** [Demoly, 2010]. Le HUB en question est structuré par l'ontologie **PRONOA2 (PROduct relationships description based On mereotopologicAI theory)** pour décrire les relations spatio-temporelles du produit et de son process [Gruhier *et al.*, 2015].

Les travaux présentés se focaliseront sur la définition de l'approche et du développement du système sensible au contexte. Ce système constitue un support d'aide à la décision pour le concepteur, lui permettant ainsi, de vérifier et de valider ses choix de conception, en prenant conscience de leurs impacts sur chaque étape de son cycle de vie. Ainsi, la thèse est organisée en six chapitres (Figure 1), la contribution scientifique étant décrite dans les chapitres 3 et 4.

Le **premier chapitre**, intitulé « *Contexte et problématique* » aborde la partie relative à la compréhension du contexte scientifique industriel pour en faire émerger la problématique générale de recherche.

Le **deuxième chapitre**, appelé « *État de l'art dans les domaines de la conception proactive augmentée au sein de la gestion du cycle de vie des produits et de la réutilisation des connaissances* » décrit un état de l'art de l'ingénierie proactive dans le PLM, de la réutilisation des connaissances et de la description des prédictions générées.

Le **troisième chapitre** présente un cadre méthodologique de contexte de conception basé sur la prédiction appelée **DAMON (Design context AugMENTatiON framework)**, afin de réutiliser efficacement les connaissances dans la conception d'assemblage. La méthode proposée consiste à exploiter des connaissances selon une typologie comprenant trois catégories (connaissances conditionnelles, déclaratives et procédurales), afin de faciliter et de rendre plus efficace les mécanismes de raisonnement au sein du système de gestion des connaissances, mais aussi de rendre plus pertinent l'instanciation des connaissances en conception.

Le **quatrième chapitre** permet d'intégrer un « *système sensible au contexte* » dans un environnement existant appelé **PEGASUS**, dans le but de capturer le contexte de conception et de diffu-

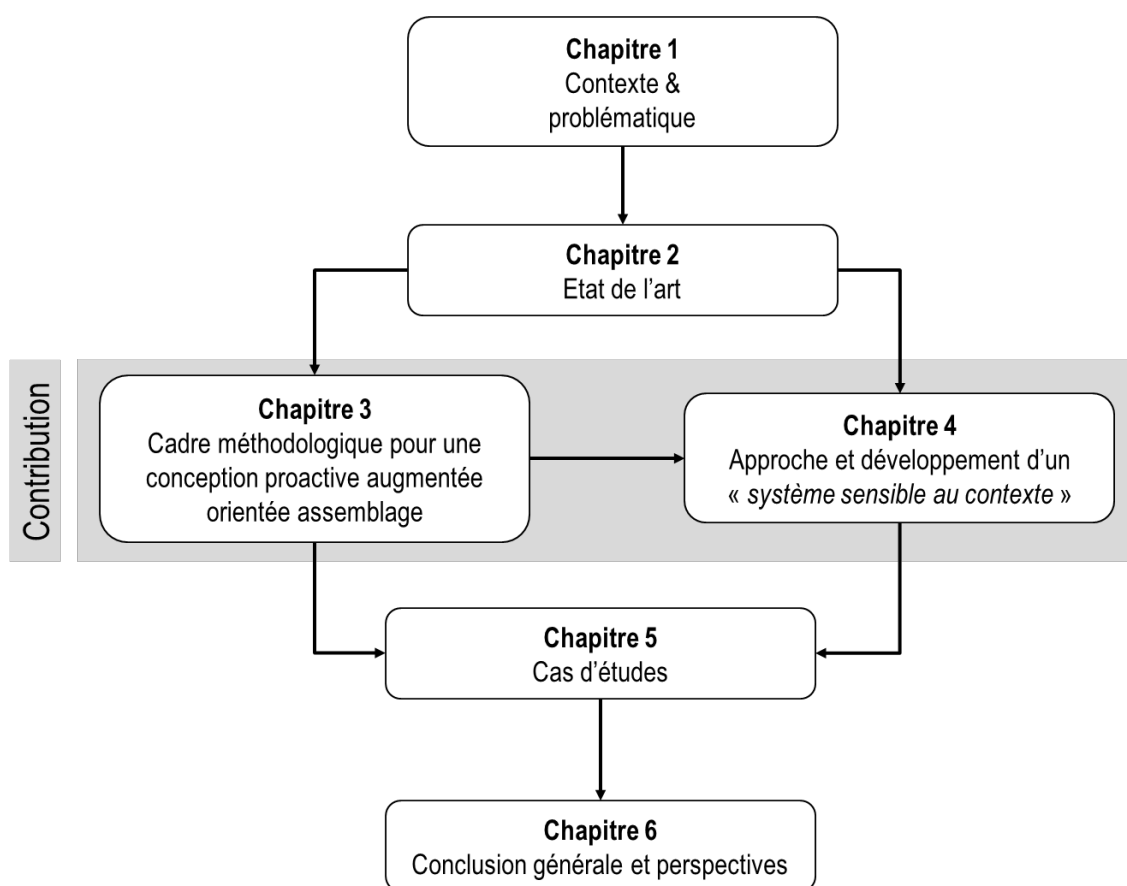


Figure 1 – Représentation synoptique de l'organisation de la thèse

ser l'information dans le système PLM. L'approche proposée, baptisée **MERCURY2** (**ManagEment appRoach of prodUct and process Relationships in assembly and design phases**), consiste à modéliser le système au travers de trois niveaux : un niveau ontologique avec **PRONOA3** (**PROduct relatioNships description based On mereotopologicAl theory**) pour décrire le « *contexte augmenté* », des unités de raisonnement **RUN** (**Reasoning UNit**) pour raisonner dans le système, et enfin une Interface Homme-Machine appelée **PEGASUS CAD Assistant** qui sert d'intermédiaire entre les concepteurs et PEGASUS.

Le **cinquième chapitre**, aborde l'application de l'approche via deux cas d'études dont l'un d'eux sera amélioré en ayant recours à de l'optimisation numérique multi-objective.

Le **dernier chapitre** (chapitre six) traduit, quant à lui, la conclusion par une synthèse sur l'avancement des travaux tout en proposant de futures pistes de réflexions et d'améliorations.

CONTEXTE ET PROBLÉMATIQUE

« *Tout obstacle renforce la détermination. Celui qui s'est fixé un but n'en change pas.* »

Léonard De Vinci

Sommaire

1.1 Enjeux industriels	2
1.2 Contexte scientifique	2
1.3 Problématique et objectifs de nos travaux de recherche	3
1.3.1 Problématique générale	3
1.3.2 Objectifs à poursuivre	4
1.4 Conclusion	6

1.1/ ENJEUX INDUSTRIELS

Dans un contexte industriel compétitif, les réductions de coûts liées aux temps de conception sont souvent prioritaires, impliquant ainsi de faire des choix importants et stratégiques pour limiter les erreurs dans la conception de produit. Les impacts associés aux décisions de conception influencent la manière dont le produit sera développé tout au long de son cycle de vie. Le produit, qu'il soit conçu d'une manière innovante ou routinière, confronte les concepteurs à des prises de décisions favorisant l'anticipation sur la configuration ou l'architecture souvent inconnue, en s'appuyant principalement sur des connaissances tacites et explicites.

La conception routinière permet notamment de se baser sur ces connaissances pour la modélisation de l'architecture du couple produit-process, avec l'appui de modèles, méthodes et outils via l'ingénierie à base de connaissances (**Knowledge Based Engineering - KBE**), et ce, au sein de la CAO. Elle permet également d'éviter la répétition des erreurs de conception en appliquant par exemple les méthodes issues du Design For X [Huang, 1996], dont le **Design For Assembly - DFA** [Boothroyd *et al.*, 1994], qui est l'une de ces composantes majeures. Le DFA est constitué d'un ensemble de règles et de méthodes qui servent à concevoir les produits/systèmes mécaniques en intégrant, très en amont du processus de conception, les contraintes liées au métier de l'assemblage.

En effet, il est parfois difficile d'interpréter les contraintes liées à un contexte métier (assemblage, fabrication, transport, maintenance, etc.), et de les comprendre en amont de la conception du produit, de son process et par conséquent d'anticiper les impacts que peuvent avoir chaque décision de conception. Ces méthodes permettent d'anticiper les difficultés inhérentes à la non prise en compte, des contraintes manufacturières en général, et des contraintes liées à l'assemblage en particulier, dans la mesure où la prise de décision pour le concepteur s'avère difficile, compte tenu de la quantité d'informations à traiter, et du manque de vision prospective sur les situations futures. De notre point de vue, il est préférable de pouvoir prédire le futur contexte dans lequel va se retrouver le produit, en ayant recours à des scénarios prédictif, afin de fournir au bon moment des informations pertinentes aux concepteurs (contexte de conception proactive du couple produit-process).

1.2/ CONTEXTE SCIENTIFIQUE

La connaissance est l'élément de base de tout système d'assistance au concepteur servant à résoudre les divers problèmes rencontrés, tels qu'une erreur de conception sur un ancien produit, un paramètre non applicable, une règle de conception obsolète, etc. Il est donc important de mettre en place une véritable gestion des connaissances qui puisse apporter des réponses à ce genre de problématiques. D'ores et déjà, de nombreuses méthodes dédiées à la gestion de connaissances existent. Elles exploitent plusieurs catégories de connaissances préalablement capturées, afin de les appréhender, et de révéler leurs fonctions : connaissances conditionnelles (quand ou pourquoi), connaissances procédurales (comment), connaissances de causalité (pourquoi), connaissances déclaratives (quoi), connaissances relationnelles (avec qui), etc. [Kim & Kim, 2011]. Ces connaissances sont souvent utilisées à un instant donné, et à la demande de l'utilisateur. La complexité de leur exploitation s'accroît lorsqu'il s'agit de les mettre en œuvre en adéquation avec le contexte.

Le terme « *prédiction* », qui se traduit par « *praedictio* » en latin, détermine le devenir de futurs événements [Pezzulo *et al.*, 2008]. Au sein du champ lexical de la prédiction figurent les termes suivants : prévoir, prévision, présage, etc., lesquels expriment un état cognitif consistant à vouloir

voir au-delà d'un instant présent. Il est possible de noter une ambiguïté avec le mot anticipation, issu de « *anticipatio* » d'origine latine, qui traduit l'action de se projeter dans l'avenir, pour représenter les résultats attendus via une action cognitive et/ou stratégique à mettre en œuvre. Par conséquent, le terme anticipation traduit un processus d'action consistant à prendre les devants pour changer le présent, alors que le terme prédiction traduit plutôt une projection sur une future situation possible, voire probable. Par conséquent, un processus de gestion de prédictions, au sein d'une ingénierie dite « *proactive* » intégrée dans une stratégie d'entreprise de type PLM, constitue une approche intéressante pour la réutilisation des connaissances en conception.

Nos travaux de recherche s'appuient sur des contributions nationales et internationales, en particulier, dans le domaine de la gestion des informations et des connaissances dédiées au couple produit-process en environnement PLM. Ils s'inscrivent au niveau national, au sein du réseau **AIP-PRIMECA** (Atelier Inter-établissement de Productivité - Pôle de Ressources Informatiques pour la MECAnique) dont la finalité est de renforcer les échanges scientifiques sur ces différentes thématiques. Nos travaux s'inscrivent également au sein de la communauté du **GDR-MACS** (Groupe De Recherche du CNRS en Modélisation, Analyse et Conduite des Systèmes dynamiques) qui se concentre, en particulier, sur la gestion du processus du cycle de vie des produits et des connaissances industrielles associées. Au sein du GDR-MACS, les travaux s'inscrivent plus spécifiquement au sein des groupes de travail l'**IS3C** (Ingénierie des Systèmes de Conception et Conduite du Cycle de Vie) et **C2EI** (Modélisation et pilotage des systèmes de Connaissances et de Compétences dans les Entreprises Industrielles). Sur le plan international, nous sommes affiliés à la **Design Society**, qui regroupe des chercheurs œuvrant dans le domaine de la conception. Nos travaux s'inscrivent également dans plusieurs autres communautés scientifiques, ou regroupements de chercheurs, par exemple :

- **L'AFIA** (Association Française pour l'Intelligence Artificielle) dédiée au développement de l'**IA** (Intelligence Artificielle) sous ses différentes formes, pour regrouper et faire croître la communauté française, afin d'en assurer la visibilité ;
- **L'IFIP WG 5.7** (International Federation of Information Processing) dédiée à la thématique des « *Systèmes de gestion de la production avancés* » (Advances in Production Management Systems).

1.3/ PROBLÉMATIQUE ET OBJECTIFS DE NOS TRAVAUX DE RECHERCHE

1.3.1/ PROBLÉMATIQUE GÉNÉRALE

Alors que le processus de capitalisation des connaissances métier commence à se généraliser au sein des entreprises, la réutilisation desdites connaissances issues des précédents projets, génère à l'heure actuelle de nombreuses difficultés du fait du manque de sensibilisation des concepteurs à ce sujet. Nous avons pu constater une évolution des pratiques dans ce domaine, grâce en particulier au développement des systèmes de gestion des informations de type PLM, en préalable à la mise en place de systèmes de gestion de connaissances. Dans les années 90, les problèmes d'interopérabilité entre l'ingénierie et la fabrication ont générés des surcoûts estimés à 1 milliard de dollars [Gregory *et al.*, 1999]. Ce type de problèmes s'est amplifié, lorsqu'il a fallu apporter de nombreuses modifications au produit, dans un contexte de conception routinière, avec à la clef l'apparition de défauts lors de transfert d'informations entre systèmes et de nouvelles exigences inhérentes à ce changement [Clarkson *et al.*, 2004]. Dans ce contexte, les entreprises sont amenées à centraliser leur savoir-faire [EADS & Mondon, 2008] [EADS, 2011], afin de gagner sur une meilleure fidélité/représentation possible lors du développement des produits [CIMDATA, 2014].

Les approches DFX [Huang *et al.*, 1999] ont vu émerger des lacunes dans le cycle de vie du produit, notamment dans le système PDM sur le processus de réutilisation des connaissances [Bilgic & Rock, 1997]. Tout ceci soulève la problématique de l'utilisation des prédictions dans la conception à l'image des méthodes utilisées depuis de nombreuses années, telles que : l'Analyse des Modes de Défaillance, de leurs Effets et de leur Criticité - **AMDEC** (Failure Mode and Effects Analysis - **FMEA**), Maison de la qualité (Quality Function Deployment - **QFD**), etc. En effet, certains auteurs estiment que les ingénieurs passent 60% de leur temps à chercher la bonne information [Ullman, 2009], quand 75% de leurs activités de conception inclut la réutilisation des connaissances existantes. D'autres auteurs évaluent à 30% le temps passé par les concepteurs à acquérir et à diffuser les connaissances au sein du processus de développement du produit [Vijaykumar & Chakrabarti, 2008]. Résultant de la diffusion partielle de la représentation des connaissances, en phase amont du processus de conception [Chandrasegaran *et al.*, 2013].

Ceci nous amène à formuler plusieurs questions, afin de pouvoir réutiliser efficacement et de manière proactive les connaissances issues des métiers situés en aval de la conception :

1. Comment rendre conscient le concepteur de l'impact de ses choix au cours du processus de conception produit ?
2. Comment réutiliser efficacement les connaissances en interaction avec les informations contenues dans le PLM ?
3. Comment générer un contexte de conception approprié basé sur des prédictions ?
4. Comment générer un « *contexte de conception augmenté* » à des fins de conception proactive du couple produit/process ?

1.3.2/ OBJECTIFS À POURSUIVRE

En se basant sur la problématique générale, formulée précédemment, et compte tenue de notre contexte scientifique et industriel, nous proposons de définir l'objectif de nos travaux de thèse de la manière suivante :

Etre capable de générer des prédictions sur les futures situations de vie du produit (assemblage, fabrication, etc.), afin d'augmenter le contexte de conception, et par conséquent, de faciliter la réutilisation (instanciations et mises à jour) au bon moment, des connaissances pertinentes du couple produit/process.

Pour aborder ces questions, il est important d'améliorer le processus de réutilisation des connaissances conforme aux intentions et au contexte de conception [Baxter *et al.*, 2007]. Pour situer nos travaux, nous nous appuyons sur la comparaison formulée par Claudia Bauzer Medeiros, qui croise processus d'utilisation des connaissances et des prédictions [Medeiros, 2009]. La Figure 1.1 illustre ce comparatif :

- **Niveau 1** : les systèmes d'aide à la décision permettent notamment d'améliorer le présent en simulant le futur ;
- **Niveau 2** : l'étude du passé, qui permet d'expliquer le présent, est l'un des processus utilisés pour les systèmes à base de connaissances ;
- **Niveau 3** : regarder dans le passé permet de prédire le futur, à l'aide d'analyse des tendances ou des systèmes de prévisions du comportement ;
- **Niveau 4** (notre proposition) : prédire le futur pour améliorer le contexte présent, actualisant ainsi connaissances issues d'expériences passées.

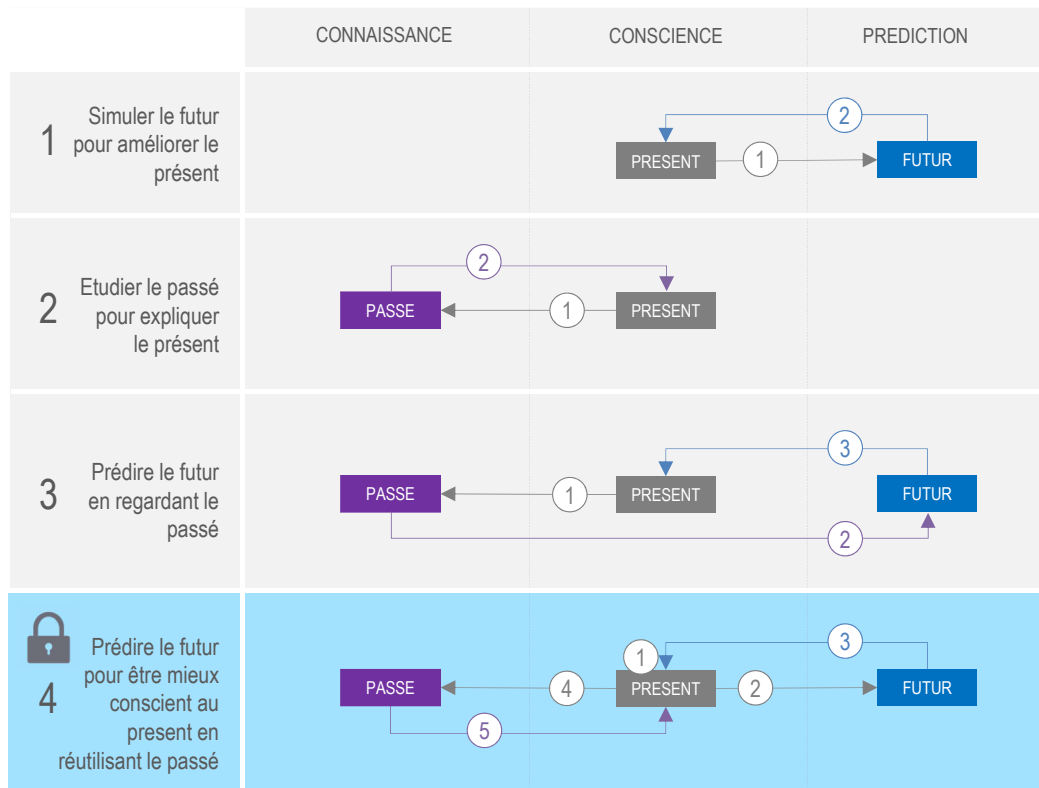


Figure 1.1 – Notre positionnement sur la réutilisation des connaissances et des prédictions

Afin d'apporter une contribution à ces différents niveaux de processus, nous proposons, comme l'illustre la Figure 1.1, un nouveau niveau (**niveau 4**) qui consiste à « *Prédire le futur pour améliorer le présent, par la réutilisation des connaissances issues d'expériences passées* », permettant ainsi, en phase amont du processus de conception, de rendre conscient le concepteur de l'impact de ses choix. Nos travaux de recherche ont pour objectif de lever le verrou scientifique correspondant au concept de « *présent augmenté* » à son existence, sa pertinence et à ses interactions.

Le concept de « *contexte augmenté* », que nous proposons, sera expliqué à travers une méthode et une approche dédiées au développement d'un système sensible au contexte, avec la mise en œuvre de deux cas d'études permettant ainsi d'illustrer la contribution de nos travaux.

Face à la problématique générale formulée aux objectifs énoncés et au verrou identifié, nous proposons les deux hypothèses suivantes qui guideront notre travail de recherche :

1. Il est possible dans un domaine de recherche restreint au domaine de la « *conception pour l'assemblage* », d'exploiter des prédictions couplées à un contexte de conception pour constituer un « *contexte augmenté* » et ainsi être capable d'effectuer une conception proactive à base de connaissances ;
2. Il est possible de définir une nouvelle génération d'outils de type « *systèmes d'aide à la décision* » exploitant le « *contexte augmenté* » permettant au concepteur d'être conscient des impacts de ses choix de conception sur les performances attendues du couple produit-process.

Au travers de la Figure 1.2, un état de l'art est proposé, portant sur les thèmes majeurs des travaux de recherche :

- Le contexte de conception proactive dans le PLM ;
- La réutilisation des connaissances ;
- La description et la génération des futurs contextes.

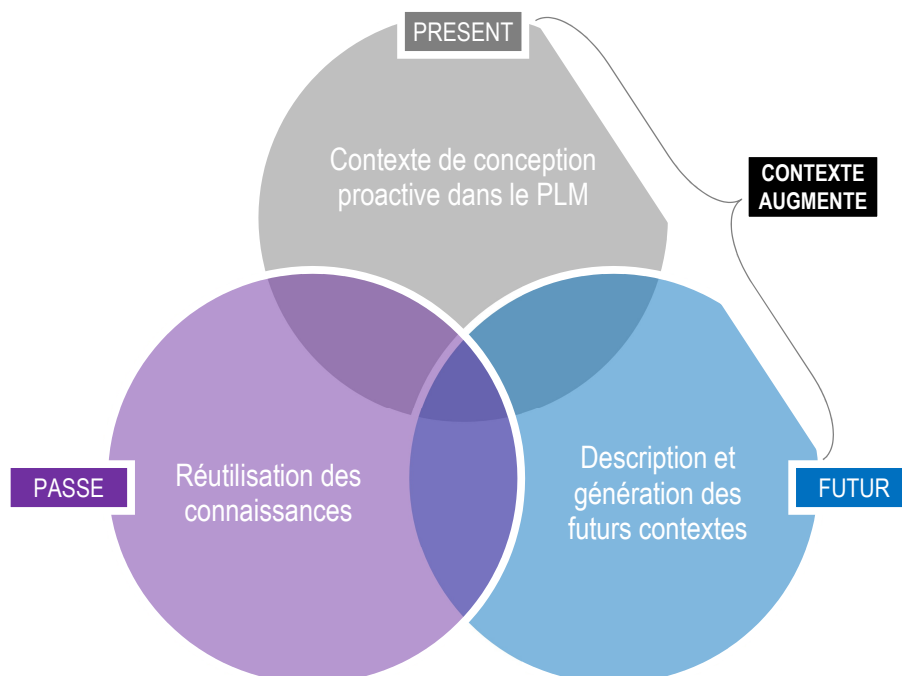


Figure 1.2 – Domaines de recherche traités dans le cadre de nos travaux de recherche

La première étape consiste à définir un cadre méthodologique dédié au concept de « *contexte augmenté* », qui doit permettre au concepteur d'exploiter un contexte de conception donné qui sera enrichi par des informations issues d'un des futurs contextes (exemple futur contexte d'assemblage du produit). Dans notre cas, le futur contexte est le résultat issu d'une génération de solutions alternatives dans un contexte de conception. Ce qui va ajouter des informations complémentaires sur l'assemblage pour faciliter l'aide à la décision du concepteur et faire émerger les connaissances de ce « *contexte augmenté* » qui pourront être exploitées plus efficacement par le concepteur. La deuxième étape consiste à appliquer le cadre méthodologique en vue de définir une approche intégrant nécessitant le développement d'un outil spécifique d'assistance aux concepteurs, ce dernier à la particularité d'être sensible au contexte en ciblant quand et pourquoi réutiliser, de manière opportune, certaines connaissances spécifiques.

1.4/ CONCLUSION

Ce chapitre a introduit l'enjeu industriel et scientifique de notre travail de recherche, en formulant une problématique liée à la réutilisation efficace et pertinente des connaissances, en phase amont du processus de conception du couple produit-process. Cette problématique constitue le point de départ de la formulation de nos objectifs de recherche, en identifiant un verrou scientifique conduisant à l'énoncé de deux hypothèses de recherche. Dans le prochain chapitre, nous allons confronter notre problématique de recherche à un état de l'art des domaines de la gestion et de

la réutilisation des connaissances, de la gestion des informations du produit sur l'ensemble de son cycle de vie, et sur l'exploitation des prédictions afin d'augmenter le contexte pour une conception proactive.

ÉTAT DE L'ART

« La connaissance s'acquiert par l'expérience, tout le reste n'est que de l'information. »

Albert Einstein

Sommaire

2.1 Introduction	10
2.2 Contexte augmenté et conception proactive pour l'assemblage	10
2.2.1 Définition et objectif du contexte de conception proactive	10
2.2.2 Sensibilité au contexte de « <i>conception pour X</i> » et intention de conception	12
2.2.3 Modélisation du contexte	17
2.2.4 Description et génération des futurs contextes	21
2.2.5 Modélisation du futur contexte	23
2.2.6 Bilan et positionnement	29
2.3 Réutilisation des connaissances	30
2.3.1 Méthodes de gestion et d'ingénierie des connaissances	32
2.3.2 Ontologie et règles de raisonnement sémantique	34
2.3.3 Modélisation de la connaissance avec le raisonnement	38
2.3.4 Bilan et positionnement	41
2.4 Gestion du Cycle de vie des Produits	42
2.4.1 Concept et objectifs du PLM	42
2.4.2 Gestion des informations et des connaissances produit-process	45
2.4.3 Modélisation du PLM	46
2.4.4 Bilan et positionnement	51
2.5 Conclusion : nécessité d'un cadre méthodologique et d'une approche	52

2.1/ INTRODUCTION

Ce chapitre expose les travaux de recherche passés et en cours dans la communauté scientifique en lien avec la problématique de conception proactive orientée assemblage à base de « *contexte augmenté* », évoquée dans le chapitre précédent. Nous aborderons, dans le prochain paragraphe, les travaux relatifs au concept de « *contexte augmenté* » emprunté au domaine de l'informatique et des sciences de l'information (pervasive and mobile computing) et à la notion de conception proactive. Nous citerons, en particulier, les concepts de sensibilité et d'intention dans les contextes de conception, ainsi que les travaux dédiés à la présentation des modèles de contexte et de prédiction pour interpréter le « *contexte augmenté* ». Le paragraphe 3 fait état des approches servant à modéliser et réutiliser les connaissances exploitant les technologies sémantiques. Le paragraphe 4 développe ensuite le concept de gestion du cycle de vie des produits, et ce, dans la perspective de ses évolutions et des challenges à venir. Pour terminer, le paragraphe 5 trace un bilan de cet état de la littérature, afin de proposer un cadre méthodologique et une approche qui puissent être originaux au regard des travaux existants. Cet état de l'art circonscrit le périmètre des problématiques connues nous permettant de positionner nos travaux de recherches.

2.2/ CONTEXTE AUGMENTÉ ET CONCEPTION PROACTIVE POUR L'ASSEMBLAGE

2.2.1/ DÉFINITION ET OBJECTIF DU CONTEXTE DE CONCEPTION PROACTIVE

A ce jour, peu de travaux relevant de l'analyse des contextes, des situations, des prédictions et des anticipations sont abordés dans le domaine du génie industriel. Par contre, dans d'autres domaines tels que la philosophie, les mathématiques, le génie logiciel voire le génie biologique, des travaux scientifiques existent et ont conduit à plusieurs publications.

Dans le domaine du génie industriel, en ce qui concerne plus spécifiquement les phases de conception de produits, voire de systèmes mécaniques, ils nécessitent la prise en compte et l'intégration de toutes les contraintes et connaissances requises dans chaque étape du cycle de vie du produit (processus métier, termes métier, règles d'experts, expériences, etc.). Ceci implique de capitaliser les connaissances en stockant des règles métiers qui rendent la réalisation des tâches de conception difficiles pour les architectes et les concepteurs produits, du fait de leur nombre et de leurs associations spécifiques à certains contextes métier [Maropoulos & Ceglarek, 2010] [Kusiak & Salustri, 2007]. Au cours des deux précédentes décennies, cette question a été abordée en complément des problématiques de l'ingénierie séquentielle et de l'ingénierie concurrente ou simultanée, toutes deux utilisées par les entreprises afin de diffuser au mieux les informations produit/process au sein des collectifs d'ingénierie tout au long du cycle de développement du produit. En effet, l'ingénierie séquentielle (ou « *sequential engineering* ») consiste à passer d'une étape du cycle de vie du produit à une autre, de manière séquentielle et cloisonnée au niveau : temporel, compétences métier, documents, etc. contribuant par exemple à ne diffuser aux services aval (production, maintenance, etc.) les données relatives au produit et à son industrialisation qu'une fois la conception détaillée terminée.

L'ingénierie concurrente (ou « *concurrent engineering* ») également qualifiée d'ingénierie intégrée, collaborative, simultanée, etc. consiste à structurer les organisations, les méthodes et les outils en vue de favoriser la transmission d'informations intermétiers, interservices et qui permet des gains de temps significatifs, des gains en qualité palliant certaines erreurs, etc. Cette évolution

vers l'ingénierie concurrente s'appuie largement sur la diffusion des informations ainsi que sur la gestion des connaissances (pour la fabrication, pour l'assemblage, etc.) pouvant même conduire à une ingénierie à base de connaissances utilisant des règles heuristiques, avec l'aide de systèmes experts et de moteurs d'inférences [Huang *et al.*, 1999]. Mais les informations qualitatives et les connaissances ne sont pas encore exploitées efficacement, en raison de leurs relations complexes ainsi que des contextes métier en constante évolution.

L'évolution actuelle tend vers une ingénierie qualifiée de proactive (ou « *proactive engineering* ») que l'on peut définir par :

L'intégration d'un contexte qualitatif aval dans les processus amont de conception, avant même que la géométrie du produit n'émerge.

Cette ingénierie proactive est une approche qui met en œuvre de manière dynamique l'anticipation des contraintes de conception, fabrication, maintenance, etc. au travers des relations que le produit a dans son cycle de vie (Figure 2.1).

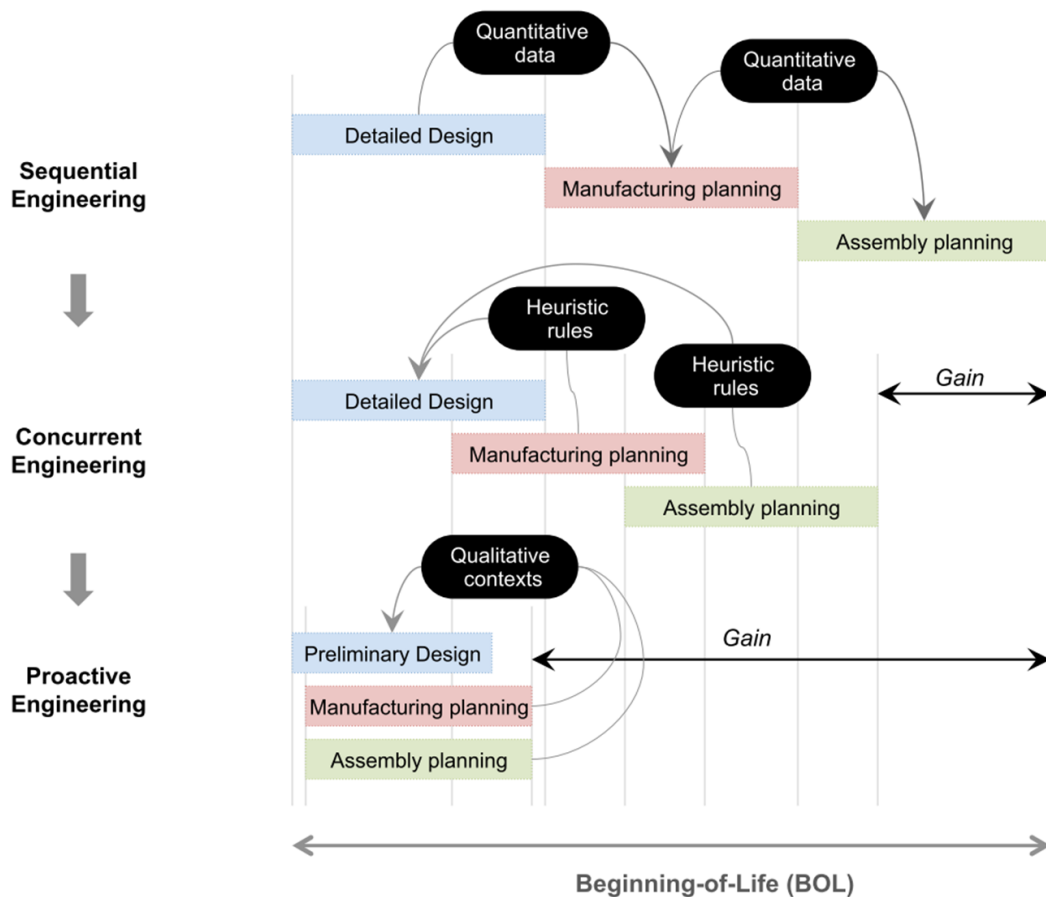


Figure 2.1 – De l'ingénierie séquentielle vers l'ingénierie proactive en début du cycle de vie [Demoly *et al.*, 2013]

L'ingénierie proactive étend donc les capacités de l'ingénierie simultanée en recouvrant les informations minimales dans la conception, afin de définir un contexte qualitatif, par une description formelle [Kim & Kim, 2011]. L'objectif de l'ingénierie proactive se définit comme la nécessité d'adapter le besoin futur des utilisateurs :

- En interprétant la compréhension des informations, pour les instancier dans un futur contexte [Boytsov & Zaslavsky, 2010];
- En gagnant du temps par la superposition des informations de la conception à la planification, en vue d'exploiter d'une manière intelligente les bonnes informations [Liu *et al.*, 2009] [Yan *et al.*, 2001].

Du point de vue de l'ingénierie mécanique, la notion de contexte a été développée par certains auteurs, comme un ensemble de contraintes qui influencent le comportement d'un système dans une tâche donnée [Bazire & Brézillon, 2005]. Cela implique la prise en compte des intentions de conception, leur représentation et leur compréhension [Chen *et al.*, 2015a].

2.2.2/ SENSIBILITÉ AU CONTEXTE DE « *conception pour X* » ET INTENTION DE CONCEPTION

La sensibilité au contexte et l'intention de conception sont des éléments importants qui servent à mettre en œuvre une approche basée sur la proactivité du contexte qualitatif. Il est ainsi nécessaire de comprendre comment capturer les intentions du concepteur, c'est-à-dire où et quand distribuer la connaissance métier dans le contexte de conception.

2.2.2.1/ INTENTIONNALITÉ

La représentation mentale de l'intention est un élément complexe qui relève de la philosophie de l'esprit. Il est possible, selon [Zamenopoulos, 2012], de définir la propriété de l'esprit afin de maintenir les états intentionnels, de la manière suivante :

Sa capacité à représenter ou à refléter des réalités existantes et non-existantes.

Ainsi, selon [Zamenopoulos & Alexious, 2007], intégrer la logique humaine pour communiquer avec la machine doit incorporer les notions d'intentions dans le contexte, afin de savoir :

- « *Ce que le concepteur veut* » ;
- « *Ce qu'il pense* » ;
- « *Ce qu'il veut concevoir* ».

Trois stratégies sont décrites par [Zamenopoulos, 2012] pour associer le terme d'intention à trois termes basiques illustrés à la Figure 2.2, ci-après.

Ces trois stratégies consistent à :

- Associer le terme « *sens* » pour spécifier les propriétés des objets ou l'état des choses dans le monde auquel le « *sens* » est attribué. Cette spécification est appelée « *sens* », « *intention* », « *connotation* », « *propriété universelle* » ou « *contenu représentatif* » ;
- Associer le terme « *sens* » à l'objet même ou à l'état d'une(des) affaire(s) dans le monde qui est attribué à une expression ;
- La spécification des propriétés d'un objet (c'est-à-dire le contenu de la représentation) détermine les conditions de vérité ou les conditions de satisfaction qui évaluent la relation entre une référence et une expression.

La capture d'intention est donc un élément important, car elle est décrite, comme un plan d'ac-

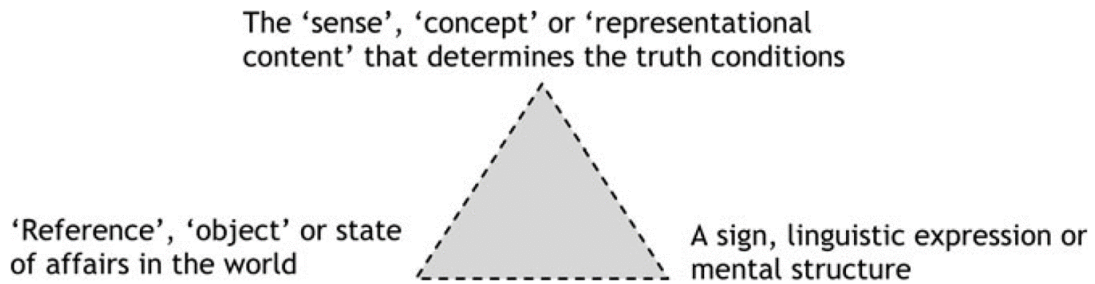


Figure 2.2 – Le triangle sémantique selon différentes théories de l'intention de conception [Zamenopoulos, 2012]

tion basé sur deux mécanismes : la causalité et l'intentionnalité [Riedl & Young, 2004]. L'intention dans la conception de produit, est considérée comme « *la capture intentionnelle d'un but* » [Ganeshan *et al.*, 1994]. La compréhension des intentions dans les systèmes CAO, est basée sur la fonctionnalité du produit et des spécifications géométriques [Zhang & Luo, 2010]. En outre, l'intention de conception doit bien être interprétée dans les systèmes d'informations, pour pouvoir assurer une sensibilité du concepteur. Elle doit être captée le plus tôt possible, afin d'adapter les connaissances adéquates au contexte lors de prise de décisions [Baxter *et al.*, 2007].

En conséquence, la collecte des données d'entrée et de sortie dans les systèmes CAO (les paramètres, le comportement, les relations, etc.), doit être basée sur un réseau de contraintes et de déduction logique [Shih & Anderson, 1997]. En général, la représentation des modèles paramétriques est basée, soit sur la collecte désordonnée, soit ordonnée des contraintes dans la conception des produits, ceci joue un rôle dans la description des intentions de conception [Shah, 1995]. L'autre aspect de l'intention de conception est la rationalité dans la prise de décisions qui se base sur la divulgation de connaissances, d'expériences, de principes, de normes, et de contraintes. Elle facilite l'anticipation pour générer des prévisions par l'action dynamique, et est considérée comme un processus qui nécessite la conscience de soi dans une situation donnée [Lini, 2013].

[Chen *et al.*, 2015a] [Chen *et al.*, 2015b] proposent un modèle de processus de conception, inspiré du modèle Fonction, Behavior, System (FBS) [Gero & Kannengiesser, 2004]. Ce modèle conceptuel comporte une transformation du besoin du concepteur vers une description fonctionnelle du produit (Figure 2.3).

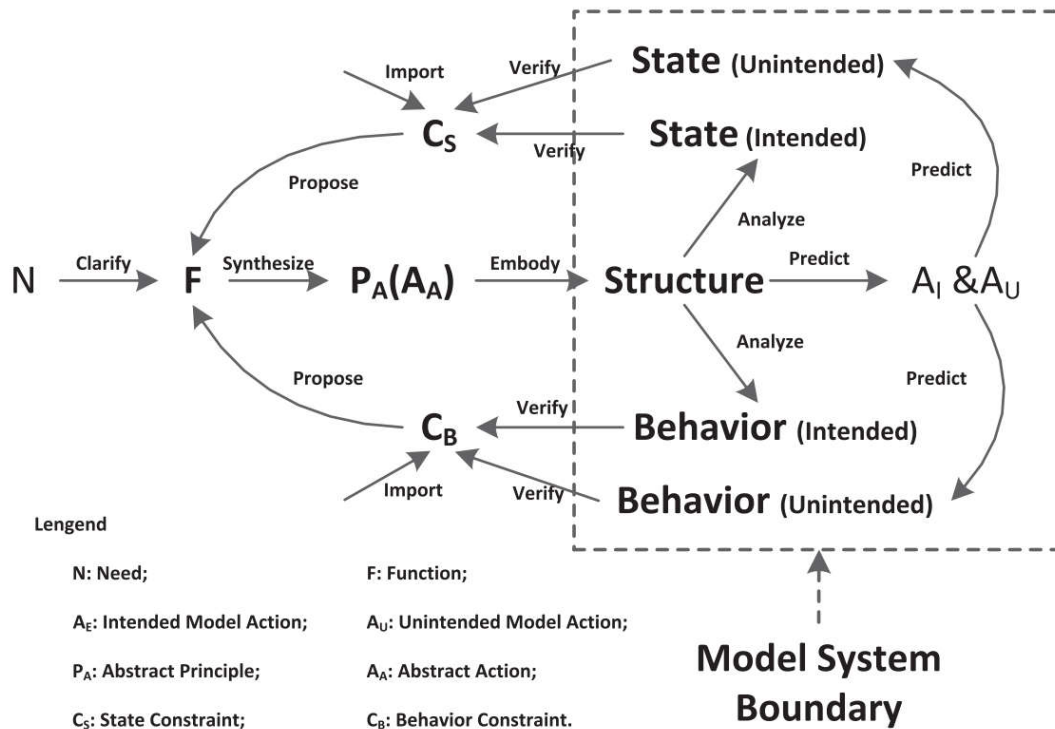


Figure 2.3 – Modèle Need-Function-Principle-System - NFPS par [Chen *et al.*, 2015a] [Chen *et al.*, 2015b]

Un concepteur peut désirer représenter une fonction avec le changement d'un environnement objectif, d'un état problématique à un état souhaitable. L'environnement est représenté par des éléments de modèles avec leurs comportements, leurs actions et les relations qu'ils ont entre eux. Après cette phase, le processus de conception peut alors passer du monde intentionnel au monde semi-objectif. Lorsqu'un besoin est transformé en une fonction (soit une description semi-objective), un concepteur ou un outil logiciel peut utiliser certaines connaissances physiques de base telles que les modèles d'actions ou de comportements, plutôt que l'expérience subjective, et ce, afin de générer un système basé sur l'un de ces modèles pour répondre au besoin. Ainsi, l'intention de conception est interprétée par l'ordinateur pour faciliter la prise de décisions le plus tôt possible [Wieggers & Vergeest, 2001], en allouant des connaissances pertinentes et contextualisées dans le processus de conception [Markus M. L., 2001].

2.2.2.2/ SENSIBILITÉ AU CONTEXTE

Au-delà de la compréhension du concept de l'intention de conception, les travaux de recherche dans la littérature se sont aussi portés sur la compréhension du contexte et de l'interprétation de sa sensibilité, afin d'en capturer l'intention. Le concept en question consiste à capturer et diffuser l'information au bon moment et à la bonne personne. Dès lors, plusieurs auteurs proposent et réaffirment ces concepts, tels que :

La « *sensibilité au contexte* » ou « *sensible au contexte* » dans le jargon informatique, utilise le contexte pour fournir des informations et/ou des services adaptés aux tâches de l'utilisateur [Abowd *et al.*, 1999].

Le système « *sensible au contexte* » est utilisé pour désigner les systèmes qui utilisent le contexte, afin de fournir des informations et/ou des services pertinents à l'utilisateur, lorsque la pertinence dépend de la tâche de l'utilisateur [Dey, 2000] [Dey, 2001].

Un système est « *sensible au contexte* » s'il peut extraire, interpréter et utiliser des informations contextuelles et adapter sa fonctionnalité au contexte d'utilisation actuel [Byun & Cheverst, 2004].

La « *sensibilité au contexte* » est la capacité de décrire le sens et d'agir sur l'information dans l'environnement de l'ordinateur tels que l'emplacement, le temps, la température ou l'identité de l'utilisateur [Lee & Lee, 2007].

Pour développer la « *sensibilité au contexte* », il faut établir des bases principales pour élaborer un système sensible au contexte (**Context-Sensitive System - CSS**) en suivant un processus de conception. Les auteurs [Viera *et al.*, 2011] apportent leur contribution à la définition relative à la manière de modéliser un CSS. En effet, les auteurs définissent le système pour :

- Fournir une feuille de route et soutenir le concepteur lors du démarrage d'un nouveau projet ;
- Proposer une séparation claire des activités contextuelles, en créant un nouveau rôle dans l'équipe de développement logiciel : le « *concepteur de contexte* » ;
- Souligner la nécessité de travailler avec les artefacts existants lors de la conception (par exemple : les exigences, les modèles conceptuels, la logique métier), au lieu de partir de zéro ;
- Couvrir les principales activités liées à la conception, en fournissant des directives pour indiquer les artefacts d'entrée/sortie et un moyen systématique pour exécuter chaque activité.

2.2.2.3/ CONCEPTION POUR X

La sensibilisation du concepteur est effectuée en amont du processus de conception afin d'anticiper la diffusion d'information dans la « *conception pour X* ». Dès lors, le concept de « *conception pour X* » a émergé (**Design For X - DFX**) et (**Design To X - DTX**) [Huang, 1996] [Andreasen and Mortensen, 1997] [Kuo *et al.*, 2001] [Holt & Barnes, 2009], et constitue un ensemble de règles permettant la prise en compte par le concepteur des contraintes issues de certains métiers impliqués tout au long du cycle de vie du produit (assemblage, fabrication, maintenance, etc.) ou encore d'objectifs ciblés, tels que les coûts (Design To Cost), les standards (Design To Standards), etc. L'efficacité de la « *conception pour X* », réside dans la réduction des temps de cycle de développement, pour améliorer la qualité et la productivité de l'acte de conception. Il permet par exemple de :

- Réduire le temps de mise sur le marché ;
- Réduire les coûts ;
- Minimiser l'impact environnemental ;
- Réduire le nombre de composants.

Une partie de cette méthodologie [Dagleish *et al.*, 2007] [Barnes *et al.*, 2004], permet d'anticiper l'impact d'un métier ou d'un objectif sur chaque étape du cycle de vie du produit. Le développement du DFX jette les premières bases d'une approche proactive, dans la mesure où elle va impacter le produit en phase amont du processus de conception, à partir de situations ou contextes futurs [Barnes, 1999]. La « *conception pour X* » peut se décliner sous plusieurs composantes, permettant d'atteindre des objectifs de coût, de standardisation, etc., et aussi facilitant la réduction des temps

de développement, tout en améliorant la qualité, en réduisant les erreurs de conception, etc. Il s'agit par exemple de (Figure 2.4) :

- La conception pour la fabrication et l'assemblage (**Design For Manufacture and Assembly - DFMA**) ;
- La conception pour le démontage (**Design For Disassembly - DFD**) ;
- La conception pour la fiabilité (**Design For Reliability - DFR**) ;
- La conception pour la maintenabilité (**Design For Maintainability - DFM**) ;
- La conception pour l'environnement (**Design For the Environment - DFE**) ;
- La conception à coût objectif (**Design To Cost - DTC**) ;
- ...

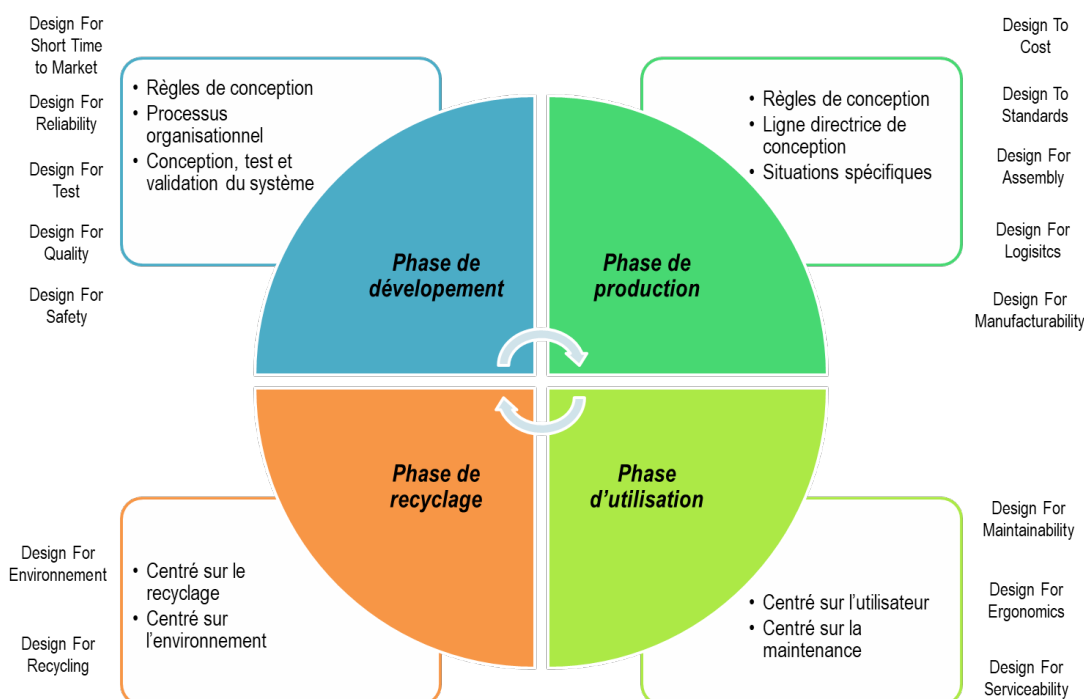


Figure 2.4 – Les méthodologies employées en conception pour X, tout au long du cycle de vie du produit selon [Bralla, 1996] [Pahl & Beitz, 1996] [Pahl *et al.*, 2007]

Une composante importante de la « *conception pour X* », concerne la conception pour la fabrication et l'assemblage (DFMA). Cette approche vise à préparer le produit pour son futur assemblage et sa future fabrication en réunissant un ensemble d'informations, de connaissances, de procédures analytiques et de recommandations, afin de favoriser la réduction des erreurs de conception dans ce domaine [Boothroyd *et al.*, 1994] [Boothroyd *et al.*, 2011]. Aujourd'hui, des approches existent et sont basées sur des lignes directrices afin de présenter au concepteur des méthodes fondées sur l'évaluation qualitative [Andreasen, 2000] et quantitative [Rampersad, 1994]. Ces dernières constituent une aide pour la conception des composants en y incorporant des règles de fabrication et d'assemblage structurées dépendant des procédés d'obtention sélectionnés. Par exemple, pour faciliter l'intégration d'auto-positionnement, il est nécessaire de minimiser les directions d'insertion, et de fournir les caractéristiques d'orientation.

La composante appelée conception pour l'assemblage (**Design For Assembly - DFA**) a permis, la mise en place de méthodes pour générer des séquences d'assemblage directement à partir d'une maquette numérique 3D. Un autre exemple consiste à introduire un raisonnement basé sur

la géométrie, afin de calculer le degré de symétrie des composants pour en faciliter l'assemblage [Jared *et al.*, 1994]. L'application des approches de « *conception pour X* », favorise ainsi, en phase amont du processus de conception, la prise de conscience du concepteur sur l'impact de ses choix de conception sur la fabricabilité et l'assemblabilité des composants de sa solution.

2.2.3/ MODÉLISATION DU CONTEXTE

2.2.3.1/ MODÈLE DE ZIMMERMANN ET AL.

La modélisation du contexte comprend plusieurs paramètres à connaître, servant de fil directeur pour le développement d'applications sensibles au contexte. Les auteurs [Zimmermann *et al.*, 2007] contribuent à définir le contexte d'individualité de l'entité. Ceux-ci proposent une base pour l'utilisation de systèmes sensibles au contexte, afin de mettre l'accent sur ses propriétés dynamiques. Ces auteurs présentent cinq catégories fondamentales de contexte qui déterminent l'espace de conception tel que décrit dans le modèle représenté à la Figure 2.5.

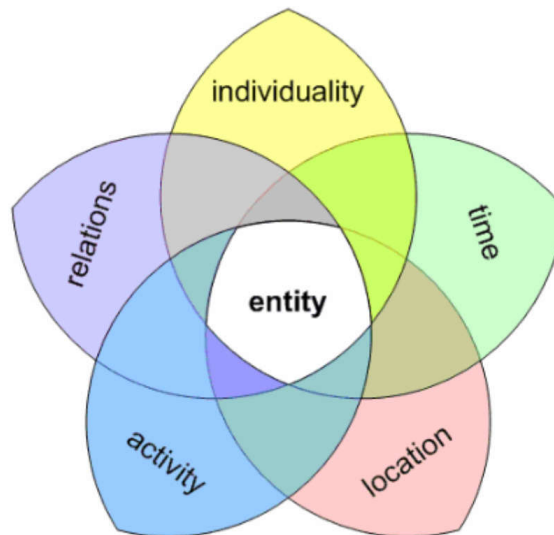


Figure 2.5 – Définition du contexte selon [Zimmermann *et al.*, 2007]

La Figure 2.5 traduit cinq types de contexte suivants :

- **Contexte d'individualité** : il fournit l'accès à des informations sur l'entité. Lesquelles sont répertoriées au rang de quatre catégories d'entités : naturelles, humaines, artificielles et de groupes ;
- **Contexte de temps** : le temps est un élément important pour la compréhension humaine, et pour favoriser la classification. Le contexte de temps facilite les calculs mathématiques lorsqu'il s'agit d'effectuer des comparatifs en représentant les intervalles de temps et d'analyse de l'historique des interactions ;
- **Contexte d'emplacement** : l'emplacement constitue l'un des paramètres exploités dans les systèmes sensibles au contexte. Il fournit des informations spatiales sur l'emplacement exact d'un système (par exemple : distance, angle, etc.), tel que le mets en œuvre le système de positionnement global (**Global Positioning System - GPS**). Une entité possède toujours un emplacement physique qualitatif, qui peut être représenté par des emplacements quantitatifs différents, en même temps que plusieurs emplacements virtuels ;
- **Contexte d'activité** : il recouvre les activités présentes et futures de l'entité. Ces activités

sont décrites par des objectifs, des tâches, et des actions explicites, ce qui lui permet de s'adapter en fonction du besoin nécessaire ;

- **Contexte de relation** : il capture les relations qu'une entité a établies avec d'autres. Ces entités sont des personnes, des choses, des dispositifs, des services ou des informations. Cet ensemble construit ainsi la structure de l'entité, faisant partie intégrante du contexte. La catégorie de relation est subdivisée en relations sociale, fonctionnelle et de composition.

Dans le domaine du génie logiciel, cette définition est proposée pour combler l'écart entre les utilisateurs et les développeurs, car elle permet une compréhension naturelle du concept de contexte pour les utilisateurs, tout en facilitant l'ingénierie informatique pour les développeurs de logiciels.

2.2.3.2/ MODÈLE DE VIERA ET AL.

Avec l'utilisation croissante du contexte dans les systèmes informatiques, il y a une nécessité de soutenir les concepteurs sur la construction de leurs applications pour y inclure le concept de contexte. [Viera *et al.*, 2011] proposent un méta-modèle au sens large qui tend à définir une terminologie liée au contexte et au système sensible de contexte, en indiquant les concepts et les activités impliqués dans le contexte.

Deux axes sont proposés :

- **Théorique** : sur l'avancée des travaux de recherche de la modélisation du contexte et du système sensible au contexte ;
- **Pratique** : sur la conception et la spécification du cadre **CEManTIKA** (Contextual Elements Modeling and Management through Incremental Knowledge Acquisition), en intégrant une partie statique d'un système sensible au contexte (pour la modélisation de la structure du contexte) et une partie dynamique (pour supporter les variations de comportement dans le contexte). Selon [Viera *et al.*, 2011], les instanciations provenant de la préférences des experts se font au travers d'un projet appelé **ICARE** (Intelligent **C**ontext **A**wareness for **R**ecommending **E**xperts).

Les principaux concepts de ce méta-modèle sont représentés par la Figure 2.6 et ci-après :

- **L'entité contextuelle (ContextualEntity)** : est une représentation concrète d'un objet du monde réel. Utilisée pour classer et décrire les individus (par exemple Personne, Professeur, Mission, etc.) ;
- **L'élément contextuel (ContextualElement)** : décrit une propriété utilisée pour caractériser une entité contextuelle. Il est identifié à partir de l'ensemble des attributs et des relations associés à une entité ;
- **Le focus (Focus)** : permet de déterminer ce qui doit être considéré dans un contexte, quand la personne exécute une tâche ou une activité, l'accent est mis sur cette tâche. Il est ainsi composé de tâches et d'agents, dont ces derniers ont un rôle défini pour cette tâche ;
- **La source du contexte (ContextSource)** : identifie comment se fait l'acquisition de l'élément contextuel, selon des sources externes.

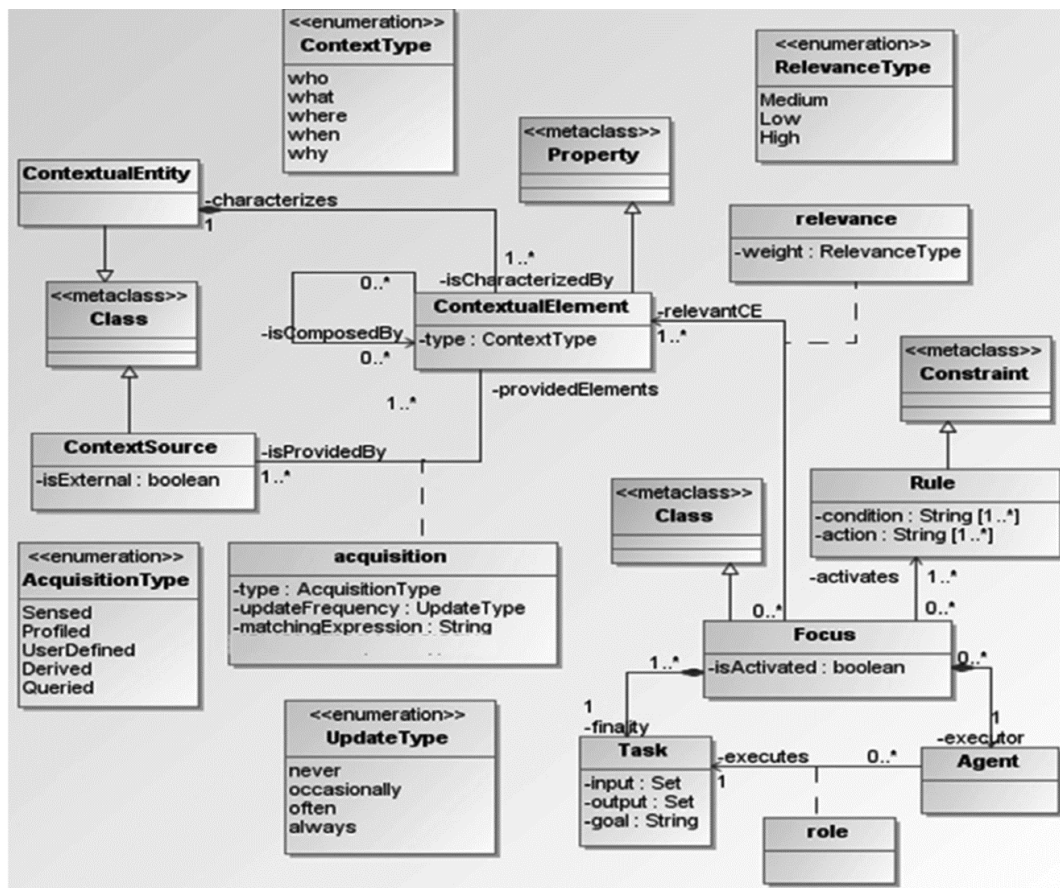


Figure 2.6 – Meta-modèle du contexte par [Viera *et al.*, 2011]

2.2.3.3/ MODÈLE DE NADOVEZA ET KIRITSIS

Un modèle de contexte basé sur l'ontologie a été proposé par [Nadoveza & Kiritsis, 2014], dont le modèle en question (Figure 2.7) s'inspire des travaux de [Zimmermann *et al.*, 2007] pour représenter le contexte selon les informations provenant de fonctions (Feature) en se focalisant sur l'utilisateur et son contexte métier.

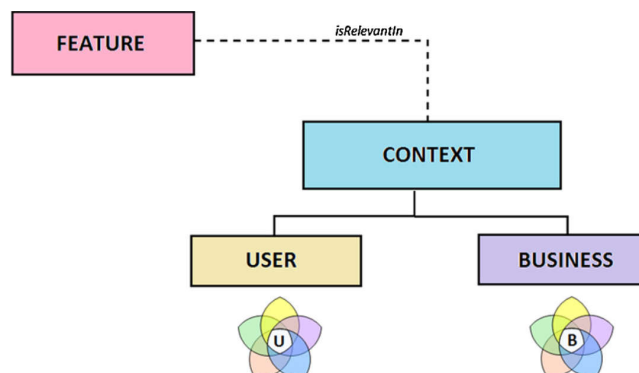


Figure 2.7 – Modèle conceptuel du contexte par [Nadoveza & Kiritsis, 2014]

Les auteurs s'inspirent du contexte métier en production pour définir une ontologie afin de diffuser, au bon moment, la bonne information. Ils proposent ainsi l'infrastructure du modèle avec des classes génériques et des propriétés pour le développement d'applications manufacturières. Divisé en deux parties, il s'agit de la définition (i) de la description du contexte de l'utilisateur de son statut, de sa localisation, de son rôle, (ii) tout en définissant le contexte métier avec les ressources allouées et l'activité.

2.2.3.4/ MODÈLE DE DHUIEB ET AL.

L'un des récents modèles de contexte, concerne celui proposé par [Dhuiieb *et al.*, 2016]. Ces derniers proposent un modèle spécifique pour améliorer l'interprétation des connaissances dans les systèmes à base de connaissances, en suggérant une structuration multi-échelle de la connaissance avec un modèle conceptuel de contexte (Figure 2.8). [Dhuiieb *et al.*, 2016] prennent en considération :

- **Le contexte opérationnel** : pour la description de l'activité et des ressources allouées ;
- **Le contexte organisationnel** : pour la relation entre les acteurs et l'équipe ;
- **Le contexte centré sur l'utilisateur** : pour la définition du profil de la personne, notamment son niveau d'expertise.

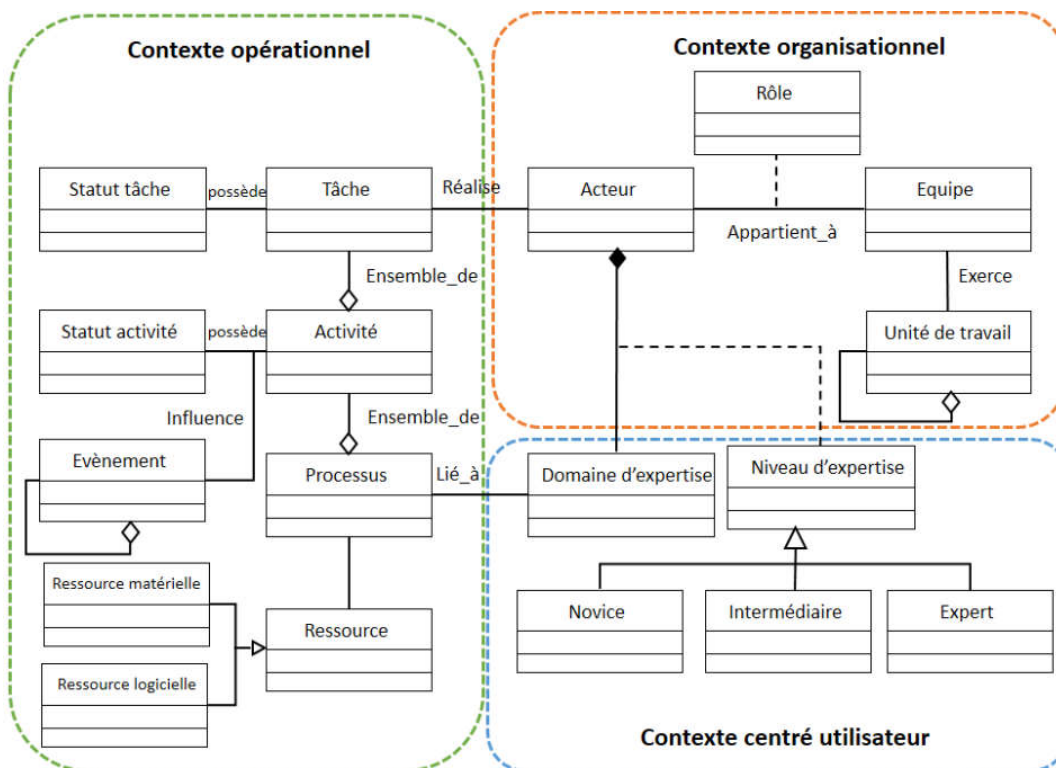


Figure 2.8 – Modèle de contexte proposé par [Dhuiieb *et al.*, 2016]

Ces travaux ainsi issus des domaines de recherches de la gestion des connaissances, de l'information ubiquitaire, et de la sensibilité au contexte, ont permis de proposer une approche cognitive, pour structurer la connaissance multi-échelle en modélisant le contexte, et en établissant un map-

page¹ du contexte avec la connaissance.

2.2.4/ DESCRIPTION ET GÉNÉRATION DES FUTURS CONTEXTES

La plupart des modèles présentés sont conformes à l'interprétation du contexte. Il est intéressant d'aborder la génération et l'anticipation du futur contexte. A ce stade, l'objectif est de faciliter la recherche de connaissances adéquates, dans le but d'anticiper à la fois le contexte mais également le futur contexte. Tout d'abord, nous distinguons deux définitions : la prédiction et l'anticipation.

La prédiction permet au processus d'anticipation de réagir au futur contexte.

Plusieurs concepts de prédiction et d'anticipation sont apparus dans l'histoire de la science dans le but de penser et de résoudre les problèmes scientifiques (par exemple : les prévisions météo, l'anticipation du besoin client, l'interprétation de la future position des astres, etc.). Ces réflexions ont permis le changement de comportement d'une personne/d'application de manière qualitative dans les domaines suivants : mathématiques, biologique, physique, philosophique et autres.

2.2.4.1/ DESCRIPTION DE LA PRÉDICTION

Depuis des décennies, l'émergence de la notion de prédiction a été discutée par Leibniz, le père de l'ère informatique [Leibniz, 1976]. Ce dernier anticipe la fondation de la société de l'information, dont [Nadin, 2010] entreprit les réflexions nécessaires pour comprendre l'anticipation dans les systèmes. Le principe fondamental de la prédiction est le *déterminisme* (dépendant ainsi d'une condition initiale à un instant t). Il décrit le principe de *causalité* (dépendant de son état passé ou de sa condition actuelle). Quant à Newton et Leibniz, ceux-ci décrivent une approche méthodologique pour prédire le mouvement des planètes et des satellites en connaissant le passé et le présent, par l'intermédiaire de mathématiques appliquées [Newton, 2008].

La théorie de la prédiction est expliquée par [Lestocart, 2008], qui interprète cette théorie, par la dérive de la mécanique statistique, également appelée en physique « *hypothèse ergodique* » du grec *ergon* : *énergie*. D'autres théories ont ainsi évoluées vers la théorie de la prédiction, dont les fondamentaux ont évolués en 1868 vers des notions irréversibles (avec l'émergence de la théorie cinématique des matériaux) aux notions probabilistes (en introduisant la théorie des probabilités basée sur la loi fondamentale de la physique, impliquant, la loi thermodynamique, le problème d'irréversibilité et la notion de temps). Conduisant aux structures dynamiques (causales et déterministes), dont [Lestocart, 2008] pose l'une des questions pour définir l'impossibilité d'un observateur, de prédire sans calcul, un état futur d'un système. Le comportement du système dynamique est donc non-linéaire et doit être représenté par des équations différentielles stochastiques (faits aléatoires).

En 1941, des réflexions provenant du mathématicien et physicien [Wiener, 1964] apparaissent, sur la systématisation de la prédiction « *optimale* ». Ses réflexions offrent la possibilité de prédire un état futur à l'aide d'équations en se basant sur l'état présent. Il faudra attendre 1975 pour que la théorie de la prédiction soit associée au processus stochastique particulier appelé « *la chaîne de Markov* ». Laquelle se base sur l'état présent, pour le calcul des séquences de variables afin d'estimer l'état futur. Elle est automatisée via le modèle de prévision proposé par [Knight, 1975], qui suggère différents états probables.

1. Processus de conversion

Pour conclure sur ces paragraphes dédiés aux définitions, la prédiction est définie comme « *une représentation d'un événement futur spécifique* » [Pezzulo *et al.*, 2008], qui propose depuis plusieurs décennies, des modèles prédictifs, des modèles d'aide à la décision ou dédiés aux raisonnements qui ont pris différentes formes telles que les algorithmes génétiques, la logique floue, les réseaux de neurones ou encore les réseaux bayésiens, avec chacun leurs spécificités.

2.2.4.2/ DESCRIPTION DE L'ANTICIPATION

La prédiction converge vers l'anticipation qui est considérée comme un processus métacognitif² ou comme une action dynamique qui fait suite aux prédictions générées. C'est donc un « *processus métacognitif qui s'inscrit dans une dynamique particulière : un processus nécessite tant une conscience de soi que de la situation, des capacités amnésiques³ immédiates et de long terme, dans un but d'économie des ressources cognitives* » [Lini, 2013].

En philosophie, [Von Glasersfeld, 1998] décrit le comportement d'un système selon trois types d'anticipations :

- **Sous la forme d'attentes implicites** qui sont une condition préalable à de nombreuses actions ;
- **Comme l'attente d'un événement futur spécifique**, basé sur l'observation d'une situation actuelle ;
- **L'anticipation d'un événement, d'une situation ou d'un but souhaité**, dont la tentative est d'atteindre la cause générée.

Depuis de nombreuses années, des travaux de recherche sont axés sur le développement de systèmes d'anticipation. Dans le domaine de la biologie, ces systèmes facilitent la compréhension de modèles prédictifs en interaction avec leurs environnements, en vue de changer leur état réel [Rosen, 2012]. Les systèmes d'anticipation peuvent également être définis comme des systèmes dont l'état actuel est déterminé non seulement par un état passé, mais aussi par d'éventuels états futurs [Nadin, 2010]. En conséquence, l'anticipation est « *considérée comme une action dynamique associée à un processus de prise de décision orienté vers l'avenir, qui nécessite une conscience de soi dans une situation donnée* » [Pezzulo *et al.*, 2008]. Fréquemment utilisée dans le domaine de l'intelligence artificielle, l'anticipation facilite les prises de décision dans l'interprétation du langage entre l'humain et la machine. Elle fait partie des sciences cognitive qui facilitent la connaissance de l'état futur dans une situation donnée.

Les systèmes apprenants (machines et/ou solveurs intégrant des mécanismes d'apprentissage) tendent vers des systèmes basés sur l'évolution d'agents, en intégrant leur curiosité et leur permettant de revenir en arrière, puis de changer le poids du réseau neuronal. Ceci permet aux agents de surveiller les états du système, en analysant les écarts entre l'état actuel et l'état futur [Schmidhuber, 1991]. Pour définir ces états dans une situation donnée, cette dernière se détermine comme « *un ensemble d'entités et d'interactions qui caractérisent de façon globale l'environnement externe dans lequel l'acteur mobilise sa compétence* » [Belkadi *et al.*, 2013]. La situation en conception est décrite par [Reymen *et al.*, 2002] comme étant la combinaison de l'état du produit à concevoir, du processus et du contexte de conception à un moment donné. Ainsi, anticiper dans

2. En psychologie, la métacognition est la « *cognition sur la cognition* ». Autrement dit, la métacognition consiste à avoir une activité mentale sur ses propres processus mentaux, c'est-à-dire « *penser sur ses propres pensées* » — <https://en.wikipedia.org/wiki/Metacognition>

3. Art de cultiver la mémoire. Moyen facilitant la conservation ou le rappel des souvenirs. Propre à aider la mémoire. Qui concerne la mémoire — <http://dictionnaire.education/fr/mnesique>.

une situation donnée permet de définir la capacité d'action sur la préparation de certains effets ou de futurs états dans l'environnement, afin de voir loin, pour prendre une décision à un moment judicieux.

2.2.5/ MODÉLISATION DU FUTUR CONTEXTE

2.2.5.1/ APPROCHE DE MAYRHOFFER

Le développement d'un cadre méthodologique est proposé par l'auteur [Mayrhofer, 2004] pour définir l'architecture du futur contexte, concernant l'utilisation d'appareils connectés. Ce dernier propose deux contributions sur la reconnaissance de la prédiction dans le contexte, à la fois pour (i) comprendre le contexte et pour (ii) prédire le futur contexte, afin d'anticiper de manière proactive l'information. L'auteur suggère deux éléments portés sur le développement et l'implémentation de l'architecture (Figure 2.9), qui doit comprendre :

- Une flexibilité définie entre plusieurs étapes via des interfaces simples, pour rendre l'architecture appropriée afin de faciliter l'arbitrage des capteurs et de la large gamme d'appareils ;
- Une séparation claire des préoccupations, pour les capteurs, les extracteurs de caractéristiques, les classificateurs, les prédicateurs et l'interaction des utilisateurs, sont totalement indépendants et interchangeables.

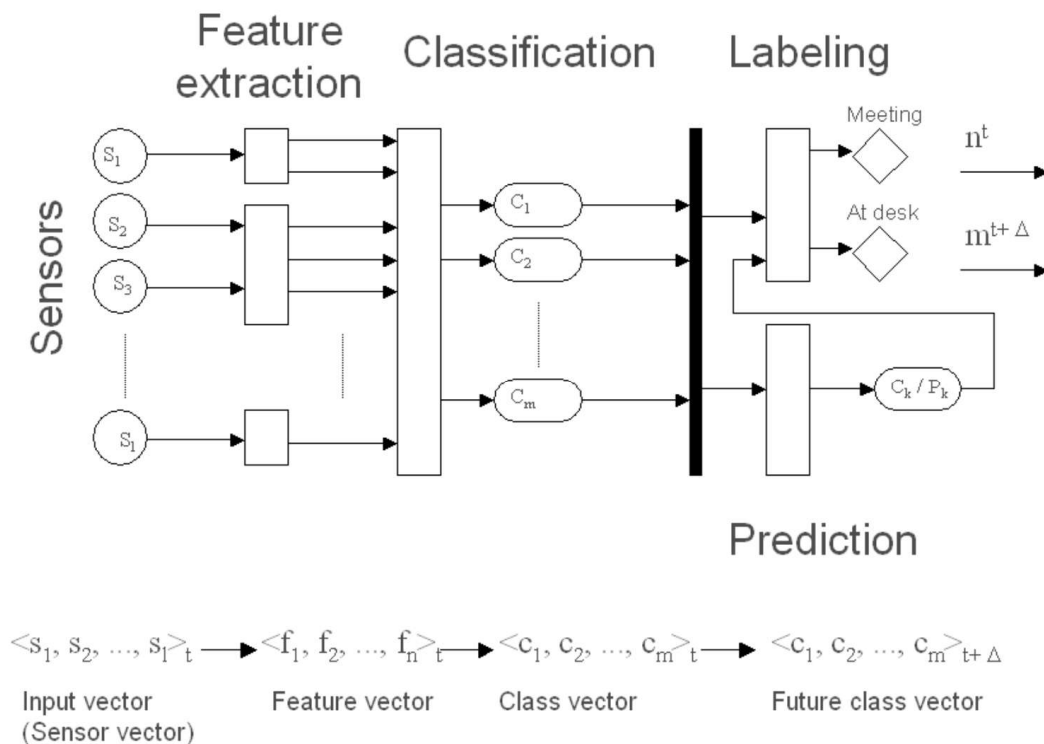


Figure 2.9 – Cadre méthodologique proposé par [Mayrhofer, 2004]

L'autre partie de la contribution comprend l'implémentation de l'architecture, avec :

- **L'utilisation de vecteurs caractéristiques hétérogènes pour la reconnaissance de contextes** : cette partie introduit une méthode, pour utiliser une large classe de fonction-

nalités afin de dériver le contexte ;

- **Prévision de contextes** : le concept prédit l'ensemble du contexte à son niveau et identifie localement les algorithmes issus des appareils fournissant les informations ;
- **Développement d'un cadre multi-plateformes adapté aux dispositifs, avec des ressources limitées.**

2.2.5.2/ APPROCHE DE ZAMENOPOULOS ET ALEXIOUS

Dans le domaine biologique, les travaux de recherche sont centrés depuis des années sur la modélisation de l'anticipation [Rosen, 1985] [Rosen, 2012]. Cet auteur [Rosen, 1985] étudie les relations entre les systèmes pour gérer les traitements complexes. Il expose deux aspects sur sa vision de l'anticipation :

- Le premier concerne sa définition des systèmes d'anticipation qui ont influencé la compréhension contemporaine de l'anticipation. L'anticipation est couplée à la capacité d'un système à contenir un modèle de lui-même et/ou de son environnement. Cette capacité permet au système d'agir non seulement selon son histoire, mais aussi en réponse à des états possibles ou futurs ;
- Le second aspect consiste à modéliser des classes de systèmes appelées métabolisme réparateur (M, R). La Figure 2.10 montre le modèle (M, R) en question. Respectivement le symbole M est le modèle de prédiction qui influence selon les données d'entrée environnante, le modèle appelé Effecteur - E , qui a comme « effet » d'influencer l'objet du système - S , à savoir le système d'anticipation.

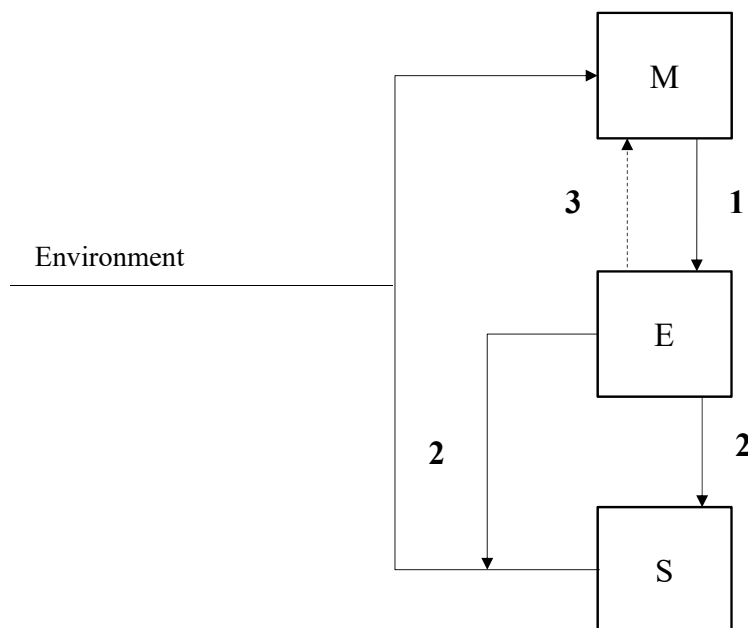


Figure 2.10 – Modèle basé sur la prédiction proposée par [Rosen, 1985]

D'une manière abstraite, les systèmes anticipatifs ont la capacité d'inclure le raisonnement humain dans la machine, afin de faciliter la prédiction pour évoluer et comprendre les futurs états. La capacité du système est basée sur la construction de théories et de modèles afin de représenter l'anticipation selon le besoin de l'utilisateur. Des divergences apparaissent sur la vision de l'anticipation proposée par Rosen. Celle de [Dubois, 1998] consiste à proposer une approche basée sur

le modèle de prédiction. Il introduit les concepts dits « *d'incursion* » (récursion implicite) et de son alternatif appelé « *d'hypercursion* » (une incursion avec de multiples solutions). L'approche consiste à décrire le système d'anticipation de Rosen nommé S , au travers d'équations, pour générer le modèle ou l'état futur du système selon le modèle prédictif M . L'état futur du système S et du modèle M est représenté dans le temps $t + \Delta t$, souvent considéré comme une fonction F . Le système S est considéré à un instant t et celui du modèle M à un stade ultérieur $t + \Delta t$ (Equation 2.1 et Equation 2.2) :

$$\frac{\Delta S}{\Delta t} = \frac{[S(t + \Delta t) - S(t)]}{\Delta t} = F[S(t), M(t + \Delta t)] \quad (2.1)$$

$$\frac{\Delta M}{\Delta t} = \frac{[M(t + \Delta t) - M(t)]}{\Delta t} = F[S(t), M(t + \Delta t)] \quad (2.2)$$

Le deuxième état est présenté avec le modèle d'environnement, noté E , dans le modèle prédictif du système S (Equation 2.3 et Equation 2.4) :

$$\frac{\Delta S}{\Delta t} = \frac{[S(t + \Delta t) - S(t)]}{\Delta t} = F[S(t), E(t + \Delta t)] \quad (2.3)$$

$$\frac{\Delta E}{\Delta t} = \frac{[E(t + \Delta t) - E(t)]}{\Delta t} = G[E(t), S(t + \Delta t)] \quad (2.4)$$

Les concepts, proposés par l'application de la méthode, permettent d'étudier et de développer une série de modèles formels, pour spécifier des équations d'ondes numériques ou la génération de fractales. Il est évident que l'anticipation ne se cantonne pas qu'au domaine biologique sur les systèmes vivants. La vision de l'anticipation est abordée avec les travaux de recherches dans la conception. [Zamenopoulos & Alexious, 2007] ont étudié cette approche en proposant un examen du concept d'anticipation et de son sens dans le contexte de conception. Les auteurs définissent les conditions pour identifier la conception comme un « *problème distinct* », sans s'engager dans une interprétation spécifique des processus, des connaissances ou des entités impliquées dans la conception. Ainsi, ils s'inspirent de la cartographie du système d'anticipation (M, R) établie par [Letelier *et al.*, 2006]. Le système vivant en question est capable d'anticiper, de se maintenir et de se rétablir lui-même (Equation 2.5) :

$$A \xrightarrow{f} B \xrightarrow{\Phi f} H(A, B) \xrightarrow{\beta b} H(B, H(A, B)) \quad (2.5)$$

Le modèle relationnel est la représentation de la propriété de réplcation engendrée par le métabolisme avec celui des réparations (ou fonctions). Soit $f : A \rightarrow B$, désigne une composante fonctionnelle d'un système qui transforme l'entrée A en sortie B . Cette composante qui représente le métabolisme. $H(A, B)$ est l'ensemble de toutes les transformations possibles de A en B , dont $\phi f : H(A, B)$ est la composante de réparation. Ainsi, le mécanisme de réplcation est dérivé des applications du métabolisme et de la réparation, représenté par $\beta b : H(A, B) \rightarrow H(B, H(A, B))$. Le mécanisme est dérivé par [Zamenopoulos & Alexious, 2007] pour introduire la notion d'héritage dans la conception. La notion d'intention du concepteur est alors évoquée, au travers d'un ensemble cohérent pour comprendre ce qu'il veut et comment l'anticiper. βb est l'état de réplcation du mécanisme. Le concept est hérité de la fonction dans l'ensemble de $H(A, B)$ sur B puis transformé par la fonction

de f . Le mécanisme de réplique est hérité de toutes les données de φf , sur la base de B . En héritant de βb ces données sont capitalisées dans le système $H(B, H(A, B))$ (Figure 2.11).

$$\begin{array}{ccc}
 H(B, H(A, B)) \cong A & \xrightarrow{f} & B \\
 & \swarrow \beta_b & \downarrow \varphi_f \\
 & & H(A, B)
 \end{array}$$

Figure 2.11 – Modèle basé sur la prédiction par [Zamenopoulos & Alexious, 2007]

Dans le domaine de la conception, cette vision d'anticipation facilite la capacité du concepteur à générer une solution technique tout en anticipant la solution par la capture de ses intentions de conception. L'intérêt réside dans la prédiction, afin de voir loin pour développer un produit durable en minimisant les erreurs de conception.

2.2.5.3/ APPROCHE DE DEMOLY ET AL.

Les auteurs [Demoly *et al.*, 2011] ont proposé une approche spécifique pour générer plusieurs possibilités de futurs contextes. Ces futurs contextes sont ici représentés par la génération de séquences d'assemblage, basée sur l'algorithme **ASDA** (**A**ssembly **S**equences **D**efinition **A**lgorithm) présenté dans la Figure 2.12. Celui-ci est fondé sur un modèle mathématique en intégrant les règles DFA, pour faciliter les décisions issues de l'ingénierie sur la définition de la structure du produit. Le cadre proposé permet d'une part, de générer des séquences d'assemblage et de les évaluer selon des critères de décisions, et d'autre part, d'apporter au concepteur des informations supplémentaires sans avoir de géométrie définie au stade préliminaire de la conception.

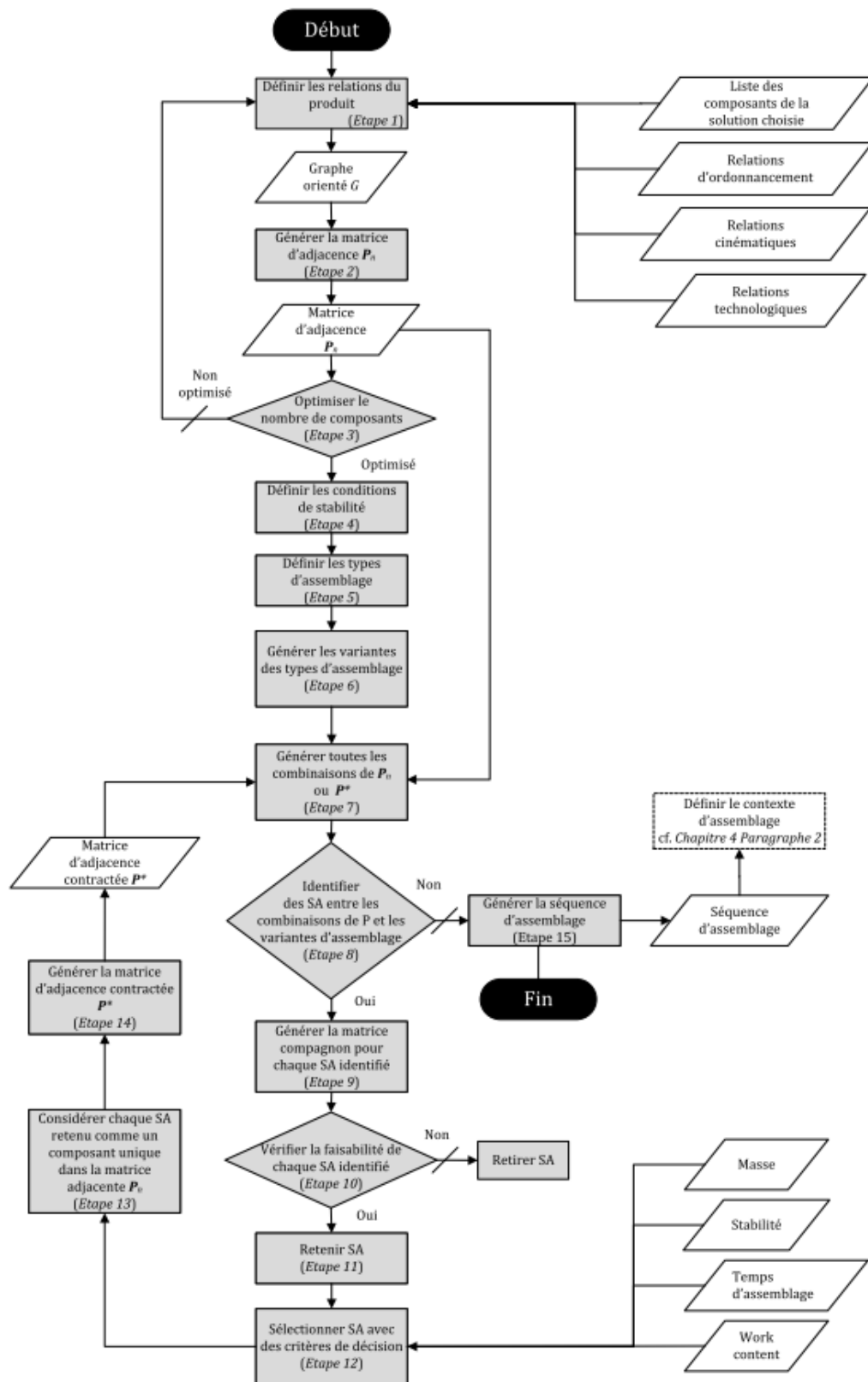


Figure 2.12 – Approche proposée par [Demoly, 2010]

2.2.5.4/ APPROCHE D'ANDOLFATTO

Toujours dans le domaine de la conception pour l'assemblage, [Andolfatto, 2013] présente une démarche dans le but de construire un avant-projet de gamme d'assemblage paramétrée, pour définir et allouer des tolérances aux variations géométriques (Figure 2.13). Fondé sur un modèle structuro-fonctionnel du produit, celui-ci est en lien avec les techniques d'assemblage.

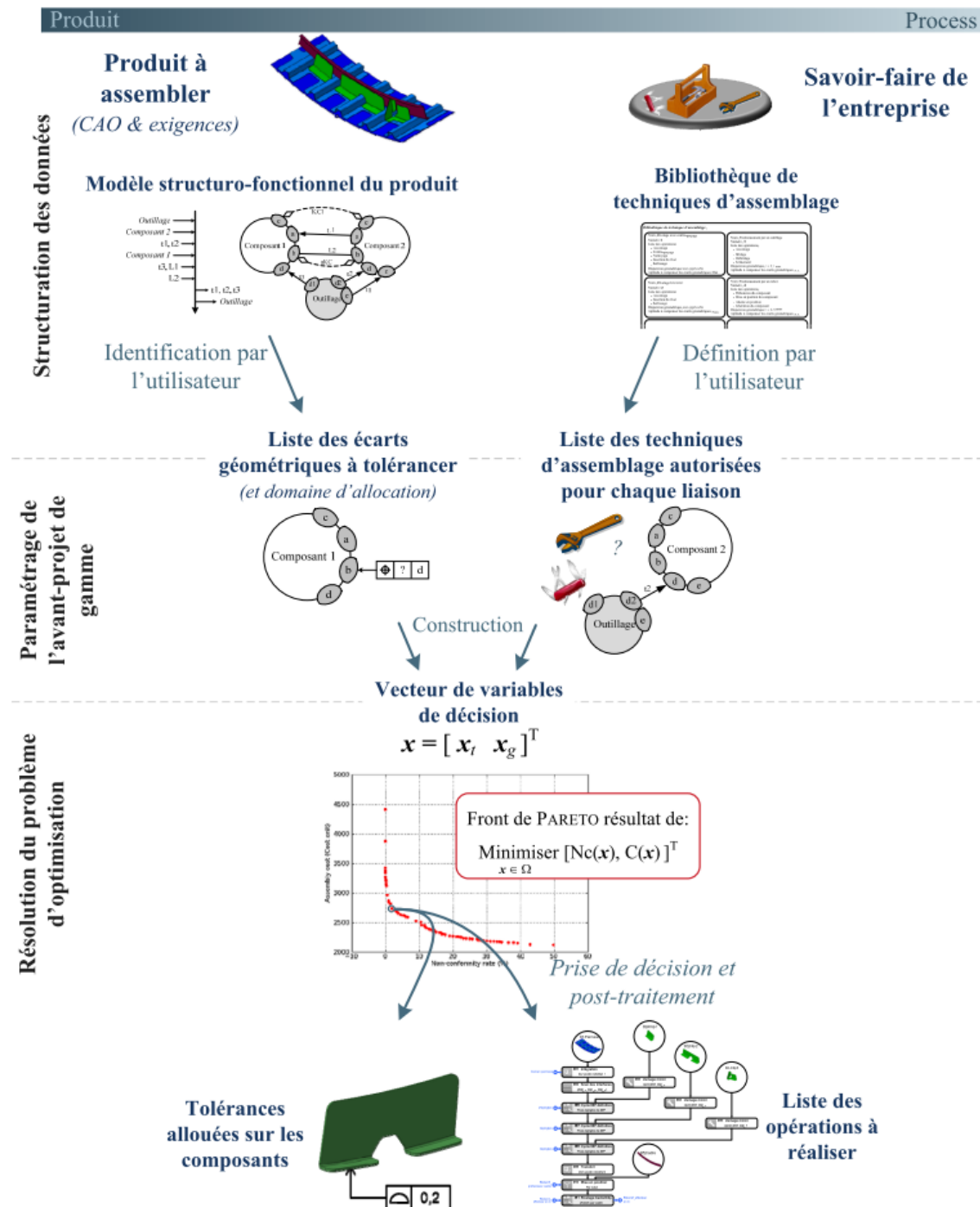


Figure 2.13 – Approche proposée par [Andolfatto, 2013]

Les travaux de l'auteur [Andolfatto, 2013] sont inspirés de [Mathieu & Marguet, 2001] [Marguet, 2001], sur la propagation des variations géométriques et de leur dispersion liées à la variété des techniques d'assemblage. La Figure 2.13, ci-dessus, représente la méthode proposée pour élaborer une gamme d'assemblage visant à minimiser le coût du produit assemblé en ciblant la réduction des non-conformités au cours de l'assemblage. Ainsi la définition du problème d'optimisation concerne la minimisation du coût, représenté par un front de Pareto. Ceci permet au concepteur de connaître le meilleur compromis pour les avant-projets de gamme issu de Pareto pour l'optimisation.

2.2.6/ BILAN ET POSITIONNEMENT

Cette section a introduit un état de l'art pour comprendre les notions inhérentes aux concepts de conception proactive, de contexte de conception, de futur contexte, d'intention, de sensibilité au contexte, qui vont nous servir à formaliser notre concept de « *contexte augmenté* ». Pour ce faire, nous avons puisé des références dans différents domaines relevant de la biologie, de la philosophie, des mathématiques, mais aussi de l'ingénierie mécanique. Cet état de l'art est synthétisé, à l'aide du Tableau 2.1 ci-dessous, intégrant les auteurs et leurs propositions, qui nous ont servi à appréhender les contours de ce que pourrait être notre concept de « *contexte augmenté* ».

Tableau 2.1 – Synthèse de la littérature scientifique dans le domaine des contextes actuels et futurs

Auteur(s)	Catégorie	Principe	Domaine d'application
[Andolfatto <i>et al.</i> , 2014]	Futur contexte	Élaboration de gamme provenant d'un futur contexte pour assurer sa solvabilité	Mécanique, production
[Demoly <i>et al.</i> , 2011]	Futur contexte	Génération du futur contexte, basé sur les relations produit	Mécanique, mathématique
[Dhuieb <i>et al.</i> , 2016]	Contexte	Structure les connaissances pour les adapter au contexte	Informatique, production
[Mayrhofer, 2004]	Futur contexte	Prédire le futur contexte pour faciliter la proactivité	Informatique
[Nadoveza & Kiritsis, 2014]	Contexte	Développement d'une ontologie basée sur le contexte	Informatique, production
[Viera <i>et al.</i> , 2011]	Contexte	Développement d'un système sensible au contexte	Informatique
[Zamenopoulos & Alexious, 2007]	Futur contexte	Modélisation de l'intention et de l'anticipation du concepteur	Mécanique, conception
[Zimmermann <i>et al.</i> , 2007]	Contexte	Définit le contexte et son utilisation	Informatique

Nous avons fait un état de l'art des différents travaux de recherches qui abordent les notions de contexte et de futur contexte pour la conception proactive. Sur cette base, force est de constater qu'il existe un besoin dans le domaine de la conception de produits, et qui consiste à mieux

connaître les intentions du concepteur et donc d'anticiper ses choix, afin d'instancier efficacement les connaissances requises. Notre positionnement au regard des autres travaux de la littérature scientifique est illustré dans la Figure 2.14. Il faut savoir que vis-à-vis de la communauté scientifique, l'ingénierie proactive intègre « *au plus tôt* » dans le processus de conception les connaissances du cycle de vie [Demoly *et al.*, 2013] décrites de manière formelle, pour proposer des solutions alternatives en contraignant les espaces de conception [Roucoules, 2007] [Martin & Roucoules, 2009]. Par ailleurs, le fait de générer des solutions alternatives liées au processus (futur contexte) permet à la fois d'évaluer, d'estimer les impacts et les conséquences d'une décision sur le produit [Bernard & Perry, 2003]. Ainsi, nous proposons de superposer, dans le domaine de la conception proactive, le « *contexte augmenté* » en articulant le contexte issu du présent, avec celui du futur, au travers d'un nouveau cadre méthodologique baptisé DAMON. La mise au point d'un « *contexte augmenté* » se focalise sur plusieurs points : l'utilisateur, l'activité, la relation entre les objets, le temps et l'emplacement ou la position de l'objet dans l'espace.

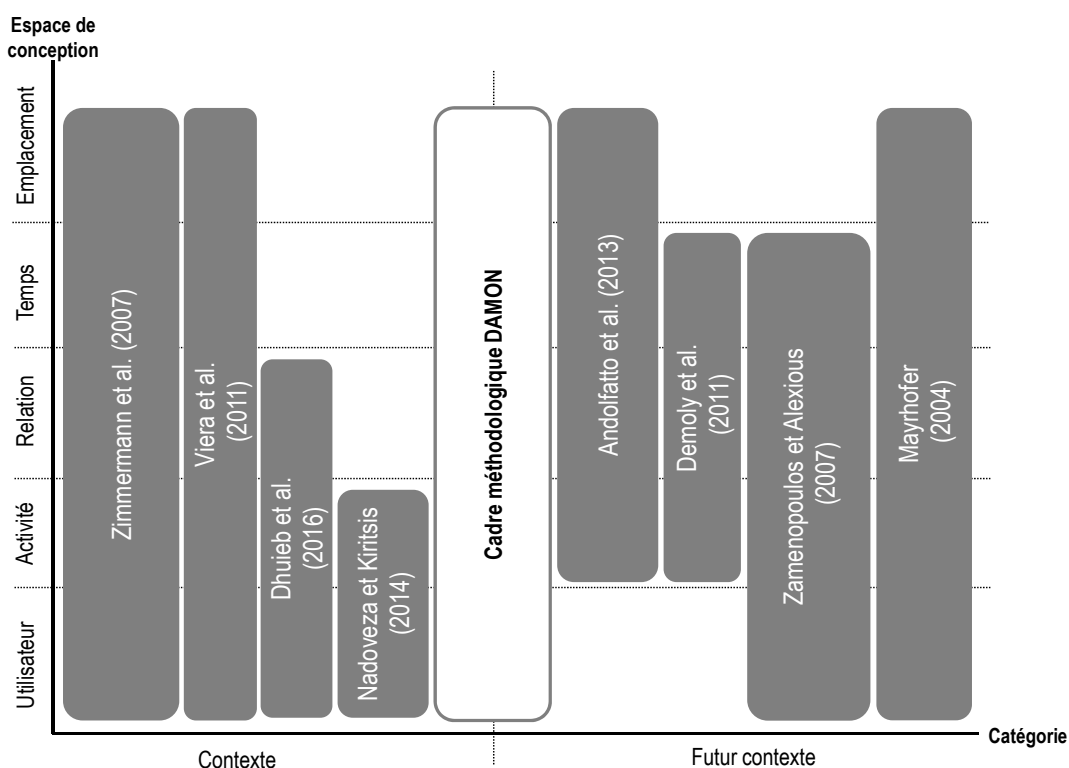


Figure 2.14 – Cartographie des domaines relatifs au contexte et futur contexte, représentant le « *contexte augmenté* »

2.3/ RÉUTILISATION DES CONNAISSANCES

Dans la mesure où l'interprétation du « *contexte augmenté* » a été passé en revue au regard des travaux de la littérature, nous allons aborder, dans la section suivante, l'état de l'art sur la réutilisation des connaissances dans le contexte.

2.3.0.1/ DÉFINITION ET OBJECTIF DE LA RÉUTILISATION DES CONNAISSANCES

La connaissance est décrite comme une notion tacite et explicite [Nonaka, 1994] [Nonaka & Takeuchi, 1995] [Newell, 1982]. Elle émerge de la mise en contexte de l'information, qui s'appuie sur les données. Autrement dit, elle est fortement imbriquée dans le sens où « *les données sont la matière première de l'information, et l'information est la matière première de la connaissance* » [Zins, 2007]. :

- La donnée est un objet réel et vérifiable, considéré comme quantitatif et objectif ;
- L'information donne un sens au contexte défini, permettant ainsi à l'utilisateur de prendre une décision à partir de connaissances induites ;
- Considérée comme une preuve viable, la connaissance est une information réelle qui est toujours vraie et vérifiée [Zins, 2007]. Dès lors, elle se caractérise comme de « *l'information qui prend un (ou plusieurs) sens dans un (ou plusieurs) contexte(s) à travers les acteurs de l'entreprise* » [Ermine, 2003].

En effet, la connaissance est répertoriée dans la pyramide appelée **DIKW** : **D**ata - **I**nformation - **K**nowledge - **W**isdom. [Ackoff, 1989] la décrit comme (Figure 2.15) :

- « *Quelques informations (I) valant une livre de données (D)* » ;
- « *Quelques connaissances (K) valant une livre d'informations (I)* » ;
- « *Quelques compréhensions (W) valant une livre de connaissances (K)* ».

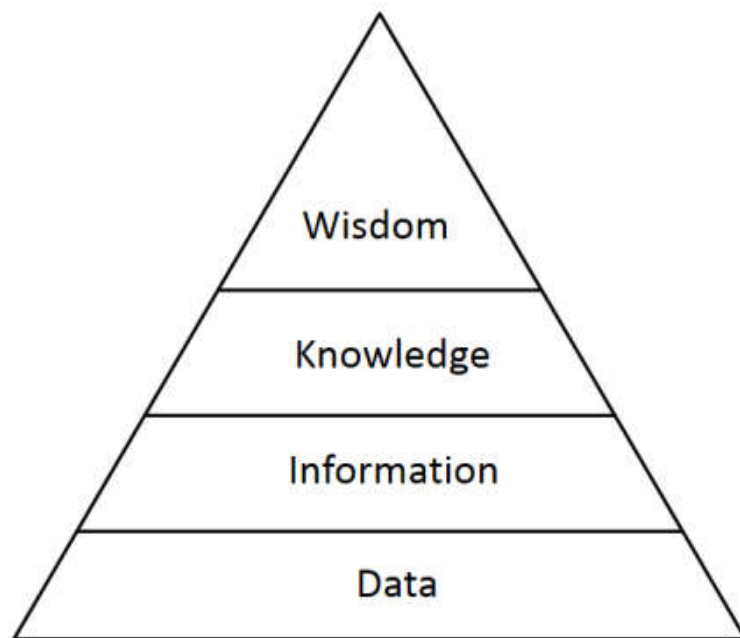


Figure 2.15 – La pyramide DIKW [Ackoff, 1989]

Capitaliser le savoir et le savoir-faire des entreprises afin de les partager, au travers de connaissances spécifiques, est un enjeu scientifique. [Song *et al.*, 2005] [Song *et al.*, 2006] décrivent la connaissance telle que **K = IESA** ou **K - Knowledge** représente la connaissance, **I** l'Information, les compétences avec **S - Skills**, le **E** comme l'Expérience et le **A** pour l'Attitude. La réutilisation des

connaissances dans les systèmes exploitent différentes approches basées sur le produit, le procédé et la tâche [Baxter *et al.*, 2007] [Baxter *et al.*, 2008]. D'autres catégories de connaissances sont employées (Figure 2.16), telle que les connaissances déclaratives (savoir-quoi), les connaissances de causalité (savoir-pourquoi), les processus intentionnels avec les connaissances procédurales (savoir-comment) et les connaissances contextuelles (savoir-quand et savoir-pourquoi) [Kim & Kim, 2011]. Puisque la connaissance peut être considérée comme une information dans son contexte, il en résulte que la représentation des connaissances dépendra à la fois du contenu mais également du contexte de l'information [Chandrasegaran *et al.*, 2013].

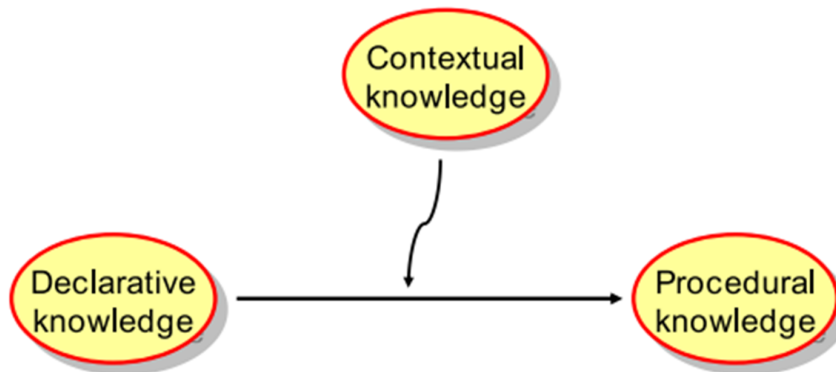


Figure 2.16 – Typologie des connaissances selon [Kim & Kim, 2011]

2.3.1/ MÉTHODES DE GESTION ET D'INGÉNIERIE DES CONNAISSANCES

Les approches présentées dans la section suivante nous aident à comprendre la manière dont les connaissances sont exploitées via les bases de connaissances (KB). A ce titre, le domaine de l'ingénierie des connaissances a reçu des efforts intenses et intéressants pour comprendre comment instancier les connaissances adaptées au contexte. L'efficacité dans leur utilisation consiste à pouvoir diffuser la bonne connaissance, à la bonne personne et au bon moment. Il convient d'abord de les capturer et de les exploiter de manière dynamique. La capture des connaissances est exprimée selon la norme ISO 15489 qui propose la méthodologie présentée à la Figure 2.17.

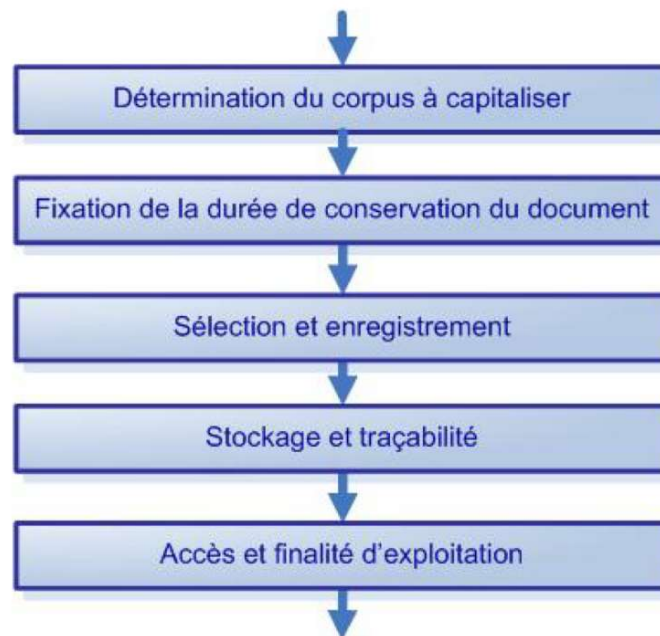


Figure 2.17 – Processus de capitalisation des connaissances basé sur la norme ISO 15489 et sur la gestion des enregistrements [Laroche, 2007]

La réutilisation de connaissances est un défi servant à faciliter la sensibilisation du concepteur pour résoudre les conflits dans le processus de conception. Ainsi, plusieurs méthodes ont vu le jour afin de pouvoir aborder cette problématique. Des approches assez connues dans le domaine de l'ingénierie des connaissances ont émergées, notamment celle de [Ermine, 2003] qui propose la méthode **MASK (Methodology for Analyzing and Structuring Knowledge)**. Cette méthode consiste à capitaliser les connaissances et à les lier dans une base de données selon plusieurs points de vue analytiques. Elle consiste ainsi à proposer des modèles pour en faciliter la conception routinière [Serrafero *et al.*, 2006].

Une autre méthode facilite la formalisation des connaissances. Elle se nomme **MOKA (Methodology and software tools Oriented to Knowledge Based engineering Applications)**. L'objectif de cette méthode sert à développer des applications KBE, en exploitant une ontologie appelée ICARE, afin de structurer les connaissances repertoriées [Klein, 2000] [Brimble & Sellini, 2000]. L'approche est également utilisée pour l'utilisation des connaissances techniques au travers de la conception dans CATIA [Skarka, 2007].

KNOMAD (Knowledge capture ; Normalisation ; Organisation ; Modeling ; Analysis ; and Delivery) est une approche pour la conception pluridisciplinaire qui a pour but d'acquérir des connaissances, de les formaliser et de les distribuer dans le cycle de vie [Curran *et al.*, 2010].

KOD (Knowledge Oriented Design) est une méthode qui permet de réunir les interviews d'experts, en se basant sur des phases de modélisation dite pratique et cognitive, afin de formaliser les recueils. Il faut savoir qu'une phase de modélisation avec l'informatique implique de favoriser l'interaction avec ces modèles [Vogel, 1991].

CommonKADS quant à lui, permet de mettre en place un système de gestion des connaissances avec l'aide d'une base de connaissances [Breuker, 1994].

Alizon *et al.* [Alizon *et al.*, 2006] ont développé une méthode appelée **R.E.U.S.E. (Reuse Existing Unit for Shape and Efficiency)** qui consiste à réutiliser les connaissances issues des informations de

fabrication dans la conception, via trois étapes : (i) l'étude de similarité, (ii) l'évaluation sur l'efficacité et (iii) la configuration. Contrairement aux autres approches, cette méthode est proposée pour être exploitée de manière générale dans l'industrie de petites ou grandes productions, et ce, dans la mesure où elle met en évidence plusieurs niveaux de réutilisation, et d'autre part, elle se base sur plusieurs critères d'informations de production.

Les approches visant à favoriser la réutilisation des connaissances sont des éléments pour lesquels il est nécessaire de détecter, capturer, formaliser et connaître le moment-clé pour leur diffusion. L'ensemble des méthodes évoquées ci-dessous sont répertoriées dans le Tableau 2.2.

Tableau 2.2 – Vue d'ensemble des méthodes de réutilisation des connaissances

Méthodes	Description	Catégorie
MOKA	Méthode de formalisation de la connaissance pour le développement des applications KB/KBE	Connaissance implicite
KNOMAD	Implémentation multidisciplinaire pour capitaliser, formaliser et diffuser des connaissances	Connaissance implicite
KOD	Développement d'une méthodologie pour les systèmes à base de connaissances : proposition d'un cadre de modélisation de la collecte de connaissances	Connaissance produit
CommonKADS	Hypothèse basée sur le partage des connaissances et qui repose sur leur réutilisation	Connaissance implicite
R.E.U.S.E	Objectif de réutilisation des connaissances issues de la production pour la conception	Connaissance manufacturière

2.3.2/ ONTOLOGIE ET RÈGLES DE RAISONNEMENT SÉMANTIQUE

Après avoir abordé les méthodes d'exploitation des connaissances, nous recensons, dans la section suivante, l'une des technologies, dite sémantique, servant à les formaliser avec l'aide d'ontologies et de règles de raisonnement.

2.3.2.1/ OBJECTIF DE L'ONTOLOGIE

Pour concevoir, exécuter et structurer les connaissances, plusieurs techniques de modélisation ont été proposées pour formaliser les connaissances. L'application du graphique conceptuel est basée sur des réseaux sémantiques considérés comme la modélisation de la « *mémoire humaine* » [Sowa, 1984]. L'ontologie est un formalisme basé sur ce graphique, qui se définit par une spécification explicite d'une conceptualisation représentée par des concepts ou des classes (entreprises, objets, etc.), elle est entre autres liée par des relations [Gruber, 1993]. Les fonctions permettent de caractériser les objets et les axiomes pour définir avec le langage naturel, la fonction des prédicats en interprétant le résultat des instances.

Dans le cadre du Web sémantique, plusieurs langues ont été développées par le consortium **W3C**⁴ [Berners-Lee *et al.*, 2001]. Aujourd'hui, la formalisation de l'ontologie s'effectue par

4. World Wide Web Consortium

l'intermédiaire du langage nommé **OWL (Web Ontology Language)**. Il permet notamment la structuration des connaissances selon des classes définies, dont les relations sont formalisées comme des contraintes et des méta-données [Monticolo, 2008] [Monticolo *et al.*, 2012a] [Monticolo *et al.*, 2012b].

L'utilisation d'ontologies couplées aux bases de connaissances permet au langage OWL interfacé avec le modèle de graphe **RDF (Resource Description Framework)**, de faciliter la communication et l'interopérabilité entre les systèmes. Représenté par la Figure 2.18 telle une forme pyramidale sur le Web sémantique, il existe plusieurs strates de cette pyramide représentant différentes technologies répertoriées au sein du web hypertextes **URI (Uniform Resource Identifier)**, **XML (Extensible Markup Language)** et l'alphabet basique comme l'Unicode. Les technologies standardisées du Web sémantique sont représentées par plusieurs couches telles que RDF, RDF schema (ou RDFS), vocabulaire d'ontologie (OWL) et la logique de description, comme le SPARQL - Protocol and RDF Query Language. La dernière couche est, entre autres, un élément étudié par la communauté scientifique, dont le but consiste à standardiser les concepts de preuve et de confiance.

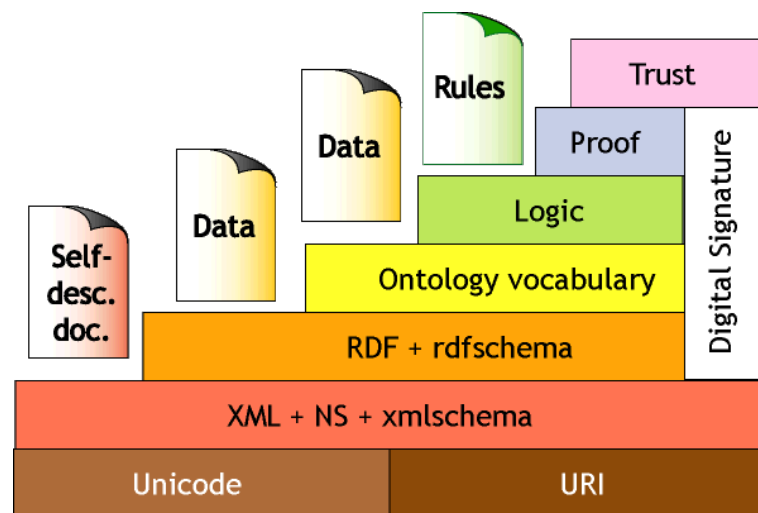


Figure 2.18 – Architecture du Web Sémantique [Antoniou & Van Harmelen, 2008]

2.3.2.2/ REPRÉSENTATION DU SYSTÈME DE RAISONNEMENT

La représentation des connaissances spécifiques (conditionnelle, déclarative, procédurale, etc.), requiert un formalisme particulier pour faciliter leur capture et leur réutilisation. La composition de différents types de connaissances facilite la différenciation et l'interaction qu'ont ces connaissances entre elles. Cette composition facilite également la compréhension pour l'agent informatique en vue de faire des inférences dans les bases de connaissances. Ces auteurs [Lehmann & Magidor, 2002] présentent l'inférence comme un processus qui consiste à obtenir une information explicite (qui était seulement implicite dans la connaissance de l'agent). Ils ont présenté une approche logique pour faciliter le raisonnement dans la base de connaissances, en favorisant les relations et les modèles par les connaissances conditionnelles (d'où l'existence du raisonnement sémantique suivant : *Si... Alors... Sinon...*).

La connaissance conditionnelle est importante car elle permet la diffusion des connaissances déclaratives et procédurales. Elle conditionne quand et pourquoi il faut les utiliser. Basées sur les schémas Si-Alors (If-Then) ou Si-Alors-Sinon (If-Then-Else), plusieurs méthodes existent pour représenter la connaissance conditionnelle qui est principalement fondée sur les graphes et la logique

[Tândăreanu & Ghindeanu, 2003]. Il est donc possible de la décrire comme telle :

La **connaissance conditionnelle** (ou connaissance contextuelle) - **Ck** : décrit le savoir « *quand* » et « *pourquoi* » il faut utiliser des connaissances déclaratives et procédurales, et surtout, quand il ne faut pas les utiliser (par exemple, avec les règles Si-Alors, etc.).

De manière similaire, la connaissance déclarative se différencie de la connaissance conditionnelle de la manière suivante :

La **connaissance déclarative** - **Dk** : décrit le savoir « *quoi* » servant à résoudre un problème (par exemple, une règle de conception, une définition, etc.).

Quant aux connaissances procédurales, leurs relations avec les connaissances déclaratives sont liées, à savoir :

La **connaissance procédurale** - **Pk** : décrit le savoir « *comment* » pour résoudre le problème en effectuant des tâches spécifiques selon une démarche processus ou de type processus (par exemple, procédure de conception, ordre de montage, bonnes pratiques, etc.).

L'exploitation des règles issues des connaissances conditionnelles doit permettre de prédéfinir les règles spécifiques pour activer, dans un premier temps, les connaissances déclaratives et par la suite les connaissances procédurales :

- **Si** l'assemblage de deux composants s'effectue par une liaison pivot **Alors** cette dernière est représentée par un cercle qui guide en rotation une pièce autour de l'axe de la liaison.

Tandis que la connaissance procédurale est conçue comme une procédure ou une action, l'exemple suivant suit la continuité de la connaissance déclarative comme :

- **Si** la liaison pivot est représentée par un cercle **Alors** le cercle (i) doit être tracé en connaissant l'équation d'un cercle, (ii) doit avoir un centre identifié, (iii) doit avoir un rayon défini, (iv) doit être placé sur un plan donné et (v) doit être fermé en connectant ses points frontières.

Les auteurs [Wray *et al.*, 2004] proposent la décomposition des connaissances pour faciliter l'encodage, afin de permettre à l'agent autonome d'être efficace dans l'exécution de ses tâches dédiées à la recherche de connaissances. L'exemple ci-dessous introduit un comparatif entre une modélisation classique de l'instruction Si-Alors (à gauche) et une modélisation décomposant des connaissances déclaratives et procédurales (à droite) pour faciliter le raisonnement de l'agent (Figure 2.19). Nous pouvons observer, à droite de la Figure 2.19, la règle Si-Alors pour coder les contraintes de lancement de missiles d'un avion de chasse avec les connaissances déclaratives, tandis que la règle suivante lie ces informations pour lancer la procédure, afin d'exécuter le lancement du missile. Cette deuxième approche est préférable à l'approche conventionnelle, car elle fournit une description claire pour faciliter l'inférence de l'agent dans le système.

Programmation avec une approche imbriquée	Programmation séparant les connaissances déclaratives et procédurales.
<pre> IF Aircraft = FA-18 altitude > 5000 Speed > 300 missile-is-locked ROE-are-met THEN Action(push-fire-button) </pre>	<pre> IF Aircraft = FA-18 THEN Fire-Constraints FA-18 Min-Fire-Altitude = 5000 Min-Fire-Speed = 300 </pre>
	<pre> IF Aircraft = ?plane Fire-constraints ?plane Min-Fire-Altitude ?min-alt Min-Fire-Speed ?min-spd Altitude > ?min-alt Speed > ?min-spd missile-is-locked ROE-are-met THEN Action(push-fire-button) </pre>

Figure 2.19 – Exemples d’une approche conventionnelle imbriquant connaissances déclaratives et procédurales (à gauche) et d’une convention de programmation (à droite), qui sépare mieux les affirmations déclaratives et les actions procédurales [Wray *et al.*, 2004]

2.3.2.3/ RÈGLES DE RAISONNEMENT SÉMANTIQUE

Les systèmes incluant un raisonnement sémantique sont peu nombreux et peu utilisés, car il existe peu d’outils commerciaux pour le soutien à la décision contextuelle lors de la phase de conception, ce qui conduit à des limitations dans l’exploration des espaces de solutions [Guindon, 1990]. La conception de systèmes complexes avec deux types de contraintes linéaires et non-linéaires, ne permet pas au concepteur de prendre en compte toutes les interactions entre les différentes variables. Utiliser des ontologies pour obtenir des connaissances adéquates à partir d’un système de base de connaissances (**Knowledge Based System - KBS**), demande de réexploiter les connaissances dans l’activité de conception, avec une application contenant un moteur de raisonnement [Reymen *et al.*, 2002].

Les moteurs d’inférences exploitent des règles spécifiques au langage sémantique, pour introduire l’ontologie. La description logique (**Description of Logic - DL**) et le langage de règles pour le web sémantique (**Semantic Web Rule Language - SWRL**), sont les principaux éléments pour vérifier la cohérence de l’information. Le DL et le SWRL permettent de mettre en place des règles complexes pour justifier et vérifier l’information, afin d’établir des relations claires entre les composants d’assemblage et leurs fonctionnalités [Kim *et al.*, 2006]. L’ontologie représente ainsi, explicitement les relations entre les pièces mécaniques pendant le processus d’assemblage [Kim *et al.*, 2006].

D’autres protocoles utilisent les règles de langage du web pour améliorer les requêtes sémantiques telles que la **Semantic Query-enhanced Web Rule Language - SQWRL**. Celles-ci sont l’évolution des requêtes SPARQL qui exploitent en complément du SWRL, les connaissances pertinentes en utilisant le langage sémantique. [Zhu *et al.*, 2012] présentent des unités de raisonnement spécifiques qualifiées comme étant un ensemble de règles et de requêtes appliquées au calcul d’inférence et basées sur des relations d’assemblage. L’auteur définit l’importance du raisonnement ontologique utilisé. En effet, il définit qu’il faut (i) concevoir et maintenir des ontologies de haute qualité, (ii) répondre aux requêtes sur les concepts et les instances de l’ontologie, (iii) intégrer et aligner plusieurs ontologies, et (iv) inférer de nouvelles informations basées sur les connaissances existantes.

Les moteurs d’inférences en question, tel que Pellet, Jena, HermiT, etc. sont déjà incorporés dans l’outil Protégé, qui constitue un logiciel dédié à l’édition d’ontologies. Par exemple, le raisonneur

HermiT 1.3.7 se différencie des autres raisonneurs par sa capacité à vérifier la cohérence du modèle de l'ontologie [Horrocks *et al.*, 2012]. Il est entièrement compatible avec le langage OWL2, dont les classes et les propriétés peuvent être définies avec l'algorithme de subsomption, qui réduit considérablement le nombre de tests de cohérence. Enfin, les réponses de requêtes SPARQL et les règles SWRL sont également prises en charge par HermiT [Glimm *et al.*, 2014]. A contrario, le raisonneur Pellet, utilise le tableau calculus qui offre comme fonctionnalité la possibilité de construire différents ensembles de règles de dérivation. Cette approche permet d'éviter au raisonneur un comportement indéterminé avec un raisonnement uniquement basé sur la logique (logic-based reasoning).

2.3.3/ MODÉLISATION DE LA CONNAISSANCE AVEC LE RAISONNEMENT

2.3.3.1/ APPROCHE ASD

Plusieurs auteurs proposent des formalismes dédiés à la modélisation de la connaissance en conception pour l'assemblage [Kim *et al.*, 2004] [Kim *et al.*, 2006] qui se basent sur les relations d'assemblages en considérant les relations spatiales en lien avec la géométrie du produit. La représentation ontologique permet de décrire des assemblages avec leurs contraintes associées. L'ontologie en question appelée **AsD (Assembly Design)** repose sur la description des assemblages issus de la théorie de la méréotopologie⁵, pour faciliter la compréhension de l'assemblage généré par le concepteur ou le gammiste [Kim *et al.*, 2004] [Kim *et al.*, 2008]. La Figure 2.20 ci-dessous illustre les classes de l'ontologie AsD.

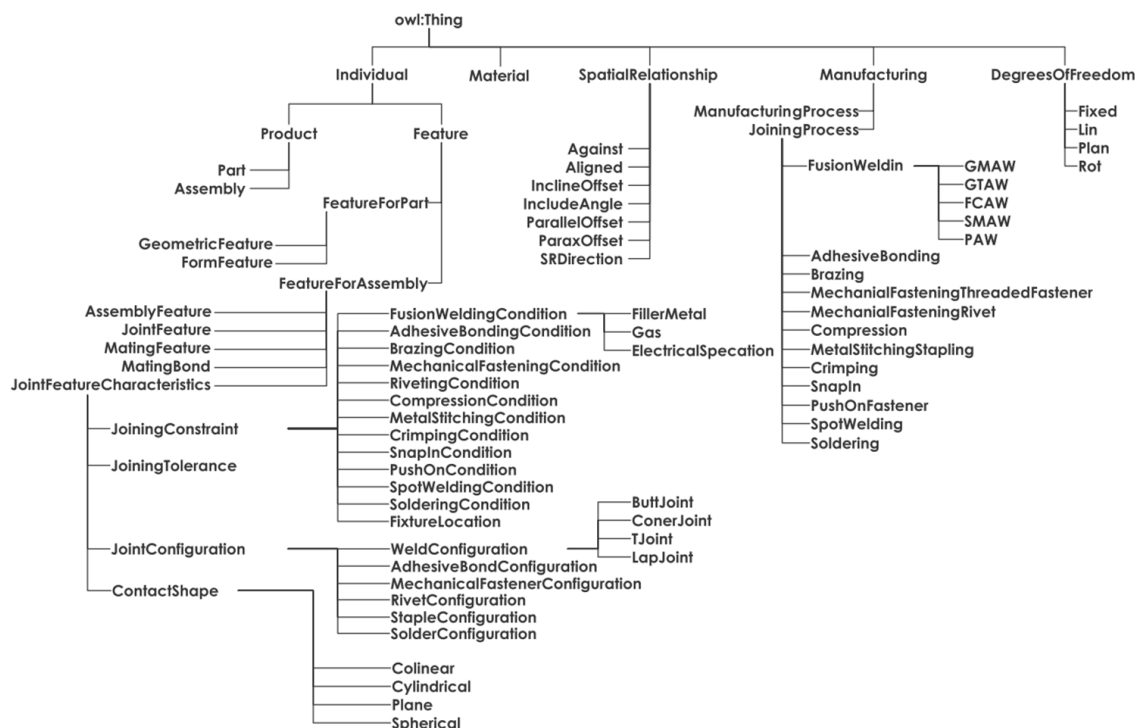


Figure 2.20 – Modèle ontologique AsD proposé par [Kim *et al.*, 2008]

Pour faciliter le partage entre les systèmes d'informations, le format de données est normalisé pour

5. Description théorique sur la nature des objets (régions), des entités qui les occupent et de leurs relations (primitives).

saisir les intentions du concepteur en lien avec les termes métiers. Ainsi l'utilisation du langage OWL, avec le formalisme du DL et du SWRL, permettent de comprendre dans la Figure 2.21 l'intention en lien avec les relations spatiales [Kim *et al.*, 2006] (par exemple, direction de référence, surface de contact, relation spatiale, degré de liberté, etc.). Les règles SWRL sont utilisées pour représenter et raisonner sur les différentes articulations entre l'assemblage et leurs relations ainsi que pour définir les termes métiers de conception. Les contraintes d'assemblage sont explicitement représentées en tant que telles et différenciées par le raisonneur.

SWRL rule:

- $\text{MatingFeature}(?x) \wedge \text{FormFeature}(?y) \wedge \text{Part}(?z) \wedge \text{belongTo}(?y, ?z) \wedge \text{hasMatingComponent}(?x, ?y) \wedge \text{hasSpatialRelationship}(?x, \text{against}) \wedge \text{hasContactShape}(?x, \text{Plane_Z}) \wedge \text{ReferenceDirection}(?x, \text{Z-Direction}) \rightarrow \text{hasDOF}(?z, \text{Plan_Z})$
 - $\text{MatingFeature}(?x) \wedge \text{FormFeature}(?y) \wedge \text{Part}(?z) \wedge \text{belongTo}(?y, ?z) \wedge \text{hasMatingComponent}(?x, ?y) \wedge \text{hasSpatialRelationship}(?x, \text{Against}) \wedge \text{hasContactShape}(?x, \text{Plane_Z}) \wedge \text{ReferenceDirection}(?x, \text{Z-Direction}) \rightarrow \text{hasOwnDOF}(?z, \text{Rot_Z})$
-

Figure 2.21 – Exemple de règles SWRL proposées par [Kim *et al.*, 2006] introduisant les concepts de : direction de référence, surface de contact, relation spatiale, degré de liberté, etc.

2.3.3.2/ APPROCHE DE ZHU ET AL.

[Zhu *et al.*, 2010] [Zhu *et al.*, 2012] cherchent à créer des applications sémantiques qui étendent les ontologies d'ingénierie avec des mécanismes de raisonnement. Ils proposent deux tâches spécifiques pour faciliter le raisonnement, en exploitant les inférences et les requêtes du SWRL et du SQWRL, couplées aux raisonneurs Jena et Pellet :

- La requête facilite la récupération des données existantes dans la base de connaissances ;
- L'inférence avancée permet de dériver de nouvelles données à travers des extensions de règles.

Les objectifs fondamentaux sont de proposer et d'exploiter un modèle qui permet de raisonner sur les ontologies d'assemblage du produit [Zhu *et al.*, 2011]. La fonctionnalité de raisonnement est proposée en deux objectifs :

- **Objectif 1** : *activer la fonctionnalité de requête directe* - en permettant une approche légère pour récupérer des informations d'assemblages CAO. Par exemple, définir les instances de composants et de contraintes dans des ontologies qui correspondent à des requêtes sur des données d'un assemblage ;
- **Objectif 2** : *activer la fonctionnalité avancée d'inférence* - en proposant des capacités d'inférence que les systèmes CAO ne peuvent pas supporter. Les connaissances nouvelles sont ainsi dérivées par l'extension des règles SWRL/SQWRL. A titre d'exemple, l'extraction des informations implicites du produit, ou bien la déduction de la contrainte d'assemblage en fonction des caractéristiques issues de la géométrie des composants qui le constituent (Figure 2.22).

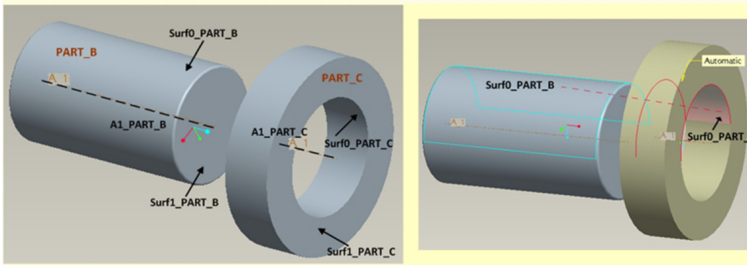
SQWRL Specification for Inferring and Retrieving Out Constraint *Insert* (1)

SQWRL(SQSP4): `sameAs(?c1,c_name1) ^ sameAs(?c2,c_name2 ^
has_feature(?c1,?fea1) ^ Protrusion(?fea1) ^
has_modelitem(?fea1,?item1) ^ Cylindrical_surface(?item1)
^ has_feature(?c2,?fea2) ^ Cut(?fea2) ^
has_modelitem(?fea2,?item2) ^ Cylindrical_surface(?item2)
→ sqwrl:select(?c1,?fea1,?item1,?c2,?fea2,?item2) ^
sqwrl:orderBy(?fea1)`

Explanation: `c_name1` is Component `c1`'s name
`c_name2` is Component `c2`'s name

Conditions: Component `c1` has a *Protrusion* feature `fea1` which contains a cylindrical surface `item1` and Component `c2` has a *Cut* feature `fea2` which also contains a cylindrical surface `item2`

Consequence: `c1` and `c2` can be mated, return `c1, fea1, item1, c2, fea2, item2`, ordered by `fea1`



I: PART_B and PART_C and Related Geometry Features

II: Assembling PART_C to PART_B in PRO/E

Figure 2.22 – Exemple de raisonnement sémantique proposé par [Zhu *et al.*, 2011]

Une architecture pour l'application de la technologie sémantique est proposée au travers de plusieurs strates. Pour cela, des unités de raisonnement sont exposées au niveau de l'assemblage et du composant présentées comme des ensembles de requêtes et d'inférences spécifiques agissant sur l'ontologie. Comparé aux systèmes CAO conventionnels, cette approche offre un accès léger aux données CAO qui sont capturées au travers des représentations explicites et formelles, en fournissant des capacités d'inférence que les systèmes CAO ne peuvent actuellement pas fournir.

2.3.3.3/ APPROCHE PRONOA2

Les auteurs [Demoly *et al.*, 2012] [Gruhier *et al.*, 2015] ont proposé une approche pour définir une ontologie concernant les relations du produit, en se basant sur une théorie de la méréotopologie appelée **JANUS** (Joined **A**wareNess and **U**nderstanding in assembly-oriented de**S**ign with mereo-topology). L'approche en question, nommée **PRONOA** (**PR**oduct relation**N**ships description based On mereotopologic**A**l theory), consiste à implémenter cette théorie dans l'ontologie pour réutiliser les informations et les connaissances dans d'autres phases du cycle de vie [Demoly *et al.*, 2012]. L'ontologie PRONOA, issue de cette approche, voit son contenu évoluer vers la description des relations spatio-temporelles entre les primitives et les régions du couple produit-process. Ainsi PRONOA2 [Gruhier *et al.*, 2015] décrit la définition et l'évolution des relations produit-process. Une méta-ontologie est proposée, pour représenter les objets (c'est-à-dire des régions) et de leurs relations (c'est-à-dire des primitives) dans les dimensions spatiales, temporelles et spatio-temporelles (Figure 2.23).

Chaque concept exploité dans l'architecture ontologique est implémenté dans l'éditeur d'ontologie Protégé, en utilisant le langage web d'ontologie OWL. Les classes sont divisées en deux sections : spatiale et temporelle. D'une part, les classes spatiales représentent la position du produit avec

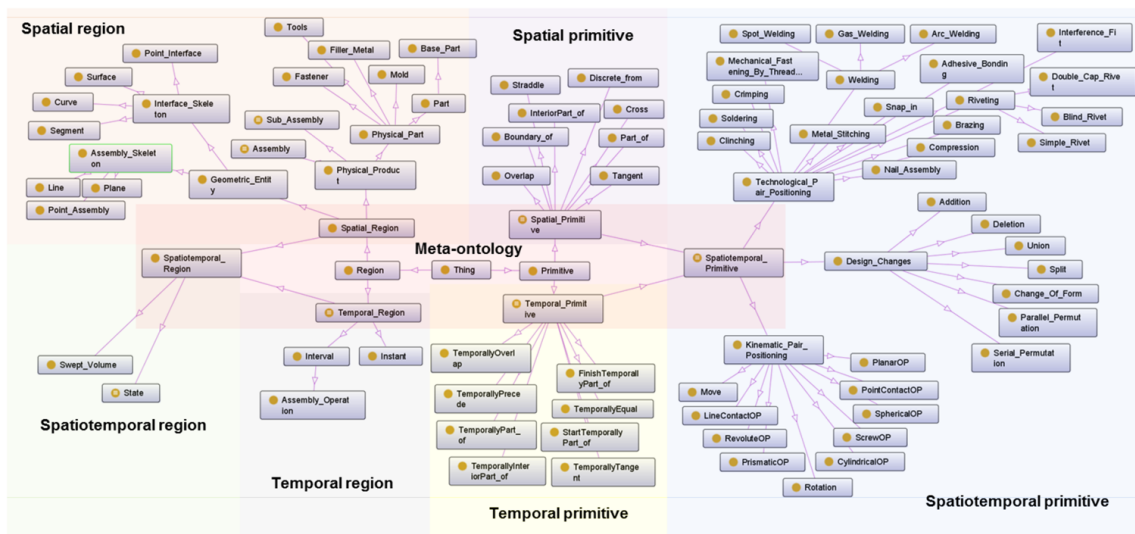


Figure 2.23 – Modèle ontologique PRONOA2 proposé par [Gruhier *et al.*, 2015]

ses relations (c'est-à-dire les entités géométriques, les contraintes d'assemblage, la fonctionnalité d'assemblage) dans l'environnement CAO. D'autre part, les classes temporelles définissent le processus d'assemblage du produit (c'est-à-dire l'opération de montage, l'intervalle entre les opérations, etc.). L'aspect spatio-temporel est la combinaison des entités du produit et du processus (à savoir, l'état dans laquelle le composant est assemblé, ou bien, le volume balayé représentant l'insertion du composant, etc.) et les relations (par exemple, le positionnement des liaisons cinématique et technologique, un changement dans la conception, etc.).

L'approche PRONOA2 permet de fournir les informations pertinentes dans l'ingénierie des produits pour faciliter l'intégration des modèles CAO. La Figure 2.24 présente les restrictions logiques de description pour le positionnement des liaisons cinématiques, afin d'exploiter l'instanciation de connaissances.

Class name	Rule type	DL rule
CylindricalOP	Cardinality	$CylindricalOP \sqsubseteq (Kinematic.Pair) \sqcap (has.Geometric.Entity = 1 Line) \sqcap (has.Geometric.Entity = 1 Surface)$
LineContactOP	Cardinality	$LineContactOP \sqsubseteq (Kinematic.Pair) \sqcap (has.Geometric.Entity = 1 Line) \sqcap (has.Geometric.Entity = 1 Segment)$
Move	Cardinality	$Move \sqsubseteq (Kinematic.Pair) \sqcap (((has.Geometric.Entity = 1 Line) \sqcap (has.Geometric.Entity = 1 Plane)) \sqcup ((has.Geometric.Entity = 1 Line) \sqcap (has.Geometric.Entity = 1 Point.Assembly)) \sqcup ((has.Geometric.Entity = 1 Plane) \sqcap (has.Geometric.Entity = 1 Point.Assembly)) \sqcup (has.Geometric.Entity = 2 Line) \sqcup (has.Geometric.Entity = 2 Plane) \sqcup (has.Geometric.Entity = 2 Point.Assembly))$
PlanarOP	Cardinality	$PlanarOP \sqsubseteq (Kinematic.Pair) \sqcap (has.Geometric.Entity = 1 Plane) \sqcap (has.Geometric.Entity = 1 Surface)$
PointContactOP	Cardinality	$PointContactOP \sqsubseteq (Kinematic.Pair) \sqcap (has.Geometric.Entity = 1 Point.Assembly) \sqcap (has.Geometric.Entity = 1 Point.Interface)$
PrismaticOP	Cardinality	$PrismaticOP \sqsubseteq (Kinematic.Pair) \sqcap (has.Geometric.Entity = 1 Line) \sqcap (has.Geometric.Entity = 1 Surface)$
RevoluteOP	Cardinality	$RevoluteOP \sqsubseteq (Kinematic.Pair) \sqcap (has.Geometric.Entity = 1 Line) \sqcap (has.Geometric.Entity = 1 Plane) \sqcap (has.Geometric.Entity = 1 Surface)$
Rotation	Cardinality	$Rotation \sqsubseteq (Kinematic.Pair) \sqcap (has.Geometric.Entity = 2 Line) \sqcap (has.Geometric.Entity = 1 Point.Assembly)$
ScrewOP	Cardinality	$ScrewOP \sqsubseteq (Kinematic.Pair) \sqcap (has.Geometric.Entity = 1 Line) \sqcap (has.Geometric.Entity = 2 Curve) \sqcap (has.Geometric.Entity = 1 Surface)$
SphericalOP	Cardinality	$SphericalOP \sqsubseteq (Kinematic.Pair) \sqcap (has.Geometric.Entity = 1 Point.Assembly) \sqcap (has.Geometric.Entity = 1 Surface)$

Figure 2.24 – Règles logiques de description (DL) dans PRONOA2 [Gruhier *et al.*, 2015]

2.3.4/ BILAN ET POSITIONNEMENT

Nous avons passé en revue les travaux de recherches abordant la notion de gestion des connaissances, et plus précisément la réutilisation des connaissances dans la conception. L'exploitation des connaissances fait partie des éléments de recherches actuels, en se concentrant non plus

sur la capture, mais sur leurs instanciations efficaces, dans un contexte difficilement compréhensible par la machine, du fait des règles métiers différentes. Selon la communauté scientifique, il est mis en évidence que la réutilisation des connaissances utilise plusieurs approches au travers de méthodes focalisées sur le produit, le process/procédé et la tâche. Ainsi, un système à base de connaissances doit faciliter l'aide à la décision, mais aussi s'adapter aux structures de travail et faciliter l'interopérabilité pour favoriser l'innovation [Perry, 2007]. Il est utile de modéliser les connaissances pour les exploiter efficacement, via des raisonnements logiques. La Figure 2.25 présente notre positionnement sur la réutilisation des connaissances avec une approche nommée MERCURY2. L'exploitation des technologies sémantiques semble être un atout pour notre problématique de diffusion de la connaissance de manière proactive pendant la conception. Notamment, pour la recherche et l'instanciation de connaissances, par des règles s'appliquant directement sur l'ontologie.

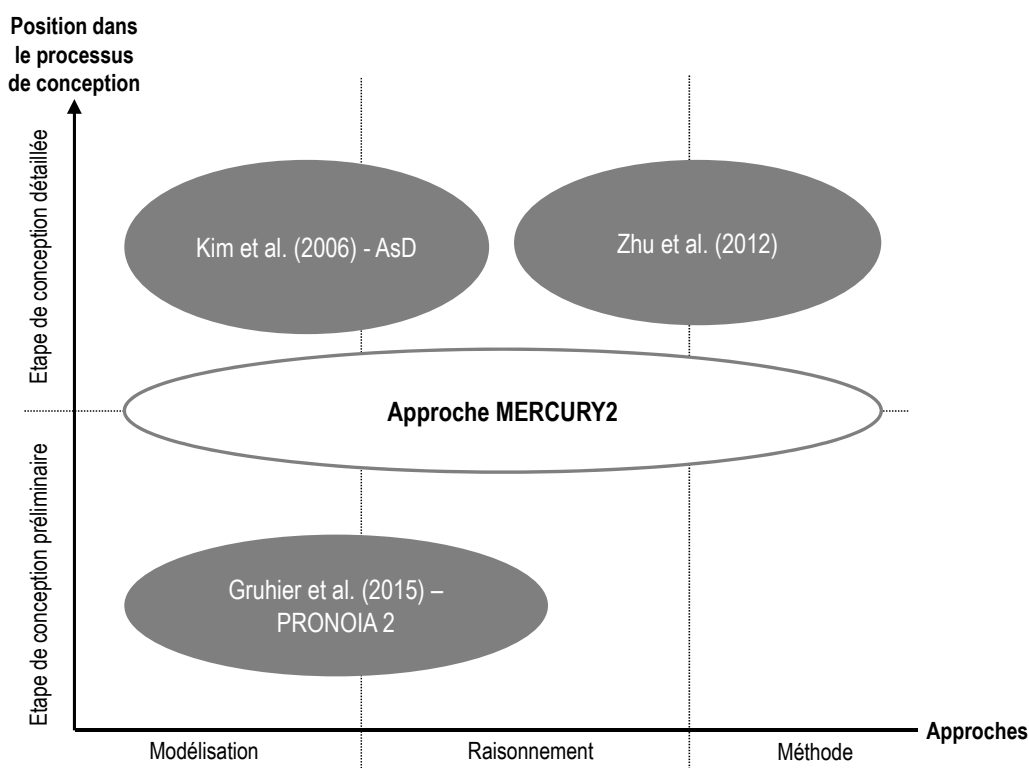


Figure 2.25 – Cartographie des approches exploitées pour réutiliser les connaissances

2.4/ GESTION DU CYCLE DE VIE DES PRODUITS

2.4.1/ CONCEPT ET OBJECTIFS DU PLM

Aujourd'hui, l'exploitation des connaissances dans le PLM (c'est-à-dire l'ensemble des données et des informations incluses), devient contraignant au vu des problèmes liés à la communication en milieu collaboratif entre chaque partie prenante, notamment dans un contexte d'ingénierie concurrente. Face à ces évolutions d'ordre méthodologique, organisationnel et technologique, des

problèmes pour formaliser les données et informations liées au couple produit-process émergent. Ainsi, plusieurs travaux de recherche dans le domaine du PLM existent et traitent de ces problématiques, déjà abordées par ailleurs dans l'état de l'art concernant la gestion des informations et des connaissances dans le cycle de vie du produit.

Le paradigme PLM est un concept, un acronyme utilisé dans le marché des Technologies de l'Information et de la Communication - **TIC** (Information and Communication Technologies - **ICT**). Il peut être considéré comme une solution de gestion des connaissances [Ameri & Dutta, 2005]. Il peut également être considéré comme une stratégie d'entreprise qui facilite la conception orientée gestion et développement de l'ingénierie [Merminod & Rowe, 2012].

Son évolution issue de l'ingénierie séquentielle dans le « *PLM 1.0* » à l'ingénierie concourante dans le « *PLM 2.0* » (selon l'éditeur de logiciels Dassault Systèmes), implique le passage d'une gestion hiérarchique à une organisation à plat, afin de réduire le temps de cycle de développement des produits et de se focaliser dans le processus de développement. L'impact du PLM dans le cycle de vie du produit, concerne par exemple, la réduction des coûts des déchets (notamment des prototypes), l'augmentation de la qualité du produit, et la réduction des erreurs de développement.

La gestion du cycle de vie des produits (**Product Lifecycle Management - PLM**) est une approche de gestion intégrée qui réunit systématiquement les données des produits des différentes étapes du cycle de vie du produit. Il constitue un ensemble cohérent de processus, de méthodes et d'outils pour gérer les informations sur les produits, tout au long de leur cycle de vie. Il ne se limite pas aux entreprises individuelles mais aussi aux réseaux de collaboration et aux chaînes d'approvisionnement [Abramovici, 2007].

Les auteurs [Terzi *et al.*, 2010] définissent les éléments fondamentaux du PLM (Figure 2.26), en montrant la stratégie employée autour du produit au cours de ses trois phases de vie : le début de vie (**Beginning Of Life - BOL**), le milieu de vie (**Middle Of Life - MOL**) et la fin de vie (**End Of Life - EOL**). Dans ce contexte, les outils, les méthodes et les systèmes d'information dédiés au PLM, exploitent les informations qui s'interfacent entre elles pour favoriser les échanges inter-métier.

L'avènement des nouvelles technologies amène l'ingénierie proactive à comprendre le contexte donné afin d'anticiper les futures évolutions du produit au travers de son développement. Pour cela, des problèmes d'interopérabilité ont été décelés au sein de la relation produit/process, dû au formalisme spécifique de chaque modèle métier, et qui, par conséquent, sont inadéquats avec le contexte de l'entreprise [Allen & Sriram, 1998].

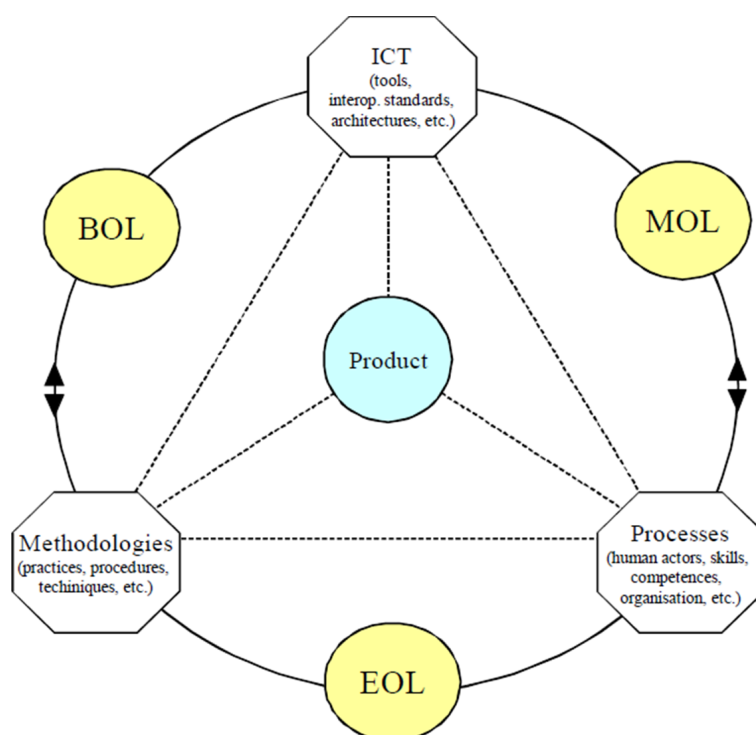


Figure 2.26 – Elements fondamentaux du PLM [Terzi *et al.*, 2010]

La Figure 2.27, proposée par les auteurs [Demoly *et al.*, 2013], représente les systèmes internes au PLM, au travers de leur utilisation sur le cycle de vie du produit et pour laquelle, plusieurs systèmes, outils et méthodes sont cartographiés. Nous identifions au début du cycle de vie, les systèmes PDM, MPM, **SCM** (**S**upply **C**hain **M**anagement), en milieu de vie, l'**ERP** (**E**ntreprise **R**essource **P**lanning), et en fin de vie, le **LCA** (**L**ife **C**ycle **A**ssessment). Ces outils et méthodes viennent compléter le PLM, tels que les outils CAO, **CAE** (**C**omputer **A**ided **E**ngineering), **CAM** (**C**omputer **A**ided **M**anufacturing), KBE, **CAPP** (**C**omputer **A**ided **P**rocess **P**lanning) en complément des approches DFX.

Les objectifs principaux du PLM consistent à améliorer la performance de développement des produits pour :

1. Eliminer les activités qui n'apportent pas de valeur ajoutée au processus de conception, en vue de réduire les coûts ;
2. Réduire les temps d'attente, pour la diffusion des connaissances afin de réduire le temps d'exploitation ;
3. Stimuler la réutilisation des connaissances pour accroître leur qualité et leur efficacité.

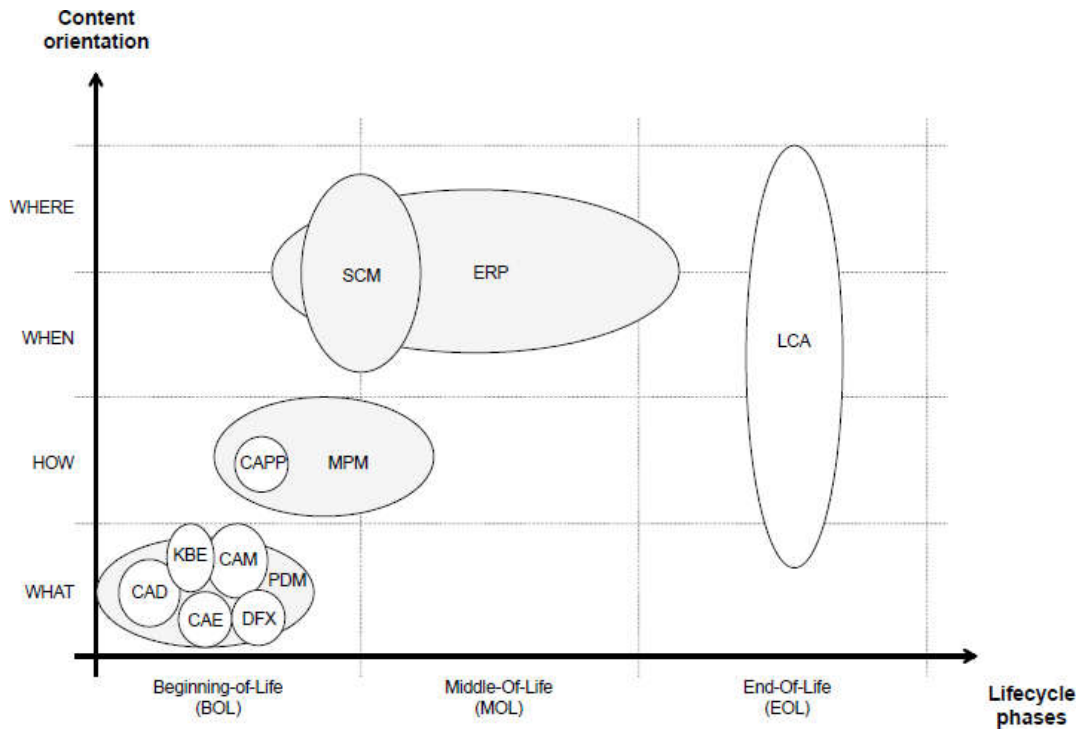


Figure 2.27 – Cartographie des systèmes PLM existants selon [Demoly *et al.*, 2013]

2.4.2/ GESTION DES INFORMATIONS ET DES CONNAISSANCES PRODUIT-PROCESS

En tant que stratégie d'entreprise intégrant de nouvelles méthodes, mais aussi de nouveaux outils et processus, le PLM est né d'un besoin industriel visant à une meilleure gestion des informations et des connaissances du couple produit/process. Historiquement, la CAO a été l'un des premiers systèmes d'ingénierie dans la conception des produits, qui se focalise sur la modélisation de maquettes numériques (Digital Mock-Up), en ayant la possibilité d'interagir avec des vues 2D ou 3D du produit. Cette possibilité conduit à l'émergence des formats standard (STEP, IGES, etc.) pour faciliter l'interopérabilité entre les outils CAO et de faciliter l'exploitation des fichiers numériques.

L'évolution de la gestion des données a permis la mise en place d'une véritable organisation des données d'ingénierie (**Engineering Data Management - EDM**) plus connue sous le terme de Gestion des Données Produit (PDM). Elle a ainsi renforcé plusieurs facteurs tels que la traçabilité, l'identification des composants, les relations produit-process, etc. L'approche PLM et les technologies associées constituent une évolution des systèmes PDM dont le cœur de fonctionnement est la gestion des liens entre les composants du produit et plus généralement de l'activité d'ingénierie selon une approche processus.

Les systèmes de gestion des données de produits (**Product Data Management - PDM**) sont des outils qui aident les ingénieurs à gérer les données et les processus liés au développement du cycle de développement [Bilgic & Rock, 1997].

Récemment les systèmes PLM exploitent les systèmes **Manufacturing Process Management - MPM**, pour faciliter la gestion des informations issues du processus d'assemblage et de fabrication [Sly, 2004].

Les systèmes (**Manufacturing Process Management - MPM**) représentent une étape visant à lier les informations de la conception du produit avec celle de la production, tout en améliorant la qualité de l'information et en réduisant le délai de mise sur le marché [Sly, 2004].

Depuis plusieurs années, l'utilisation du concept de PLM a été couplé à d'autres types de gestion des données, y compris celle des connaissances (**Knowledge Management - KM**), qui, conjointement avec le PDM et le MPM, rendent la gestion des produits et du process plus efficaces sur la distribution des connaissances. Le type de gestion est itératif (c'est à dire une répétition du processus de développement), liant plusieurs acteurs, services et fonctions, dans le but de mettre en œuvre une base structurée afin de faciliter la communication, et par conséquent la performance du processus d'ingénierie.

La gestion des connaissances (**Knowledge Management - KM**) est un processus qui identifie, capture et augmente la connaissance collective dans une organisation [Krogh, 1998]. Celle-ci augmente la capacité d'innovation et de réactivité [Hackbarth, 1998].

L'application de la gestion des connaissances en association avec un système de PLM est une démarche souvent utilisée pour concevoir et optimiser les produits en termes de qualité, de coûts et de délais. L'anticipation des caractéristiques du produit constitue l'un des éléments exploités pour augmenter sa capacité de personnalisation [Schuh *et al.*, 2008]. La collaboration entre toutes les parties prenantes, permet d'accroître l'efficacité dans le processus de conception des produits avec l'aide d'une démarche de gestion des connaissances (KM).

L'évolution du PLM tend à ouvrir de nouvelles voies qui, à long terme, convergeront vers un système optimal de communication s'appuyant sur une véritable gestion de l'information. Il deviendra une plateforme interopérable (ou HUB) permettant l'intégration de processus neutre [Liu *et al.*, 2009]. L'interconnexion entre le système PLM et le système KM élargit les possibilités d'exploiter les technologies sémantiques et ontologiques, pour une meilleure compréhension des informations et des connaissances. Ces changements affecteront principalement la gestion du cycle de vie en début et en milieu du cycle de vie (BOL et MOL).

2.4.3/ MODÉLISATION DU PLM

Les évolutions liées au développement d'un système technique sont devenues de plus en plus complexes à gérer, tant par le nombre important de composants d'un produit, que par les relations avec les autres services ou systèmes. Depuis les années 2000, la hausse de compétitivité des pays émergents a accéléré le développement du PLM, ce qui implique des méthodes et des moyens [Schuh *et al.*, 2008] servant à développer l'aspect collaboratif et innovant [Sharma, 2005]. Tout ceci est impossible à défaut d'optimisation efficace de la gestion des flux et de la modélisation effective des processus [Ahmad *et al.*, 2005] [Allweyer, 2010] [Hossain *et al.*, 2012], l'échange des informations/connaissances au sein d'une architecture PLM. La Figure 2.28 suivante présente les bases de la modélisation d'un système PLM.

Force est de constater que l'évolution du PLM est non seulement d'ordre sociale et technologique, mais également intimement liée au processus collaboratif d'ingénierie. Le PLM est donc destiné à faciliter la communication entre les parties prenantes. Cette évolution permet d'étudier, dans la section suivante, la modélisation du PLM au travers de notre problématique de recherche dédiée à la gestion proactive, intelligente et collaborative de l'exploitation des connaissances.

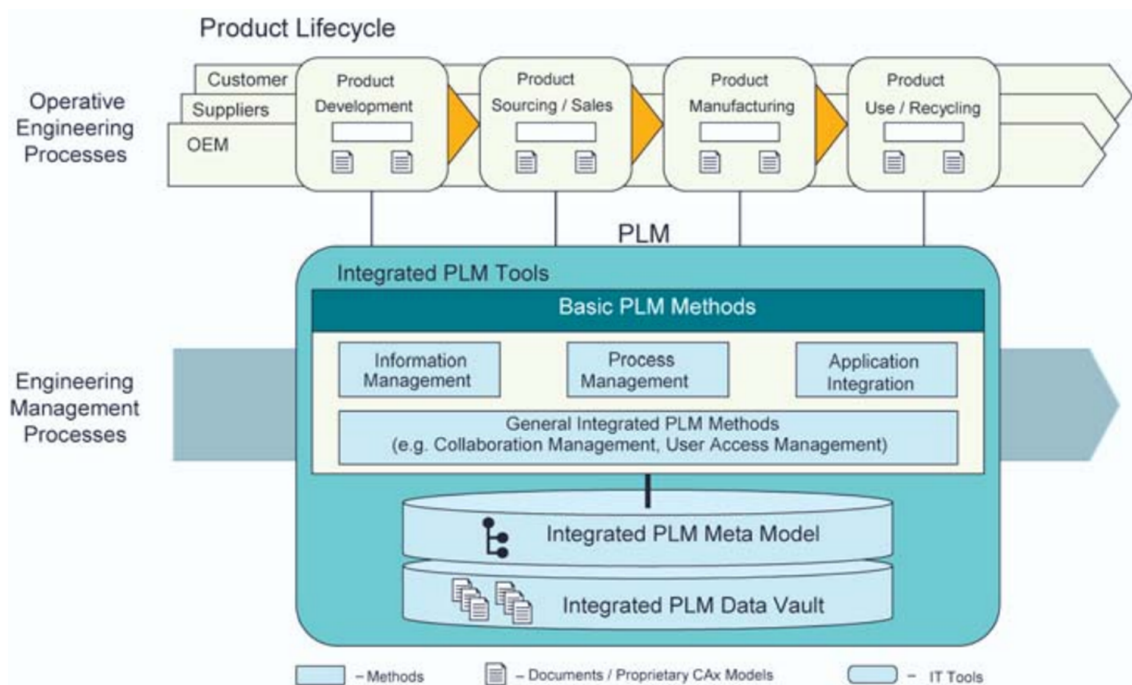


Figure 2.28 – Elements de base pour modéliser l'approche de gestion du cycle de vie des produits [Abramovici, 2007]

2.4.3.1/ GESTION INTELLIGENTE

La gestion intelligente consiste à lier l'intelligence aux informations des produits en vue d'encourager la propagation des informations tout au long du cycle de vie et d'en fournir les informations utiles, notamment sur les impacts environnementaux [Främling *et al.*, 20013]. Plusieurs approches permettent la modélisation des produits et des processus dans le développement de produits telles l'approche sur les propriétés guidées du développement produit (**Property-Driven Development/Design - PDD**) [Weber, 2003] (Figure 2.29). Le concept consiste à différencier la représentation des propriétés en les formalisant. Les caractéristiques décrivent la structure du produit, sa forme, sa cohérence avec la matière. Ainsi, les propriétés décrivent le comportement du produit. Le PDD relie alors deux relations : les caractéristiques et les propriétés pour mesurer et tester plusieurs outils et méthodes, afin de soutenir l'analyse d'ingénierie telle que les estimations, les expériences, les interviews clients, les tests expérimentaux, les diagrammes, les tables et les simulations.

L'état 0 correspond au PDM aujourd'hui ;

L'état 1 ajoute la capture des propriétés ;

L'état 2 ajoute le soutien de la gestion des processus et des ressources ;

L'état 3 ajoute la capture et la gestion des propriétés du cycle de vie ;

L'état 4 ajoute la capture et la gestion des moyens et des ressources.

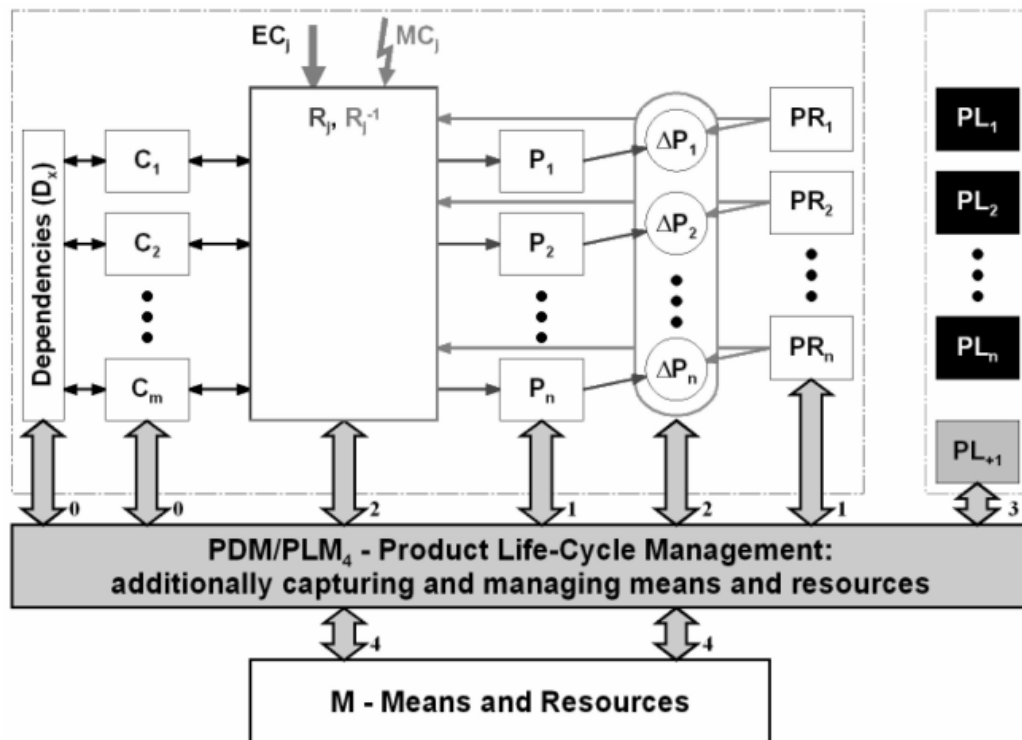


Figure 2.29 – Evolution du PDM et du PLM basés sur le PDD [Weber, 2003]

2.4.3.2/ GESTION DES CONNAISSANCES

La gestion des connaissances exploite les systèmes KM et PDM pour permettre d'atteindre une gestion organisée des flux de connaissance structurées et non-structurées, en complément des flux d'information mis en jeu lors de conception du couple produit-process. Les connaissances non structurées découlent de plusieurs études destinées à recueillir les bonnes informations/connaissances (élicitation, capture, exploitation, traçabilité) requises pour effectuer l'acte de conception et de pouvoir assurer leur réutilisation. [Chandrasegaran *et al.*, 2013]. Plusieurs concepts proposent des conceptions paramétriques du produit, au travers d'application dédiées à la conception de produits industriels [Gómez *et al.*, 2013]. La récupération des informations au travers des outils de CAO, tels que CATIA ou CREO, explorent plusieurs alternatives pour le processus de conception et d'assemblage. D'un point de vue général, les connaissances de l'entreprise sont divisées en deux types (i) les connaissances opérationnelles et (ii) les connaissances spécifiques [Garetti *et al.*, 2005] :

- Les connaissances opérationnelles traitent des pratiques utilisées dans l'entreprise afin d'exécuter les différents types de processus métier (des procédés techniques aux procédés managériaux) ;
- Les connaissances spécifiques traitent des produits et des procédés spécifiques liés au savoir-faire métier, utilisés dans les processus de fabrication (par exemple, la mise en forme d'un matériau destiné à équiper deux moteurs différents).

L'approche PROMISE en est un exemple (Figure 2.30). Proposée par [Kiritsis, 2013], cette approche consiste, avec l'aide des technologies sémantiques, à développer une nouvelle méthode et

un outil dans la gestion du cycle de vie des actifs, dont le but est d'explorer (i) l'optimisation des actifs, (ii) de lier les connaissances à la conception et à la fabrication, et enfin (iii) de proposer une approche holistique du cycle de vie pour les chaînes d'approvisionnement en boucle fermée .

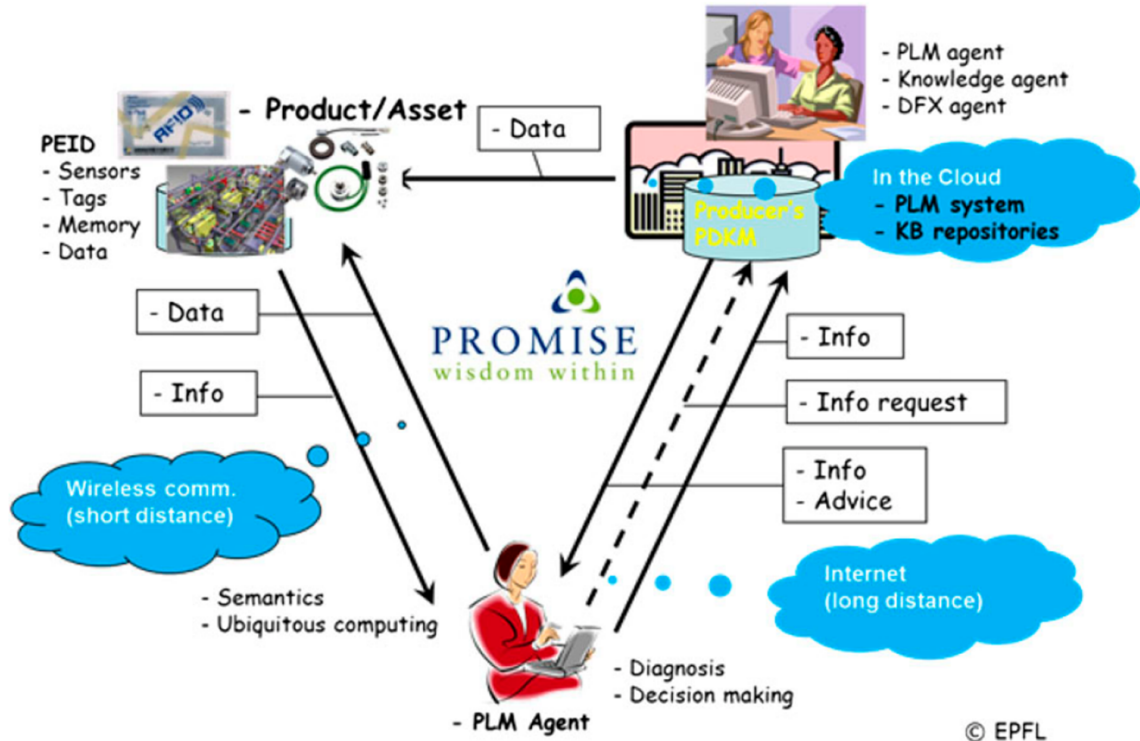


Figure 2.30 – Approche PROMISE [Kiritsis, 2013]

Les auteurs [Petrazoller *et al.*, 2014] proposent un HUB baptisé « *Incubateur* » dont la finalité est de centraliser les informations du produit (Figure 2.31). Ce HUB communique avec le PDM pour comprendre et raisonner, afin de proposer des espaces de conceptions (tels que les squelettes d'assemblages, les volumes-enveloppe, etc.). L'application facilite la réactivité et l'agilité dans l'ingénierie de conception avec l'aide d'outil CAO. Le HUB permet entre autres d'instancier des connaissances dans le système d'information (PDM) ou dans la CAO en relation avec la base de connaissances.

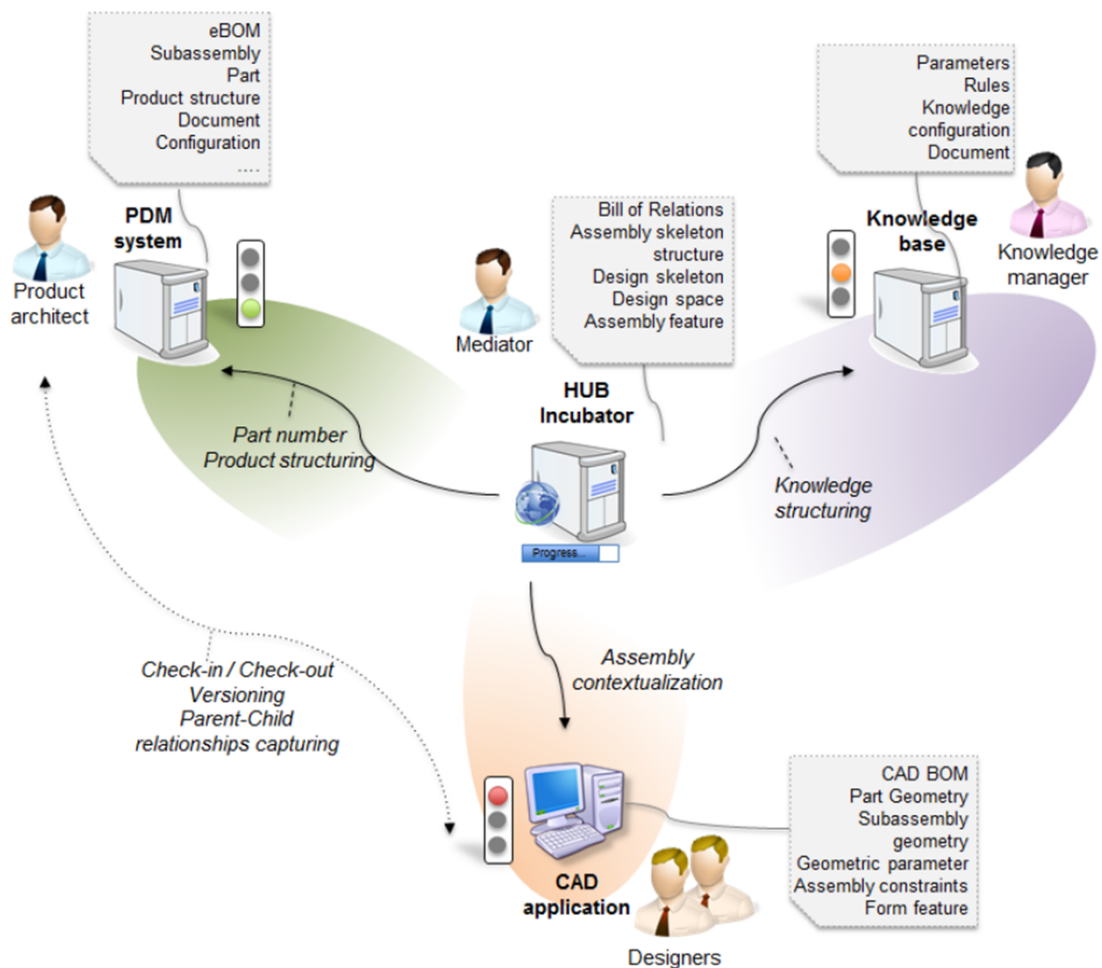


Figure 2.31 – Concept du HUB Incubateur [Petrazoller *et al.*, 2014]

2.4.3.3/ GESTION COLLABORATIVE

La gestion collaborative permet une meilleure gestion inter ou intra entreprise(s) entre les parties prenantes impliquées dans la gestion du cycle de vie des produits. Plusieurs avantages peuvent ainsi être retenus, tels que l'augmentation de la qualité et du nombre de documents dédiés à la conception des processus, de la construction et du contrôle de la nomenclature (**Bill Of Materials - BOM**), de la gestion des processus pour approuver les changements, et du dépôt de documents électroniques.

L'une des nombreuses approches de collaborations est basée selon [Sharma, 2005], autour de trois thèmes appelés **Collaboration**, **Produit**, **Innovation** (**Collaborative Product Innovation - CPI**) (Figure 2.32). La collaboration interne et externe des entreprises, est un point phare du PLM, car les parties prenantes sont en mesure de communiquer et de s'informer efficacement sur les données Produit-Process grâce à la mise en œuvre du PLM, facilitant grandement les échanges. La mise en œuvre comprend également des services standards pour échanger des données avec le PDM [Gunpinar & Han, 2008].

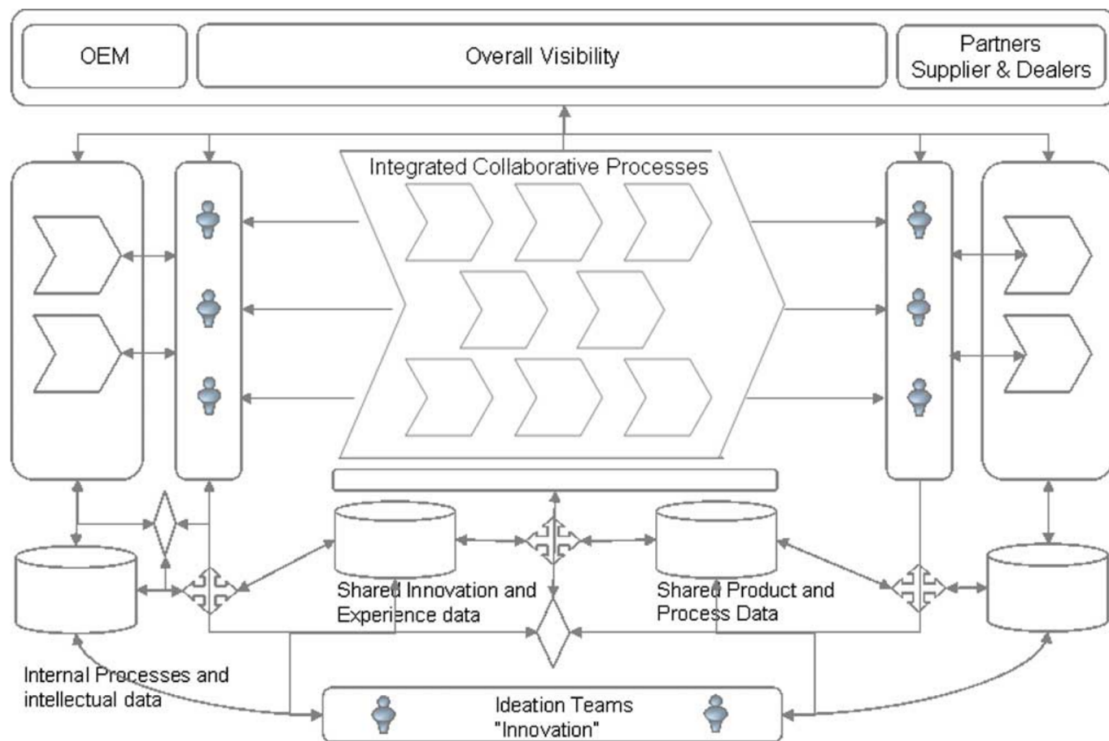


Figure 2.32 – Collaborative Product Innovation [Sharma, 2005]

Une autre approche permet d'évaluer les avantages de la mise en œuvre du PLM. Il est proposé un cadre méthodologique pour mesurer les bénéfices de la technologie produite, ainsi que la compréhension des résultats, pour faire le lien avec les changements dans les organisations [Cantamessa *et al.*, 2012]. Ces différences sont également étudiées par les auteurs [Merminod & Rowe, 2012], autour d'un terme appelé « *glitch* ».

A ce jour, le PLM est encore orienté vers la collaboration des parties prenantes. L'évolution de celui-ci tend à mettre l'accent sur l'utilisation de manière intelligente des connaissances en interprétant le contexte donné, ainsi que leur logique d'interaction envers d'autres données.

2.4.4/ BILAN ET POSITIONNEMENT

L'étude des travaux de recherche existants dans le domaine de la gestion du cycle de vie des produits met en évidence la nécessité de gérer les informations naissantes dans les systèmes d'information PLM, dont le double objectif est de gérer les informations, et de réutiliser efficacement leurs diffusions dans un contexte de conception. La Figure 2.33 représente le positionnement proposé pour nos travaux de recherche. Le but final est de proposer une articulation autour de la réutilisation des connaissances issues du DFA dans un système PLM avec comme plateforme : une base de connaissances (KB), une base de conception (représentée par un système PDM), et une base de prédiction (système MPM). Tout en communiquant avec l'outil CAO, un médiateur gèrera et facilitera l'envoi des informations et des connaissances, au sein d'une application développée au sein du laboratoire appelée PEGASUS.

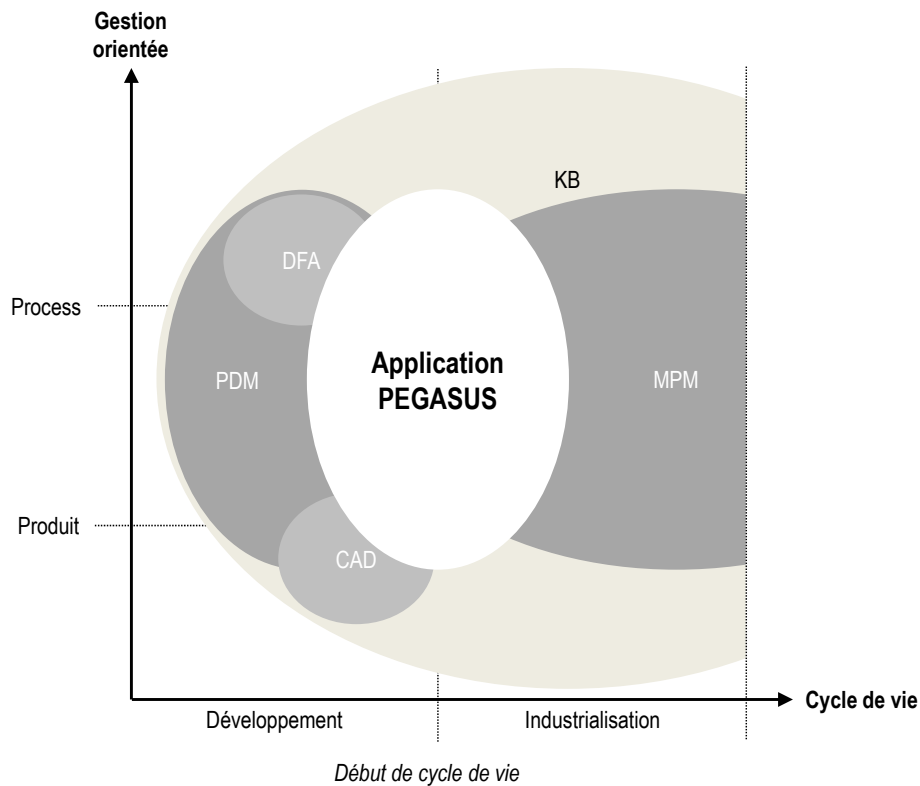


Figure 2.33 – Cartographie des systèmes PLM en début de cycle de vie et positionnement de l'outil PEGASUS

2.5/ CONCLUSION : NÉCESSITÉ D'UN CADRE MÉTHODOLOGIQUE ET D'UNE APPROCHE

Dans le cadre de ce chapitre de thèse, un état de l'art a d'abord été effectué sur la notion de « *contexte augmenté* » en interprétant la modélisation du contexte et du futur contexte, pour comprendre les intentions du concepteur. Ainsi, les caractéristiques attendues du « *contexte augmenté* » consistent à superposer de manière harmonieuse les informations du contexte présent et celles du futur contexte. Ces caractéristiques s'apparentent à l'utilisateur (la capture de son intention), l'activité (le processus de conception dans une situation donnée), la relation entre les objets, le temps et l'emplacement ou la position de l'objet dans l'espace.

Par la suite, une étude bibliographique a été effectuée dans les domaines de la réutilisation des connaissances en exploitant les technologies sémantiques ainsi dans le domaine de la gestion des informations d'un système sur l'ensemble de son cycle de vie. Plusieurs approches et modèles ont été évoqués, permettant de faire émerger, plus spécifiquement, notre positionnement, dont l'objectif est de proposer un cadre méthodologique et une approche servant à exploiter et réutiliser les connaissances, avec l'aide d'un système sensible au contexte issu du PLM. Le noyau du système sera géré par un médiateur qui a pour fonction de gérer et diffuser les connaissances avec les règles sémantiques, de générer un contexte de conception approprié tout en interprétant les intentions et les situations. En tant que tel, chacun des trois domaines devrait faciliter la prise de conscience du concepteur, en diffusant de manière pertinente, les connaissances aux concepteurs, avec l'aide des autres parties prenantes.

*2.5. CONCLUSION : NÉCESSITÉ D'UN CADRE MÉTHODOLOGIQUE ET D'UNE APPROCHE*⁵³

Dans notre proposition favorable à une approche proactive, nous allons nous concentrer sur l'un des cycles de vie du produit qui concerne l'assemblage. Il est donc nécessaire de consolider le contexte augmenté orienté sur ce cycle de vie, en superposant les relations spatiales et temporelles. Nous verrons donc une théorie spécifique à cette consolidation afin de permettre la recherche de connaissances en lien avec la situation de conception. Ainsi, les éléments nécessaires à la réutilisation des connaissances dans un outil CAO facilitera l'aide à la décision des concepteurs en ayant une diffusion pertinente de ces dernières.

CADRE MÉTHODOLOGIQUE DU CONTEXTE DE CONCEPTION AUGMENTÉ ORIENTÉ ASSEMBLAGE - DAMON

« La conscience est un trait d'union entre ce qui a été et ce qui sera, un pont jeté entre le passé et l'avenir. »

Henri Bergson

Sommaire

3.1 Introduction	56
3.2 Cadre méthodologique - DAMON	58
3.2.1 Objectif du cadre méthodologique DAMON	58
3.2.2 Vue d'ensemble du cadre méthodologique DAMON	61
3.3 Description de DAMON	63
3.3.1 Définition des relations du produit via le graphe orienté (étape 1)	63
3.3.2 Génération de la planification des séquences d'assemblage avec les relations composants à composants (étape 2)	64
3.3.3 Actualisation du contexte de conception (étape 3)	66
3.3.4 Requête des connaissances possibles (étape 4)	71
3.3.5 Instanciation des connaissances appropriées dans l'activité de conception (étape 5)	77
3.4 Conclusion	82

3.1/ INTRODUCTION

Ces travaux de recherche reposent sur un état de l'art dans le domaine de la réutilisation des connaissances, de la modélisation du contexte et du futur contexte, ainsi que de la gestion du cycle de vie des produits, afin de mettre en évidence les besoins scientifiques et leurs limites. Parmi les nombreuses questions de recherche dédiées à la conception proactive du couple produit-process, nous nous focaliserons sur la réutilisation des connaissances qui garantit la sensibilisation des concepteurs dans leurs activités de conception sur l'impact de leurs choix de conception sur une partie restreinte du cycle de vie du produit : l'assemblage.

La Figure 3.1 présente ce cadre au travers de trois strates : connaissance d'assemblage, contexte de conception pour l'assemblage et information de planification d'assemblage (prédiction) qui représentent les fondements de cette approche. En effet, le contexte est augmenté grâce aux relations d'assemblage dans l'environnement CAO, avec les informations issues des gammes de montage servant à exploiter des connaissances décomposées en trois éléments distincts. De fait, lesdites connaissances (connaissance conditionnelle - Ck, connaissance déclarative - Dk et connaissance procédurale - Pk) seront alors exploitées dans le contexte de conception augmenté.

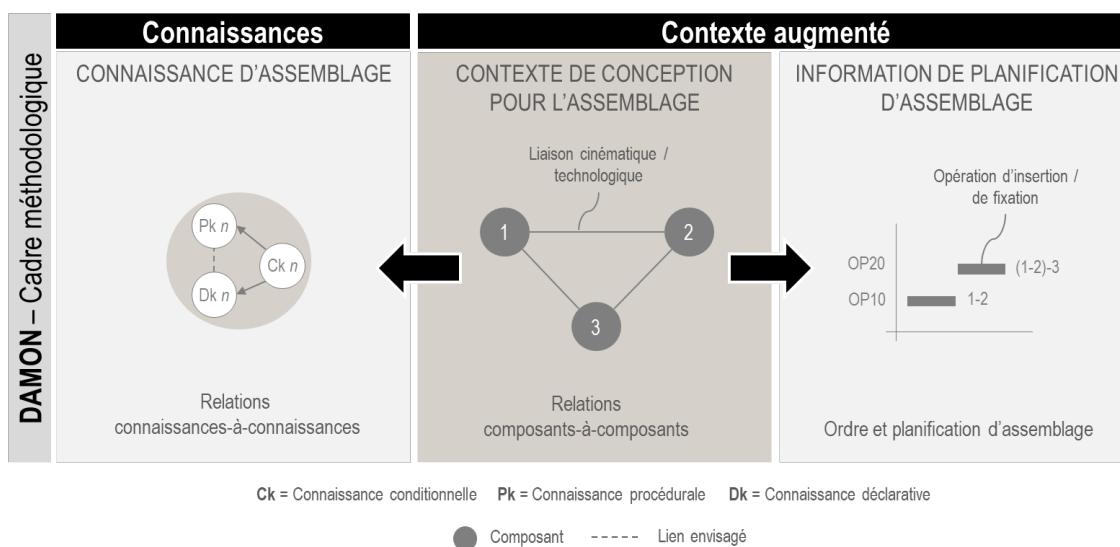


Figure 3.1 – Introduction du cadre méthodologique DAMON

Dès lors, il est nécessaire de distinguer ces connaissances pour faciliter non seulement la compréhension du concepteur mais aussi l'inférence au système d'information, et ce, en vue de cibler et répondre aux besoins. La Figure 3.2 présente l'intérêt de différencier ces connaissances de manière pertinente. L'objectif de cette distinction est double. En effet, elle tend à la fois à fluidifier le raisonnement pour diffuser les connaissances adéquates au contexte de conception et à l'assemblage, mais également à sensibiliser le concepteur sur le « *savoir-quoi* » (Dk) et le « *savoir-comment* » (Pk), grâce au « *savoir-quand* » et au « *savoir-pourquoi* » (Ck).

Actuellement, la réutilisation des connaissances en conception s'effectue avec l'ajout de règles et de paramètres par l'intermédiaire de templates ou d'applications externes à la CAO. Dès lors, le raisonnement servant à instancier les connaissances ne différencie pas le « *savoir-quoi* » et le « *savoir-comment* ». Le raisonnement en question, via les règles « *Si-Alors* » (If-Then) voire

« *Si-Alors-Sinon* » (If-Then-Else) soient en l'occurrence les Ck qui intègrent tout aussi bien les Dk que les Pk. Ainsi, notre contribution, consiste à des informations d'assemblage superposées aux informations de conception, afin de permettre la recherche de connaissances plus adaptée au contexte.

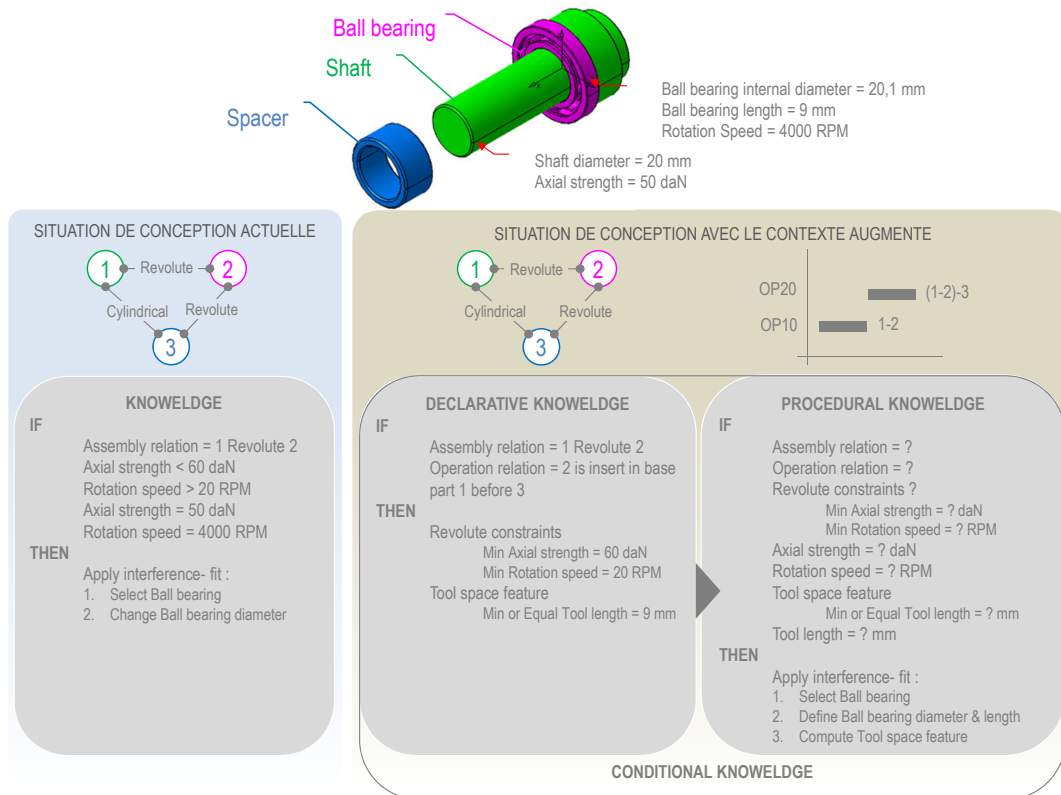


Figure 3.2 – Comparatif entre la situation de conception actuelle et la situation de conception dotée d'un « *contexte augmenté* »

Par exemple, la règle « *Si-Alors* » raisonnera d'abord sur Dk pour inférer des éléments géométriques, des paramètres avec des relations et des annotations. Sur ces éléments, viendra s'ajouter ensuite, une procédure pour les implémenter dans le « *contexte augmenté* » avec Pk. L'objectif de tout ceci étant de ne pas « *noyer* » les concepteurs de connaissances inappropriées au contexte. Il est important de comprendre que chaque étape de l'approche est essentielle pour exploiter les connaissances dans le PLM. Par conséquent, le but de nos travaux consiste à accroître le contexte afin de réutiliser de manière granulaire la connaissance et de faciliter ainsi, le raisonnement dans un système PLM.

Le chapitre actuel décrit en détail le cadre méthodologique appelé **DAMON**, dans le domaine de la conception pour l'assemblage, en se focalisant sur le lien entre le « *contexte augmenté* » et les connaissances. La présentation du cadre méthodologique est détaillée dans la première sous-section afin de rappeler ses objectifs dans sa vue d'ensemble. La seconde sous-section se base sur DAMON pour aborder en détail les étapes importantes. Nous verrons l'importance des liens entre les informations issues de la conception (présent) et celles issues de l'assemblage (futur), pour aboutir à la réutilisation des connaissances (passé).

3.2/ CADRE MÉTHODOLOGIQUE - DAMON

3.2.1/ OBJECTIF DU CADRE MÉTHODOLOGIQUE DAMON

Le cadre méthodologique proposé s'inspire des situations temporelles exprimées par [Bergson, 1919]. Interpelé par la conscience Henri Bergson, la définit de la manière suivante : « *la conscience est un trait d'union entre ce qui a été et ce qui sera, un pont jeté entre le passé et l'avenir* ». Celle-ci est donc une mémoire qui conserve et accumule le passé dans le présent. Au travers de la Figure 3.3, nous interprétons les situations temporelles (passé, présent, futur), pour les transposer à notre besoin. Ainsi, le présent est le point de départ servant à augmenter le futur. Le présent est constitué d'informations, d'exigences et de contraintes, etc. Les éléments du présent se superposent à ceux du futur généré (lequel est représenté par les domaines issus du cycle de vie du développement produit comme l'assemblage ou la fabrication). L'ensemble des éléments du présent et du futur viennent augmenter les informations à la fois du contexte de conception et de la prédiction. Ainsi, regarder dans le passé, les connaissances telles que des règles et des bonnes pratiques en lien avec cette augmentation, pour faciliter la prise de conscience à l'instant présent.

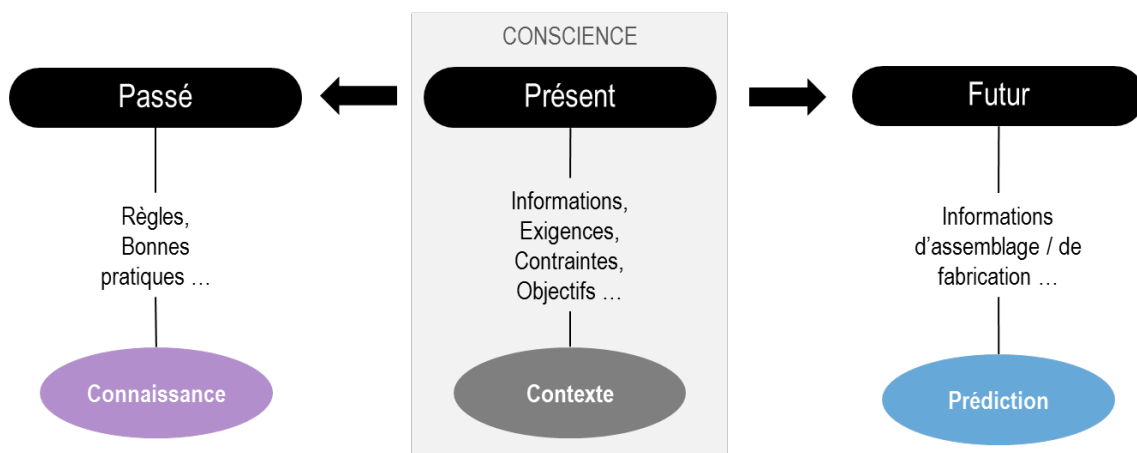


Figure 3.3 – Les trois situations temporelles appliquées au cadre méthodologique proposé

Le cadre méthodologique proposé et baptisé **DAMON** (**D**esign context **A**ugmentati**O**N framework) est décrit dans le domaine de la conception orientée vers l'assemblage. Ce cadre méthodologique pourrait-être étendu vers d'autres contextes en lien avec le cycle de vie du produit (fabrication, maintenance, utilisation, vente, etc.), cependant dans le cadre de notre travail de thèse, nous nous focaliserons uniquement sur le contexte d'assemblage. DAMON est composé de cinq grandes étapes présentées dans l'ordre chronologique par (1) définir, (2) générer, (3) actualiser, (4) requêter et (5) instancier. Toutes ces étapes sont illustrées dans la Figure 3.4 pour décrire les mécanismes de raisonnement avec la base de connaissances, la base de données de conception et la base de prédiction.

Ainsi, les objectifs du cadre méthodologique DAMON consistent à :

- Assurer la sensibilisation du concepteur sur les problématiques d'assemblage pendant son activité de conception ;
- Augmenter le contexte de conception en superposant les informations issues de l'ordre et de la planification d'assemblage, dans la continuité des travaux établis au sein de

notre laboratoire sur la théorie JANUS, dédiée à la « *méréotopologie spatiotemporelle* » [Gruhier *et al.*, 2015];

- Faciliter la compréhension du système d'information en impliquant un raisonnement lié à des connaissances décomposées en trois catégories, afin de garantir une granularité au « *juste nécessaire* » et une meilleure réutilisation des connaissances ;
- Impliquer plusieurs acteurs métiers tels que l'architecte produit, le gammiste, l'ingénieur process, le gestionnaire de connaissances dans le but de diffuser les informations et les connaissances aux concepteurs, par l'intermédiaire d'un médiateur dans un système PLM ;
- Établir des bases pour modéliser le raisonnement et permettre qu'il soit exploité par les systèmes d'informations, des solveurs, des moteurs d'inférence, etc. d'en interpréter les intentions et la situation dans la conception.

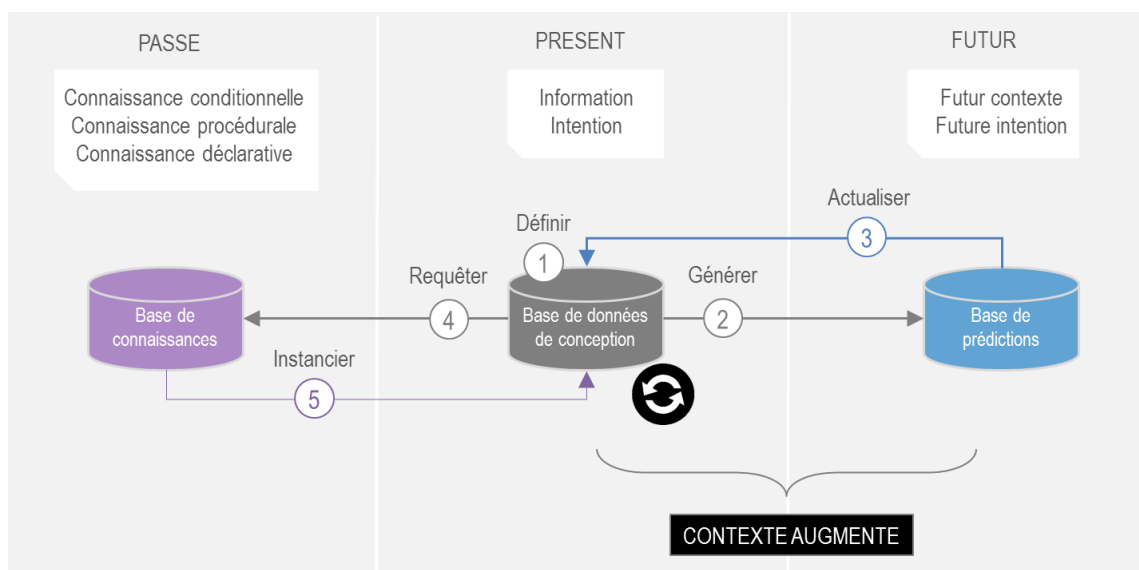


Figure 3.4 – « *Contexte augmenté* » pour la réutilisation des connaissances

La Figure 3.4 représente les différentes étapes de la définition du « *contexte augmenté* » pour la réutilisation des connaissances. Premièrement, (1) la définition de l'intention et de l'information est nécessaire pour générer le contexte de conception dans le présent. La deuxième étape consiste (2) à générer le futur contexte et les futures intentions. L'une des prédictions sélectionnées (3) actualise la base de contexte, en superposant au présent, le contexte avec le futur contexte, formant ainsi, le « *contexte augmenté* ». Dès lors, l'information issue du produit, qualifiée de « *spatiale* », est représentée par le contexte de conception, et est mise à jour avec les informations représentant le process considéré comme « *temporel* ». Ces informations, en devenant « *spatio-temporelles* » (4), permettent d'envoyer des requêtes dans la base de connaissances pour instancier les connaissances appropriées (5) dans le contexte de conception augmenté. La notion de « *contexte augmenté* » est désormais définie comme suit :

Une situation dans laquelle le contexte est complété par des prédictions, afin d'activer les connaissances appropriées, d'accroître la sensibilisation et la prise de conscience du concepteur sur l'impact de ses choix de conception sur le futur cycle de vie du produit.

L'activité de conception correspond à la phase où les connaissances sont instanciées, comme le

traduit la Figure 3.5 qui positionne cette activité de conception entre chaque situation de conception¹.

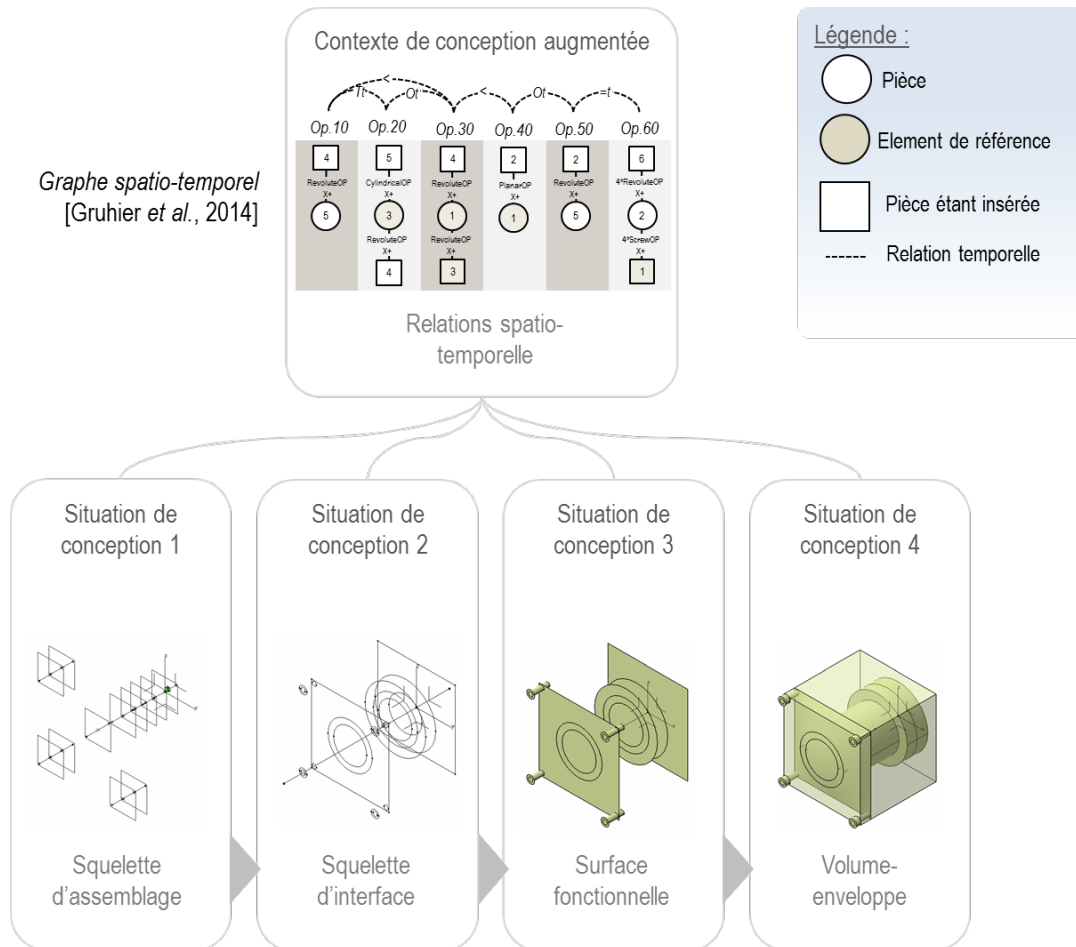


Figure 3.5 – Situation de conception en lien avec le contexte de conception augmentée

La situation de conception est définie comme « *une instance à un instant donné des informations contextuelles pertinentes* » [Dhouieb, 2016]. Dans notre cas, la situation est un instant donné qui représente le contexte augmenté de conception sur la base des relations du graphe spatio-temporel [Gruhier et al., 2014], autrement dit des relations de composants à composants (spatiales) superposées avec la planification de séquence d'assemblage (temporelles), afin de définir les espaces de conception (c'est-à-dire la définition des squelettes d'assemblage, des squelettes d'interface, des surfaces fonctionnelles et des volume-enveloppes). En d'autres termes, la situation de conception correspond à la représentation de l'intention du concepteur dans le contexte CAO. Les connaissances sont ainsi suggérées pendant le processus de conception, dans l'environnement CAO en partant des squelettes d'assemblage jusqu'aux volumes-enveloppe. L'intérêt des espaces de conception réside dans l'instanciation d'entités géométriques qui se cumulent (lignes, plans, surfaces, volumes, etc.), décrits par l'architecte produit et définis par le(s) concepteur(s),

1. Pour des raisons de visibilité, nous avons caché certains éléments géométriques dans chaque situation de conception. Par exemple, les plans n'apparaissent pas dans la situation de conception 2, dont les surfaces fonctionnelles en dépendent

dont le seul but est de délimiter les contraintes nécessaires à la conception de produits pour l'assemblage. Ceux-ci sont de plus en plus contraints au fil de l'évolution de la situation de conception, avec l'ajout de relations et de leurs paramètres, dont certaines connaissances ont des règles d'accessibilité, etc. (exemple : la délimitation du volume-enveloppe ne doit pas dépasser la hauteur de la surface pour des raisons d'accessibilité). Tout ceci va restreindre le concepteur lorsqu'il voudra enfreindre ses règles, d'où l'apparition d'un message d'erreur pour le prévenir. La sous-section suivante présente le cadre principal proposé illustrant d'un point de vue global l'articulation entre le « *contexte augmenté* » et la réutilisation des connaissances dans l'activité de conception.

3.2.2/ VUE D'ENSEMBLE DU CADRE MÉTHODOLOGIQUE DAMON

La Figure 3.6 décrit les étapes importantes et essentielles afin de réutiliser les connaissances. Les détails de ces étapes sont transposés dans le logigramme (Figure 3.7).

- **1. Définir** : à partir de la définition du contexte, l'architecte produit détermine les relations composants à composants avec les informations supplémentaires représentées par les liaisons cinématiques et technologiques ;
- **2. Générer** : la séquence d'assemblage est choisie par le gammiste en établissant les contraintes de précédence afin de générer la séquence en question en se basant selon le contexte de production. De ce choix est créée la planification d'assemblage (ou gamme de montage) pour laquelle des opérations d'assemblage sont complétées par l'ingénieur process (telles que les opérations de manipulation, d'insertion, de fixation, de contrôle, etc.). Dans notre cas, nous exploiterons seulement les opérations d'insertion et de fixation. Ces informations sont compilées et attribuées à chaque relation entre composants ;
- **3. Actualiser** : les informations de conception et d'assemblage sont compilées, réunissant les relations produit-process au sein d'un graphe spatio-temporel. Ce graphe comprend les liaisons de positionnement cinématique/technologique ainsi que la planification de la séquence d'assemblage. Le médiateur est l'initiateur de cette compilation ;
- **4. Requêter** : sur la base de ces informations, les connaissances (Ck, Pk et Dk) sont requêtées dans la base de connaissances, en recherchant les connaissances candidates avec le graphe spatio-temporel (contenant l'ordre d'assemblage, le type de liaisons cinématiques et technologiques, les pièces stabilisées, etc.). L'inférence propose sous forme de fonctions d'assemblage, des annotations (pour expliquer la règle de conception et la manière de l'appliquer), des relations avec les paramètres (soumis à des contraintes d'assemblage), auxquelles sont ajoutées des entités géométriques ;
- **5. Instancier** : les connaissances appropriées sont allouées au graphe spatio-temporel et instanciées dans l'activité de conception. La connaissance choisie par le concepteur est instanciée directement dans l'environnement CAO sur les espaces de conception. Un flux fonctionnel interprète l'intention du concepteur par le médiateur afin de les instancier de façon itérative.

Le cadre méthodologique DAMON est ainsi détaillé au travers des situations temporelles décrites dans les sous-sections suivantes (Figure 3.7).

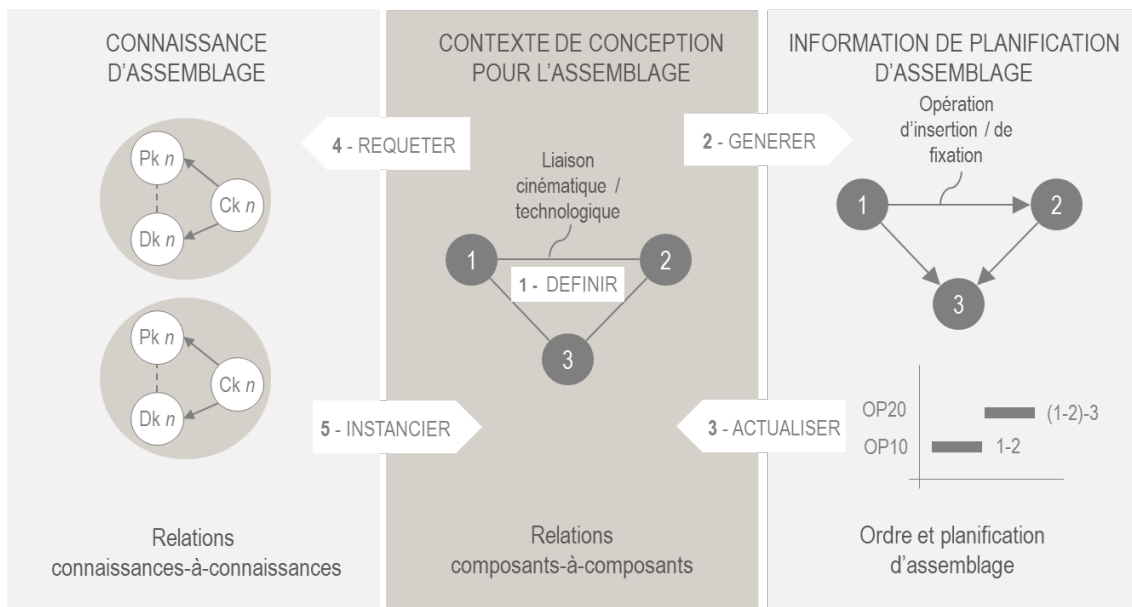


Figure 3.6 – Vue d'ensemble du cadre méthodologique DAMON

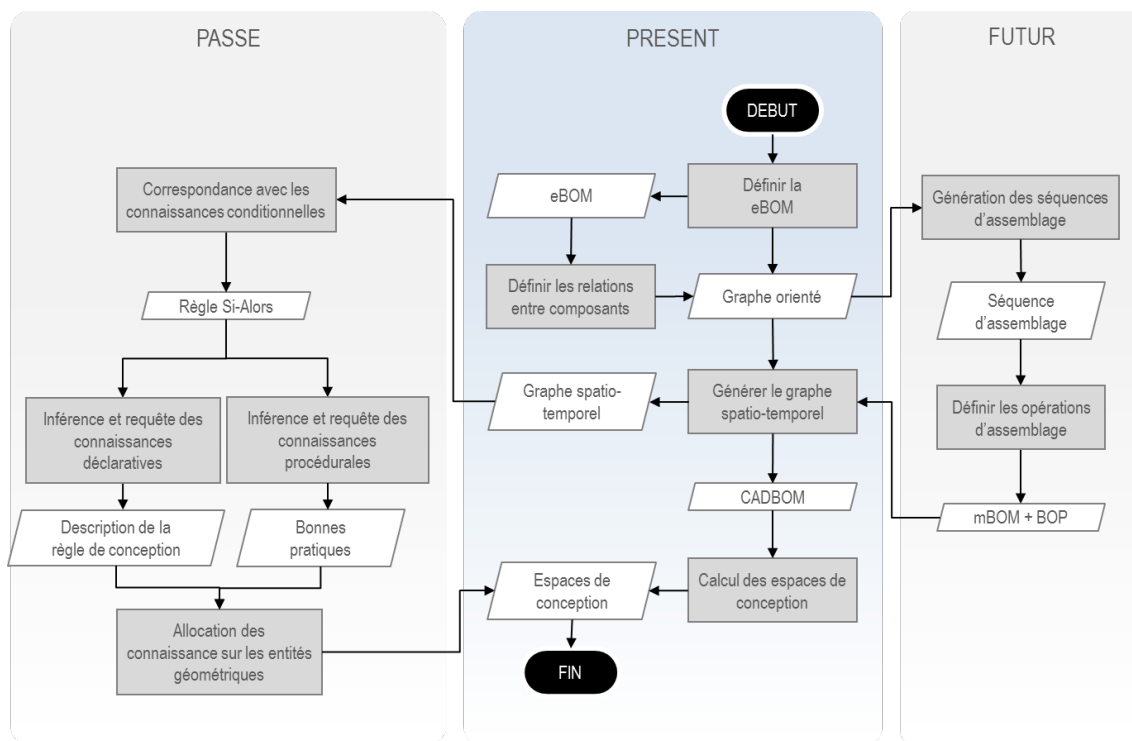


Figure 3.7 – Logigramme décrivant les différentes étapes du cadre méthodologique DAMON

Les parties prenantes ont des rôles bien définis pour interagir avec le médiateur, afin de proposer au concepteur les informations et les connaissances pertinentes pour son activité de conception. Le Tableau 3.1 suivant décrit le rôle de chaque partie prenante ainsi que leurs fonctions respectives.

Tableau 3.1 – Parties prenantes identifiées

Parties prenantes	Rôle / objectif	Fonctions
Architecte produit	Définir l'architecture du produit	Applique les fonctions et les exigences fonctionnelles du produit ; Définit les relations entre les composants ; Définit le contexte de conception. Définit la structure du produit.
Gammiste	Définir la gamme d'assemblage du produit	Génère et évalue la séquence d'assemblage admissible ; Définit le contexte de fabrication ; Sélectionne les sous-assemblages. Définit les contraintes de précédence.
Ingénieur process	Définir les technologies d'assemblage standards	Définit les opérations d'assemblage. Crée la gamme de montage.
Concepteur	Concevoir et définir la géométrie du produit	Définit les espaces de conception. Décrit le contexte de conception.
Gestionnaire des connaissances	Gérer les connaissances	Capture et met à jour les nouvelles connaissances. Vérifie et évalue la consistance des connaissances.
Médiateur	Gérer et diffuser les informations et les connaissances	Génère le contexte de conception approprié. Interprète les intentions et situations pour diffuser les connaissances.

3.3/ DESCRIPTION DE DAMON

3.3.1/ DÉFINITION DES RELATIONS DU PRODUIT VIA LE GRAPHE ORIENTÉ (ÉTAPE 1)

L'architecte produit définit les relations d'assemblage en suivant l'approche de conception descendante (Top-Down design) [Demoly *et al.*, 2011]. Il commence par énumérer les composants via la nomenclature produit (eBOM), dans laquelle il définit les liaisons mécaniques avec leurs fonctions et leurs exigences. Celles-ci sont définies dans un graphe orienté entre les contacts de chaque composant (Figure 3.8). Ainsi des relations de contact sont proposées entre composants avec des informations supplémentaires (liaisons cinématiques et technologiques) :

- **Contact** : relation de contact physique entre deux composants ;
- **Liaison cinématique** : information supplémentaire sur la relation de contact qui permet la description des contraintes de degrés de liberté (rotation et translation) pour chaque composant du produit ;
- **Liaison technologique** : information complémentaire sur la relation de contact qui permet la définition de l'assemblabilité du produit et donc sur la relation d'assemblage entre deux composants.

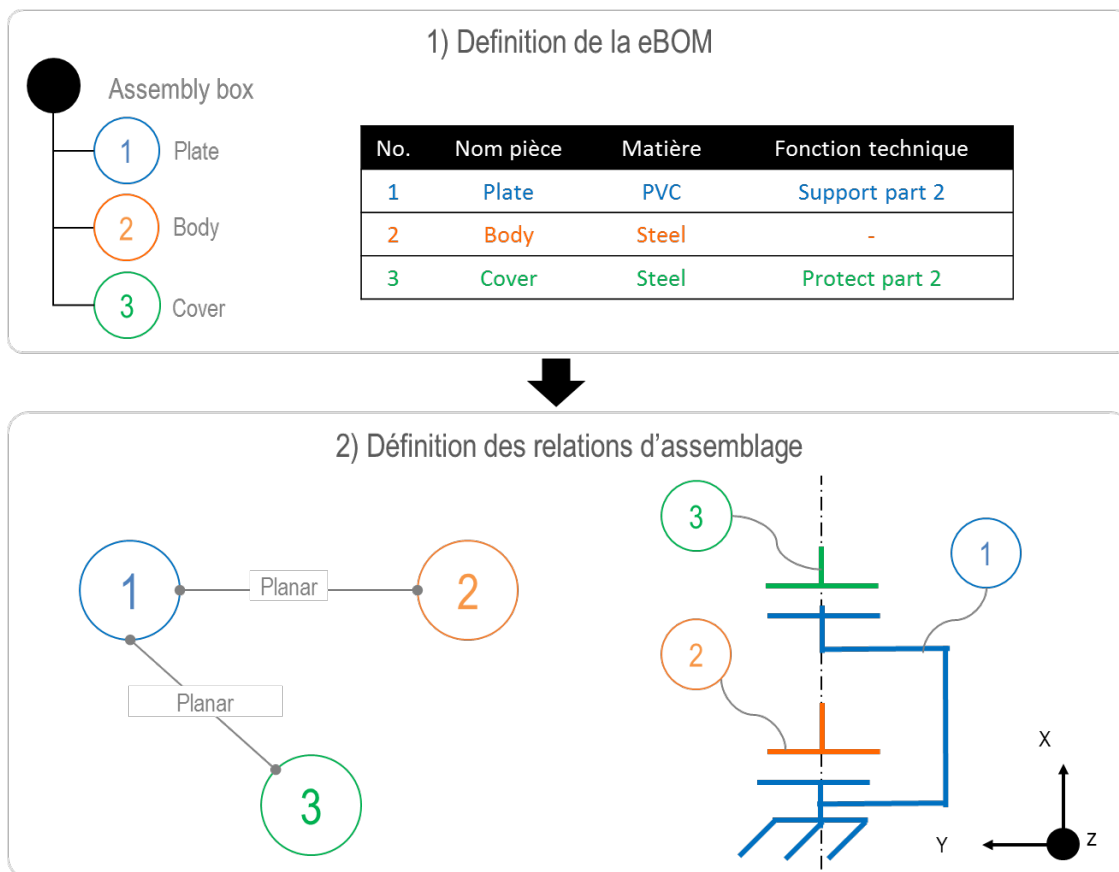


Figure 3.8 – Définition du contexte de conception - cas d'application choisi d'une boîte

La définition du contexte de conception est nécessaire pour générer le contexte futur. La section suivante présente la génération de la planification de la séquence d'assemblage basée sur le graphe orienté.

3.3.2/ GÉNÉRATION DE LA PLANIFICATION DES SÉQUENCES D'ASSEMBLAGE AVEC LES RELATIONS COMPOSANTS À COMPOSANTS (ÉTAPE 2)

Sur la base des relations de précédence, le futur contexte est généré en suggérant plusieurs solutions d'assemblage (Figure 3.9). L'algorithme ASDA développé au sein de notre laboratoire par les auteurs [Demoly *et al.*, 2011] est utilisé pour générer plusieurs séquences d'assemblage dans lesquelles des informations spécifiques sont mises en évidence :

- **L'ordre d'assemblage** : cette information décrit l'ordre chronologique dans lequel les composants du produit sont positionnés et assemblés, impliquant par conséquent l'ordonnement des opérations d'assemblage ;
- **Le type de sous-assemblage (série, parallèle, entremêlé, inclusif)** : cette information permet de décrire la nature du sous-assemblage liée au réseau de relations entre les composants, afin d'affecter des opérations d'assemblage ;
- **La dimension du sous-assemblage** : cette information décrit le nombre de composants contenus dans chaque sous-assemblage ;
- **Le niveau de sous-assemblage** : cette information présente les différents états du produit (configuration d'assemblage) dans le processus d'assemblage. De ce fait, nous aurons la

vision de l'imbrication des sous-ensembles du produit, et ainsi un aperçu de la complexité du produit ;

- **La pièce stabilisée (ou élément de référence)** : cette information décrit, pour chaque sous-assemblage et chaque niveau de sous-assemblage, un composant ou un sous-assemblage, stabilisé par un posage, sur lequel les autres composants/sous-ensembles devront se positionner et s'assembler progressivement. Cet élément est alors considéré comme primaire dans une ou plusieurs opérations d'assemblage consécutives. Il joue un rôle de support pour les autres composants/sous-ensembles et garde une orientation constante ;
- **Les directions d'assemblage** : cette information décrit les directions d'insertion en fonction de la stabilité des sous-assemblages. Par cette information, il est possible de déterminer pour une séquence d'assemblage générée le nombre de réorientations à effectuer.

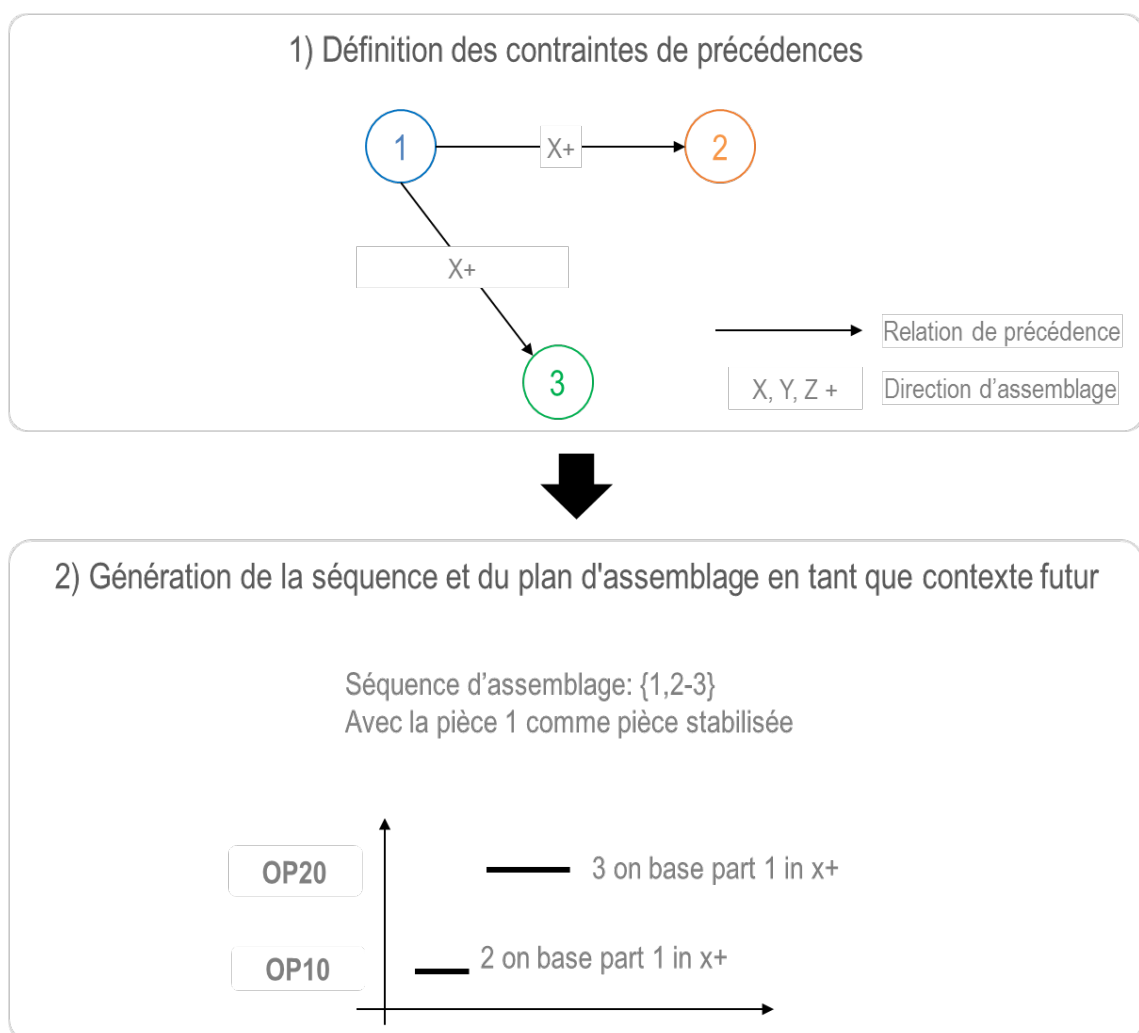


Figure 3.9 – Génération du scénario d'assemblage

Le fonctionnement de l'algorithme consiste à générer une matrice d'adjacence afin d'optimiser le nombre de composants et de définir les conditions de stabilité, les types d'assemblage, ainsi que les variantes. La matrice va générer toutes les combinaisons possibles en proposant les sous-assemblages sur plusieurs niveaux. La séquence d'assemblage est sélectionnée par le gammiste

puis complétée par l'ingénieur process. Le choix de la séquence d'assemblage est soumis à plusieurs critères de décisions en lien avec les contraintes de production. Ces choix dépendent notamment du temps d'assemblage et de la montabilité du produit avec les outils de production. Ainsi, une nomenclature de fabrication (**manufacturing Bill Of Materials - mBOM**) et des processus (**Bill Of Process - BOP**) est créée pour chaque relation en complément d'autres informations (structure d'assemblage, exigence de capacité, graphe de précédence, poste de travail, ressource, plan de montage, instruction, etc.) [Kretschmer *et al.*, 2017]. Le paragraphe suivant explique l'agrégation des informations afin de les formaliser pour représenter le « *contexte augmenté* ».

3.3.3/ ACTUALISATION DU CONTEXTE DE CONCEPTION (ÉTAPE 3)

Toutes les informations sont synthétisées dans un graphe spatio-temporel (Figure 3.10) afin de sensibiliser le concepteur sur l'impact de ses choix de conception de l'assemblage du produit. Pour la génération du graphe spatio-temporel, il est d'abord nécessaire de réaliser le modèle de données du système (graphe orienté et eBOM), pour ensuite créer le modèle de temporalité (séquence d'assemblage et mBOM+BOP). Dans ce cas, les relations de composants à composants (spatiales) et la planification de séquence d'assemblage (temporelles) sont superposées via les régions (objet) et leurs relations (primitives). Ces éléments sont issus de la théorie de la méréotopologie appelée **JANUS** (**J**oined **A**ware**N**ess and **U**nderstanding in assembly-oriented de**S**ign with mereotopology) [Demoly *et al.*, 2012] [Gruhier *et al.*, 2014]. Elle consiste à décrire et à établir les relations entre les objets et les primitives relatives aux informations produit-process.

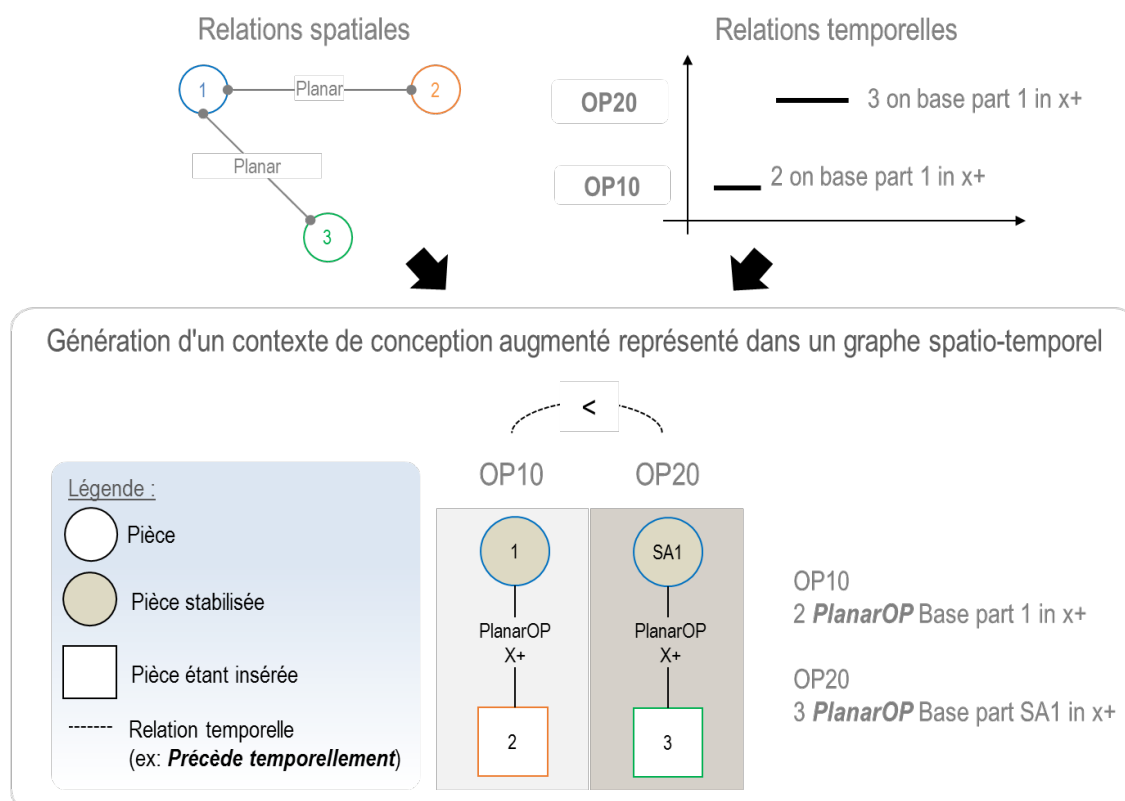


Figure 3.10 – Les informations spatiale et temporelle se chevauchent afin de former le « *contexte augmenté* »

Le graphe spatio-temporel, issu de la théorie JANUS consiste à connaître la définition dans l'espace (les composants mécaniques) et le temps (les opérations d'assemblage) l'assemblage des pièces. Les relations temporelles fournissent des informations sur l'ordre de montage entre chaque opération. Chacune de ces opérations est liée, par exemple, à des relations dénommées **Précédés temporellement (Temporally Precedes)** désignés par le sigle $<$, ou encore par le lien **équivalents Temporellement (Temporally Eqal)** symbolisés par le sigle $= t$. Une autre représentation concerne la pièce stabilisée dans laquelle les autres composants sont insérés. Au cours de l'activité de modélisation, les espaces de conception sont générés et alloués sur le graphe spatio-temporel au fur et à mesure que le concepteur les définit dans chaque situation de conception (Figure 3.11).

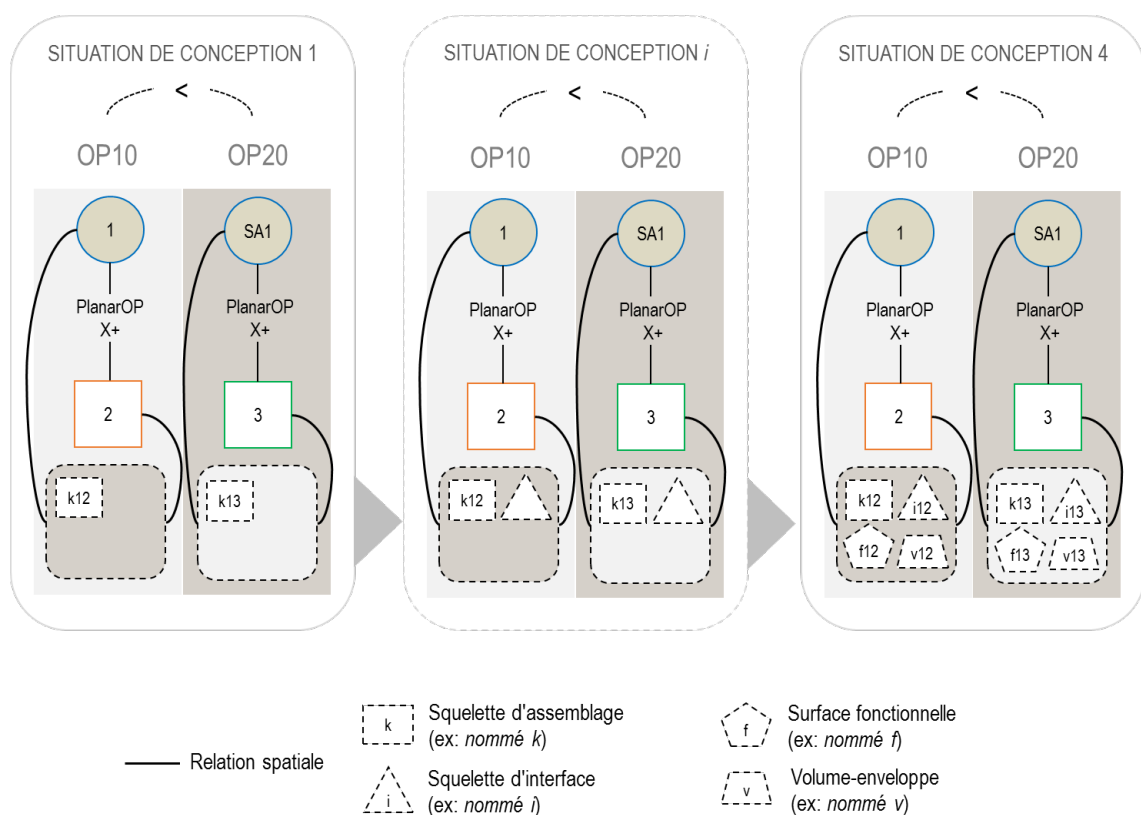


Figure 3.11 – Génération des entités géométriques dans chaque situation

Chaque espace de conception est généré dans l'environnement CAO, selon les intentions décrites par l'architecte produit. Ainsi, les espaces concernés sont définis par les concepteurs au fur et à mesure de l'évolution de la conception. C'est sur la base de ces éléments que les connaissances sont instanciées.

Les définitions des entités des espaces de conception sont basées sur la description spatio-temporelle de JANUS. Une telle description dite de positionnement des liaisons cinématique/technologique a été définie par les auteurs [Demoly *et al.*, 2012] [Gruhier *et al.*, 2014], nous avons donc étendu les définitions précédentes dans la Figure 3.12. L'exemple concerne un assemblage pivot en déplacement représenté par un volume balayé. Ce volume est défini par des régions

temporaires, c'est-à-dire que la pièce insérée va se déplacer au fur et à mesure selon la décomposition de ces régions. La liaison pivot (« *revoluteOP* » en anglais), se rattache au sigle OP signifiant opération. Cette liaison est représentée par des éléments géométriques pour définir l'espace de conception de la liaison. Autrement dit, elle se compose de deux squelettes d'assemblage (ligne et plan perpendiculaires), de quatre squelettes d'interface (deux cercles appartenant au plan, un cercle ou un rectangle appartenant au plan et un segment appartenant à la ligne), deux surfaces fonctionnelles (un cylindre et un circulaire) ainsi que de trois volumes-enveloppe (deux cylindres et un cylindre ou un cuboïde).

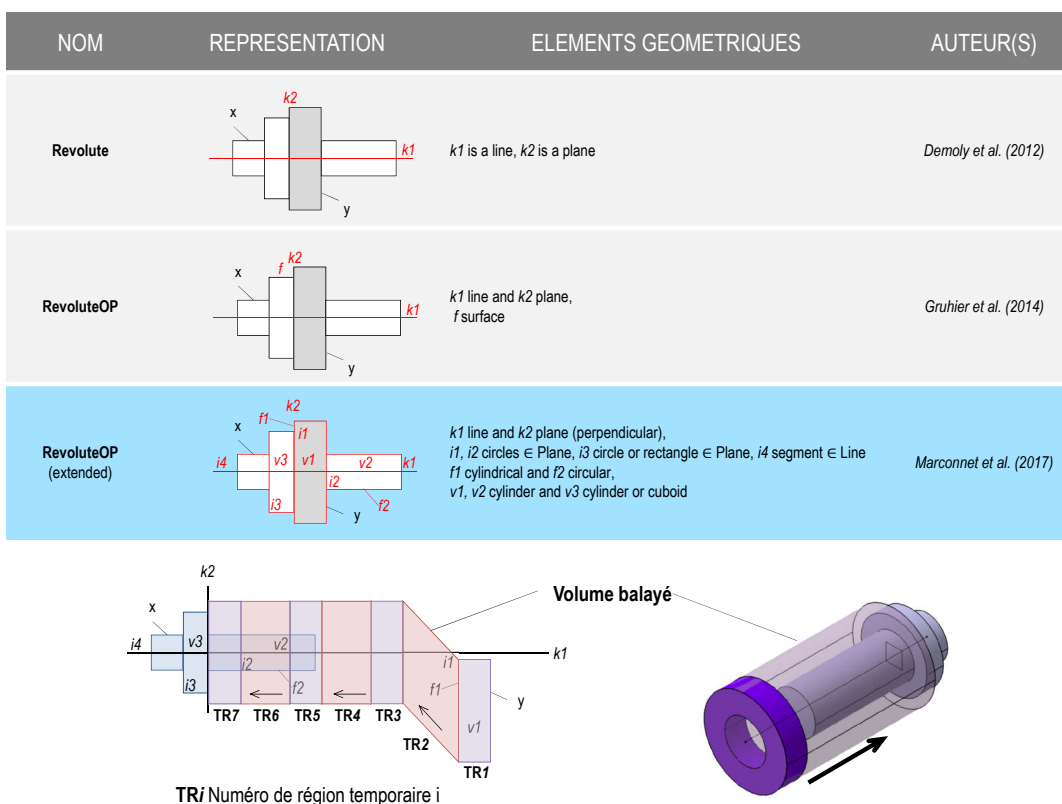


Figure 3.12 – Notre proposition d'amélioration de la théorie JANUS

La Figure 3.13 et la Figure 3.14 décrivent les espaces de conception pour les liaisons de positionnement cinématique/technologique que nous exploitons dans les travaux de recherche. En se basant sur les éléments géométriques, les connaissances sont requêtées et allouées en fonction de leur adéquation à chaque situation. La section suivante présente le raisonnement nécessaire pour chercher et inférer les connaissances appropriées.

POSITIONNEMENT LIAISON CINEMATIQUE	REPRESENTATION	RELATIONS SPATIO- TEMPORELLE	SITUATION 1 SQUELETTE D'ASSEMBLAGE (k)	SITUATION 2 SQUELETTE D'INTERFACE (i)	SITUATION 3 SURFACE FONCTIONNELLE (f)	SITUATION 4 VOLUME-ENVELOPPE (v)	REPRESENTATION CAO
RevoluteOP			k1 Line k2 Plane (Perpendicular)	i1, i2 Circle ∈ Plane i3 Circle or rectangle ∈ Plane i4 Segment ∈ Line	f1 Cylindrical f2 Circular	v1, v2 Cylinder v3 Cylinder or cuboid	
PrismaticOP			k1 Line k2 Plane (Perpendicular)	i1 Rectangle ∈ Plane	f1 Rectangular	v1 Cuboid	
ScrewOP			k1 Line k2 Plane (Perpendicular)	i1 Circle ∈ Plane i2 Segment ∈ Line i3 Circle or rectangle ∈ Plane	f1 Cylindrical	v1 Cylinder v2 Cylinder or cuboid	
CylindricalOP			k1 Line k2 Segment ∈ Line	i1 Circle	f1 Cylindrical	v1 Cylinder v2 Cylinder or cuboid	
SphericalOP			k1 Point	i1 Circle	f1 Spherical	v1 Cylinder v2 Cylinder or cuboid	
PlanarOP			k1 Plane (Perpendicular)	i1 Circle or rectangle ∈ Plane i2 Circle or rectangle ∈ Plane	f1 Circular or rectangular f2 Circular or rectangular	v1 Cylinder or cuboid	

Figure 3.13 – Description des définitions des espaces de conception pour le positionnement des liaisons cinématiques

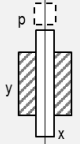
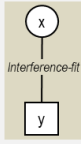
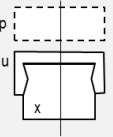
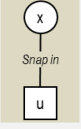
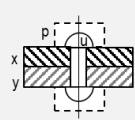
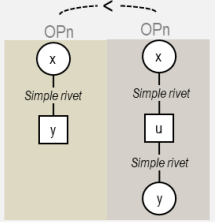
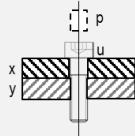
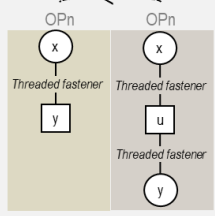
POSITIONNEMENT LIAISON TECHNOLOGIQUE	REPRESENTATION	RELATIONS SPATIO-TEMPORELLE	SITUATION 1 SQUELETTE D'ASSEMBLAGE (k)	SITUATION 2 SQUELETTE D'INTERFACE (i)	SITUATION 3 SURFACE FONCTIONNELLE (f)	SITUATION 4 VOLUME-ENVELOPPE (v)
Interference-fit			k1 Line k2 Segment ∈ Line	i1 Circle	f1 Cylindrical	v1 Cylinder v2 Cylinder or cuboid
Snap in			k1 Line k2 Plane (Perpendicular)	i1 Rectangle ∈ Plane	f1 Rectangular	v1 Cuboid
Simple rivet			k1 Plane (Perpendicular) k2 Line k3 Plane (Perpendicular) k4 Line k5 Plane (Perpendicular)	i1 Circle or rectangle ∈ Plane i2 Circle or rectangle ∈ Plane i3, i4 Circle ∈ Plane i5 Circle or rectangle ∈ Plane i6 Segment ∈ Line i7, i8 Circle ∈ Plane i9 Circle or rectangle ∈ Plane i10 Segment ∈ Line	f1 Circular or rectangular f2 Circular or rectangular f3 Cylindrical f4 Circular f5 Cylindrical f6 Circular	v1 Cylinder or cuboid v2, v3 Cylinder v4 Cylinder or cuboid v5, v6 Cylinder v7 Cylinder or cuboid
Threaded fastener			k1 Plane (Perpendicular) k2 Line k3 Plane (Perpendicular) k4 Line k5 Plane (Perpendicular)	i1 Circle or rectangle ∈ Plane i2 Circle or rectangle ∈ Plane i3, i4 Circle ∈ Plane i5 Circle or rectangle ∈ Plane i6 Segment ∈ Line i7 Circle ∈ Plane i8 Segment ∈ Line i9 Circle or rectangle ∈ Plane	f1 Circular or rectangular f2 Circular or rectangular f3 Cylindrical f4 Circular f5 Cylindrical	v1 Cylinder or cuboid v2, v3 Cylinder v4 Cylinder or cuboid v5 Cylinder v6 Cylinder or cuboid

Figure 3.14 – Description des définitions des espaces de conception pour le positionnement des liaisons technologiques

3.3.4/ REQUÊTE DES CONNAISSANCES POSSIBLES (ÉTAPE 4)

Les informations provenant du graphe spatio-temporel permettent d'inférer et de requêter à la fois sur les relations spatiales et temporelles pour allouer les connaissances (c'est-à-dire D_k , P_k , et C_k) (Figure 3.15). Le médiateur recherche toutes les relations candidates, pour lesquelles le gestionnaire de connaissances a formalisé et stocké les connaissances. Les conditions de la recherche sont établies sur l'intention et la situation représentées par le positionnement des liaisons cinématiques/technologiques ainsi que l'ordre d'assemblage. Si les conditions définies par le gestionnaire de connaissances correspondent entre le graphe spatio-temporel et les connaissances, alors les connaissances sont ainsi greffées au fur et à mesure de l'avancée du concepteur. En conséquence, la définition des entités des espaces de conception représente les intentions du concepteur dans l'environnement CAO.

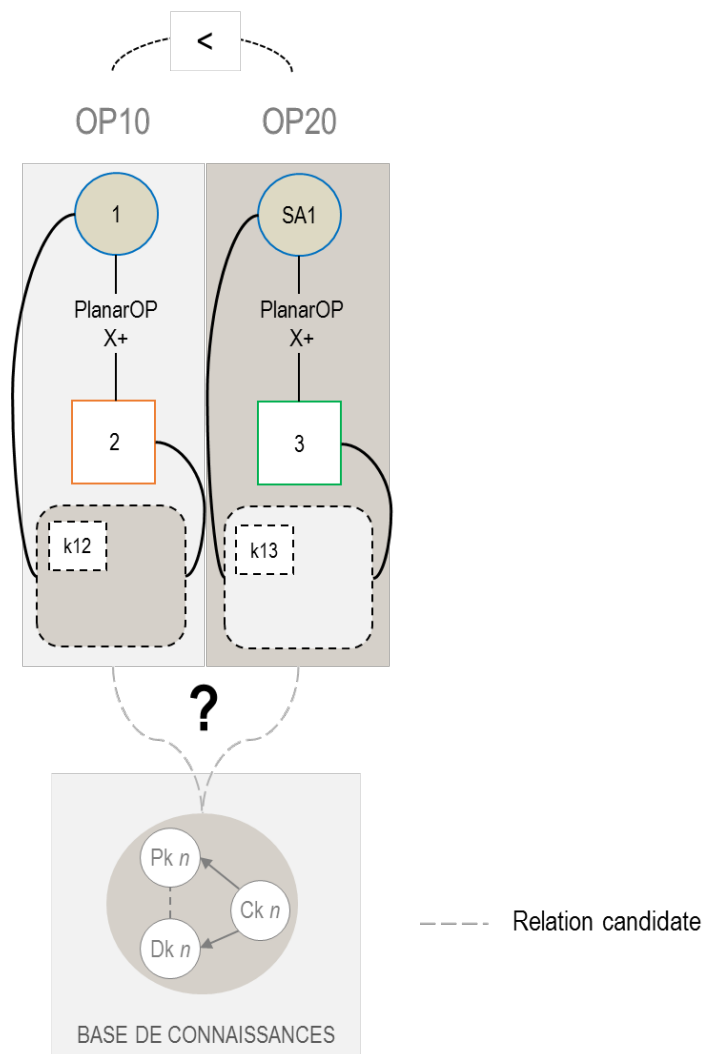
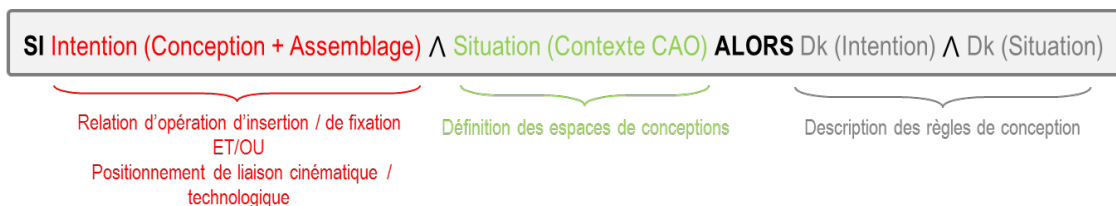


Figure 3.15 – Correspondance des relations candidates entre les connaissances et le graphe spatio-temporel

Ces conditions de recherche sont nécessaires pour savoir quand et pourquoi utiliser les différents

types de connaissance : Dk et Pk via Ck (Figure 3.16). Sur la base des règles Si-Alors (If-Then), le premier raisonnement commence par l'envoi de requêtes pour trouver Dk, afin de trouver la description des règles de conception appropriée. Les résultats obtenus, permettent avec le second raisonnement d'extraire la Pk correspondante, pour indiquer comment implémenter Dk dans le « *contexte augmenté* ». Celui-ci propose les bonnes pratiques (ou procédures) pour appliquer Dk. Ck est le mécanisme qui raisonne autour de Dk et Pk, représentées par les règles Si-Alors, en effectuant des inférences et des requêtes selon l'intention et la situation du concepteur dans l'environnement CAO. La capitalisation des règles et des requêtes est actuellement écrite et formalisés par des spécialistes sur la base de retours d'experts. Pour vérifier la cohérence de ces règles, elles sont testées dans la CAO sur quatre types de cas d'étude d'assemblage représentant un assemblage en série, un assemblage entremêlé, un assemblage inclusif/obligatoire et un assemblage en parallèle. Ces quatre versions représentent différentes manières d'insérer des composants.

1. PREMIER RAISONNEMENT AVEC Dk



2. DEUXIEME RAISONNEMENT AVEC Pk



Figure 3.16 – Composition d'une connaissance conditionnelle - Ck

La Figure 3.17 et la Figure 3.18 traduisent plusieurs exemples de connaissances capitalisées qui représentent des fonctions d'assemblage. Les connaissances sont décomposées pour faciliter l'articulation entre Dk et Pk via le formalisme Ck. Par exemple, la fonction d'accessibilité (accessibility feature) présente une connaissance qui facilite l'insertion du composant, lorsque celui-ci est entouré de parois ne permettant pas de le voir ou de le manipuler. Ainsi, cette fonctionnalité génère un volume pour fournir cet accès et la vision nécessaire à l'opérateur pour insérer le composant.

Ici, le premier raisonnement consiste à rechercher tous les éléments liés à la fonction d'accessibilité. Celle-ci comprend deux annotations, un volume généré et deux relations avec leurs paramètres. Les deux annotations contiennent la description de la règle de conception et la bonne pratique pour l'appliquer. Celles-ci doivent s'instancier sur une vue de face (ici sur le volume-enveloppe représenté par un cylindre ou un cuboïde). Le volume-enveloppe en question a besoin d'éléments de squelette d'assemblage (une ligne et un plan perpendiculaire) pour être généré. Au volume-enveloppe viennent s'ajouter des relations et des paramètres qui vont limiter la modélisation des squelettes d'interface avec la règle qui s'applique à la fois à la largeur (Width) mais aussi à la hauteur (Height). Tout ceci est à prendre à considération avec la relation : 7 mm < largeur et/ou hauteur < 80 mm. Cette relation est issue de règles ergonomiques pour le passage de la main.

Quant au second raisonnement, il prévoit d'appliquer tous ces éléments dans le « *contexte augmenté* » en se basant sur le résultat de la Dk, pour inférer et formuler une requête sur l'ordre ou

la procédure d'application de ces éléments. Ainsi, l'application s'effectue en sélectionnant dans un premier temps, les éléments géométriques issus de la situation de conception à savoir, la ligne et le plan perpendiculaire pour générer le volume. Puis, la prochaine étape concerne l'instanciation du cylindre ou du cuboïde, suivi de l'ajout des relations et des paramètres sur le volume, et enfin de l'ajout des annotations sur le volume. La dernière étape consiste à calculer et vérifier par le concepteur les paramètres de largeur et de hauteur.

NOM	DESCRIPTION - ILLUSTRATION	DESCRIPTION DE LA RÉGLE DE CONCEPTION (Dk)	BONNE PRATIQUE (Pk)
Accessibility feature		<p>IF</p> <p>(OPn : y is insert in base part x : All kinematic pair positioning) temporarily precedes (OPn : z is insert in base part x : All kinematic pair positioning) with the same assembly direction</p> <p>AND</p> <p>Assembly skeleton elements: 1 line and 1 plane (perpendicular)</p> <p>THEN</p> <p>1 annotation on front view (volume): Accessibility feature represented by a volume, is apply in order to ease access and vision of part</p> <p>AND</p> <p>2 relations and parameters: Width (mm) = 7mm < Width < 80mm Height (mm) = 7mm < Height < 80mm</p> <p>AND</p> <p>1 cylinder or cuboid volume ∈ plane following assembly direction</p>	<p>IF</p> <p>1 annotation on front view (volume): Accessibility feature represented by a volume, is apply in order to ease access and vision of part</p> <p>AND</p> <p>2 relations and parameters: Width (mm) = 7mm < Width < 80mm Height (mm) = 7mm < Height < 80mm</p> <p>AND</p> <p>1 cylinder or cuboid volume ∈ plane following assembly direction</p> <p>THEN</p> <p>1 annotation on front view (volume):</p> <ol style="list-style-type: none"> Select line and plane Generate cylinder or cuboid volume Add relations and parameters Width and Height Add annotations on front view Compute accessibility feature
Chamfer feature		<p>IF</p> <p>OPn : y is insert in x : CylindricalOP or RevoluteOP or ScrewOP</p> <p>AND</p> <p>Volume-envelope element: 1 cylinder or cuboid</p> <p>THEN</p> <p>1 annotation on front view (volume): Chamfer feature is apply in order to ease insertion of insert part</p> <p>AND</p> <p>2 relations and parameters: Diameter 1 (mm) < Diameter 2 (mm) Clearance (mm) = (Diameter 2 (mm) - Diameter 1 (mm)) / Diameter 2 (mm)</p> <p>AND</p> <p>1 chamfer feature on volume, following assembly direction</p>	<p>IF</p> <p>1 annotation on front view (volume): Chamfer feature is apply in order to ease insertion of insert part</p> <p>AND</p> <p>2 relations and parameters: Part diameter 1 (mm) < Diameter 2 (mm) Clearance (mm) = (Diameter 2 (mm) - Diameter 1 (mm)) / Diameter 2 (mm)</p> <p>AND</p> <p>1 chamfer feature on volume, following assembly direction</p> <p>THEN</p> <p>1 annotation on front view (volume):</p> <ol style="list-style-type: none"> Select axial face of volume x Select radial face of volume Generate chamfer feature Add relations and parameters Clearance, Diameter 1 and Diameter 2 Add annotations on front view Compute chamfer feature
Self-aligning feature		<p>IF</p> <p>(OPn : y is insert in Base part x : PlanarOP) temporarily precedes (OPn : u is insert in parts x and y : CylindricalOP or RevoluteOP or ScrewOP)</p> <p>OR</p> <p>(OPn : y is insert in Base part x : PlanarOP) temporarily precedes (OPn : u is insert in part x : CylindricalOP or RevoluteOP or ScrewOP) temporarily precedes (Op.n : u is insert in part y : CylindricalOP or RevoluteOP or ScrewOP)</p> <p>AND</p> <p>Assembly skeleton elements: 1 line and 1 plane (perpendicular)</p> <p>Interface skeleton element: 1 circle or rectangle ∈ Plane</p> <p>THEN</p> <p>1 annotation on front view (surface): Self-aligning feature is apply in order to ease alignment of insert part</p> <p>AND</p> <p>1 circular or rectangular surface following assembly direction</p>	<p>IF</p> <p>1 annotation on front view (surface): Self-aligning feature is apply in order to ease alignment of insert part</p> <p>AND</p> <p>1 circular or rectangular surface following assembly direction</p> <p>THEN</p> <p>1 annotation on front view (surface):</p> <ol style="list-style-type: none"> Select line and plane Coincide circle or rectangle interface Add annotations on front view Generate circular or rectangular surface following assembly direction
Reorientation feature		<p>IF</p> <p>(OPn : u is insert in base part x : All kinematic pair positioning) temporarily precedes (OPn : v is insert in base part x : All kinematic pair positioning) with the same assembly direction</p> <p>OR</p> <p>(OPn : u and v is insert in base part x : All kinematic pair positioning) with the same assembly direction</p> <p>AND</p> <p>Assembly skeleton elements: 1 line or 1 plane (perpendicular) or 1 point</p> <p>THEN</p> <p>1 annotation on front view (skeleton): Reorientation of base part x is apply in order to reorient parts, by following the same assembly direction with the insert parts</p> <p>AND</p> <p>1 plane (perpendicular) and 1 line constraint in the positive assembly direction of the assembly coordinate system fixture</p> <p>1 assembly coordinate system fixture</p>	<p>IF</p> <p>1 annotation on front view (skeleton): Reorientation of base part x is apply in order to reorient parts, by following the same assembly direction with the insert parts</p> <p>AND</p> <p>1 plane (perpendicular) with line constraint in the positive assembly direction of the assembly coordinate system fixture</p> <p>1 assembly coordinate system fixture</p> <p>THEN</p> <p>1 annotation on front view (skeleton):</p> <ol style="list-style-type: none"> Select line or plane (perpendicular) or point Generate plane (perpendicular) and line Coincide plane Generate assembly coordinate system fixture Coincide assembly skeleton axis system of fixture with assembly skeleton axis system Add annotations on front view Change orientation of assembly skeleton axis system

Figure 3.17 – Représentation des connaissances (1)

NOM	DESCRIPTION - ILLUSTRATION	DESCRIPTION DE LA RÈGLE DE CONCEPTION (Dk)	BONNE PRATIQUE (Pk)
Joining path feature		<p>IF</p> <p>OPn : y is insert in base part x : RevoluteOP</p> <p>AND</p> <p>Assembly skeleton elements: 1 line and 1 plane</p> <p>Functional surface elements: 1 circular or rectangular or cylindrical</p> <p>THEN</p> <p>1 annotation on front view (volume): Joining path feature: reduces the time and the friction of the insert part</p> <p>AND</p> <p>1 relation and parameters: Diameter 1 (mm) > Diameter 2 (mm)</p> <p>AND</p> <p>1 plane (perpendicular) following assembly direction</p> <p>1 cylinder or cuboid volume ∈ Plane following assembly direction</p>	<p>IF</p> <p>1 annotation on front view (volume): Joining path feature reduces the time and the friction of the insert part</p> <p>AND</p> <p>1 relation and parameters: Diameter 1 (mm) > Diameter 2 (mm)</p> <p>AND</p> <p>1 plane (perpendicular) following assembly direction</p> <p>1 cylinder or cuboid volume ∈ Plane following assembly direction</p> <p>THEN</p> <p>1 annotation on front view (volume):</p> <ol style="list-style-type: none"> Select plane Coincide circular or rectangular or cylindrical interface Generate cylinder or cuboid volume Add relation and parameters Diameter 1 and Diameter 2 Add annotations on front view Compute joining path feature
Tool space feature		<p>IF</p> <p>(OPn : y is insert in base part x : All technological pair positioning) temporarily precedes (OPn : u is insert in x and y : All technological pair positioning) with the same assembly direction</p> <p>OR</p> <p>(OPn : u is insert in x and y : All technological pair positioning) with the same assembly direction</p> <p>AND</p> <p>Assembly skeleton elements: 1 line and 1 plane</p> <p>Interface skeleton elements: 3 circle or rectangle ∈ Plane</p> <p>THEN</p> <p>1 annotation on front view (volume): Apply tool space feature to facilitate the space available for the tool</p> <p>AND</p> <p>2 relations and parameters: Length 1 (mm) ≥ Length 2 (mm) Width (mm) ≥ Diameter (mm)</p> <p>AND</p> <p>1 cylinder or cuboid volume ∈ Plane following assembly direction</p>	<p>IF</p> <p>1 annotation on front view (volume): Apply tool space feature to facilitate the space available for the tool</p> <p>AND</p> <p>2 relations and parameters: Length 1 (mm) ≥ Length 2 (mm) Width (mm) ≥ Diameter (mm)</p> <p>AND</p> <p>1 cylinder or cuboid volume ∈ Plane following assembly direction</p> <p>THEN</p> <p>1 annotation on front view (volume):</p> <ol style="list-style-type: none"> Select line and plane Coincide circle or rectangle interface Generate cylinder or cuboid volume Add relations and parameters Length 1, Length 2, Width, and Diameter Add annotations on front view Compute tool space feature
Tapered end feature		<p>IF</p> <p>OPn : y is insert in base part x : Interference-fit</p> <p>AND</p> <p>Assembly skeleton element: 1 line</p> <p>Interface skeleton elements: 1 circle or rectangle ∈ Plane</p> <p>THEN</p> <p>1 annotation on front view (surface): Tapered end feature must be apply in order to ease alignment of insert part</p> <p>AND</p> <p>1 circular or rectangular or cylindrical on 1 circle or rectangle ∈ Plane</p>	<p>IF</p> <p>1 annotation on front view (surface): Tapered end feature must be apply in order to ease alignment of insert part</p> <p>AND</p> <p>1 circular or rectangular or cylindrical on 1 circle or rectangle ∈ Plane</p> <p>THEN</p> <p>1 annotation on front view (surface):</p> <ol style="list-style-type: none"> Select circle or rectangle interface Generate circular or rectangular surface Add annotations on front view
Rivet-to-edge-dimension feature		<p>IF</p> <p>(OPn : y is insert in base part x : Simple rivet) temporarily precedes (OPn : u is insert in x and y : Simple rivet)</p> <p>AND</p> <p>Assembly skeleton elements: 1 line and 1 plane</p> <p>Interface skeleton elements: 1 circle or rectangle ∈ Plane</p> <p>THEN</p> <p>1 annotation on front view (surface): Apply rivet-to-edge-dimension feature to give the minimum space available for the tool</p> <p>AND</p> <p>1 relation and parameter: 1,5mm to 8 x Distance (mm)</p> <p>1 offset constraint between Line and Surface</p> <p>AND</p> <p>1 circular or rectangular or cylindrical on 1 circle or rectangle ∈ Plane</p>	<p>IF</p> <p>1 annotation on front view (surface): Apply rivet-to-edge-dimension feature to give the minimum space available for the tool</p> <p>AND</p> <p>1 relation and parameter: 1,5mm to 8 x Distance (mm)</p> <p>1 offset constraint between Line and Surface</p> <p>AND</p> <p>1 circular or rectangular or cylindrical on 1 circle or rectangle ∈ Plane</p> <p>THEN</p> <p>1 annotation on front view (surface):</p> <ol style="list-style-type: none"> Select line and plane Generate circular or rectangular surface Add relation and parameter Distance Add annotations on front view Compute rivet-to-edge-dimension feature

Figure 3.18 – Représentation des connaissances (2)

Plusieurs exemples sont présentés dans la Figure 3.19 où la géométrie diffère selon les spécificités du graphe spatio-temporel (ordre d'assemblage, liaison cinématique/technologique, etc.). A titre d'illustration, l'exemple avec la fonction d'auto-alignement propose une différence notable, en fonction de la pièce stabilisée et de la direction d'assemblage entre l'exemple 1 et 2, sur la génération de protubérances facilitant l'insertion de la pièce u. Une surface fonctionnelle est générée pour représenter cette protubérance en fonction du sens d'insertion. Nous pouvons ainsi noter que la géométrie est influencée selon ses spécificités.

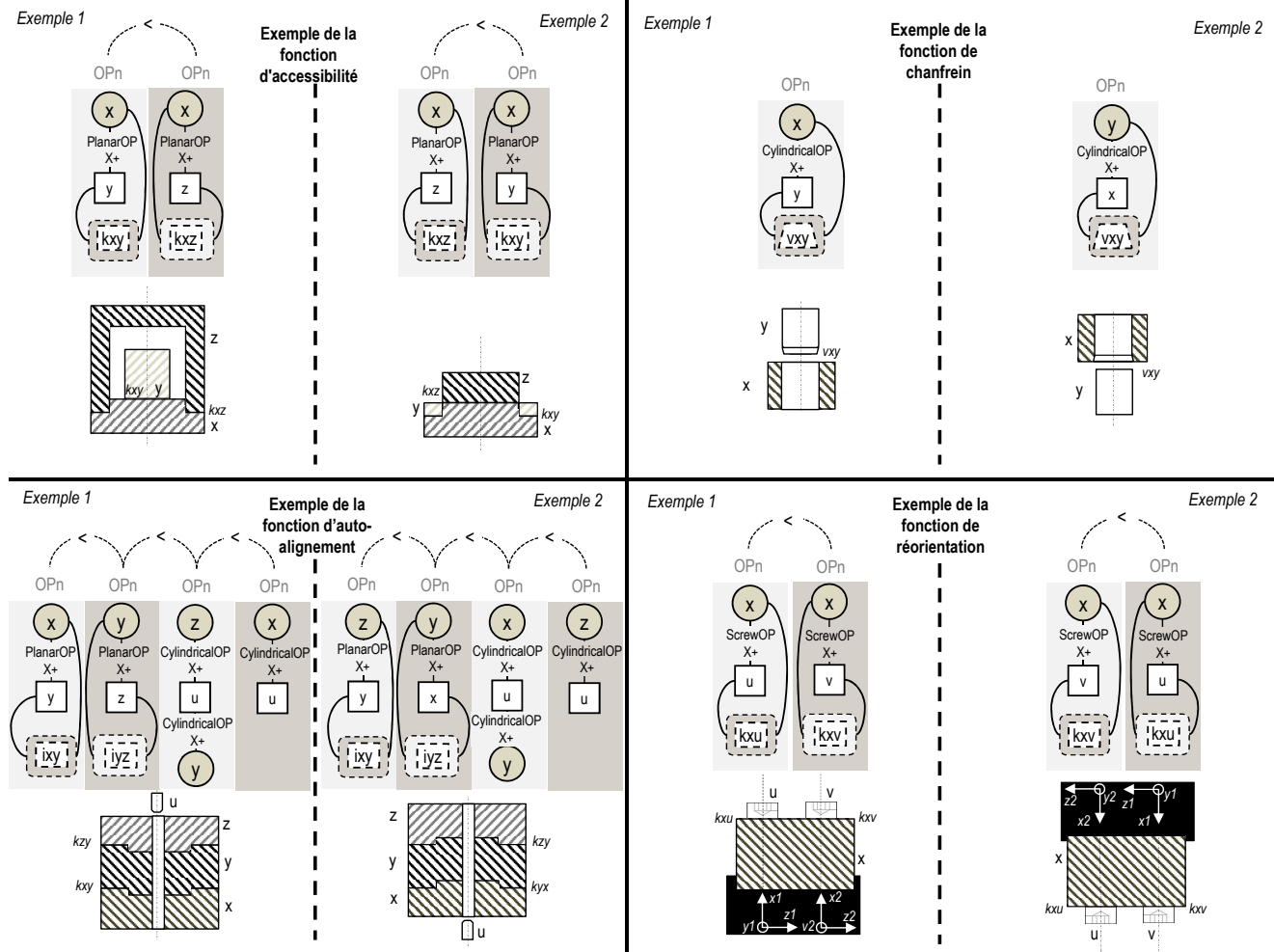


Figure 3.19 – Exemples de spécifications appliquées avec les connaissances

Pour éviter une sur-représentation des connaissancesinstanciées sur les relations du graphe spatio-temporel, il est nécessaire de gérer leur activation ou leur suppression au travers de règles. Nous exploiterons le chaînage de règles spécifiques pour l'instanciation de certaines connaissances (Figure 3.20). Pour comprendre l'intérêt du chaînage, un exemple simplifié est présenté. Il consiste à activer une autre fonction (ici la fonction de chanfrein) sur la fonction initiale d'auto-alignement :

Si

(OP10 : La pièce y est insérée dans la pièce stabilisée x) qui précède temporellement (OP20 : la pièce u insérée dans les pièces x et y)

Alors

La fonction d'auto-alignement est appliquée pour faciliter l'alignement de la pièce insérée u.

La fonction de chanfrein peut ainsi être activée :

Si

(OP10 : La pièce y est insérée dans la pièce stabilisée x) qui précède temporellement (OP20 : la pièce u insérée dans les pièces x et y)

Et

La fonction d'auto-alignement est appliquée pour faciliter l'alignement de la pièce d'insertion u.

Alors

La fonction de chanfrein est appliquée pour faciliter l'insertion de la pièce u

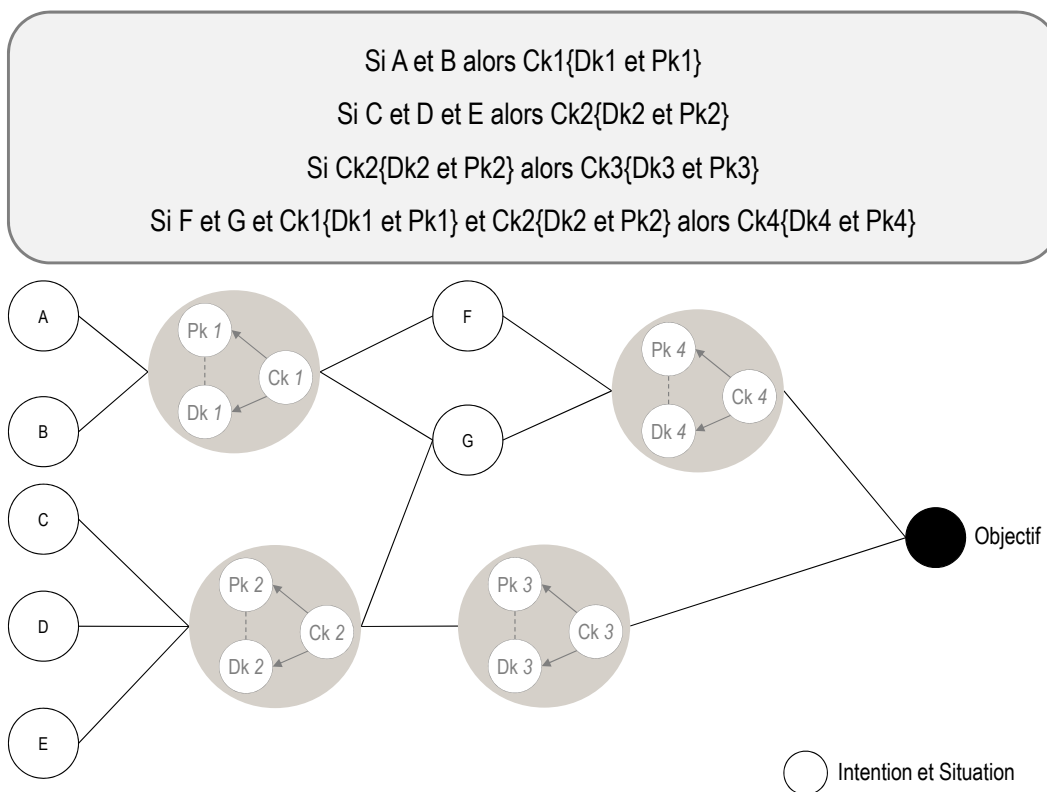


Figure 3.20 – Chaînage de règles

Plusieurs règles sont établies afin de décider l'instanciation ou non des connaissances.

Règle 1 : Concaténation des mêmes connaissances

Si

La même connaissance est instanciée dans la même opération

Alors

Concaténer ladite connaissance en une seule connaissance sur la pièce insérée

Règle 2 : Instanciation de la fonction de chanfrein

Si

La fonction d'auto-alignement est instanciée dans une opération

Alors

Instancier la fonction de chanfrein dans la même opération pour la pièce insérée

Règle 3 : Suppression de la fonction d'accessibilité

Si

La fonction d'accessibilité est instanciée dans la même opération que la fonction d'espace outil

Alors

Supprimer la fonction d'accessibilité

Dans la section suivante, nous identifierons la définition des espaces de conception dans le graphe spatio-temporel pour allouer et instancier la connaissance correspondante.

3.3.5/ INSTANCIATION DES CONNAISSANCES APPROPRIÉES DANS L'ACTIVITÉ DE CONCEPTION (ÉTAPE 5)

Le concepteur a l'opportunité de prendre conscience des connaissances pertinentes à réutiliser. Pour cela, les connaissances sont proposées sous forme de templates² construits en fonction de l'évolution de l'activité de conception. De tels templates viennent se greffer sur les éléments géométriques des espaces de conception (Figure 3.21).

2. terme anglais qui désigne un document type représenté par un gabarit ou un patron

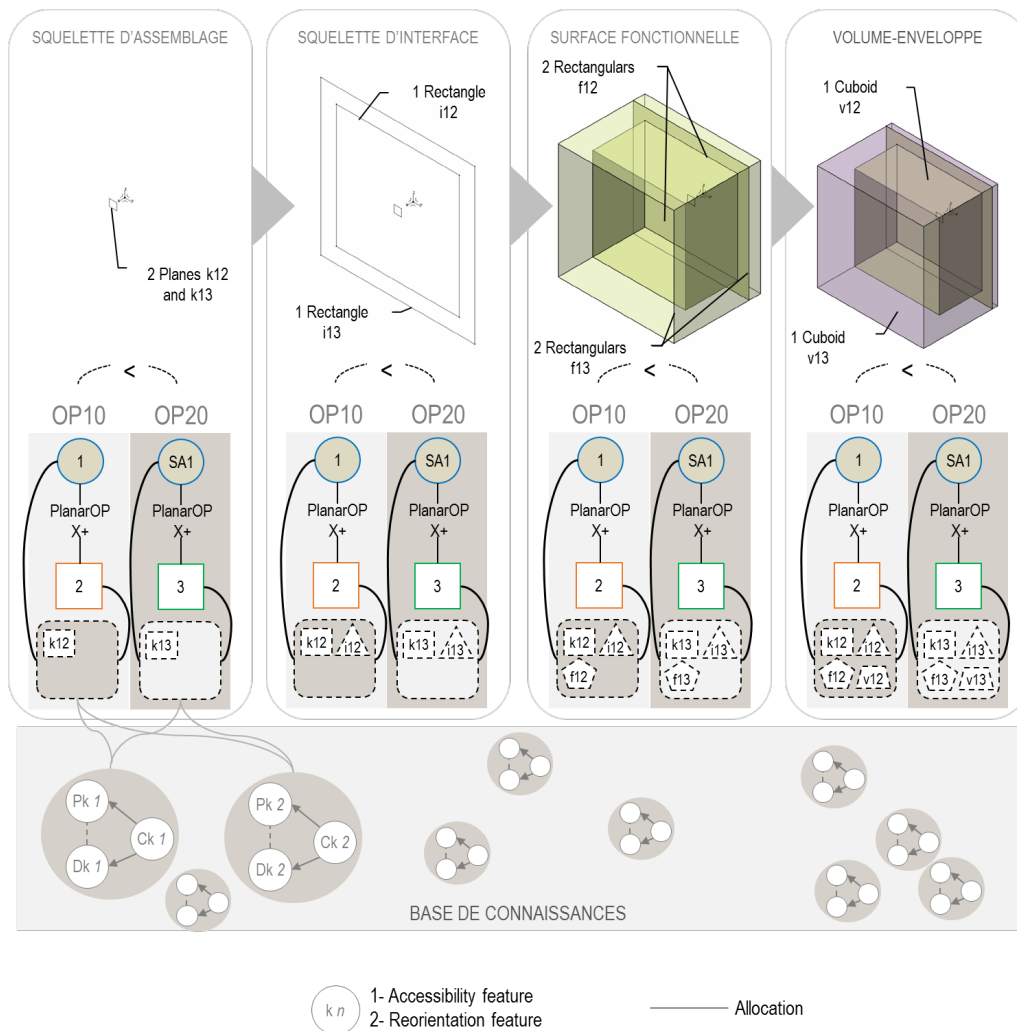


Figure 3.21 – Allocation des connaissances sur le graphe spatio-temporel

La Figure 3.22 montre l'intérêt de gérer le cycle de vie des connaissances [Skarka, 2007], à la fois dans la base de connaissances mais aussi dans les systèmes d'ingénierie à base de connaissances (**Knowledge Based Engineering - KBE**). En effet, l'identification, la capture et la validation sont faites par le gestionnaire de connaissances dans CrossRoads en se basant sur les règles d'experts. Les spécialistes et les experts définissent les processus permettant de réaliser un choix technologique en se basant sur les exigences, les intentions de conception et le savoir-faire existant. Les spécialistes créent les algorithmes en utilisant les règles d'activation des connaissances définies précédemment, et les experts établissent les données de sortie. Pour cela, un document appelé fiche d'aide à la décision (**Decision Support Sheet - DSS**), donne des indications sur les connaissances à sélectionner, la fiche concernée est principalement rattachée à une famille de produits. Cette aide à la décision correspond à une succession de règles métier permettant de guider l'utilisateur vers un choix technologique pertinent en fonction de son besoin. Ainsi, la vérification de la cohérence et de la consistance des connaissances est réalisée à la fois de manière semi-automatique selon une classification du produit, de son process et des connaissances en lien, mais aussi une série de questions proposées aux experts, pour affiner leurs choix [Vernier, 2014]. La gestion des templates dans le KBE [Kuhn, 2012], permet de les tester, de les packager en re-

groupant les annotations, les règles de conception, les paramètres et les entités géométriques, puis de les déployer en évaluant leurs pertinences, sur quatre cas d'études représentant différents types d'assemblages : série, entremêlé, inclusif/obligatoire, parallèle. Les connaissances sont instanciées dans l'activité de conception, ici représentées par la Figure 3.23. Au cours de l'activité de conception, le concepteur reçoit les connaissances entre chaque situation. Pour cela, le template sélectionné vient compléter la CADBOM avec ses ensembles d'éléments (géométriques, annotations, paramètres et relations). Il est ainsi guidé via les annotations avec les bonnes pratiques pour appliquer la règle de conception. La capitalisation de ces connaissances consiste ici à formaliser ces ensembles d'éléments au travers d'un outil de gestion des configurations de connaissances (**Knowledge Configuration Management - KCM**) développé au sein de notre laboratoire et baptisé **KrossRoads** [Vernier, 2014]. Celui-ci implique la capture et la réalisation de différentes configurations de connaissances. Malheureusement, cette application ne peut pas rechercher automatiquement la configuration adéquate de connaissances, sans l'aide du gestionnaire de connaissances. C'est pourquoi, nous exploitons cet outil, pour intégrer le raisonnement et faciliter son utilisation dans la CAO. La Figure 3.24 représente l'interface de **KrossRoads** avec l'outil de conception **CATIA V5** pour la formalisation des connaissances et leurs instanciations dans le contexte de conception augmenté. À chaque situation, la connaissance est instanciée en utilisant les templates **KBE Power Copy** (tels que définis dans **CATIA V5 Knowledgeware**) afin d'influencer directement la phase de modélisation CAO.

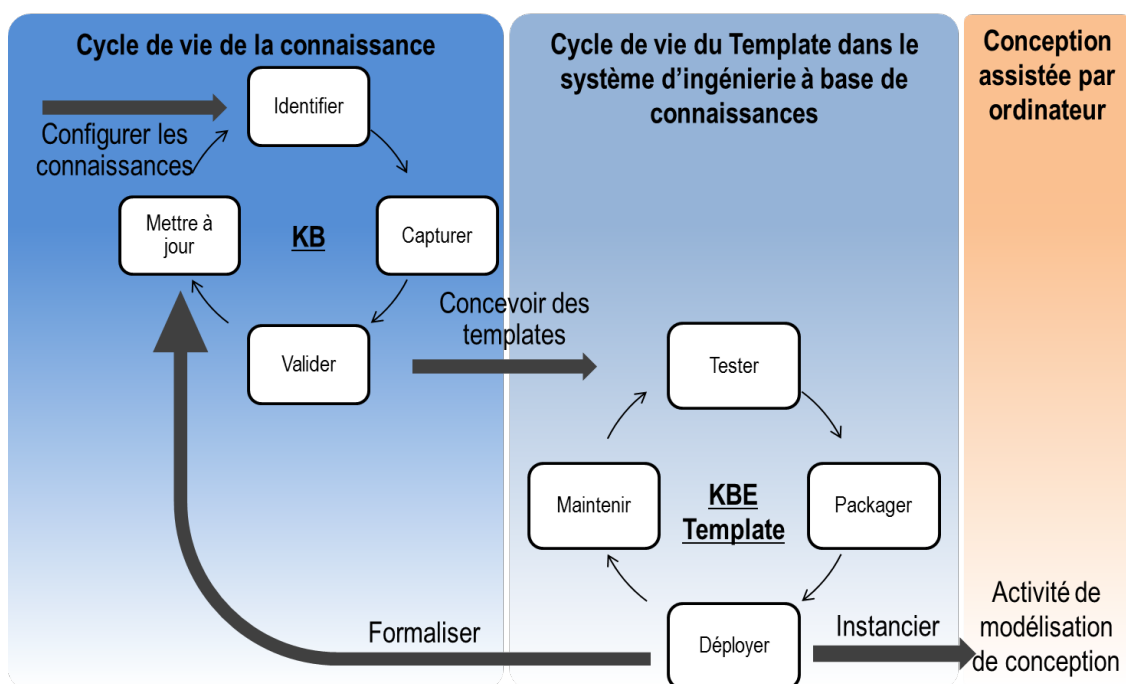


Figure 3.22 – Gestion du cycle de vie de la connaissance

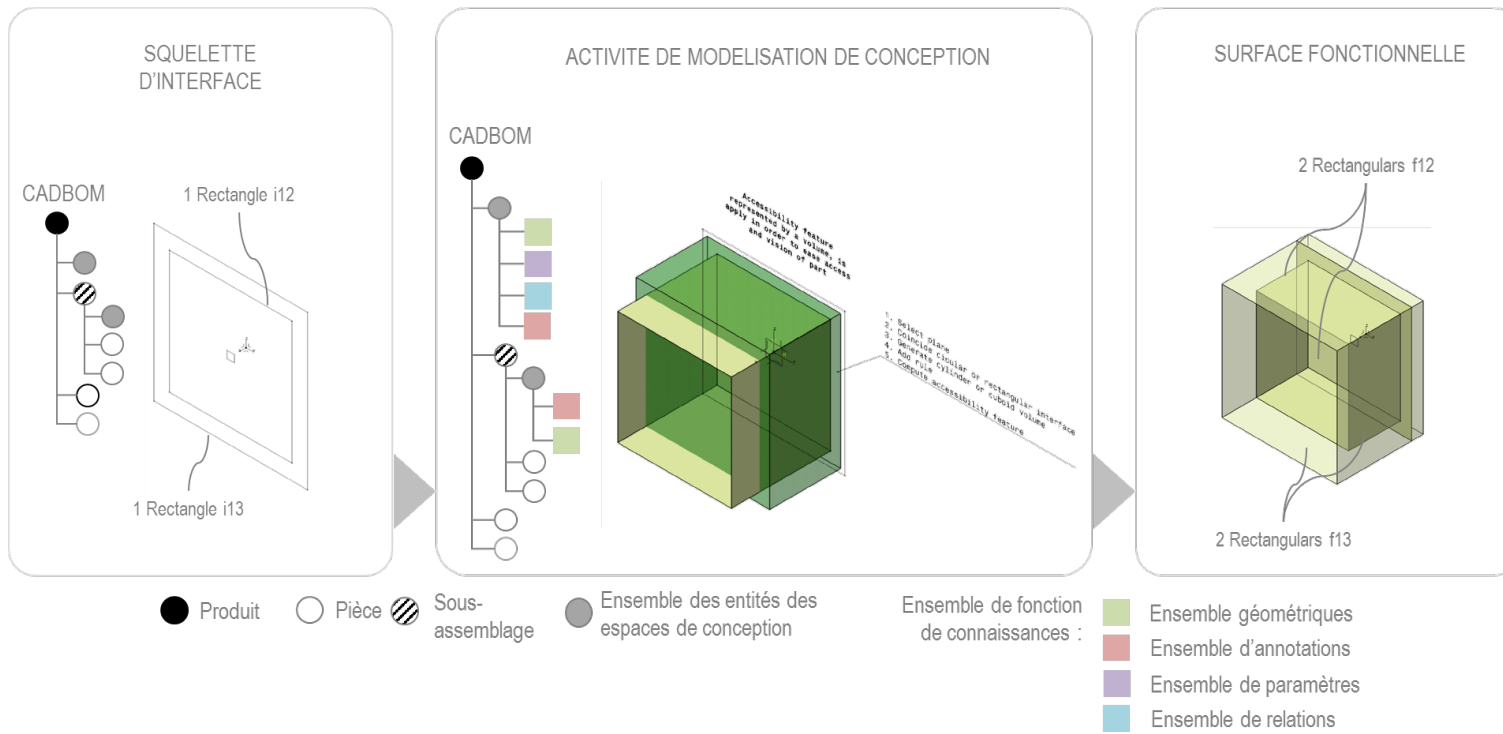


Figure 3.23 – Etapes d’instanciation des connaissances pendant l’activité de conception

KrossRoads :
Gestionnaire de configurateur de connaissances

CATIA :
CAO

Annotations environnement CAO

Information
Name : Tool space feature

Description :
Apply tool space feature to facilitate the space available for the tool
Dk

Formula :
general.columnname Actions
(OPn : y is insert in base part x : All technologi...
Assembly skeleton elements: 1 line and 1 plane
Interface skeleton elements: 3 circle or rectan...
Add formula

Best practice :
Pk
1. Select line and plane
2. Coincide circle or rectangle interface
3. Generate cylinder or cuboid volume
4. Add relations and parameter Length 1, Length 2, Width and Diameter
5. Add annotations on front view
6. Compute tool space feature

Formula :
general.columnname Actions
Length 1 (mm) >= Length 2 (mm)
Width (mm) >= Diameter (mm)
Length 1 (mm)
Length 2 (mm)
Width (mm)
Diameter (mm) **Dk**
Add formula

Picture :
Ck
Volume
Width
Length 2
Length 1
Diameter
1 line and 1 plane
3 circle or rectangle
e Plane
Delete
DCS validation cycle : Yes No
Cancel Save draft Launch validation process

Apply tool space feature to facilitate the space available for the tool

1. Select line and plane
2. Coincide circle or rectangle interface
3. Generate cylinder or cuboid volume
4. Add relations and parameter Length 1, Length 2, Width and Diameter
5. Add annotations on front view
6. Compute tool space feature

Figure 3.24 – Formalisation des templates de connaissances dans KrossRoads pour les instancier dans CATIA

La définition du produit final nous permet de comparer la sensibilisation du concepteur, avec et sans la planification de la séquence d'assemblage (Figure 3.25). Le changement géométrique du produit permet de noter, la prise de conscience sur les possibilités d'assembler le produit. Nous considérons la notion de « *sensibilisation* » comme :

L'ensemble des connaissances proposées au concepteur lors de son activité de conception.

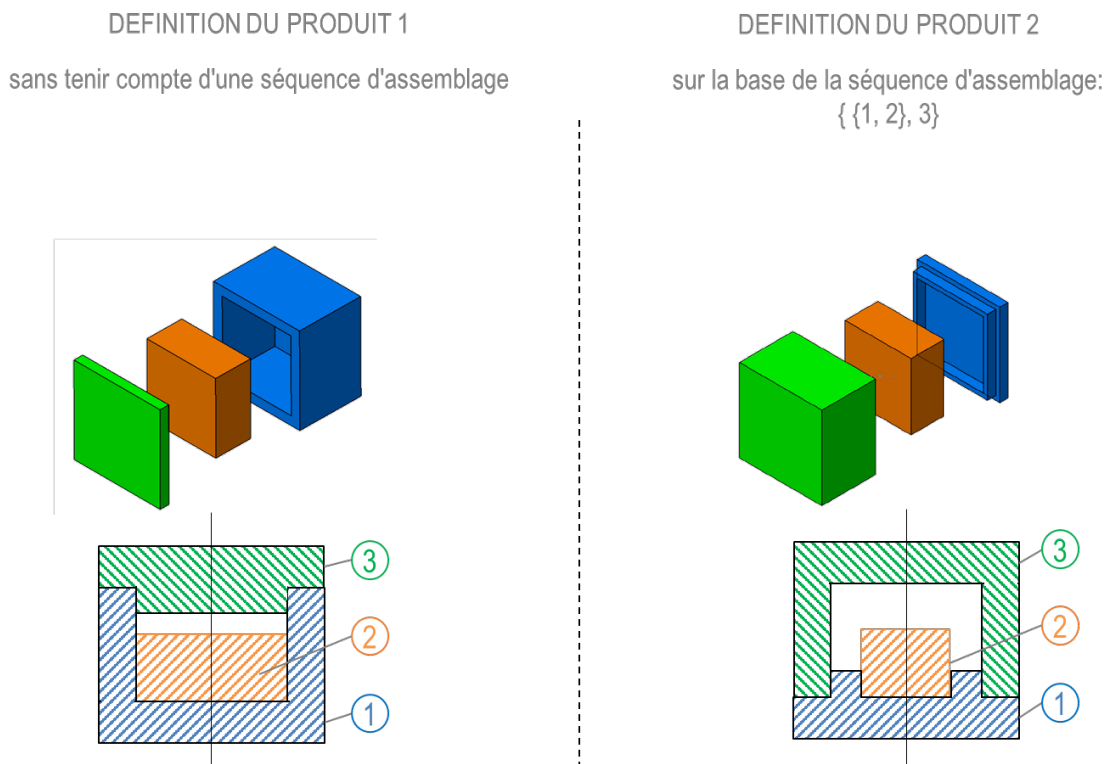


Figure 3.25 – Comparatif des différentes itérations de la boîte

En effet, les connaissances instanciées différemment sur les composants, vont influencer les relations géométriques entre les pièces. Nous pouvons voir une évolution de la conception du produit suite à l'application des connaissances suggérées dans l'activité de conception pour démontrer l'utilité de DAMON. Le comparatif en question essaie de montrer de manière simplifiée, l'application du cadre méthodologique notamment sur la géométrie du produit. En effet, la différence entre la définition des deux produits sur la prise en compte de la séquence d'assemblage, qui influence la géométrie des composants no.1 et no.3, facilite l'accès d'insertion du composant no.2. Nous évaluerons plus tard les performances de DAMON sur le cas d'étude de la pince de désincarcération.

3.4/ CONCLUSION

DAMON permet de mettre en place, au stade préliminaire de la conception, une méthode pour favoriser une réutilisation appropriée de la connaissance dans le « *contexte augmenté* ». Nous nous sommes concentrés sur les liens entre les relations d'assemblage et les informations sur la planification des séquences d'assemblage, afin d'accroître le contexte de conception et de mieux

réutiliser les connaissances métier. Nous avons pour cela, exploité plusieurs concepts et avons fait le lien avec des travaux déjà existants. Nous avons donc généré le futur pour aider à la décision de solutions alternatives grâce à l'algorithme ASDA, issu des travaux [Demoly, 2010]. Nous avons superposés les informations issues du futur avec celles du présent en se basant sur la théorie spatio-temporelle JANUS des auteurs [Demoly, 2010] et [Gruhier *et al.*, 2015]. Et enfin, nous avons fait le lien avec ces concepts pour proposer un raisonnement susceptible d'aller chercher les connaissances en adéquation avec les espaces de conception.

Ce cadre aide les concepteurs à prendre conscience de la conception orientée assemblage pour accroître les possibilités de la conception de produits. La connaissance instanciée comprend des contraintes d'assemblage, qui délimite les espaces de conception avec des relations et des paramètres. Le concepteur a ainsi une vision, grâce aux annotations et aux géométries générées, des contraintes d'assemblage, pour faire ses choix de conception. Il pourra ainsi choisir, quelle connaissance instancier en fonction de son besoin.

La diffusion des connaissances a été affinée tout au long du contexte de conception augmenté, afin de permettre une meilleure réutilisation des connaissances dans l'activité de conception. Elles ont été décomposées en trois catégories (Ck, Dk, et Pk), pour favoriser la recherche, et la pertinence de leur instanciations. Cette approche proactive répond aux défis globaux liés à la réutilisation du savoir [Chandrasegaran *et al.*, 2013]. L'étude explique l'efficacité de l'approche en montrant différentes définitions de produits, avec une diffusion granulaire des connaissances dans chaque situation.

Le chapitre suivant abordera le développement d'une approche pour favoriser l'intégration informatique de DAMON. Nous aborderons ainsi, avec la technologie sémantique, un HUB de conception, pour faciliter la compréhension entre le système d'information PLM et les intentions des parties prenantes.

APPROCHE DE MODÉLISATION D'UN SYSTÈME SENSIBLE AU CONTEXTE - MERCURY2

« Science sans conscience, n'est que ruine de l'âme. »

François Rabelais

Sommaire

4.1 Introduction	86
4.2 Approche - MERCURY2	87
4.2.1 Vue d'ensemble et objectif de l'approche - MERCURY2	87
4.2.2 Étapes de modélisation	90
4.3 Architecture de l'application sémantique	91
4.3.1 Vue d'ensemble de l'architecture retenue pour l'application sémantique . .	91
4.3.2 Conception de la couche interface applicative (étapes 1 à 2)	94
4.3.3 Conception de la couche à base d'ontologie (étapes 3 jusqu'à 8)	99
4.3.4 Conception de la couche logique de raisonnement (étapes 9 jusqu'à 11) .	109
4.4 Conclusion	116

4.1/ INTRODUCTION

L'implémentation du cadre méthodologique DAMON est présentée dans ce chapitre, via une approche appelée MERCURY2 qui constitue une évolution de l'approche MERCURY proposée par [Gruhier *et al.*, 2017], représentée par la Figure 4.1. Dans le premier paragraphe, nous abordons l'objectif de l'approche proposée en vue de développer un système sensible au contexte et expliquons ensuite l'architecture proposée pour cette implémentation.

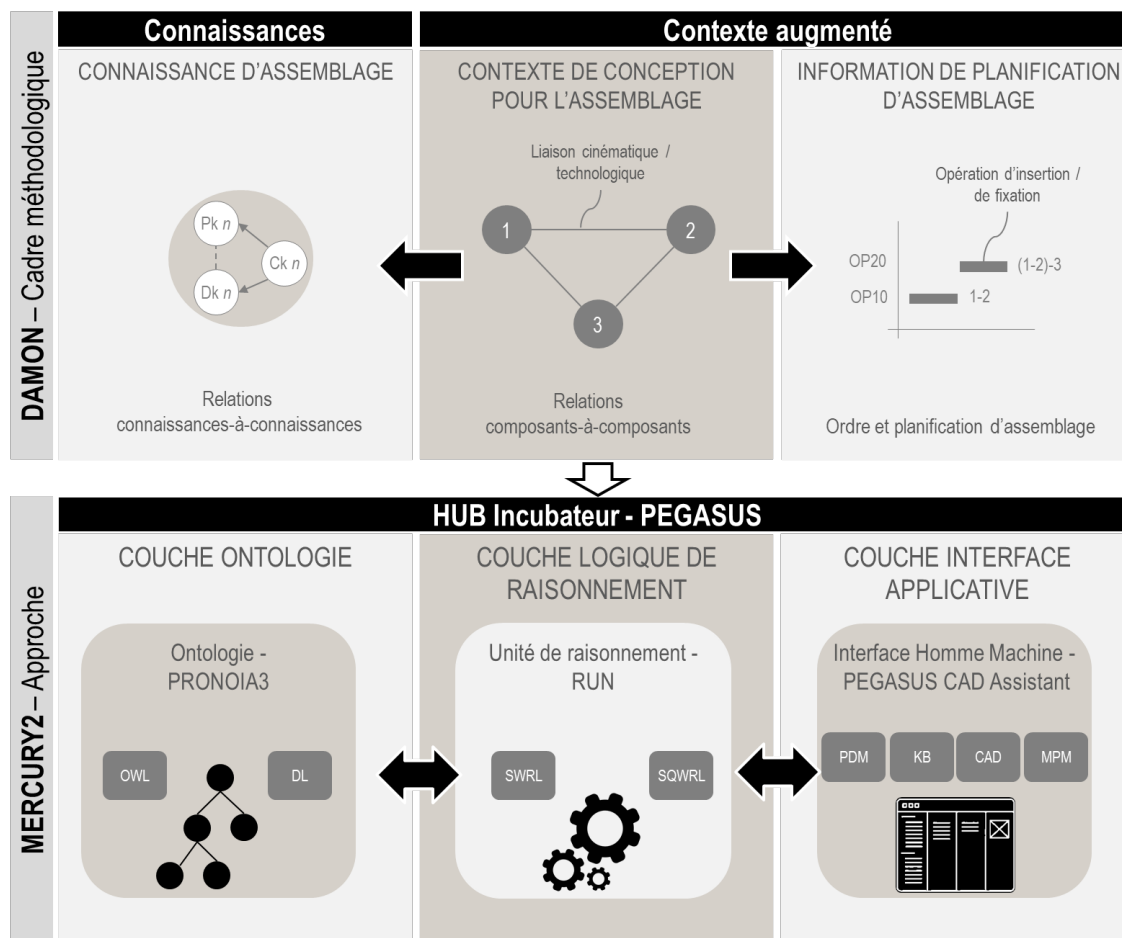


Figure 4.1 – Introduction de l'approche MERCURY2

En effet, il est possible de définir sur la base des spécifications fonctionnelles, le rôle et les tâches des parties prenantes qui sont amenés à interagir au travers d'une Interface Homme Machine - IHM appelée PEGASUS CAD Assistant dans un HUB incubateur de conception PLM appelé PEGASUS. Ce dernier a comme fonctionnalité de centraliser toutes les informations nécessaires provenant des systèmes PDM, MPM, KB, et CAO, pour instancier les connaissances. Pour cela, il est important de définir un modèle qui permettra d'encadrer les classes et relations et ce, grâce à la technologie sémantique avec l'éditeur d'ontologie Protégé. Nous utiliserons le langage d'ontologie web (Ontology Web Language - OWL) et la description logique (Description Logic - DL) de l'ontologie formelle PRONOA3. L'élément, qui permet l'interaction entre les informations et connaissances, est le moteur du système, qui, sur la base des conditions définies dans le cadre DAMON, permet

de les transposer dans le HUB pour fournir au médiateur le raisonnement sémantique. Le système d'inférence applique les protocoles de raisonnement sémantique pour former des unités de raisonnement RUN, à l'aide de langage de règles pour le web sémantique (**Semantic Web Rules Language - SWRL**) et de langage de requête de règles pour le web sémantique (**Semantic Query-enhanced Web Rule Language - SQWRL**).

4.2/ APPROCHE - MERCURY2

4.2.1/ VUE D'ENSEMBLE ET OBJECTIF DE L'APPROCHE - MERCURY2

L'approche présentée, suit le cadre DAMON qui permet via l'ontologie formelle PRONOA3, les unités de raisonnement RUN et l'interface PEGASUS CAD Assistant de faciliter l'interopérabilité entre les systèmes dans le PLM. Cet assistant propose des connaissances adaptées à chaque situation de conception. Cette approche MERCURY2 comprend à la fois la formalisation de l'ontologie, son raisonnement et son interfaçage avec les parties prenantes. Elle fait partie d'un projet beaucoup plus vaste appelé incubateur de conception ou PEGASUS (Figure 4.2) visant à améliorer la coordination des flux d'informations dans le contexte de conception orienté assemblage. Un développement supplémentaire doit être fait pour assurer des interfaces transparentes de communication avec les systèmes PDM, MPM, KB et les applications CAO. L'ontologie PRONOA3 (**PRO**duct **relatio**Nships description based **On** mereotopolog**Ic**AI theory) permet au système de comprendre le contexte de conception et de réutiliser des connaissances sur les objets et leurs interconnexions. Le rôle principal de PEGASUS est de :

- Gérer (générer et capturer) des contextes de conception pour les concepteurs mécaniques ;
- Saisir les intentions de conception pendant les activités de modélisation de CAO ;
- Instancier les connaissances appropriées aux situations de conception.

La Figure 4.2 met en évidence l'application du HUB qui gère l'information d'ingénierie et de fabrication à partir d'un point central. L'application permet de communiquer avec :

1. Le système PDM qui s'appuie sur la gestion de la nomenclature produit (eBOM), des sous-assemblages, des composants et de tout élément relatif à la structuration du produit et du contexte ;
2. Le système MPM qui est en charge de la gestion de la mBOM et la BOP, de la construction de plan d'assemblage, de la génération de séquence d'assemblage avec toutes les informations visant à générer et à définir une gamme de montage ;
3. Le système KB qui est géré par le gestionnaire de connaissances afin de les capturer et les instancier. Elles sont composées de connaissances conditionnelles, procédurales et déclaratives ;
4. L'application CAO qui consiste à capturer et à définir la géométrie du produit et de ses composants. Les concepteurs peuvent définir les contraintes d'assemblage, paramétrer le produit et exploiter les fonctionnalités de forme (form feature) ;
5. Le HUB géré par le médiateur concilie la BOR pour la diffusion des espaces de conception ainsi que les fonctions d'assemblage (assembly feature).

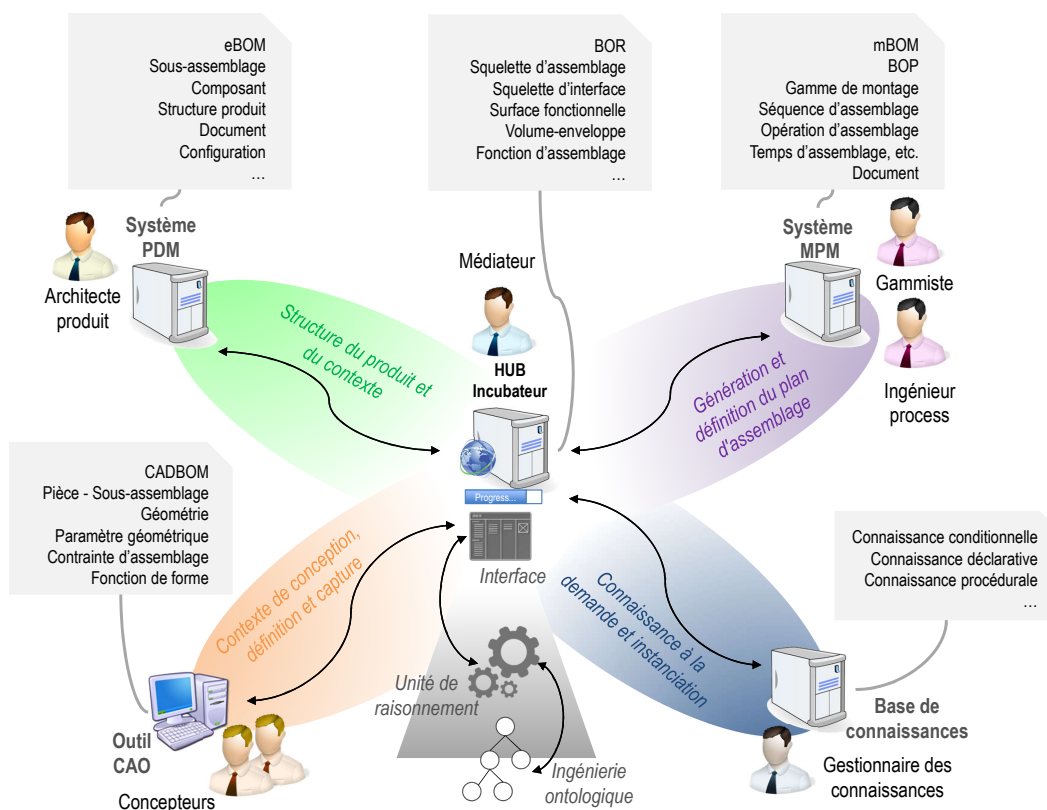


Figure 4.2 – Vue d'ensemble du HUB Incubateur - PEGASUS

La Figure 4.3 présente la centralisation et la conciliation des informations dans le HUB. La BOR gère et centralise toutes les informations provenant des systèmes PDM, MPM, CAO et KB.

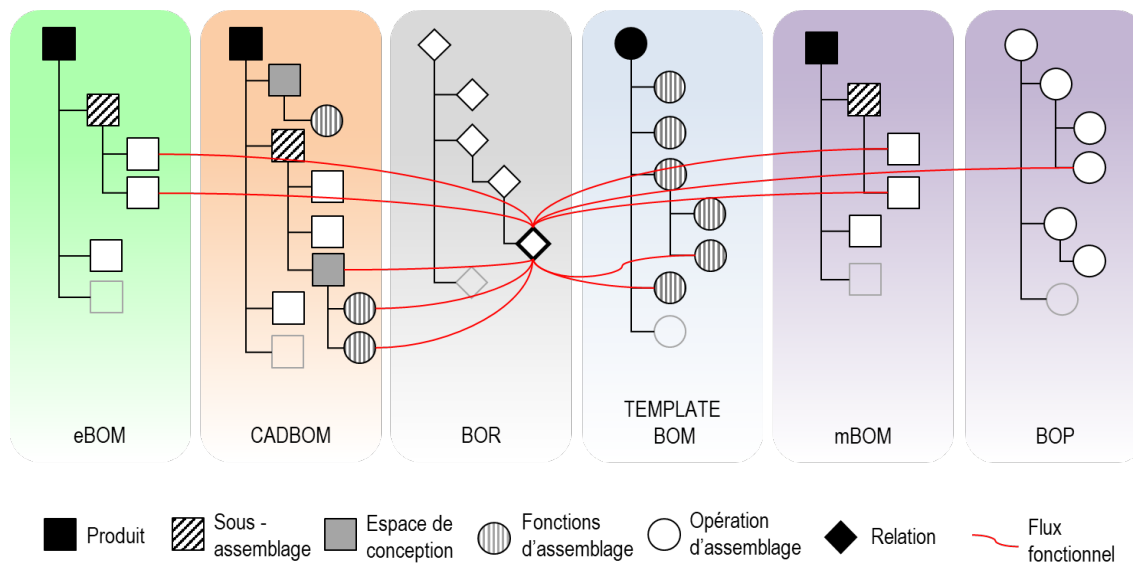


Figure 4.3 – Réconciliation des systèmes PDM, MPM, CAO et KB avec la Bill Of Relations (BOR)

Toutes les informations et les connaissances sont mises à jour par le médiateur au travers de la BOR. La diffusion de ces informations est représentée par étapes dans la Figure 4.4, qui pour chaque processus de conception du produit permet notamment de structurer des données pour la eBOM, la mBOM, le template BOM, la BOR et la CADBOM. Le flux fonctionnel d'informations est structuré de manière à rationaliser et faciliter le transfert des connaissances aux parties prenantes. La BOR et la BOP permettent alors de compléter les structures du produit, de la CAO, et des opérations de montage dans leurs systèmes respectifs [Petrazoller *et al.*, 2014] .

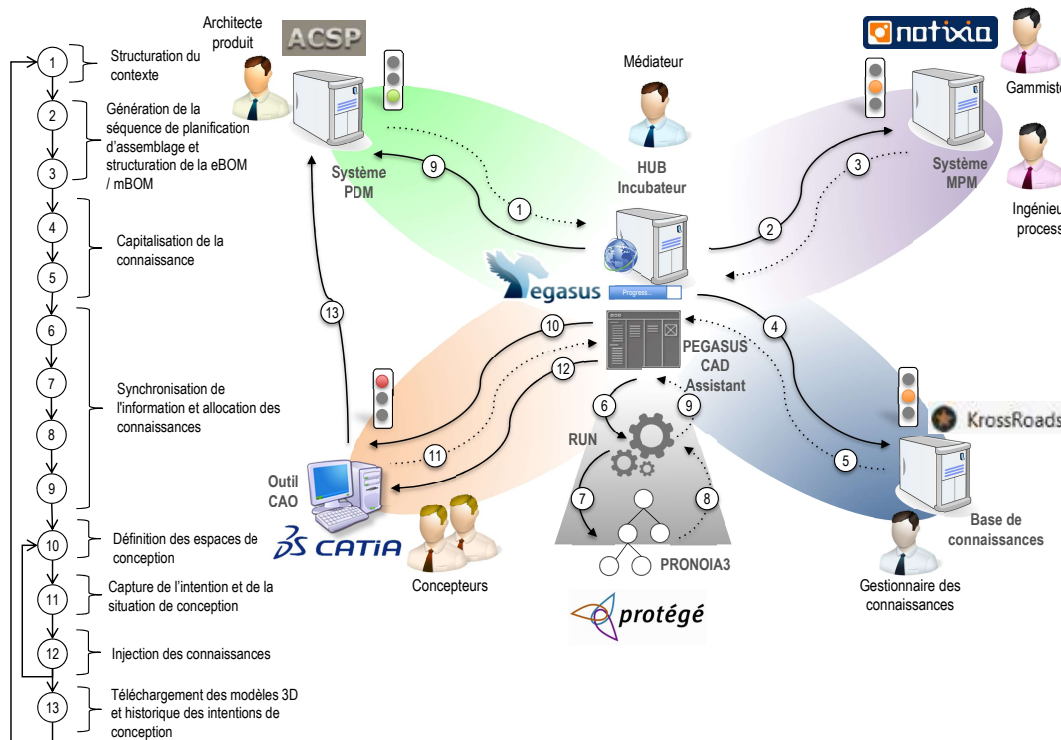


Figure 4.4 – Flux d'informations et des connaissances avec le HUB

Dès lors, le rôle du médiateur est de détecter un événement (par exemple, la création d'une partie mécanique dans le PDM). L'événement ainsi créé est détecté par PEGASUS qui représente l'interface du système PLM dans la gestion des informations techniques du couple produit-process. Le médiateur interprète l'information et crée un fichier .xml, qui se diffuse vers les autres systèmes d'information. Nous exploitons le système PDM **ACSP** (**A**telier **C**oopératif de **S**uivi de **P**rojet), qui permet l'interopérabilité entre le système de gestion de données techniques (PDM) et les autres outils, tels que l'aide à la décision et ceux servant à la spécification de la structure du produit, afin de créer la planification de la gamme d'assemblage représentée par le système MPM **NOTIXIA**. Le logiciel **NOTIXIA** permet de gérer les informations des données process en fournissant toutes les informations de la gamme, des séquences et des opérations d'assemblage associées. Sur ces informations, le gestionnaire de connaissances, peut capitaliser les connaissances dans le gestionnaire de configuration de connaissances **KrossRoads**, afin d'établir les relations avec les éléments du « *contexte augmenté* ». **KrossRoads** permet notamment la formalisation et la centralisation du savoir faire, mais également l'instanciation des connaissances en contexte projet avec la création

des configurations de connaissances, ainsi que la génération et l'utilisation des applications métier en se basant sur les configurations de connaissances. Enfin, le médiateur crée au fur et à mesure les instances de l'ontologie PRONOA3, pour interpréter la situation et les intentions de conception. La cohérence de l'information est vérifiée avec la description logique (DL) contenue dans l'ontologie PRONOA3. Les unités de raisonnement, vont alors corréliser les informations provenant de la CAO, pour effectuer des requêtes dans KrossRoads en vue de cibler les connaissances possibles. Elles sont proposées aux concepteurs sous forme d'annotations dans CATIA via l'interface PEGASUS CAD Assistant. Nous avons sélectionné ces technologies car elles permettent de gérer en toute indépendance et de manière proactive les informations/connaissances du couple produit-process dans la mesure où il convient d'intégrer et d'uniformiser via le HUB, les approches historiquement proposées par le laboratoire. Notre originalité est d'intégrer un raisonnement visant à comprendre et instancier les informations et les connaissances dans la conception.

4.2.2/ ÉTAPES DE MODÉLISATION

L'approche MERCURY2 permet donc d'établir les bases de développement servant à gérer les connaissances et à les instancier dans le HUB. Pour cela, plusieurs étapes nécessitent d'être réalisées. Ces étapes, utiles à la modélisation du HUB sont définies (Figure 4.5), afin d'identifier : les profils des parties prenantes qui vont interagir avec le système, la modélisation de l'ontologie formelle et le raisonnement sémantique.

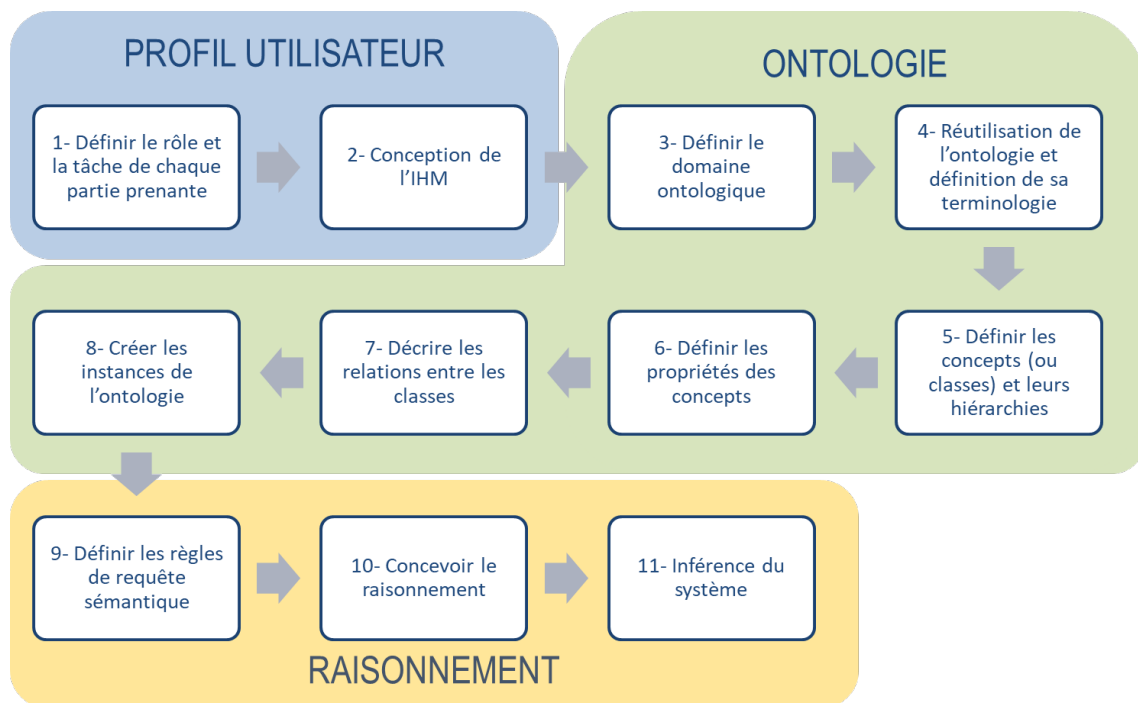


Figure 4.5 – Processus de conception du HUB illustrant le découpage entre la conception du profil utilisateur, la conception de l'ontologie et du volet raisonnement

La section suivante est divisée en trois parties principales inspirées des recommandations formulées par [Zhu *et al.*, 2011] pour construire un système de raisonnement. Dans notre cas, les exigences pour modéliser le HUB sont inspirées de celles proposées par [Codina & Ceccaroni, 2010] :

- La description des profils de chaque partie prenante utilisant le système en vue de saisir leurs intentions dans une situation de conception ;
- Une ontologie décrivant la taxonomie des classes et leur description, utilisée par le HUB ;
- Une couche logique de raisonnement sémantique exploitant les connaissances et les diffusant aux parties prenantes de manière intelligente grâce à diverses applications de règles et de requêtes sémantiques.

4.3/ ARCHITECTURE DE L'APPLICATION SÉMANTIQUE

4.3.1/ VUE D'ENSEMBLE DE L'ARCHITECTURE RETENUE POUR L'APPLICATION SÉMANTIQUE

La modélisation du HUB est composée de trois couches qui sont inspirées de [Zhu *et al.*, 2011]. La Figure 4.6 représente ces couches contenant l'ontologie, la logique de raisonnement et l'interface d'application.

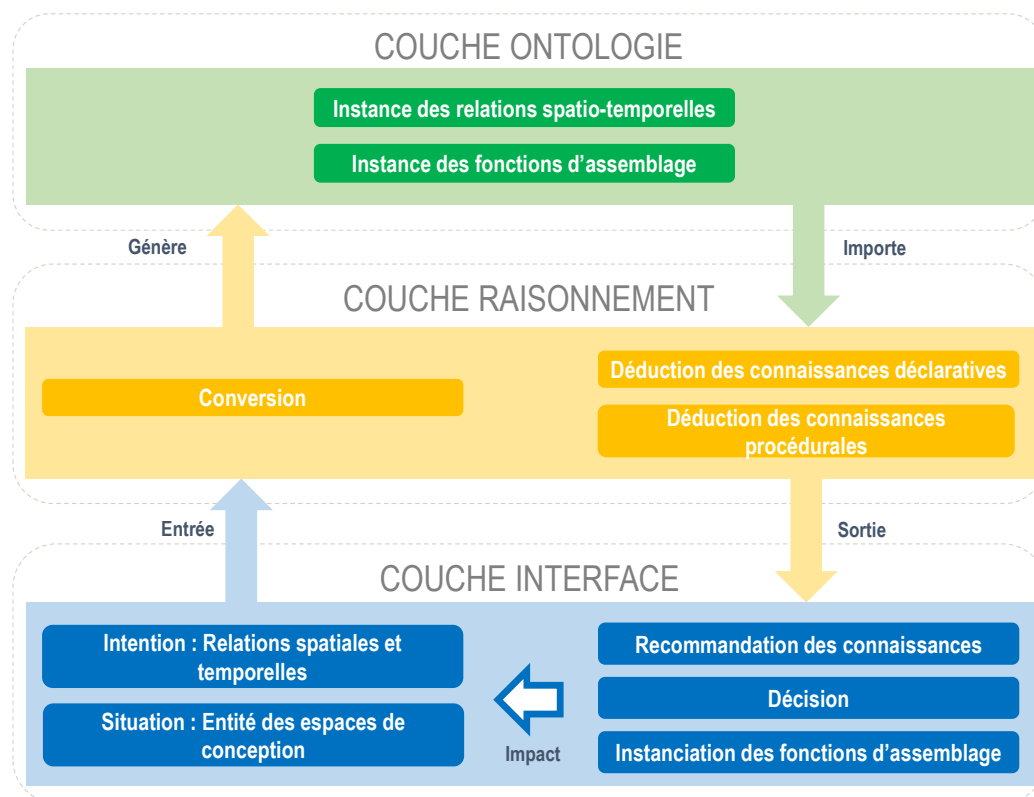


Figure 4.6 – Différentes couches composent le HUB

Les diverses couches composant le HUB communiquent selon les entrées et sorties des informations provenant du « *contexte augmenté* ». Celles-ci comprennent à la fois les informations sur les relations d'insertion et de fixation, mais aussi celles sur les liaisons de positionnement cinématique et technologique. La couche d'interface décrite à la Figure 4.6 précédente permet de faire

l'intermédiaire entre le système et les parties prenantes. L'Interface **Homme-Machine (IHM)** sert entre-autres à générer des spécifications dans l'activité de conception.

La couche de raisonnement logique supportée par le SWRL et le SQWRL vient épauler le moteur d'inférence avec le raisonneur DL (DL reasoner). Pour notre cas, nous exploiterons le raisonneur Pellet qui offre la particularité de supporter OWLAPI, et de faire des requêtes directement dans la base de connaissances. Le raisonnement concerne la représentation des règles Si-Alors, sous forme d'unité de raisonnement. Celles-ci sont la formalisation informatique des connaissances conditionnelles, qui iront d'abord inférer les Dk, pour effectuer des requêtes sur les Pk associées.

La couche d'ontologie, qui est décrite dans les logiques de description DL, comprend, outre leurs instances respectives, la terminologie des concepts et leurs relations dans le domaine de l'assemblage. C'est la raison pour laquelle, nous aborderons en détail la conception de ces couches, pour construire une IHM efficace. Nous complétons ensuite le modèle ontologique existant PRONOA2 grâce à, l'intégration de terminologies, de classes et de leurs propriétés, amis aussi de relations supplémentaires, pour générer les instances relatives au « *contexte augmenté* ».

La Figure 4.7 représente le déroulement du processus destiné à instancier les éléments dans les modèles CAO. Ainsi, l'ontologie PRONOA3, une fois réalisée, est importée dans un gestionnaire d'ontologie (ou parseur) appelé PRONOA3 Manager. L'intégration de l'ontologie, pour la population des données produit et la récupération du résultat de l'inférence repose sur la librairie OWLAPI. Suite au chargement de données produit (fichier .xml), la nouvelle API permet d'intégrer des instances dans une ontologie avec les données produit, telles que définies dans l'ontologie PRONOA3. Elle inclue le raisonneur Pellet, ainsi que la conversion objet-classe entre PRONOA3 et CATIA V5. L'IHM PEGASUS CAD Assistant permet aux concepteurs d'instancier les éléments géométriques et les templates, puis les synchroniser.

Le HUB consiste en une API Java qui permet d'une part, l'ingestion des données produit étendues et l'intégration de ces informations dans l'ontologie PRONOA3, via la nouvelle structure de données (document au format .xml), et d'autre part, l'import/export des informations issues de l'inférence dans un nouveau fichier au même format. Les informations ainsi inférées peuvent ensuite être exploitées par PEGASUS CAD Assistant et propagées au niveau du logiciel de CAO. Le flux bidirectionnel de données est défini par deux phases (Figure 4.7) :

- Une phase d'import de données produit vers la base de connaissances. Les données produit doivent être collectées par PRONOA3 Manager, et transformées dans un format adapté à la population de l'ontologie, (c'est-à-dire à la création d'individus et de relations liés au produit en cours de conception). Cette phase se déroule à l'interface de l'outil PEGASUS CAD Assistant et du gestionnaire de l'ontologie ;
- Une phase d'export des données produit étendues avec le résultat de l'inférence vers le logiciel de CAO. Il s'agit d'un processus qui fonctionne dans un sens opposé à celui décrit précédemment : de PRONOA3 Manager à PEGASUS CAD Assistant.

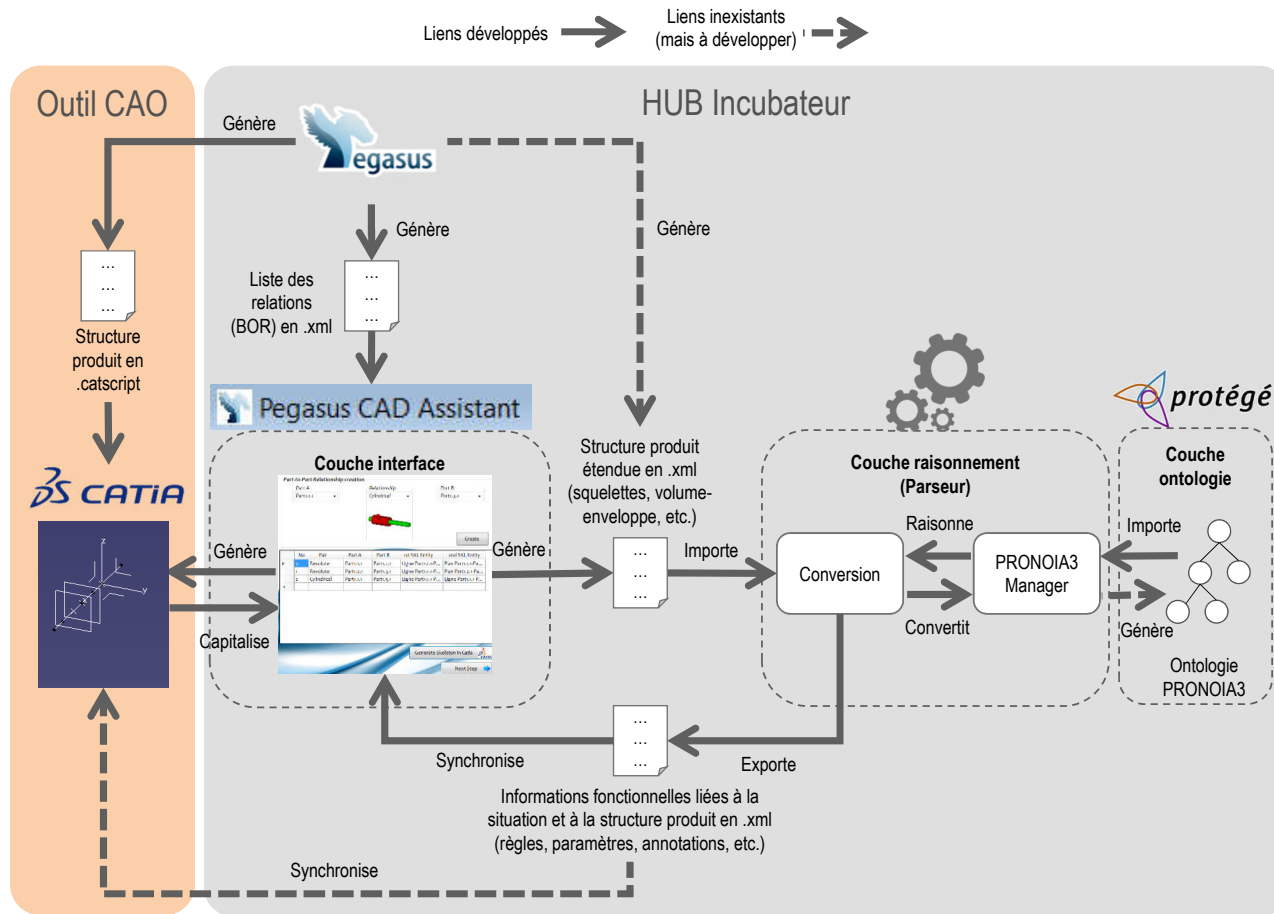


Figure 4.7 – Processus pour instancier les éléments dans la conception CAO

Le gestionnaire de l'ontologie a donc deux objectifs. Tout d'abord, l'intégration de données issues de la conception, à savoir, les informations sur la composition et la structuration du produit, dans une base de connaissances comprenant l'ontologie PRONOIA3. Le gestionnaire transforme les données reçues et instancie la base de connaissances avec les données transformées, ensuite, il est possible d'effectuer un renvoi de données vers CATIA, par ajout d'informations structurelles et fonctionnelles issues du processus d'inférence dans la base de connaissances.

Pour effectuer ces tâches, il est nécessaire de définir un schéma de données interprétable par le gestionnaire, à partir de PEGASUS CAD Assistant. Dans la première phase, le schéma de données décrit la structure étendue du produit (squelette du produit et données géométriques), et dans la deuxième phase le schéma de données est enrichi par le résultat de l'inférence, (connaissances supplémentaires liées à la situation). Ce schéma de données peut être partagé des deux côtés du processus, par PEGASUS CAD Assistant et par le gestionnaire de l'ontologie. Il est important de noter que le schéma de données permet d'abstraire la communication entre les différents composants logiciels de l'architecture. Ainsi, si l'intégration de données ne doit plus provenir de PEGASUS CAD Assistant mais d'un autre logiciel, le gestionnaire de l'ontologie permet toujours l'intégration des données produit et l'export des résultats de l'inférence.

4.3.2/ CONCEPTION DE LA COUCHE INTERFACE APPLICATIVE (ÉTAPES 1 À 2)

Le rôle et la tâche des parties prenantes sont définis afin de connaître le type d'intention et la situation que le système sensible au contexte peut capturer. Ceux-ci sont représentés via le diagramme de cas d'utilisation de la Figure 4.8.

Les fonctions suivantes sont décrites comme étant liées aux mécanismes développés dans le cadre méthodologique DAMON :

- Appliquer les fonctions et les exigences fonctionnelles du produit ;
- Définir les relations entre les composants ;
- Définir le contexte de conception ;
- Définir la structure du produit ;
- Générer et évaluer la séquence d'assemblage admissible ;
- Définir le contexte de production ;
- Sélectionner les sous-assemblage ;
- Définir les contraintes de précédence entre les opérations d'assemblage ;
- Définir les technologies d'assemblage standards ;
- Définir les opérations d'assemblage ;
- Créer la gamme de montage ;
- Définir les squelettes d'assemblage, leurs squelettes d'interfaces, leurs surfaces fonctionnelles, et leurs volumes-enveloppe ;
- Décrire le contexte de conception ;
- Capturer et mettre à jour les nouvelles connaissances ;
- Vérifier et évaluer la consistance des connaissances ;
- Générer le contexte de conception approprié ;
- Interpréter les intentions et suggérer les bonnes connaissances au concepteur, et ceci au bon moment.

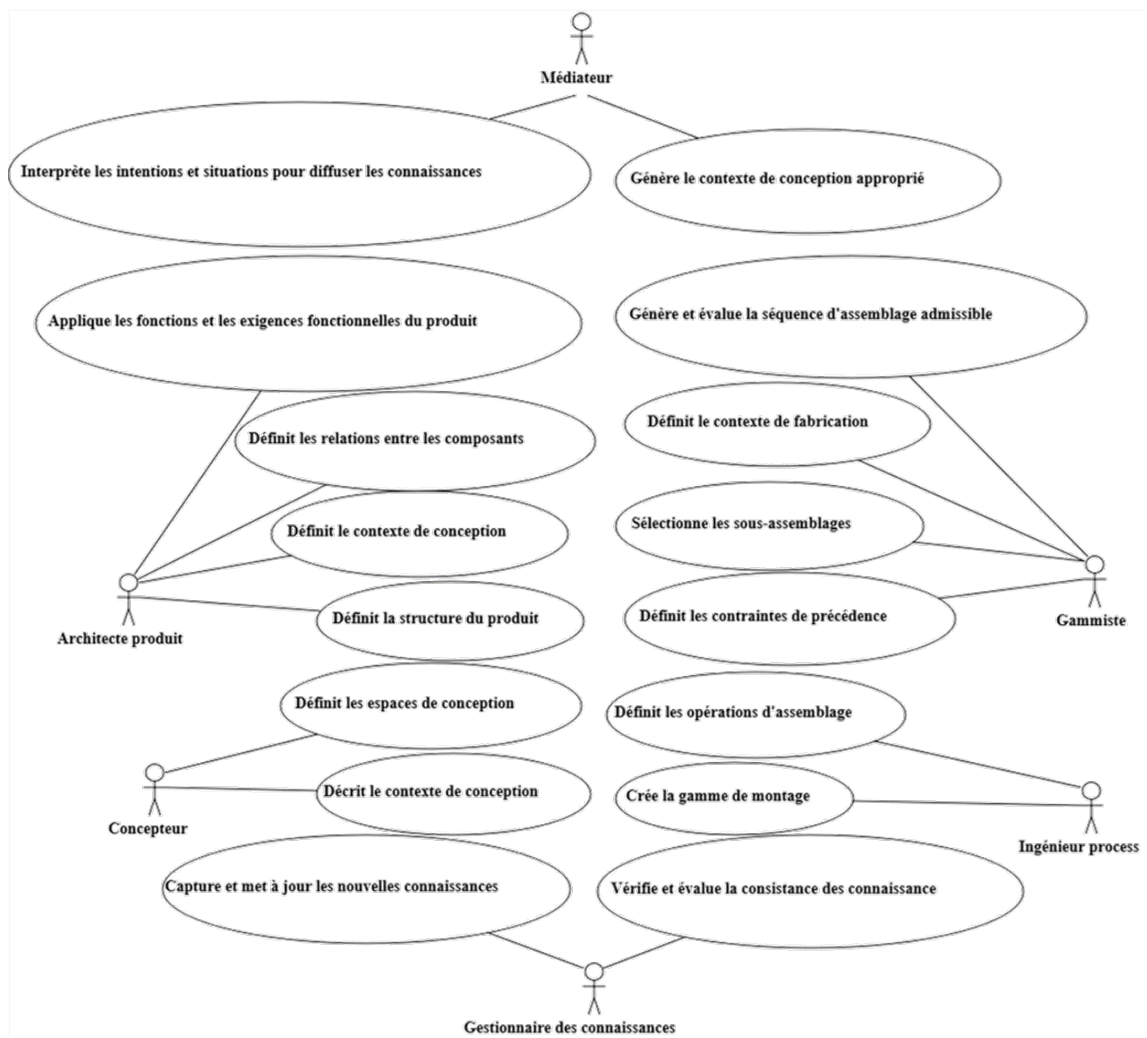


Figure 4.8 – Diagramme de cas d'utilisation du HUB PEGASUS

Le diagramme des cas d'utilisation établit les bases pour le diagramme UML de séquence (Figure 4.9). Ce dernier définit les différentes étapes des actions à réaliser entre chaque partie prenante et leur mise en application. Une fois les rôles et les tâches de chaque partie prenante définis, il est important de circonscrire leur interaction avec le système. Cette interaction avec l'interface permettra de capturer l'intention et la situation d'entrée des informations qu'il est nécessaire d'obtenir en sortie du processus. Grâce aux spécifications fonctionnelles issues des diagrammes présentés, nous pouvons établir la conception des interfaces afin que les parties prenantes puissent interagir avec le HUB.

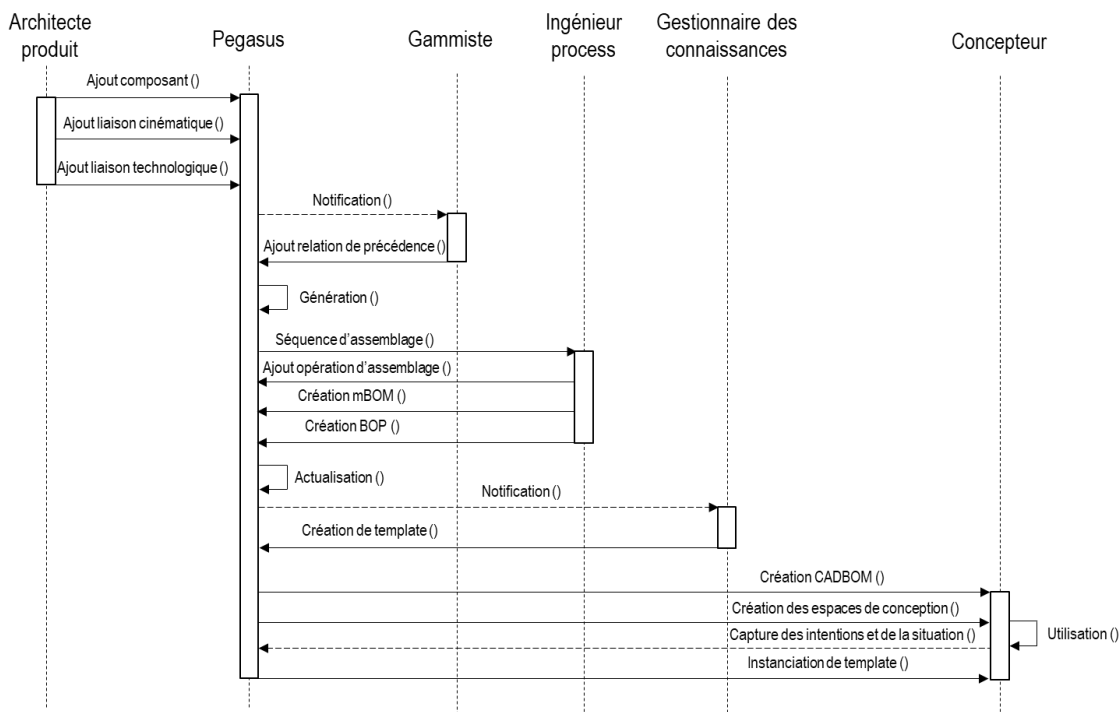


Figure 4.9 – Diagramme UML de séquence décrivant le déroulement des actions dans le HUB PEGASUS

La Figure 4.10, la Figure 4.11 et la Figure 4.12 représentent les principales interfaces qui servent d'assistants aux concepteurs. Divisée en trois parties, (i) vue contextuelle (contextual view), (ii) vue comportementale (behavioral view), et (iii) vue géométrique (geometric view) :

- La vue contextuelle capitalise les informations d'entrée, la concaténation des relations d'assemblage avec celle de la planification des séquences d'assemblage pour former un graphe spatio-temporel ;
- L'onglet vue comportementale, concerne la recommandation qui, basée sur ces informations, liste les connaissances représentées par des templates alloués au graphe spatio-temporel. Chaque template suggère des règles de conception pour l'assemblage via des bonnes pratiques. En conséquence, les templates contiennent des ensembles géométriques, des relations avec leurs paramètres associés, et des annotations qui se greffent sur la géométrie. Le concepteur a le choix d'instancier ou non le template sélectionné, ces choix influencent l'impact sur l'instanciation ou non d'autres connaissances, selon les règles de chaînage préalablement définies ;
- La partie finale dans l'onglet vue géométrique, est relative à la définition des entités géométriques du produit dans l'espace de conception de CATIA (position, longueur, forme, etc.).

Dans la section suivante, nous présentons dans un premier temps, la définition des classes (région) et de leurs relations (primitives) de l'ontologie formelle, et dans dans un second temps, la description de ces classes et de leurs relations.

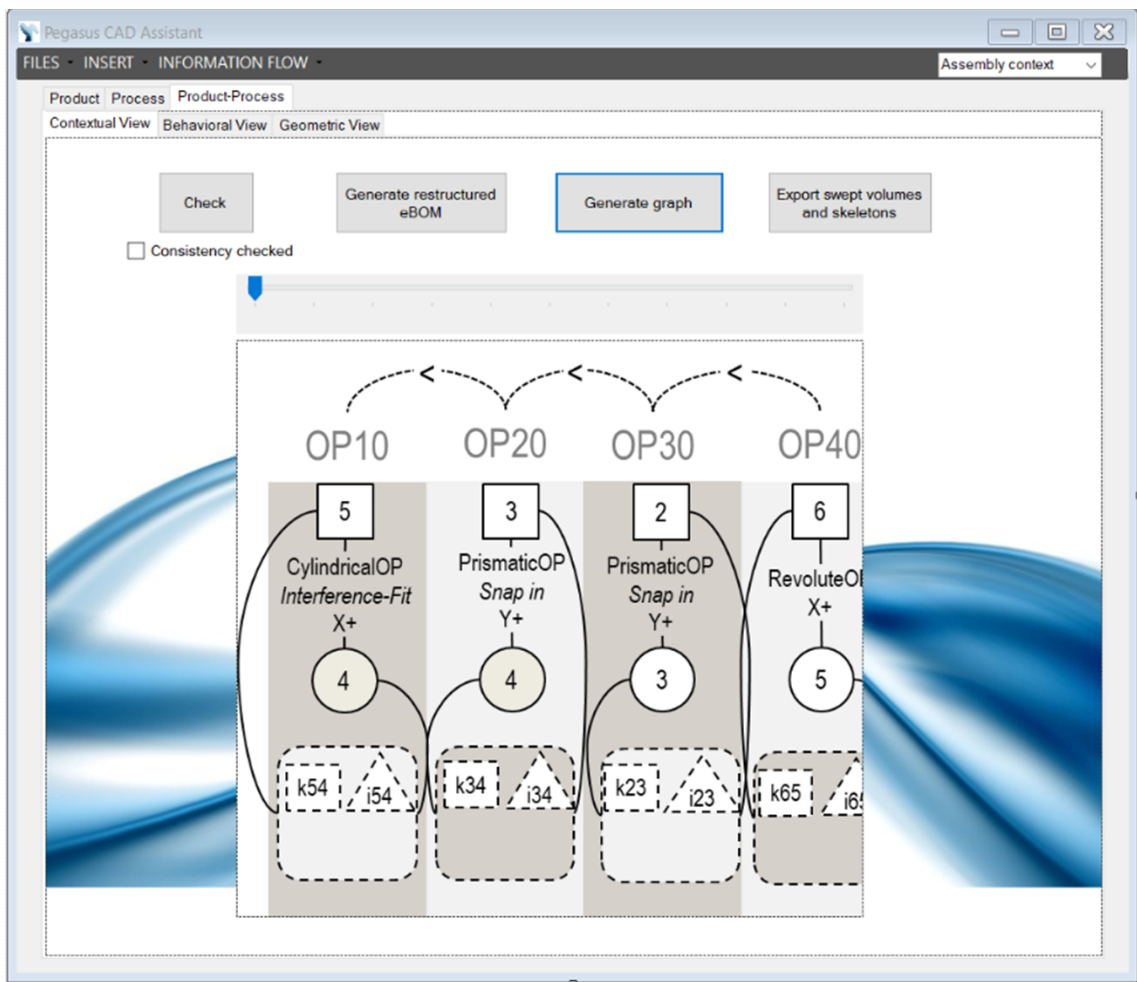


Figure 4.10 – Interface de PEGASUS CAD Assistant : vue contextuelle

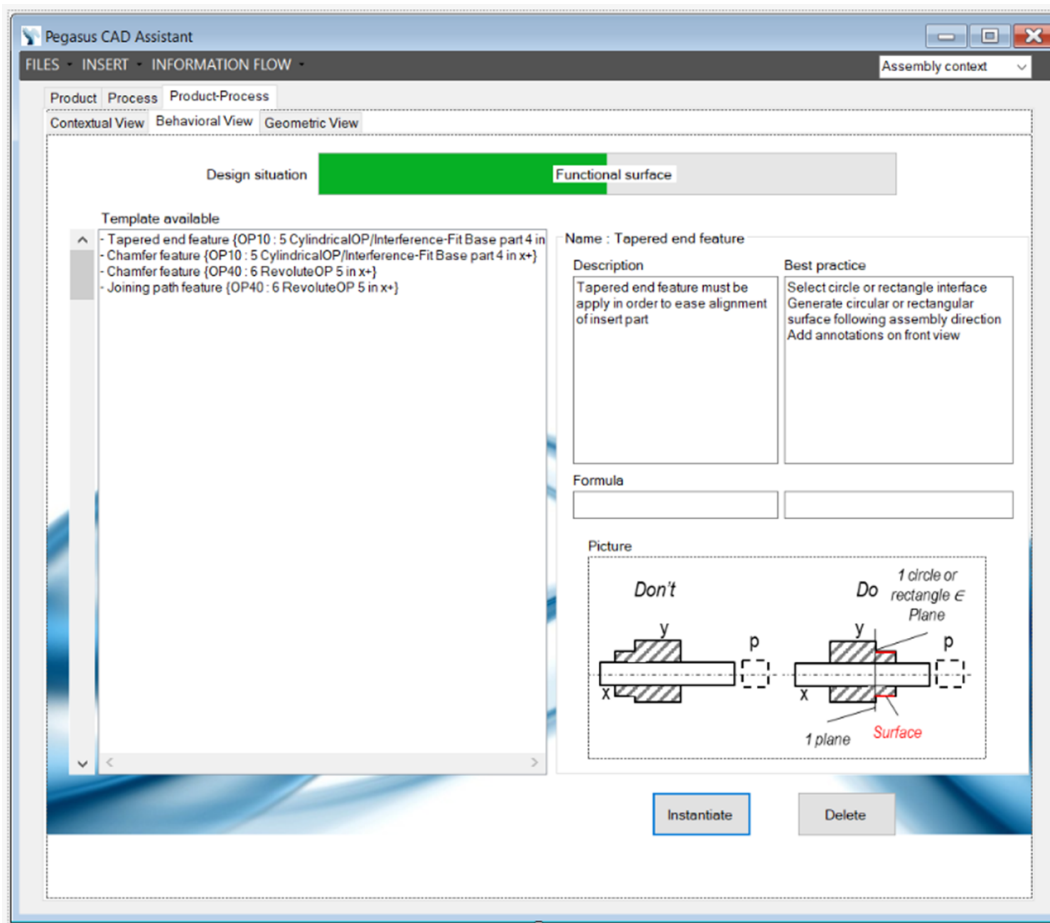


Figure 4.11 – Interface de PEGASUS CAD Assistant : vue comportementale

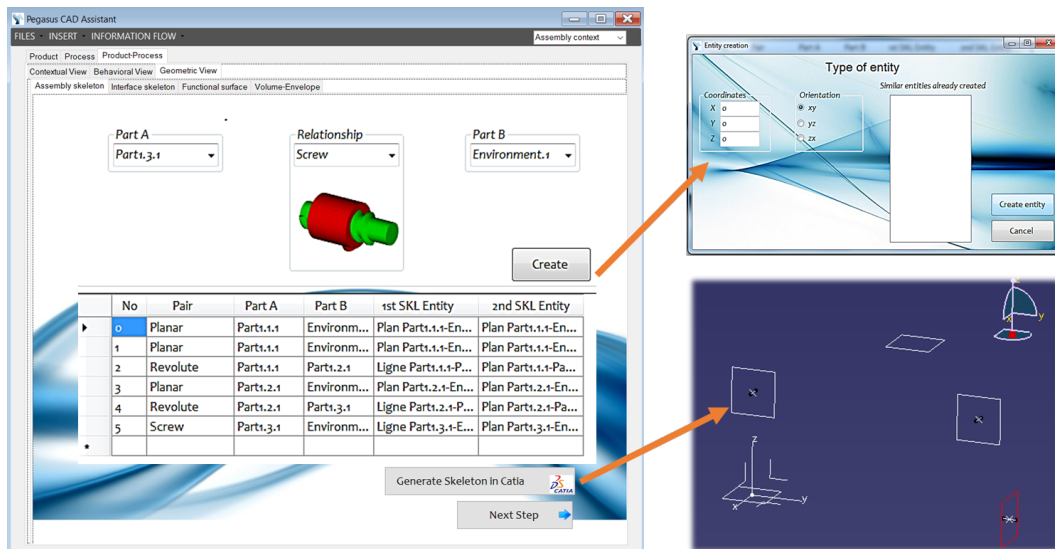


Figure 4.12 – Interface de PEGASUS CAD Assistant : vue géométrique

4.3.3/ CONCEPTION DE LA COUCHE À BASE D'ONTOLOGIE (ÉTAPES 3 JUSQU'À 8)

Il est essentiel de présenter le développement de l'ontologie pour le « *contexte augmenté* », afin de concevoir un système intelligent, dédié à la gestion des connaissances. La réutilisation d'une ontologie existante est recommandée pour structurer, organiser et exploiter les connaissances. L'ontologie PRONOA3 (Figure 4.13) est fondée sur la théorie JANUS, qui a comme fonction de décrire la définition et l'évolution des relations produit-process, elles-mêmes basées sur la méréotopologie spatio-temporelle. La théorie JANUS est dérivée de la norme ISO 15926 [Standard ISO 15926, 2003], qui soutient les changements de données dans leur diffusion auprès de plusieurs systèmes. PRONOA3 comprend ainsi une méta-ontologie, une ontologie de domaine, ainsi que l'application ontologique :

- La **méta-ontologie** est la justification de la modélisation des méta-classes, représentant des objets (c'est-à-dire les régions) et des relations (à savoir les primitives) dans les dimensions spatiales, temporelles et spatio-temporelles ;
- L'**ontologie de domaine** contient des concepts ou des classes divisés en deux sections : spatiales et temporelles. Les classes spatiales représentent la position du produit et ses relations (c'est-à-dire l'entité géométrique, les contraintes et les fonctions ou features d'assemblage dans l'environnement CAO, tandis que les classes temporelles définissent le process du produit (par exemple, les opérations de montage). Le graphe spatio-temporel est la combinaison des entités produit-process (autrement dit, de leur évolution dans le temps au cours de la conception : volume balayé, et les changements apportés) et des relations associées (à savoir le positionnement des liaisons cinématiques/technologiques, le changement dans la conception, etc.) ;
- L'**ontologie d'application** permet de capturer et d'externaliser les connaissances explicites via les instances issues des classes d'objets.

Chaque concept exploité dans l'architecture ontologique est implémenté avec l'éditeur d'ontologies Protégé, en utilisant le plugin OWL [Knublauch *et al.*, 2004]. Il faut savoir que Protégé est un outil Java open source qui fournit une architecture extensible pour la création d'applications personnalisées basées sur la connaissance, et qui, contrairement aux **API (Application Program Interface)** OWL, a la particularité de contourner les **RDF (Ressource Description Framework)** pour fournir des services basés sur OWL. En effet, Protégé permet d'appliquer des requêtes **SPARQL (SPARQL Protocol and RDF Query Language)**, de créer des constructions OWL, de créer des axiomes et d'exécuter des inférences. Dans notre cas, l'ontologie présentée est codée en OWL DL.

Notre choix s'est porté sur Protégé car il peut être étendu pour créer de véritables applications à base de connaissances en utilisant un moteur d'inférence à la fois pour raisonner et déduire de nouveaux faits via l'application de règles sur l'ontologie et ses instances (méta-raisonnement). Il fournit un ensemble d'opérateurs pour la combinaison des classes dans la modélisation de l'ontologie comme l'intersection, l'union, et la négation. Pour les besoins de notre recherche, nous n'allons pas modifier la méta-ontologie issue des précédents travaux réalisés au sein de notre laboratoire. Nous nous focaliserons sur la couche de l'ontologie de domaine et la couche de l'ontologie d'application. Nous utiliserons une convention des dénominations suivantes : les noms des classes, écrits en lettres majuscules/minuscules, avec la police Arial (par exemple, *Physical_Product*), les noms des attributs et des relations sont en majuscule/minuscule Courier Nouveau (par exemple, *is_fixed_to*), tandis que les noms des instances sont en italique (par exemple, *Part_1*).

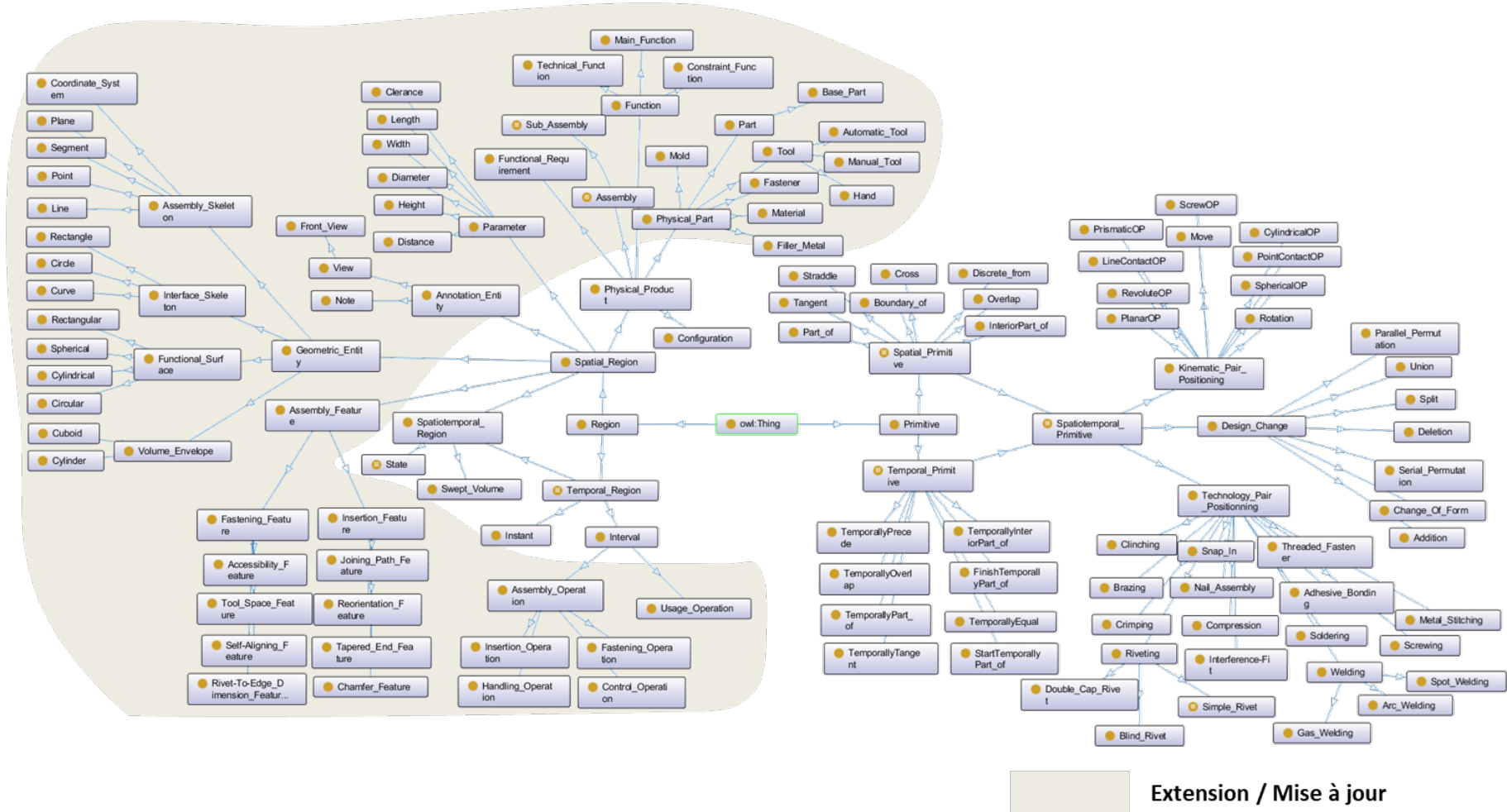


Figure 4.13 – Modèle ontologique de PRONOA3

Plusieurs étapes sont nécessaires pour définir l'ontologie. Nous nous inspirerons de la méthode de [Monticolo, 2016] qui propose la conception d'une ontologie en se basant sur l'existant. Tout d'abord, nous définissons l'ontologie de domaine, qui constitue la première étape permettant de l'identifier et de l'associer à une terminologie. Ensuite, nous détaillerons les classes des concepts, leurs hiérarchies et leurs propriétés. L'étape suivante consiste à décrire leurs relations, afin de créer les instances. Il est important d'évoquer une ontologie avec une représentation formelle car elle doit tenir compte de la description des concepts, des relations et des attributs selon le langage naturel. Les tableaux présentés ci-dessous sont retranscrits tels qu'ils sont incorporés dans le système, c'est-à-dire en anglais.

L'analyse terminologique (Tableau 4.1), est effectuée pour être enrichie en utilisant l'ontologie existante, tout en considérant trois situations possibles :

- Un terme correspond à un concept. Cette situation est la plus simple, elle permet d'étiqueter directement le concept choisi ;
- Un terme correspond à plusieurs concepts, sources de confusion car utilisé dans plusieurs arcs de l'ontologie. Dans ce cas, un effort est nécessaire pour expliquer la signification du terme dans sa description en langage naturel ;
- Plusieurs termes correspondent à un concept. Cette situation est fréquente, en particulier en cas de réutilisation d'ontologies existantes. Les termes sont alors des synonymes, hyperonymes et hyponymes.

Tableau 4.1 – Extrait des concepts et des terminologies associés

Concepts issus de PRONOA3 (classe)	Concepts liés à l'ontologie existante	Synonymes, hyperonymes, hyponymes ajoutés
Geometric_Entity	assignment, catalog, description	blueprint, specs, stipulation, condition, designation, detail, item, particular
Assembly_Sequence	continuity, precedence, priority, chain, procession, progression, succession, series	arrangement, array, disposal, disposition, distribution, ordering, order, setup
Assembly_Operation	drill, routine, form, manner, method, mode, system, technic, way, approach	course, process, procedure, proceeding
Assembly_Feature	badge, indication, sign, emblem, symbol, token	affection, attribute, attribution, character, criterion, diagnostic, differentia, characteristic
...

Nous devons spécifier pour chaque concept un thésaurus (classification et description de chaque concept). Pour ce faire, les trois tableaux suivants ont été élaborés : Tableau 4.2, Tableau 4.3 et Tableau 4.4 :

- Le tableau des concepts spécifiés, qui regroupe chaque concept par un nom unique (classe). Dans ce tableau, sont représentés : le domaine d'utilisation du concept, le concept supérieur associé (Super Class), les synonymes identifiés, une description en langage naturel et le nom de la partie prenante qui a publié le concept ;

- Le tableau des relations, qui définit les relations possibles entre les concepts spécifiés, comprenant notamment, le concept source (domaine), le concept cible (portée), la description en langage naturel, ainsi que le nom de la partie prenante qui a défini la relation ;
- Le tableau d'attributs, qui spécifie les propriétés d'une classe et du concept. En outre, il spécifie également leur type de valeur, une description en langage naturel et le nom de la partie prenante qui a spécifié l'attribut.

Tableau 4.2 – Extrait du tableau de spécification des concepts

Classe	Domaine	Super classe	Définition de la langue naturelle	Synonyme	Partie prenante
Physical_- Part	Physical_Pro- duct	Region	one of the parts of something (such as a system or mixture) : an important piece of product	Component	Product architect, Designer
Functional_- Requirement	Physical_Pro- duct	Region	the functionality which accompanies a program and states in detail what is to be performed by the system	Need, Specification	Product architect
Kinematic_- Pair_Positioning	Spatiotemporal_ Primitive	Primitive	connection between two parts with a specified constraint on their movement	Motion	Product architect
...

Tableau 4.3 – Extrait du tableau des propriétés des objets

Propriété de l'objet (relation)	Inverse de	Domaine	Portée	Définition de la langue naturelle	Partie prenante
has_Geometric_Entity	is_composed_of	Spatiotemporal_Primitive	Geometric_Entity	Relation describing spatiotemporal primitive composed of geometric entity	Product architect
is_composed_of	has_Geometric_Entity	Geometric_Entity	Spatiotemporal_Primitive	Relation describing geometric entity which is composed of spatialtemporal primitive	Product architect
has_Kinematic_Pair_Positioning	is_relation_of	Part	Kinematic_Pair_Positioning	Relation describing one of part-to-part relationships	Product architect
is_relation_of	has_Kinematic_Pair_Positioning	Kinematic_Pair_Positioning	Part	Relation describing a kinematic pair positioning in accordance with the part	Product architect
has_fixation	is_fixed_to	Fastening_Feature	Part	Relation describing the allocation of a fastening feature on parts	Process engineer, Assembly planner
is_fixed_to	has_fixation	Part	Fastening_Feature	Relation describing the allocation between two parts of the assembly structure	Process engineer, Assembly planner
...

Tableau 4.4 – Extrait du tableau des propriétés des données

Propriété des données (attribut)	Concept associé (domaine)	Valeur (portée)	Définition de la langue naturelle	Partie prenante
Part_number	Part	Integer	Increment number of part	Product architect
Part_name	Part	String	Name of part	Product architect
Technology_Pair_Positioning_name	Technology_Pair_Positioning	String	Name of the technology relation	Product architect, Process engineer
Plane_number	Plane	Integer	Increment number of plane	Product architect, Process engineer
...

Pour finaliser l'ontologie, il est important d'apporter des informations complémentaires en imposant des restrictions. Pour cela des règles sont présentées dans le Tableau 4.5, le Tableau 4.6, le Tableau 4.7 et le Tableau 4.8, qui définissent les règles DL mises en œuvre dans PRONOIA3. Elles représentent l'évolution des règles émises dans PRONOIA2 et tendent à compléter les classes ayant des propriétés restrictives. Le but consiste à les écrire dans l'ontologie (c'est-à-dire que la propriété est concernée par les classes dans la méta-ontologie, alors que la propriété sera consignée dans le fichier connexe). L'objectif est de décrire les restrictions aussi élevées que possible dans la hiérarchie des classes et d'aborder le concept supérieur pour éviter les répétitions [Gruhier *et al.*, 2015].

Tableau 4.5 – Règles DL de PRONOA3 décrivant les liaisons cinématiques

Nom de la classe	Type de règle	Règle DL
RevoluteOP	Cardinality	$RevoluteOP \sqsubseteq Kinematic_Pair_Positionning$ $(has_Assembly_Skeleton = 1 Line) \sqcap (has_Assembly_Skeleton = 1 Plane) \sqcap (has_Interface_Skeleton = 2 Circle) \sqcap (has_Interface_Skeleton = 1 Circle \sqcup Rectangle) \sqcap (has_Interface_Skeleton = 1 Segment) \sqcap (has_Functional_Surface = 1 Cylindrical) \sqcap (has_Functional_Surface = 1 Circular) \sqcap (has_Volume - Envelope = 2 Cylinder) \sqcap (has_Volume - Envelope = 1 Cylinder \sqcup Cuboid)$
PrismaticOP	Cardinality	$PrismaticOP \sqsubseteq Kinematic_Pair_Positionning$ $(has_Assembly_Skeleton = 1 Line) \sqcap (has_Interface_Skeleton = 1 Rectangle) \sqcap (has_Functional_Surface = 1 Rectangular) \sqcap (has_Volume - Envelope = 1 Cuboid)$
ScrewOP	Cardinality	$ScrewOP \sqsubseteq Kinematic_Pair_Positionning$ $(has_Assembly_Skeleton = 1 Line) \sqcap (has_Assembly_Skeleton = 1 Plane) \sqcap (has_Interface_Skeleton = 1 Circle) \sqcap (has_Interface_Skeleton = 1 Segment) \sqcap (has_Interface_Skeleton = 1 Circle \sqcup Rectangle) \sqcap (has_Functional_Surface = 1 Cylindrical) \sqcap (has_Volume - Envelope = 1 Cylinder) \sqcap (has_Volume - Envelope = 1 Cylinder \sqcup Cuboid)$
CylindricalOP	Cardinality	$CylindricalOP \sqsubseteq Kinematic_Pair_Positionning$ $(has_Assembly_Skeleton = 1 Line) \sqcap (has_Assembly_Skeleton = 1 Segment) \sqcap (has_Interface_Skeleton = 1 Circle) \sqcap (has_Functional_Surface = 1 Cylindrical) \sqcap (has_Volume - Envelope = 1 Cylinder) \sqcap (has_Volume - Envelope = 1 Cylinder \sqcup Cuboid)$
SphericalOP	Cardinality	$SphericalOP \sqsubseteq Kinematic_Pair_Positionning$ $(has_Assembly_Skeleton = 1 Point) \sqcap (has_Interface_Skeleton = 1 Circle) \sqcap (has_Functional_Surface = 1 Spherical) \sqcap (has_Volume - Envelope = 1 Cylinder) \sqcap (has_Volume - Envelope = 1 Cylinder \sqcup Cuboid)$
PlanarOP	Cardinality	$PlanarOP \sqsubseteq Kinematic_Pair_Positionning$ $(has_Assembly_Skeleton = 1 Plane) \sqcap (has_Interface_Skeleton = 1 Circle \sqcup Rectangle) \sqcap (has_Interface_Skeletone = 1 Circle \sqcup Rectangle) \sqcap (has_Functional_Surface = 1 Circular \sqcup Rectangular) \sqcap (has_Functional_Surface = 1 Circular \sqcup Rectangular) \sqcap (has_Volume - Envelope = 1 Cylinder \sqcup Cuboid)$

Tableau 4.6 – Règles DL de PRONCIA3 décrivant les liaisons technologiques

Nom de la classe	Type de règle	Règle DL
Interference-Fit	Cardinality	$Interference - Fit \sqsubseteq Technology_Pair_Positionning \sqcap (has_Assembly_Skeleton = 1 Line) \sqcap (has_Assembly_Skeleton = 1 Segment) \sqcap (has_Interface_Skeleton = 1 Circle) \sqcap (has_Functional_Surface = 1 Cylindrical) \sqcap (has_Volume - Envelope = 1 Cylinder) \sqcap (has_Volume - Envelope = 1 Cylinder \sqcup Cuboid)$
Snap_In	Cardinality	$Snap_In \sqsubseteq Technology_Pair_Positionning \sqcap (has_Assembly_Skeleton = 1 Line) \sqcap (has_Assembly_Skeleton = 1 Plane) \sqcap (has_Interface_Skeleton = 1 Rectangle) \sqcap (has_Functional_Surface = 1 Rectangular) \sqcap (has_Volume - Envelope = 1 Cuboid)$
Simple_Rivet	Cardinality	$Simple_Rivet \sqsubseteq Technology_Pair_Positionning \sqcap (has_Assembly_Skeleton = 2 Line) \sqcap (has_Assembly_Skeleton = 3 Plane) \sqcap (has_Interface_Skeleton = 1 Circle \sqcup Rectangle) \sqcap (has_Interface_Skeleton = 1 Circle \sqcup Rectangle) \sqcap (has_Interface_Skeleton = 1 Circle \sqcup Rectangle) \sqcap (has_Interface_Skeleton = 1 Circle \sqcup Rectangle) \sqcap (has_Interface_Skeleton = 4 Circle) \sqcap (has_Interface_Skeleton = 2 Segment) \sqcap (has_Functional_Surface = 1 Circular \sqcup Rectangular) \sqcap (has_Functional_Surface = 1 Circular \sqcup Rectangular) \sqcap (has_Functional_Surface = 2 Cylindrical) \sqcap (has_Functional_Surface = 2 Circular) \sqcap (has_Volume - Envelope = 1 Cylinder \sqcup Cuboid) \sqcap (has_Volume - Envelope = 1 Cylinder \sqcup Cuboid) \sqcap (has_Volume - Envelope = 4 Cylinder)$
Threaded_Fastener	Cardinality	$Threaded_Fastener \sqsubseteq Technology_Pair_Positionning \sqcap (has_Assembly_Skeleton = 2 Line) \sqcap (has_Assembly_Skeleton = 3 Plane) \sqcap (has_Interface_Skeleton = 1 Circle \sqcup Rectangle) \sqcap (has_Interface_Skeleton = 1 Circle \sqcup Rectangle) \sqcap (has_Interface_Skeleton = 1 Circle \sqcup Rectangle) \sqcap (has_Interface_Skeleton = 1 Circle \sqcup Rectangle) \sqcap (has_Interface_Skeleton = 3 Circle) \sqcap (has_Interface_Skeleton = 2 Segment) \sqcap (has_Functional_Surface = 1 Circular \sqcup Rectangular) \sqcap (has_Functional_Surface = 1 Circular \sqcup Rectangular) \sqcap (has_Functional_Surface = 2 Cylindrical) \sqcap (has_Functional_Surface = 1 Circular) \sqcap (has_Volume - Envelope = 1 Cylinder \sqcup Cuboid) \sqcap (has_Volume - Envelope = 1 Cylinder \sqcup Cuboid) \sqcap (has_Volume - Envelope = 1 Cylinder \sqcup Cuboid) \sqcap (has_Volume - Envelope = 3 Cylinder)$

Tableau 4.7 – Règles DL de PRONOA3 décrivant les fonctions d'assemblage (1)

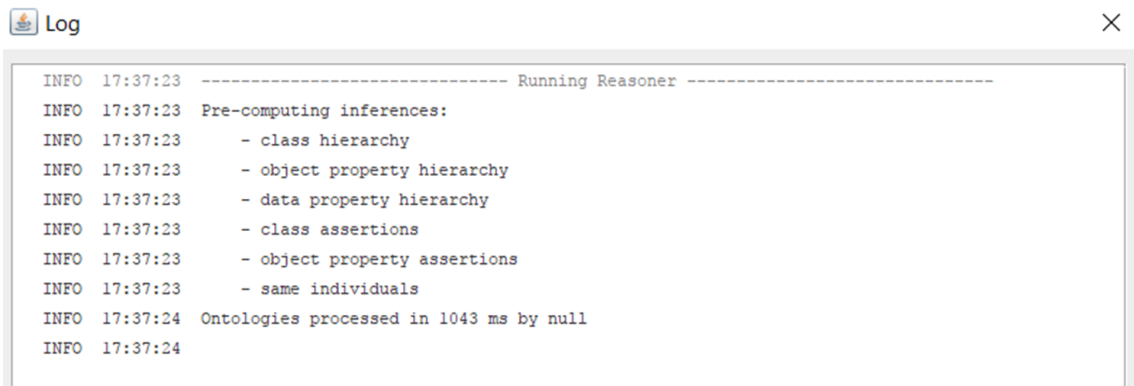
Nom de la classe	Type de règle	Règle DL
Accessibility_ Feature	Cardinality	$Accessibility_Feature \sqsubseteq Fastening_Feature$ $(has_Volume - Envelope = 1 \quad Cylinder \sqcup Cuboid)$ $(has_Parameter = 1 \quad Width) \sqcap (has_Parameter = 1 \quad Height)$ $(has_Annotation_Entity \forall Accessibility_Design_Rule_Annotation) \sqcap$ $(has_Annotation_Entity \forall Accessibility_Best_Pratice_Annotation) \sqcap$ $(has_Annotation_Entity = 1 \quad Front_View)$
Rivet-To- Edge_ Dimension_ Feature	Cardinality	$Rivet - To - Edge_Dimension_Feature \sqsubseteq$ $Fastening_Feature \sqcap (has_Functional_Surface =$ $1 \quad Circular \sqcup Rectangular) \sqcap (has_Parameter =$ $1 \quad Distance) \sqcap (has_Annotation_Entity \quad \forall \quad Rivet -$ $To - Edge_Dimension_Design_Rule_Annotation) \sqcap$ $(has_Annotation_Entity \quad \forall \quad Rivet - To -$ $Edge_Dimension_Best_Pratice_Annotation) \sqcap$ $(has_Annotation_Entity = 1 \quad Front_View)$
Self- Aligning_ Feature	Cardinality	$Self - Aligning_Feature \sqsubseteq Fastening_Feature$ $(has_Functional_Surface = 1 \quad Circular \sqcup$ $Rectangular) \sqcap (has_Annotation_Entity \quad \forall \quad Self -$ $Aligning_Design_Rule_Annotation) \sqcap$ $(has_Annotation_Entity \quad \forall \quad Self -$ $Aligning_Best_Pratice_Annotation) \sqcap$ $(has_Annotation_Entity = 1 \quad Front_View)$
Tool_ Space_ Feature	Cardinality	$Tool_Space_Feature \sqsubseteq Fastening_Feature$ $(has_Volume - Envelope = 1 \quad Cylinder \sqcup Cuboid) \sqcap$ $(has_Parameter = 2 \quad Length) \sqcap (has_Parameter =$ $1 \quad Width) \sqcap (has_Parameter = 1 \quad Diameter) \sqcap$ $(has_Annotation_Entity \forall Tool_Space_Design_Rule_Annotation) \sqcap$ $(has_Annotation_Entity \forall Tool_Space_Best_Pratice_Annotation) \sqcap$ $(has_Annotation_Entity = 1 \quad Front_View)$

Tableau 4.8 – Règles DL de PRONOA3 décrivant les fonctions d'assemblage (2)

Nom de la classe	Type de règle	Règle DL
Chamfer_ - Feature	Cardinality	$Chamfer_Feature \sqsubseteq Insertion_Feature \sqcap (has_Volume - Envelope = 1 \quad Chamfer) \sqcap (has_Parameter = 2 \quad Diameter) \sqcap (has_Parameter = 1 \quad Clearance) \sqcap (has_Annotation_Entity \forall Chamfer_Design_Rule_Annotation) \sqcap (has_Annotation_Entity \forall Chamfer_Best_Pratice_Annotation) \sqcap (has_Annotation_Entity = 1 \quad Front_View)$
Joining_ - Path_ Feature	Cardinality	$Joining_Path_Feature \sqsubseteq Insertion_Feature \sqcap (has_Assembly_Skeleton = 1 \quad Plane) \sqcap (has_Volume - Envelope = 1 \quad Cylinder \sqcup Cuboid) \sqcap (has_Parameter = 2 \quad Diameter) \sqcap (has_Annotation_Entity \forall Joining_Path_Design_Rule_Annotation) \sqcap (has_Annotation_Entity \forall Joining_Path_Best_Pratice_Annotation) \sqcap (has_Annotation_Entity = 1 \quad Front_View)$
Reorientation_ Feature	Cardinality	$Reorientation_Feature \sqsubseteq Insertion_Feature \sqcap (has_Assembly_Skeleton = 1 \quad Line \sqcap 1 \quad Coordinate \quad Axis \quad System) \sqcap (has_Annotation_Entity \forall Reorientation_Design_Rule_Annotation) \sqcap (has_Annotation_Entity \forall Reorientation_Best_Pratice_Annotation) \sqcap (has_Annotation_Entity = 1 \quad Front_View)$
Tapered_ - End_ Feature	Cardinality	$Tapered_End_Feature \sqsubseteq Insertion_Feature \sqcap (has_Functional_Surface = 1 \quad Circular \sqcup Rectangular \sqcup 1 \quad Cylindrical) \sqcap (has_Annotation_Entity \forall Tapered_End_Design_Rule_Annotation) \sqcap (has_Annotation_Entity \forall Tapered_End_Best_Pratice_Annotation) \sqcap (has_Annotation_Entity = 1 \quad Front_View)$

C'est dans ce contexte que nous pouvons vérifier la consistance des classes. En effet, une classe consistante peut à la fois dire si elle peut avoir une instance ou s'il s'agit d'une classe vide (équivalent en OWL à owl :nothing). Par ailleurs, une ontologie est dite consistante si elle a un modèle, à savoir une interprétation qui satisfait tous ses axiomes. Toutefois, une ontologie peut être consistante et avoir une classe inconsistante. Il suffit simplement que la classe inconsistante soit interprétée comme l'ensemble vide. Pour cela, nous avons exploité Pellet pour vérifier la consistance de l'ontologie (Figure 4.14).

Une fois l'ontologie réalisée, les instances sont créées afin d'exploiter le mécanisme d'inférence des règles. Il est important d'exploiter le système d'inférence avec une ontologie formelle, car la déduction logique, l'induction et l'abduction doivent, selon la situation, répondre aux intentions du concepteur.



```

INFO 17:37:23 ----- Running Reasoner -----
INFO 17:37:23 Pre-computing inferences:
INFO 17:37:23   - class hierarchy
INFO 17:37:23   - object property hierarchy
INFO 17:37:23   - data property hierarchy
INFO 17:37:23   - class assertions
INFO 17:37:23   - object property assertions
INFO 17:37:23   - same individuals
INFO 17:37:24 Ontologies processed in 1043 ms by null
INFO 17:37:24

```

Figure 4.14 – Résultat du test de consistance de PRONOA3 effectué avec l'éditeur d'ontologie Protégé et le moteur d'inférence Pellet

4.3.4/ CONCEPTION DE LA COUCHE LOGIQUE DE RAISONNEMENT (ÉTAPES 9 JUSQU'À 11)

Pour faciliter la compréhension du contexte de conception augmenté, nous avons défini des unités de raisonnement appelées **RUN (Reasoning UNit)**, qui décomposeront les connaissances D_k et P_k . Ces unités sont les éléments clés comprenant un ensemble de règles et de requêtes pour le moteur d'inférence. Ces unités sont formalisées manuellement par un spécialiste qui traduit les règles d'experts suivant le langage sémantique spécifique.

De plus, le moteur d'inférence est lui-même fondé sur des informations relatives au graphe spatio-temporel et à la définition des entités des espaces de conception, dans l'environnement CAO.

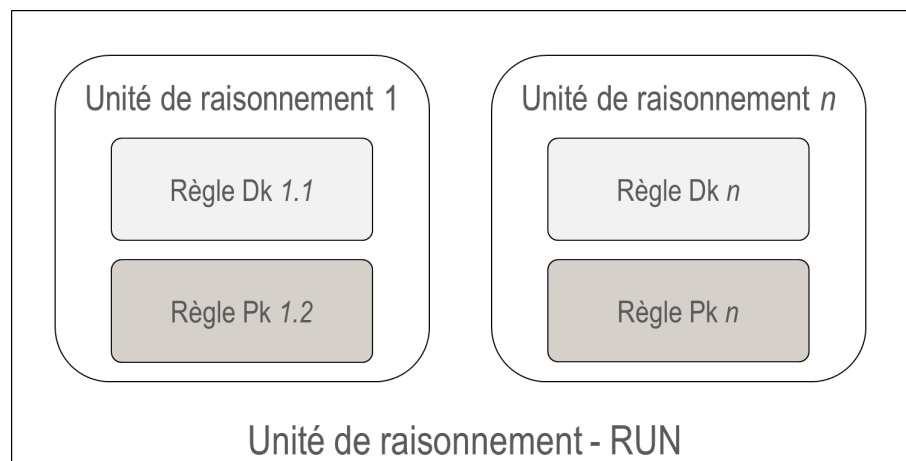


Figure 4.15 – Introduction des unités de raisonnement RUN

Ainsi, les unités de raisonnement RUN (Figure 4.15) sont appliquées sur la sémantique des données du produit limitées au domaine de l'assemblage. Celles-ci comprennent des règles à la fois pour les D_k et les P_k , lesquelles sont la représentation des C_k . Elles comprennent également l'identification, l'allocation et l'instanciation, des éléments issus des fonctions d'assemblage (assembly feature), ainsi que leur impact dans la conception, tels que présentés dans le Tableau 4.9

récapitulant les unités de raisonnement. Nous allons nous focaliser sur l'une d'entre elles, à savoir l'identification dans le « *contexte augmenté* » de conception et l'instanciation de la fonction « *dimension du bord de rivet* » (rivet-to-edge-dimension feature) pour l'exemple illustratif retenu : le cas d'un système de fixation par rivet de deux tôles (cf Figure 4.16 ci-dessous). Nous représentons les conditions préalablement remplies issues du Ck, afin de les transposer en règles SWRL et SQWRL. Nous appelons *S* les spécifications détaillant ces conditions.

Tableau 4.9 – Sommaire des unités de raisonnement - RUN

Unité de raisonnement RUN	Situation	Capabilité
RUN1	Squelette d'assemblage	Identification et instanciation de la fonction d'accessibilité (accessibility feature)
RUN2	Volume-enveloppe	Identification et instanciation de la fonction de chanfrein (chamfer feature)
RUN3	Squelette d'assemblage et squelette d'interface	Identification et instanciation de la fonction d'auto-alignement (self-aligning feature)
RUN4	Squelette d'interface	Identification et instanciation de la fonction de réorientation (reorientation feature)
RUN5	Squelette d'assemblage et surface fonctionnelle	Identification et instanciation de la fonction de réduction du chemin d'insertion (joining path feature)
RUN6	Squelette d'assemblage et squelette d'interface	Identification et instanciation de la fonction de l'espace outil (tool space feature)
RUN7	Squelette d'assemblage et squelette d'interface	Identification et instanciation de la fonction d'extrémité (tapered end feature)
RUN8	Squelette d'assemblage et squelette d'interface	Identification et instanciation de la fonction de dimension du bord de rivet (rivet-to-edge-dimension feature)

Nous appliquons ainsi des règles de langage pour l'inférence avec SWRL, les règles de langage pour la requête avec le SQWRL et utilisons le moteur de raisonnement Pellet. [Zhu *et al.*, 2011] ont formulé plusieurs précisions concernant :

- **SWRL** : SWRL est un langage de règles s'appuyant sur OWL qui lui, constitue un langage de modélisation de l'information basé sur la logique. OWL DL est un sous-ensemble du langage de l'ontologie OWL et repose conceptuellement sur la logique de description (DL). Il profite de la sémantique bien définie de DL, des propriétés formelles, et des algorithmes de raisonnements connus [Baader *et al.*, 2003] ;
- **SQWRL** : il s'agit d'un langage de requêtes qui est basé sur le SWRL, alors que ce dernier est un langage de règles, et non de requêtes. Au contraire, SPARQL est un protocole simple de langage de requêtes RDF défini par le W3C [W3C, 2013]. SQWRL a la particularité, par rapport à SPARQL d'interroger les ontologies OWL et de bien interpréter la sémantique associée [O'Connor & Das, 2009] ;
- **Pellet** : plusieurs moteurs de raisonnement prennent en charge le mécanisme de raisonnement ontologique. Cependant, certains raisonneurs de DL, comme FACT++ et Racer, ne suffisent pas à comprendre les ontologies générales de OWL DL. De nos jours, en tant que moteur d'inférence, Pellet constitue une solution intéressante, car il peut déterminer si une ontologie donnée est cohérente et il identifie les relations de consistance entre les concepts et les autres fonctions [Dentler *et al.*, 2010].

Présentation de l'unité de raisonnement 8 - RUN8 Nous allons nous concentrer sur l'une de ces unités de raisonnement, la RUN8, dans laquelle nous nous proposons d'étudier les étapes de raisonnement. Nous partons du postulat que les informations issues du « *contexte augmenté* » sont : les informations sur les relations d'assemblage, l'ordre d'assemblage entre les composants, les entités géométriques définissant les espaces de conception, etc. Dans ce cas, nous identifions le contexte de conception augmenté et générons la fonction « *dimension du bord de rivet* » rivet-to-edge-dimension feature) (Figure 4.16).

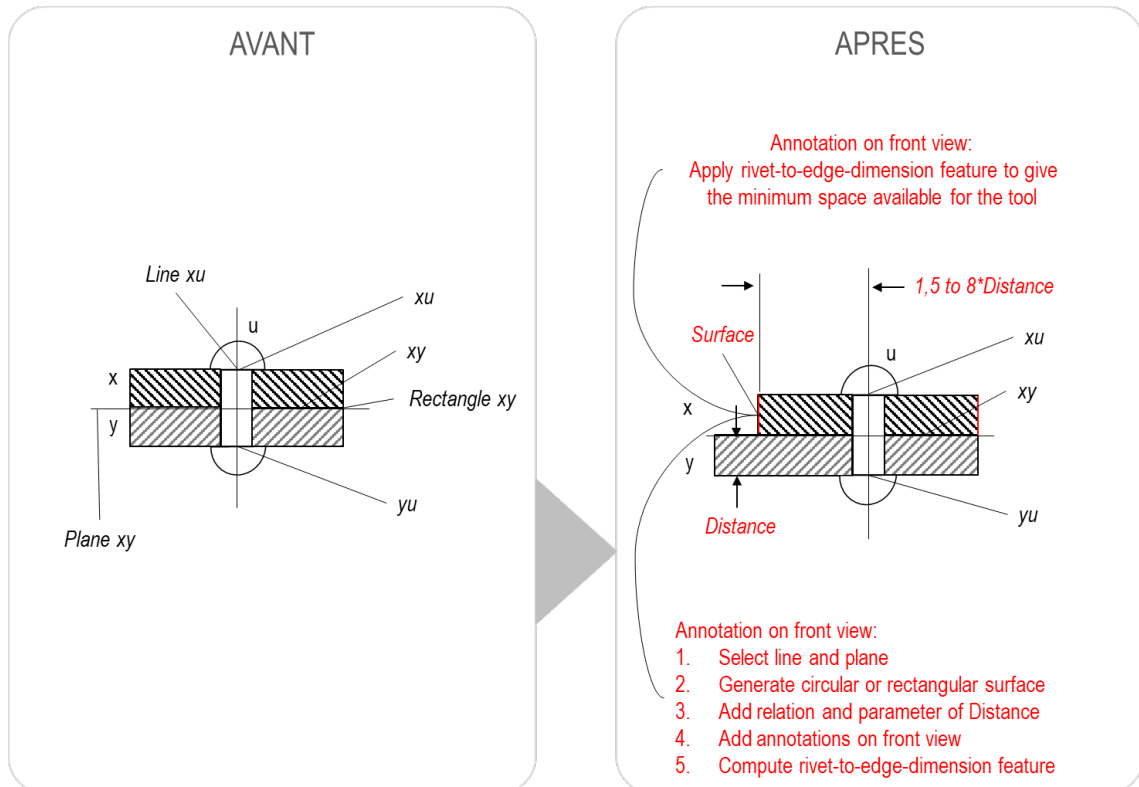


Figure 4.16 – But de la fonction de dimension du bord de rivet

Cette fonction a la particularité de contraindre au travers d'une règle de conception, en faisant coïncider son paramètre de Distance entre l'axe traversant le rivet u et la pièce x (ligne xu) avec la surface de la pièce x . Le paramètre de décalage doit être compris entre 1,5mm, puis doit être multiplié par 8mm avec le paramètre Distance. Le but de cette fonction est de faciliter le passage de l'outil permettant de riveter. Ainsi, les instances de l'ontologie PRONOA3 (Figure 4.17), permettent d'identifier précisément les espaces de conception générés, pour connaître à la fois la surface fonctionnelle à générer sur le squelette d'assemblage composé d'une ligne et d'un plan, et du squelette d'interface composé d'un rectangle ou d'un cercle. Le paramètre de Distance est relié à la ligne et au rectangle contraints par la règle de conception. Viendront s'ajouter deux annotations sur la vue de face de la surface fonctionnelle, afin de préciser aux concepteurs la description de la fonction, ainsi que la bonne pratique pour l'appliquer.

—	has individual
—	has subclass
—	has_Assembly_Operation
—	has_Assembly_Skeleton
—	has_Interface_Skeleton
—	has_Technology_Pair_Positionning
—	is_inserted_into
—	Sub-Assembly_is_composed_of_Physical_Part
—	TemporallyPrecede_follow_Assembly_Operation
—	TemporallyPrecede_precede_Assembly_Operation

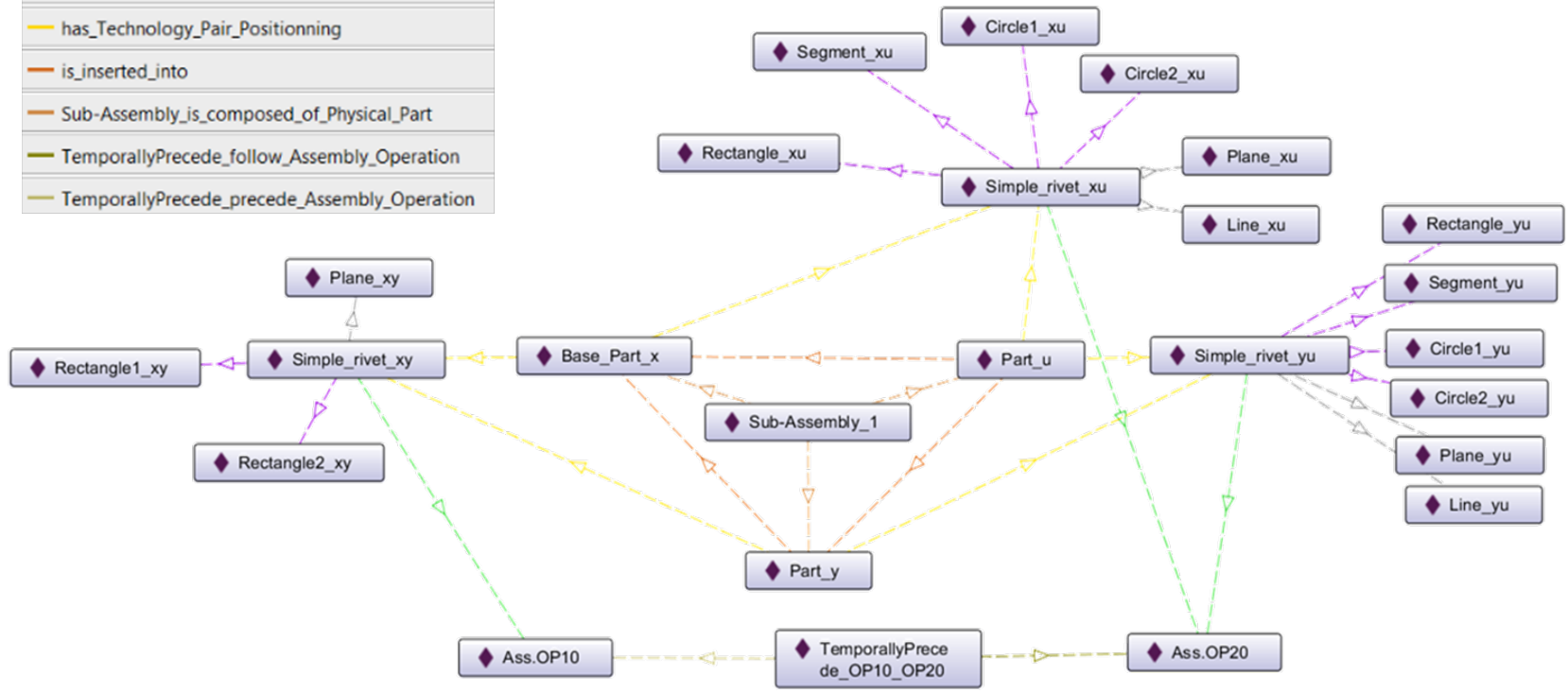


Figure 4.17 – Instances de départ de l'ontologie, représentant le « contexte augmenté », avant l'instanciation de la fonction

Nous allons décomposer l'unité de raisonnement en deux parties : les règles Dk 1.1 et Dk 1.2 seront d'abord présentées. Elles consistent à identifier et allouer les éléments (paramètres, annotations, éléments géométriques, et relations) de la fonction sur les espaces de conception. Tandis que les règles Pk 1.1 et Pk 1.2, sur la base de l'allocation précédente, vont être alloués, les éléments de la fonction selon un ordre bien défini.

Spécification 1 - Description en langage naturel de la règle Dk 1.1 :

Si

L'OP10 consiste à insérer la **pièce y** dans la **pièce stabilisée x** avec *comme relation technologique de positionnement xy* le **rivetage simple xy**, avec *comme squelette d'assemblage* : **1 plan xy**, et *comme squelette d'interface* : **1 rectangle xy**.

L'OP20 précède temporellement l'OP10 qui consiste à insérer la **pièce u** dans la **pièce stabilisée x** et la **pièce y** avec *comme relations technologiques de positionnement* le **rivetage simple xu et yu**, avec *comme squelette d'assemblage* : **1 ligne xu**.

La **fonction de dimension du bord de rivet** contient *comme surface fonctionnelle* : **1 extrusion rectangulaire**, avec *comme paramètre* : **1 distance**, et avec *comme entité d'annotation* : **1 vue de face** comprenant deux notes.

Alors

La situation de conception a *comme entrée* la **fonction dimension de bord de rivet** avec le **plan**, la **ligne** et le **rectangle**, dont le **rivetage simple xy** a *comme surface fonctionnelle* l'**extrusion rectangulaire** généré, et avec *comme entité d'annotation*, la **vue de face**. Cette dernière avec la **ligne** ont *comme paramètre* la *Distance*.

S1 - Règle SWRL pour identifier et allouer la fonction :

$$\begin{aligned}
 & \text{Simple_Rivet}(?spxu) \quad \wedge \quad \text{Simple_Rivet}(?spxy) \quad \wedge \quad \text{Simple_Rivet}(?spsyu) \quad \wedge \\
 & \text{has_Technology_Pair_Positionning}(?y, ?spsyu) \wedge \text{has_Technology_Pair_Positionning}(?u, ?spxu) \wedge \\
 & \text{has_Technology_Pair_Positionning}(?u, ?spsyu) \wedge \text{has_Technology_Pair_Positionning}(?x, ?spxu) \wedge \\
 & \text{has_Technology_Pair_Positionning}(?x, ?spxy) \wedge \text{has_Technology_Pair_Positionning}(?y, ?spxy) \wedge \\
 & \text{Part}(?u) \quad \wedge \quad \text{Part}(?y) \quad \wedge \quad \text{Base_Part}(?x) \quad \wedge \quad \text{Line}(?li) \quad \wedge \quad \text{Plane}(?pl) \quad \wedge \quad \text{Rectangle}(?rect) \quad \wedge \\
 & \text{has_Assembly_Skeleton}(?spxy, ?pl) \quad \wedge \quad \text{has_Assembly_Skeleton}(?spxy, ?li) \quad \wedge \\
 & \text{has_Interface_Skeleton}(?spxy, ?rect) \quad \wedge \quad \text{Assembly_Operation}(?opi) \quad \wedge \\
 & \text{Assembly_Operation}(?opj) \quad \wedge \quad \text{has_Assembly_Operation}(?spxu, ?opj) \quad \wedge \\
 & \text{TemporallyPrecede_follow_Assembly_Operation}(?tp, ?opi) \quad \wedge \quad \text{TemporallyPrecede}(?tp) \quad \wedge \\
 & \text{TemporallyPrecede_precede_Assembly_Operation}(?tp, ?opj) \wedge \\
 & \text{has_Assembly_Operation}(?spxy, ?opi) \quad \wedge \quad \text{has_Assembly_Operation}(?spsyu, ?opj) \quad \wedge \quad \text{Rivet} - \\
 & \text{To} - \text{Edge_Dimension_Feature}(?rted) \quad \wedge \quad \text{has_Functional_Surface}(?rted, ?rectangu) \quad \wedge \\
 & \text{has_Parameter}(?rted, ?dist) \quad \wedge \quad \text{Distance}(?dist) \quad \wedge \quad \text{has_Annotation_Entity}(?rted, ?fv) \quad \wedge \\
 & \text{Front_View}(?fv) \rightarrow \text{has_input}(?rted, ?pl) \wedge \text{has_input}(?rted, ?li) \wedge \text{has_input}(?rted, ?rect) \wedge \\
 & \text{has_Functional_Surface}(?spxy, ?rectangu) \quad \wedge \quad \text{has_Parameter}(?dist, ?rectangu) \quad \wedge \\
 & \text{has_Parameter}(?dist, ?li) \wedge \text{has_Annotation_Entity}(?rectangu, ?fv)
 \end{aligned}$$

Spécification 2 - Description en langage naturel de la règle Dk 1.2 :**Si**

La situation de conception a *comme entrée* la **fonction de dimension du bord de rivet** avec le **plan**, la **ligne** et le **rectangle**, dont le **rivetage simple xy** a *comme surface fonctionnelle* l'extrusion **rectangulaire** généré, et avec *comme entité d'annotation*, la **vue de face**. Cette dernière avec la **ligne** ont *comme paramètre* la *distance* et comme règle : $Distance \geq 1,5$ and $Distance \leq 8 \times Distance$.

Alors

Allouer entre la **ligne** et l'extrusion **rectangulaire**, la *contrainte de décalage*.

S2 - Règle SWRL pour vérifier le paramètre Distance et allouer si le paramètre est vrai, la contrainte de décalage :

$$Distance(?dist) \wedge has_Parameter(?dist,?li) \wedge has_Parameter(?dist,?rectangu) \wedge swrlb : multiply(?dist,8) \wedge swrlb : lessThanOrEqual(?dist,8) \wedge swrlb : greaterThanOrEqual(?dist,1.5) \rightarrow has_offset_constraint(?li,?rectangu)$$
Spécification 3 - Description en langage naturel de la règle Pk 1.1 :**Si**

La situation de conception a *comme entrée* la **fonction de dimension du bord de rivet** avec le **plan**, la **ligne** et le **rectangle**, dont le **rivetage simple xy** a *comme surface fonctionnelle* l'extrusion **rectangulaire** généré, et avec *comme entité d'annotation*, la **vue de face**. Cette dernière avec la **ligne** ont *comme paramètre* la *distance*.

Alors

Le **rectangle**, la **ligne** et le **plan** sont d'abord *appliqués avant* l'extrusion **rectangulaire**, et qui est lui-même *appliqué avant* la **distance**. Cette dernière est *appliquée avant* la **vue de face**.

S3 - Règle SWRL pour allouer la procédure d'application des éléments de la fonction de dimension du bord de rivet :

$$has_input(?rted,?rect) \wedge has_input(?rted,?li) \wedge has_Annotation_Entity(?rectangu,?fv) \wedge has_input(?rted,?pl) \wedge has_Parameter(?dist,?rectangu) \wedge has_Parameter(?dist,?li) \wedge has_Functional_Surface(?spxy,?rectangu) \rightarrow is_apply_before(?rect,?rectangu) \wedge is_apply_before(?li,?rectangu) \wedge is_apply_before(?pl,?rectangu) \wedge is_apply_before(?rectangu,?dist) \wedge is_apply_before(?dist,?fv)$$

Spécification 4 - Description en langage naturel de la règle Pk 1.2 :**Si**

Le **rectangle**, la **ligne** et le **plan** sont d'abord *appliqués avant* l'extrusion **rectangulaire**, et qui est lui-même *appliqué avant* la **distance**. Cette dernière *est appliquée avant* la **vue de face**.

Alors

Sélectionner les éléments appliqués, et trier selon la fonction.

S4 - Requête SQWRL pour chercher l'ordre d'application de la fonction :

```
Rivet - To - Edge_Dimension_Feature(?rted) ∧ is_apply_before(?rect, ?rectangu) ∧
is_apply_before(?li, ?rectangu) ∧ is_apply_before(?pl, ?rectangu) ∧
is_apply_before(?rectangu, ?dist) ∧ is_apply_before(?dist, ?fv) →
select(?li, ?rect, ?pl, ?rectangu, ?dist, ?fv), orderBy(?rted)
```

Ci-dessous (Figure 4.18), nous constatons le résultat des instances allouées et ordonnées selon leur ordre d'application. Il faut noter que leur ordre d'application suit la bonne pratique également instanciée sur la vue de face de la surface rectangulaire. Cela permet aux concepteurs de connaître les étapes, pour appliquer correctement la fonction.

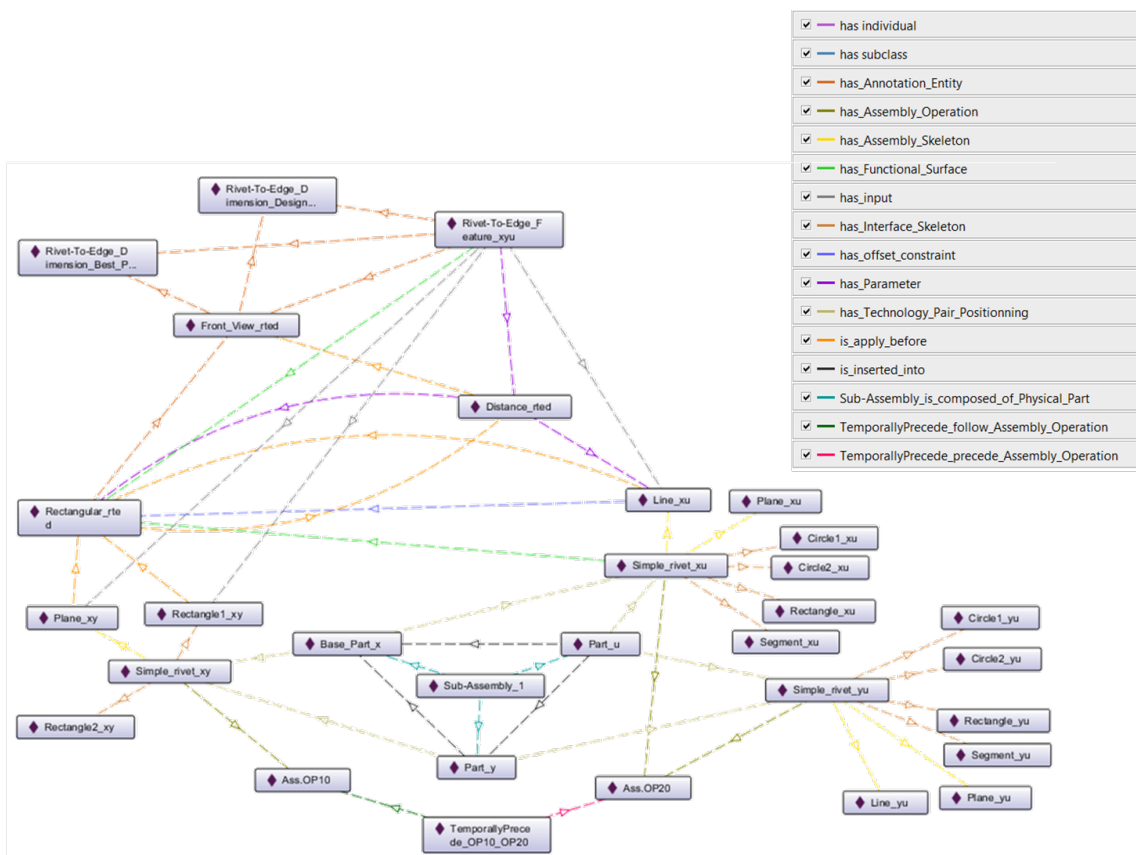


Figure 4.18 – Résultat des instances pour l'unité de raisonnement 8

4.4/ CONCLUSION

Ce chapitre a introduit l'approche MERCURY2, fondée sur les relations issues du « *contexte augmenté* » et de la situation définie par les espaces de conception. Il est basé sur le cadre méthodologique DAMON du chapitre 3 précédent, afin d'intégrer dans un HUB incubateur appelé PEGASUS, composé de plusieurs couches : l'ontologie PRONOA3, les unités de raisonnement RUN et l'interface PEGASUS CAD Assistant.

En effet, la présentation de l'architecture pour le développement d'un système sensible au contexte pour l'utilisation du « *contexte augmenté* », a permis de voir l'effet dans la conception, sur trois points principaux, (i) le fait de définir les parties prenantes, afin de comprendre leurs rôles et de saisir leurs intentions dans la situation de conception ; (ii) décrire une ontologie pour la formaliser et l'exploiter dans un système PLM ; (iii) et exploiter les connaissances via le moteur de raisonnement au travers de règles et de requêtes sémantiques.

CAS D'ÉTUDES ILLUSTRANT LE CADRE MÉTHODOLOGIQUE DAMON ET L'APPROCHE MERCURY2

*« Mieux vaut une conscience tranquille qu'une destinée prospère. J'aime mieux un bon sommeil
qu'un bon lit. »*

Victor Hugo

Sommaire

5.1 Introduction	118
5.2 Premier cas d'étude : Pince de désincarcération	118
5.2.1 Présentation	118
5.2.2 Description	119
5.3 Deuxième cas d'étude : Conception d'un drone montre	124
5.3.1 Présentation	124
5.3.2 Description	126
5.3.3 Démarche d'optimisation numérique pour améliorer l'efficacité de l'approche	135
5.3.4 Description du problème d'optimisation	136
5.4 Conclusion	142

5.1/ INTRODUCTION

Dans ce chapitre nous introduisons deux cas d'étude avec le cadre méthodologique DAMON et l'approche MERCURY2, pour améliorer la réutilisation des connaissances eu égard à la sensibilisation des concepteurs sur l'impact de leur choix de conception quant à la situation future du couple produit-process.

5.2/ PREMIER CAS D'ÉTUDE : PINCE DE DÉSINCARCÉRATION

5.2.1/ PRÉSENTATION

Dans cette section, nous nous focalisons sur l'application du cadre méthodologique DAMON dans un cas d'étude mécanique. Nous commençons en premier lieu par définir les objectifs de conception d'assemblage. Le produit étudié représente une pince de désincarcération dont la principale fonction est de permettre au sauveteur, d'accéder à la victime prisonnière de la carrosserie déformée. Plusieurs composants sont répertoriés avec les matériaux associés (Tableau 5.1). Une fois les composants définis, le fonctionnement du mécanisme est illustré à l'aide d'un schéma cinématique (Figure 5.1).

Tableau 5.1 – Liste des composants et des matériaux associés pour le cas d'étude de la pince

No.	Nom	Matériaux
1	Corps	EN AW-7075
2	Tige de vérin	36 Ni Cr Mo 16
3	Bielle droite	36 Ni Cr Mo 16
4	Lame droite	51 Cr V 4
5	Lame gauche	51 Cr V 4
6	Bielle gauche	36 Ni Cr Mo 16
7	Axe central	X 30 Cr 13
8	Axe épaulé droit	X 30 Cr 13
9	Axe épaulé gauche	X 30 Cr 13
10	Axe de tige	X 30 Cr 13

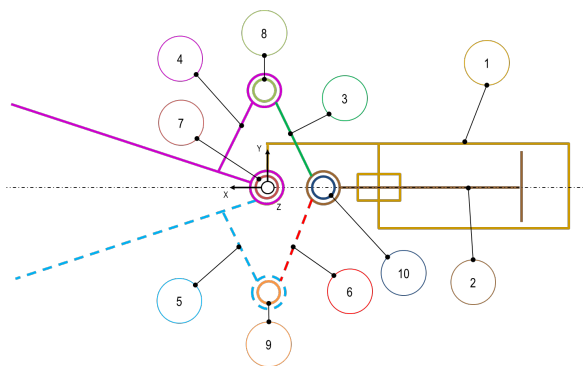


Figure 5.1 – Schéma cinématique de la pince de désincarcération étudiée

5.2.2/ DESCRIPTION

Ce cas d'application mécanique est illustré dans le paragraphe suivant le cadre méthodologique DAMON afin de comprendre les différences de géométrie induites par la planification de la séquence d'assemblage sélectionnée. Nous évaluerons l'instanciation des connaissances au travers de différentes itérations du produit.

Définition des relations d'assemblage Les relations d'assemblage sont établies en déterminant les liens entre chaque composant et leurs liaisons cinématiques/technologiques (Figure 5.2). Au préalable, on peut réaliser un schéma de principe pour modéliser le schéma cinématique et sur cette base réaliser un graphe orienté des liaisons concernées.

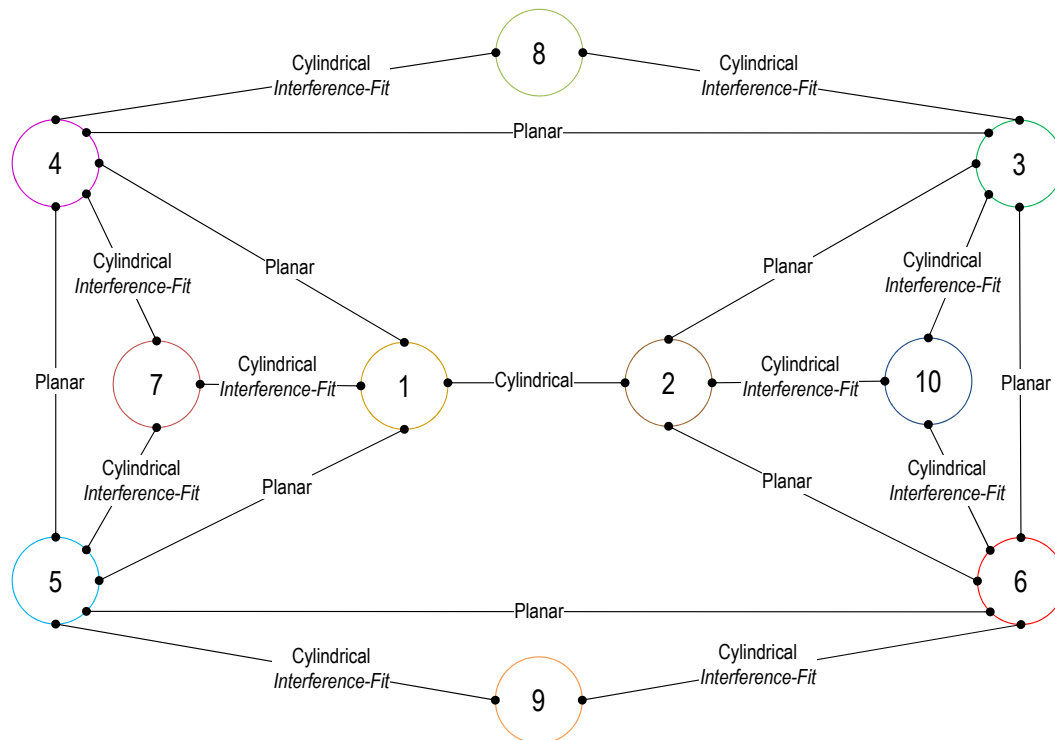


Figure 5.2 – Relations composant-à-composant de la pince de désincarcération

Génération de la séquence d'assemblage La Figure 5.3 montre les étapes servant à générer le futur contexte. Dans ce cas, le choix est axé sur la séquence qui décrit un ordre d'assemblage avec deux sous-ensembles. Les choix interviennent dans un contexte de Lean Manufacturing en exploitant la méthode **TPM (Total Productive Maintenance)**. Ce dernier a pour objectif d'augmenter le temps de fonctionnement des moyens de production également appelé temps d'ouverture. Parmi l'ensemble des gammes d'assemblage possible, les critères de choix sont les suivants :

- Les volumes de production/jour ;
- Le temps ouvert de travail (en minute) ;
- Le Design Takt Time (période de cadencement exprimée en seconde/unité) ;
- Le Taux de Rendement Synthétique (qui identifie et traite les causes de perte en production en pourcentage).

La séquence sélectionnée est la suivante :

$$\{7, 8, 9, \{1, 4, 5\}, \{2, 3, 6\}, 10\}$$

Cette séquence intègre les 2 sous-assemblages qui sont décrits ci-dessous :

$$\begin{cases} SA1 = \{2, 3, 6\}; \\ SA2 = \{1, 4, 5\}. \end{cases}$$

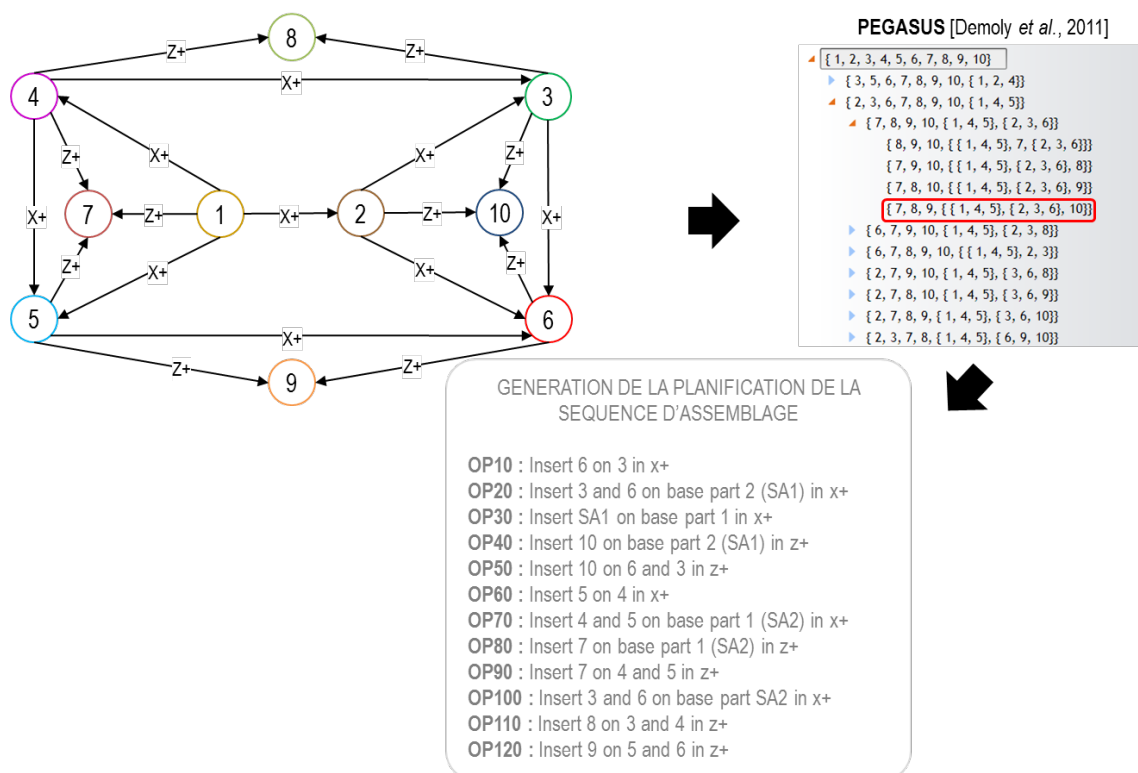


Figure 5.3 – Génération et sélection de planification de séquence d'assemblage avec PEGASUS

Génération du graphe spatio-temporel La planification de la séquence d'assemblage est superposée aux relations d'assemblage afin d'actualiser le graphe spatio-temporel. Les connaissances sont instanciées et proposées au concepteur lors de son activité de conception (Figure 5.4).

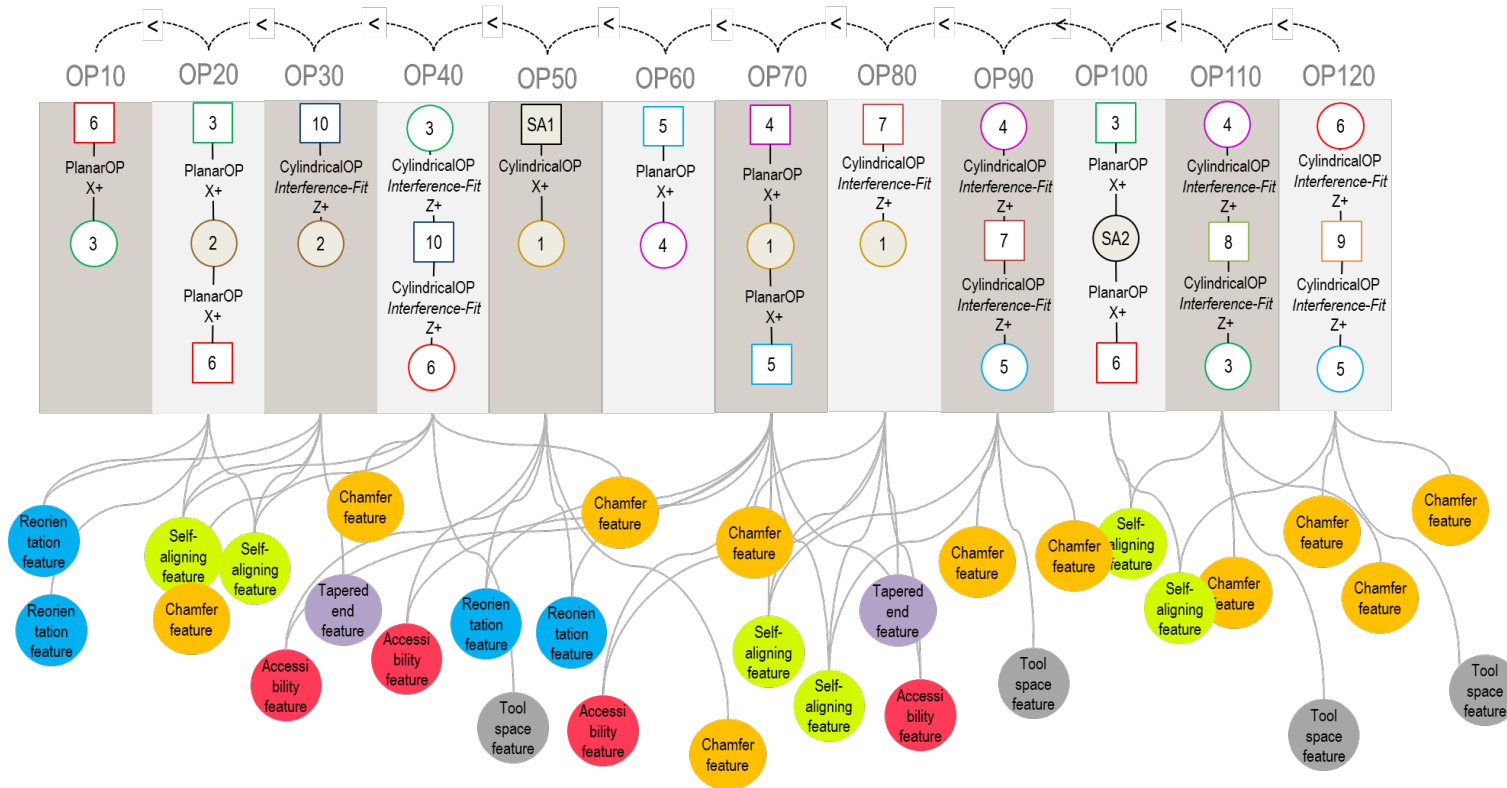


Figure 5.4 – Graphe spatio-temporel avec les connaissances allouées

Instanciation des connaissances Dans la Figure 5.5, une connaissance est suggérée au concepteur. Cette connaissance représente la fonction d'auto-alignement qui est appliquée à l'une des parties de la lame et de la bielle. Le concepteur reçoit cette connaissance et décide de l'instancier. Une fois la connaissance générée dans l'environnement de conception, ici le modèleur CAO CATIA v5, le concepteur peut décider de modifier les paramètres et les relations (longueur, diamètre, etc.) sur les entités géométriques concernées.

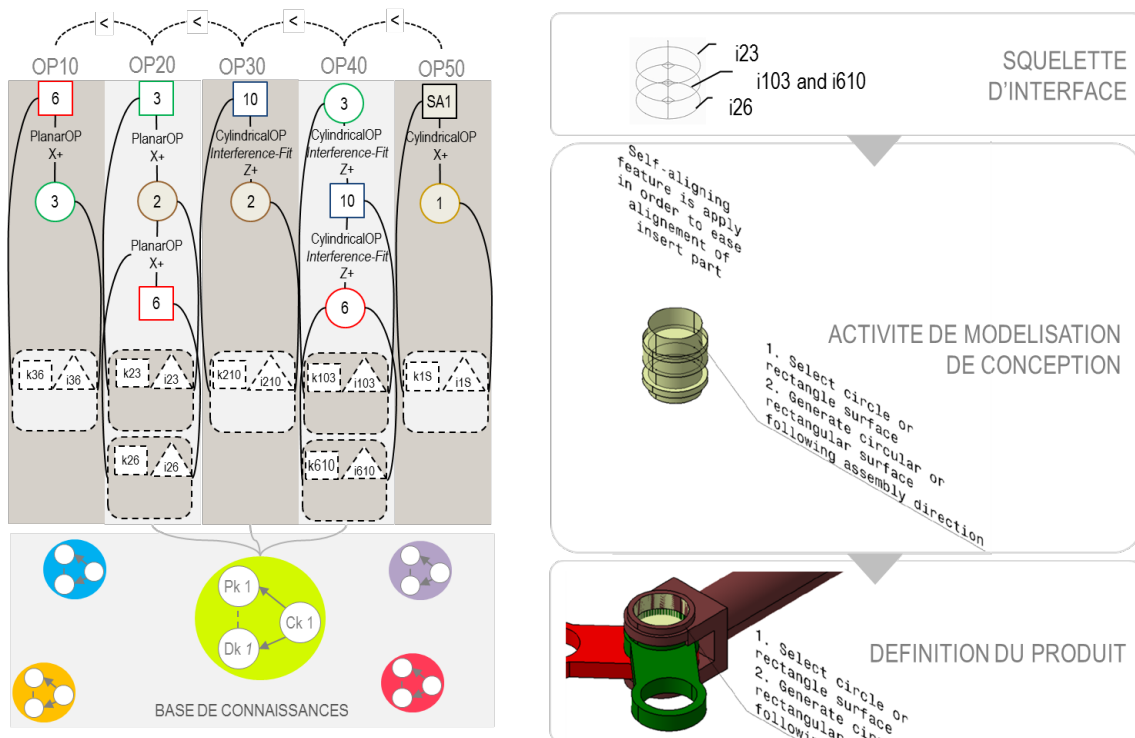


Figure 5.5 – Exemple de l'instanciation d'une connaissance représentant la fonction d'auto-alignement

Les connaissancesinstanciées sont identifiées en cours de conception de produit. Une fois les connaissances utilisées, nous pouvons, par exemple, voir les modifications opérées sur la pince (Figure 5.6). Nous remarquons que celle-ci est influencée par les connaissancesinstanciées selon les sous-assemblages et le sens d'insertion des composants.

Comparatif des définitions de produit La comparaison entre toutes les versions de la pince de désincarcération, nous permet de comparer l'instanciation des connaissances, pour sensibiliser le concepteur, en prenant conscience du graphe spatio-temporel (Figure 5.5). La Figure 5.7 détaille le nombre de connaissancesinstanciées dans chaque situation de conception.

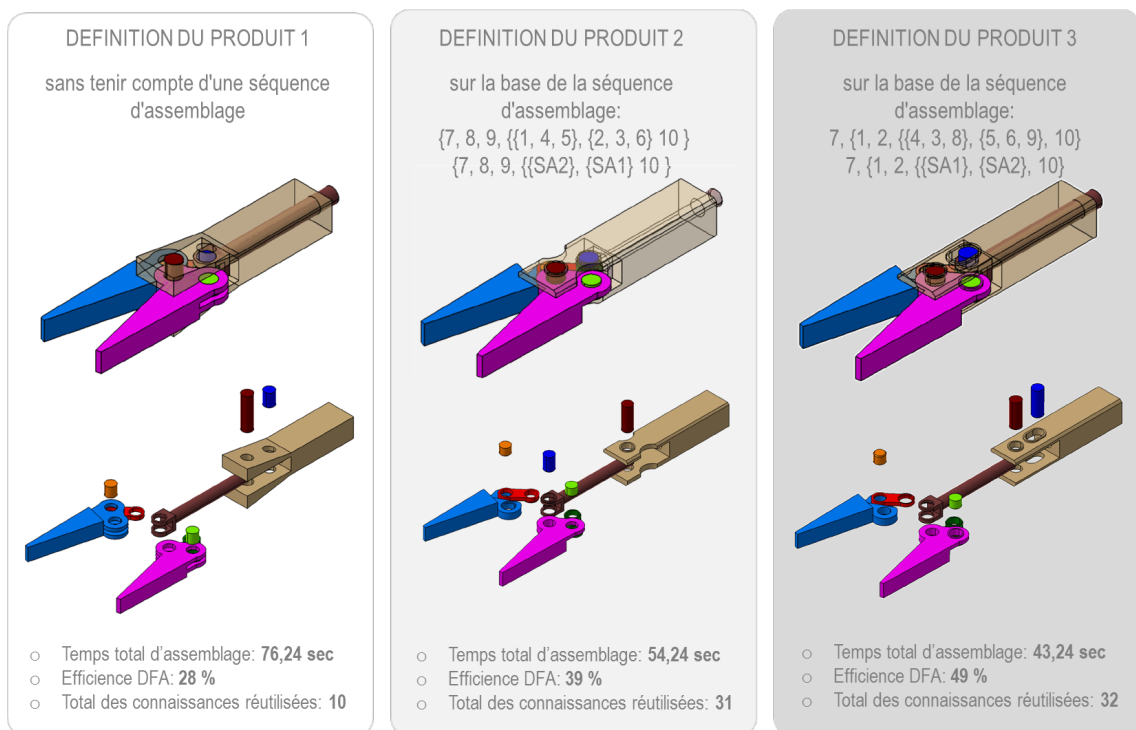


Figure 5.6 – Comparatif des différentes itérations de la pince de désincarcération

Pour soutenir la comparaison, nous avons utilisé des indicateurs dérivés du DFA, en analysant l'efficacité de l'assemblage. L'analyse DFMA [Boothroyd *et al.*, 2011] aide les concepteurs, à évaluer les erreurs d'assemblage et de fabrication, sur la géométrie du produit. Elle consiste à rechercher via une ligne directrice, toutes les contraintes du process en question. Nous utilisons l'analyse DFMA, pour optimiser les critères cités avant. Les itérations de pince sont évaluées, en utilisant un indice qualifié d'efficacité de conception (Design efficiency), nommée E (Equation 5.1) :

$$E = \frac{N_{min} \times t_a}{T_{total}} \quad (5.1)$$

Où :

- N_{min} : Nombre minimum théorique de composants ;
- t_a : Temps basique d'assemblage, c'est le temps moyen pour assembler une pièce qui ne présente aucune difficulté de manipulation, d'insertion ou de fixation (environ 3 sec.) ;
- T_{total} : Temps estimé pour compléter l'assemblage du produit, représenté par :
Le nombre d'instance de composants dans le produit × (Temps manuel de manipulation + Temps manuel d'insertion).

Selon l'analyse DFMA, si l'efficacité d'un produit est supérieure à 60%, ce dernier sera considéré comme un assemblage efficace ($E > 60\%$). Ainsi, le comparatif met en évidence plusieurs solutions différentes en termes de géométrie et de temps d'assemblage. Par exemple, nous pouvons observer que l'insertion de l'axe de la tige (composant no.10) et de l'axe central (composant no.7) ont permis une modification concernant l'accès de l'outil ou de la main avec le template d'accessibilité :

- Pour la définition du produit no.2, l'ordre d'assemblage a permis d'insérer le composant

- no.10 avant l'insertion de la tige du vérin (composant no.2) et après le composant no.7 ;
- Tandis que pour le produit no.3, les composants no.10 et no.7 sont insérés en même temps, réduisant par là, le temps d'assemblage et leur aisance à être directement insérés.

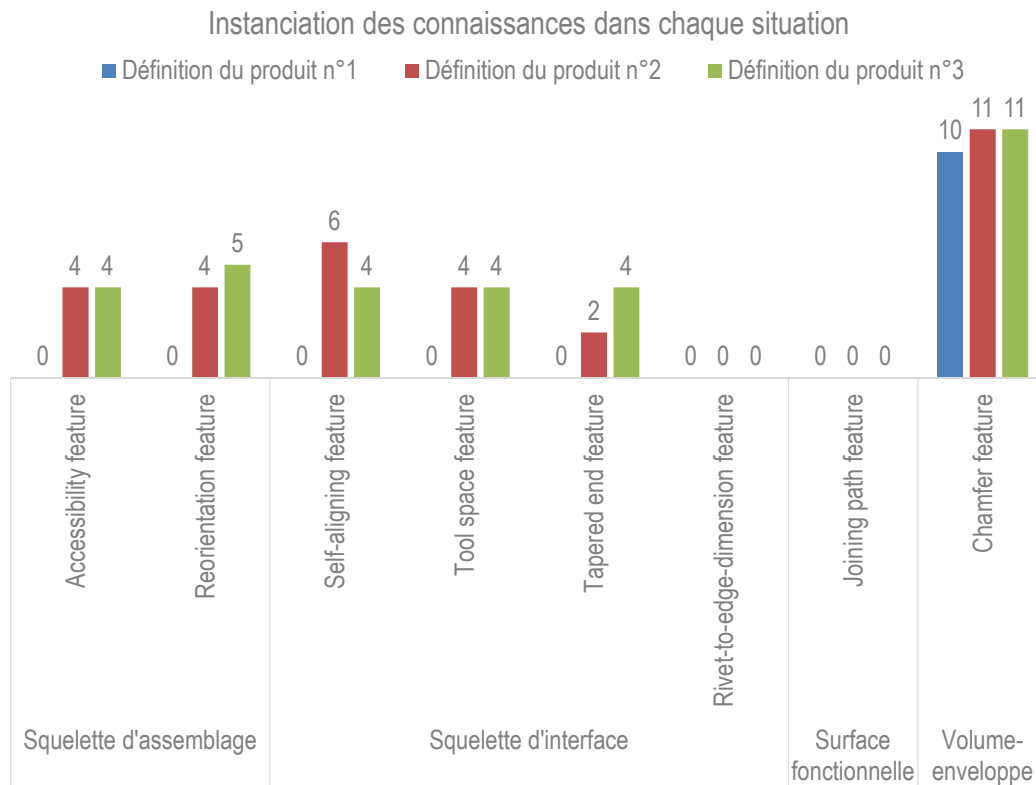


Figure 5.7 – Comparatif des connaissances instanciées en cours de conception

Cette comparaison met en avant, plusieurs points : (i) les connaissances instanciées sont plus nombreuses par rapport au produit no.1, ce qui démontre la prise de conscience du concepteur lors de l'activité de conception. (ii) Les connaissances réutilisées dans le contexte de conception augmenté ont été diffusées de manière granulaire au sein de chaque situation, à la différence du premier produit, qui se voit attribuer seulement la connaissance de la fonctionnalité de chanfrein, lorsque la géométrie émerge (situation volume-enveloppe). (iii) Cette diffusion ciblée, évite de « noyer » le concepteur parmi toutes les connaissances, en lui offrant plusieurs situations de conception, il passe ainsi moins de temps à faire le tri sur les connaissances utiles.

5.3/ DEUXIÈME CAS D'ÉTUDE : CONCEPTION D'UN DRONE MONTRE

5.3.1/ PRÉSENTATION

L'objectif de ce cas d'étude est d'appliquer l'approche MERCURY2 sur un système mécatronique de drone doté du mode *Follow Me*¹ [Marconnet *et al.*, 2017]. Inspiré du projet Nixie² le drone appelé

1. en français : suis-moi

2. <http://flynixie.com/>

DWARF (Drone WAtch Recorder Follower), a comme particularités : d'enregistrer, d'assister et de suivre son propriétaire tout en étant attaché à son poignet telle une montre connectée. Il existe des produits qui sont capables de suivre et de filmer leur propriétaire (appelé « *Follow Me Drones* » ou « *Flying Selfie Bots* » [Schneider, 2015]). Ce type de drone permet à l'utilisateur d'enregistrer des photos/vidéos, tout en contrôlant manuellement ou automatiquement le drone, pour les « *activités extrêmes* » (sport, sécurité, défense, etc.). Une autre utilisation consiste à aider les utilisateurs à créer différents points de vue en photographie (par exemple des Selfies³) de manière autonome et à distance. Les principales fonctionnalités du drone-montre sont les suivantes (Figure 5.8) :

- Voler pour filmer et photographier l'utilisateur ;
- Transporter le drone comme une montre lorsqu'il n'est pas utilisé ;
- Utilisez la technologie « *Follow Me* » pour filmer indépendamment l'utilisateur et le suivre pendant ses activités.

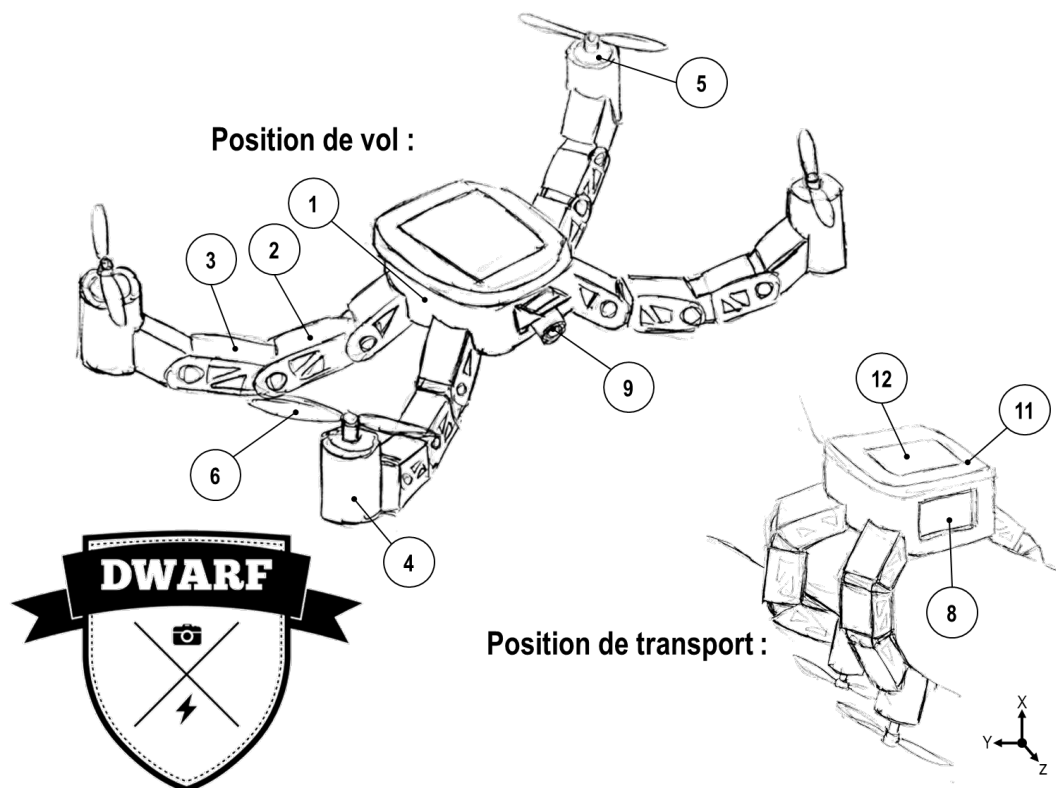


Figure 5.8 – Concept du prototype de drone-montre

La conception du produit commence par l'étude des besoins du client, de la rédaction des spécifications fonctionnelles, de la poursuite par la recherche de solutions et de sa modélisation en vue de son prototypage. Le prototype visé doit pouvoir suivre et filmer l'utilisateur, être porté comme une montre connectée tout en évitant les obstacles. Toutes ces fonctions doivent être remplies avec une autonomie suffisante (minimum 10 minutes). Le concept présenté à la (Figure 5.8) montre le corps principal séparé en 4 branches qui se déploient en position de vol. La liste des composants est définie dans le Tableau 5.2.

3. Autoportraits en français

Tableau 5.2 – Liste des composants pour le cas d'étude du drone-montre DWARF

No.	Nom
1	Chassis
2	Link1 (Lien1)
3	Link2 (Lien2)
4	Motor Mount (Maintien du moteur)
5	Motor (Moteur)
6	Helice
7	Control Board (Carte de contrôle)
8	Battery
9	Camera
10	GPS
11	Shell (Coque)
12	LCD Screen (Ecran LCD)
13	LED Ring (Anneau de lumière)

En anticipant, l'évolution future de ces types de concept, nous décidons de concevoir le produit en suivant l'approche MERCURY2.

5.3.2/ DESCRIPTION

Définition de la nomenclature initiale dans la plateforme ACSP En raison de la symétrie du produit, l'eBOM a été réalisé pour seulement un quart du produit. La nomenclature établie (Figure 5.9) est renseignée par l'architecte produit dans la plateforme ACSP (étape 1 de la Figure 4.4). La nomenclature est créée afin de générer plusieurs solutions d'assemblage dans l'étape suivante.

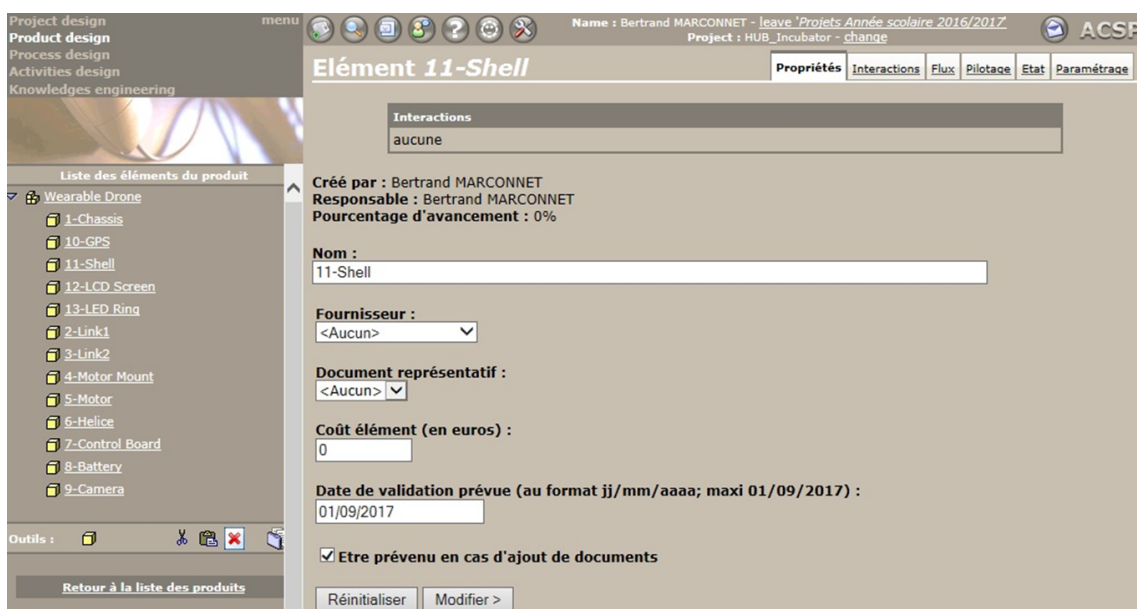


Figure 5.9 – Définition de la eBOM, dans le PDM de la plateforme ACSP

Définition des séquences d'assemblage avec l'application PEGASUS et NOTIXIA Sur la base du fichier généré (fichier .xml) depuis la plateforme ACSP, le graphe orienté est défini par l'architecte produit. Le gammiste établit les relations de précédence en détaillant la direction d'assemblage des composants. Une liste des séquences d'assemblage possibles est générée (Figure 5.10), pour permettre au gammiste de sélectionner l'une d'entre elles.

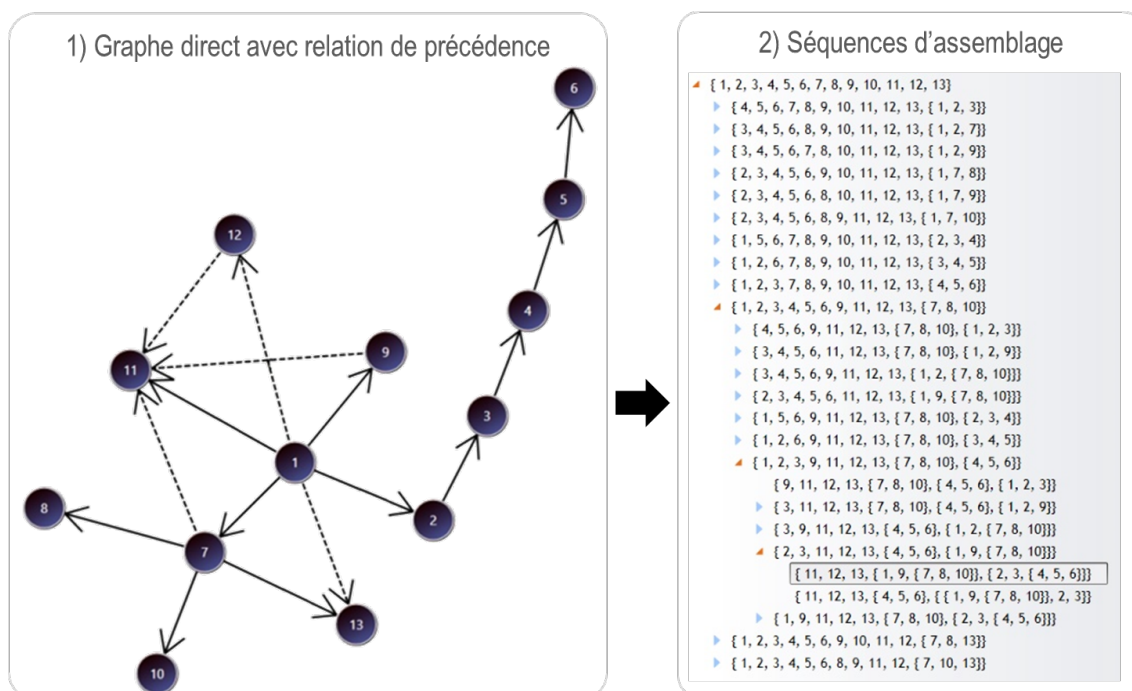


Figure 5.10 – Génération des séquences admissibles d'assemblage par l'application PEGASUS

La séquence d'assemblage est sélectionnée selon le contexte et l'objectif de fabrication (temps d'assemblage, Taux de Rendement Synthétique⁴, etc.). Dans notre cas, le TRS de base sélectionné est de 85%. Il n'est pas obligatoire d'en choisir mais il sert à affiner la sélection des séquences. Ainsi, la séquence sélectionnée est la suivante :

$$\{11, 12, 13, \{1, 9, \{7, 8, 10\}\}, \{2, 3, \{4, 5, 6\}\}\}$$

La séquence en question a été simplifiée, car la pièce 1 du châssis, a été placée dans une position différente, afin de favoriser l'insertion du cordon élastique dans les bras du drone. Le sous-assemblage SA1 {2, 3, {4, 5, 6}} a été simplifié en tant que {2, 3, 4, 5, 6}.

4 sous-assemblages sont définis :

4. ou TRS, est un indicateur pour estimer le taux d'utilisation des machines

$$SA1 = \{2, 3, 4, 5, 6\};$$

$$SA2 = \{7, 8, 10\};$$

$$SA3 = \{1, SA1\};$$

$$SA4 = \{9, SA2, SA3\}.$$

La séquence d'assemblage finale retenue est alors :

$$\{11, 12, 13, \{9, \{7, 8, 10\}, \{1\{2, 3, 4, 5, 6\}\}\}\}$$

Le résultat de la séquence est envoyé à l'ingénieur process. La séquence en question est exportée dans le système NOTIXIA (étape 2 de la Figure 4.4), afin d'établir les opérations d'assemblage (BOP) avec la mBOM (Figure 5.11).

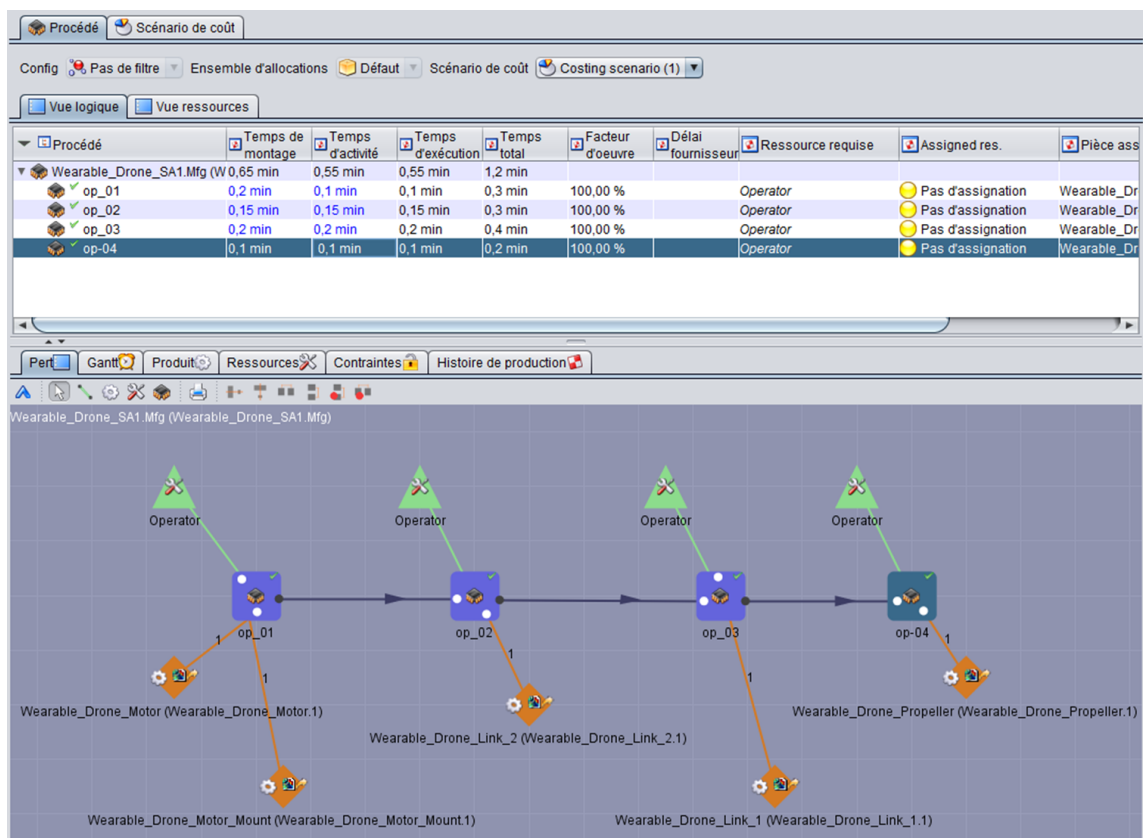


Figure 5.11 – Définition des opérations d'assemblage dans NOTIXIA

La définition des opérations d'assemblage va permettre d'apporter des contraintes sur l'insertion et la fixation des composants. L'ingénieur process établit ainsi, la planification de la séquence d'assemblage.

Contextualisation des informations du couple produit-process dans PEGASUS CAD Assis-

tant Les informations du couple produit-process sont renvoyées dans PEGASUS (étape 3 de la Figure 4.4), effectuant ainsi la conciliation de la eBOM et de la mBOM + BOP (Figure 5.12), avec les liens de composition issus de la séquence d'assemblage, afin de superposer les informations d'assemblage et de conception, au travers du graphe spatio-temporel (Figure 5.13).

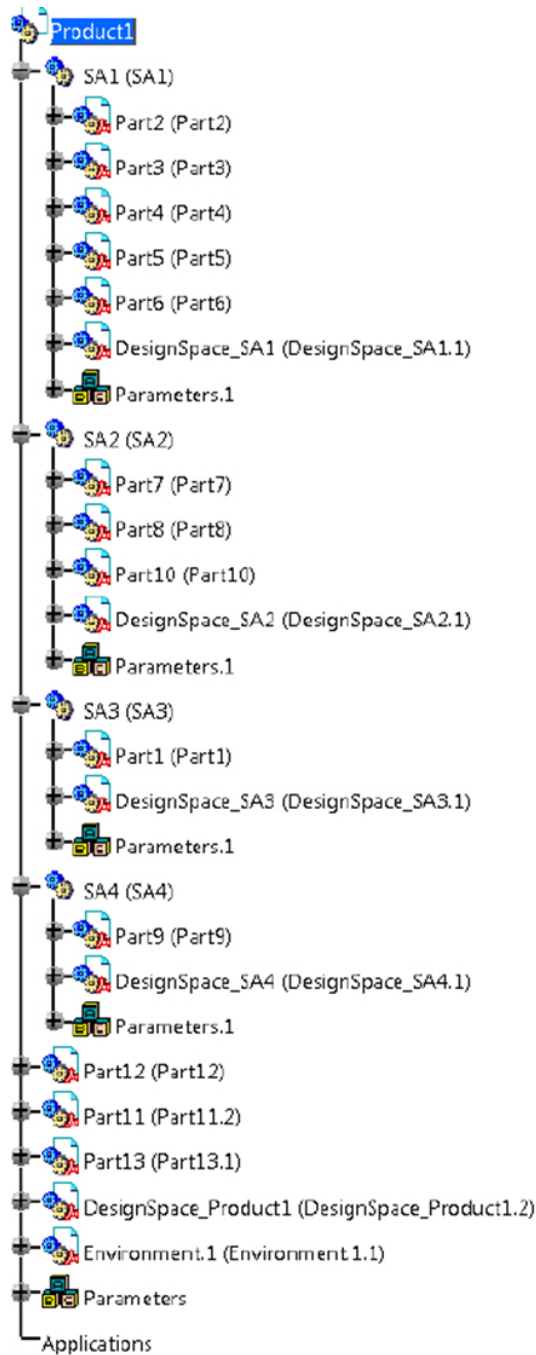


Figure 5.12 – Restructuration automatique de la eBOM dans l'outil CAO

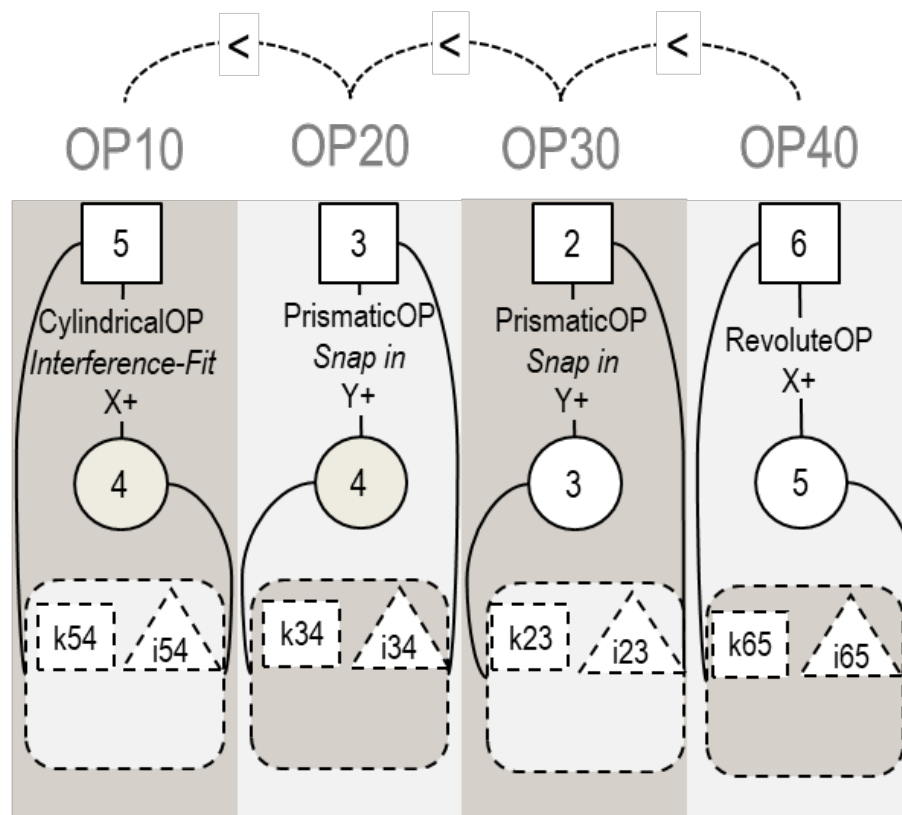


Figure 5.13 – Graphe spatio-temporel du bras du drone

Les instances sont aussi créées dans les classes de l'ontologie PRONOI3, afin de faciliter le raisonnement (Figure 5.14).

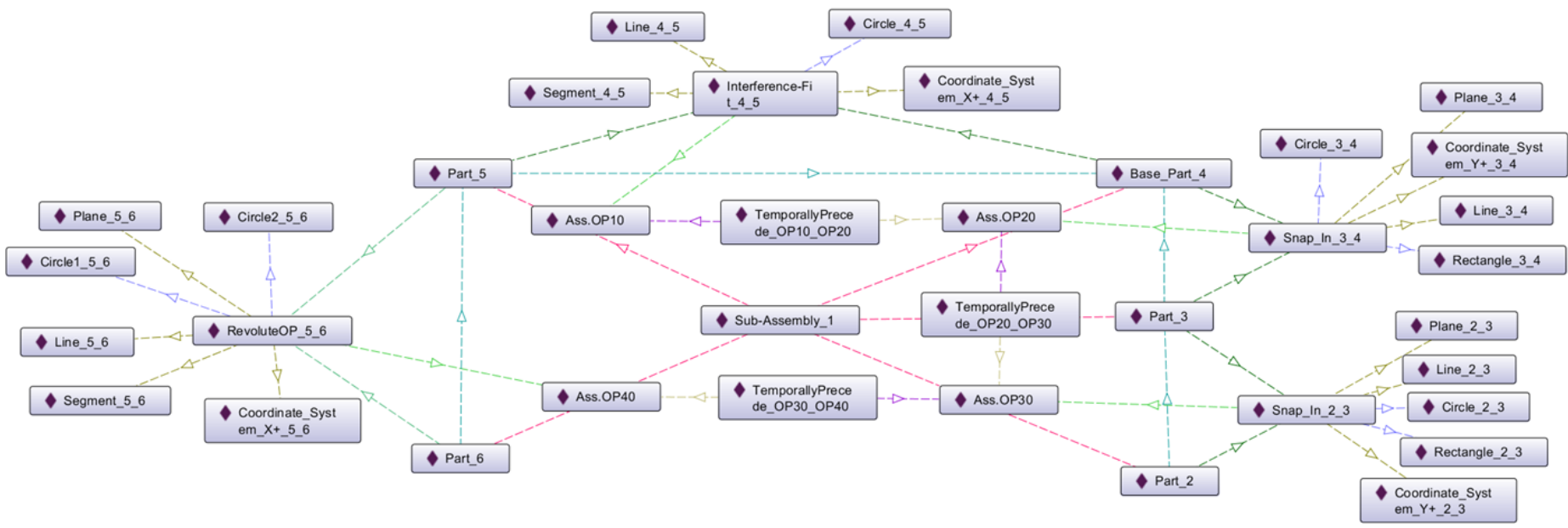


Figure 5.14 – Instances des éléments issus du « contexte augmenté » du bras du drone

Capitalisation et recherche de connaissances dans KrossRoads PEGASUS envoie la mise à jour des informations issues du graphe spatio-temporel, afin de fournir au gestionnaire de connaissances les éléments nécessaires à la capitalisation et diffusion des connaissances illustré par la Figure 5.15 (étapes 4 et 5 de la Figure 4.4).

The screenshot displays the KrossRoads interface for configuring a rule. On the left, a sidebar contains icons for Dashboard, Projects, Products, Processes, DCS, Rules, and Variables. The main area is divided into a search panel and a rule configuration panel. The search panel includes a 'New rule' button, a 'Modification date' filter, and dropdown menus for 'Product', 'Process', and 'Area'. Below these is a 'Search' button and a table of rules. The 'Tapered end feature' rule is selected and highlighted in blue. Below the table is a 'View Selected DCS' button. The rule configuration panel on the right has tabs for 'Summary', 'Edit', 'Team', and 'Versions'. The 'Summary' tab is active, showing the rule title 'Tapered end feature (V1.5)' and a description: 'Tapered end feature must be apply in order to ease alignment of insert part'. Below this is a diagram illustrating the 'Don't' and 'Do' conditions for the tapered end feature. The 'Don't' diagram shows a cross-section of a part with a tapered end, labeled with 'y' and 'p'. The 'Do' diagram shows the same part with a tapered end, labeled with 'y', 'p', '1 circle or rectangle \in Plane', and '1 line Surface'. Below the diagram is a tree view of the 'Area' hierarchy, showing 'Europe' (with sub-items 'Beaulieu', 'Stadeln', 'Torsas', 'Toledo'), 'Asia', and 'US'. At the bottom, there is a 'Formula' section and a table for 'DCS' with columns 'Status', 'Name', and 'Actions'. The table is currently empty, showing 'No records to display.'

Figure 5.15 – Synthèse des connaissances capitalisées dans KrossRoads

La Figure 5.15 représente ici la synthèse des connaissances capitalisées dans KrossRoads, afin de permettre à PEGASUS d'appliquer les unités de raisonnement RUN, pour inférer et requêter les connaissances adéquates pour le concepteur.

Raisonnement et définition des entités des espaces de conception dans CATIA V5 avec PEGASUS CAD Assistant

A ce stade (étape 6 jusqu'à l'étape 13 de la Figure 4.4), le concepteur commence la définition des espaces de conception via l'IHM PEGASUS CAD Assistant, en instanciant les fonctions d'assemblage. Pour notre exemple, la fonction d'extrémité (tapered end feature) est allouée sur le bras du drone (Figure 5.16). Les squelettes d'assemblage sont ici représentés par la première itération du concept en montrant explicitement les différentes parties placées sur les divers éléments du squelette : par exemple, les liens (Link1 et Link2) sont placés sur leurs axes et limités par un seul plan (Figure 5.17).

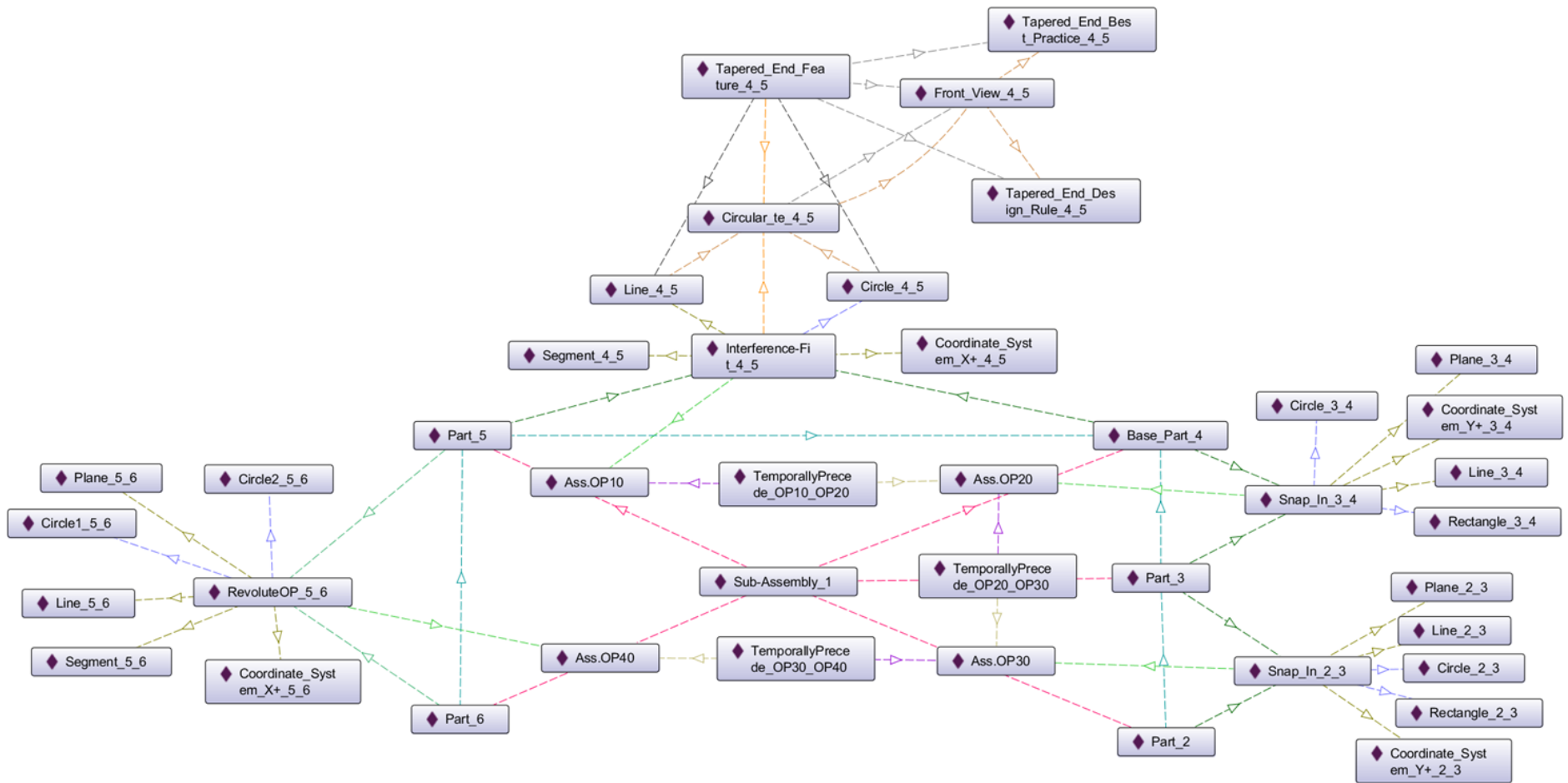


Figure 5.16 – Allocation de la fonction tapered end feature sur le bras du drone

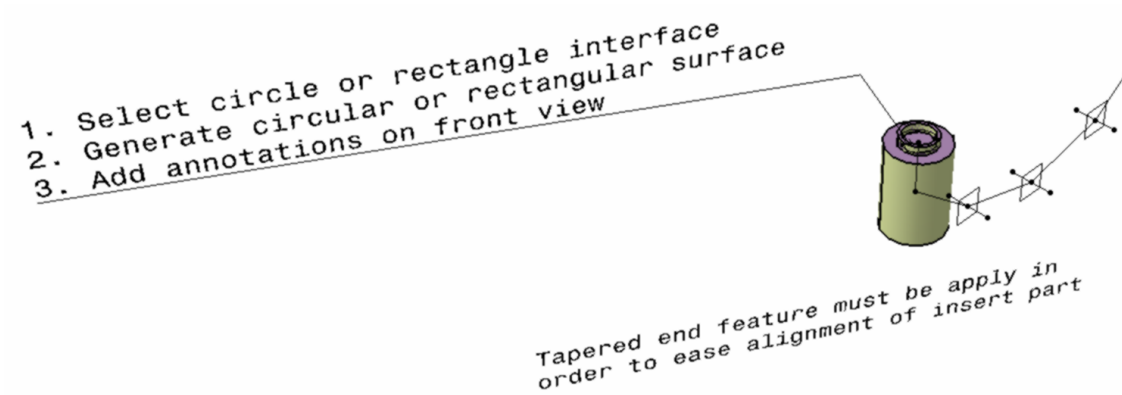


Figure 5.17 – Espace de conception du moteur et porte-moteur sur le bras du drone

Au travers de l'IHM PEGASUS CAD Assistant, les intentions du concepteur et de sa situation sont capturées, puis envoyées dans le HUB, pour rechercher les connaissances possibles au « *contexte augmenté* » de conception. La Figure 5.18, présente l'exemple de l'application de la fonction d'accessibilité et de chanfrein pour faciliter l'insertion entre les liens. Nous pouvons observer une évolution concernant la géométrie des pièces au fur et à mesure de la prise en compte des contraintes d'assemblage. La génération du modèle CAO s'effectue au fur et à mesure de l'avancée du concepteur dans chaque situation de conception, car il va en premier lieu, définir les squelettes d'assemblage (leur position dans l'environnement CAO), puis proposer les templates en lien dans l'IHM. C'est pourquoi, les espaces de conception suivants : squelettes d'interface, surfaces fonctionnelles et volume-enveloppes sont restreints en fonction de ces choix de conception et de l'instanciation des templates.

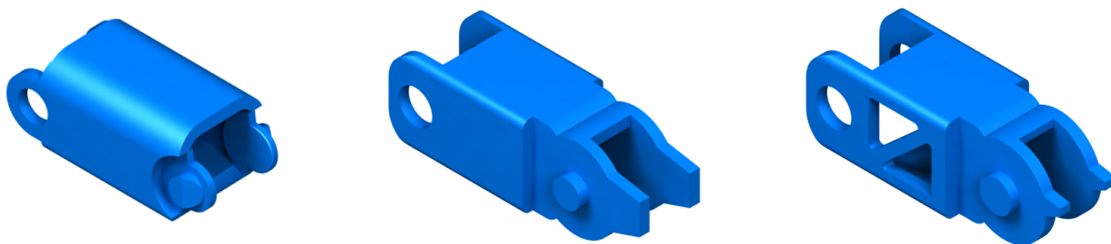


Figure 5.18 – Evolution des composants avec les connaissances

Premier bilan L'approche Top-Down design MERCURY2 appliquée dans l'environnement de type PLM, a permis de favoriser la diffusion de connaissances orientées assemblage sur le drone. La première itération du prototype de drone est illustrée dans la Figure 5.19, avec l'une des connaissances allouée qui est précisée sous forme d'annotation, directement sur la partie du « *Bras* » du produit « *Follow Me Drone* ». Concrètement, il est nécessaire de compter une demi-journée pour construire une connaissance et la règle qui y est associée, et ce, respectivement par le gestionnaire de connaissances et le spécialiste. Quant au domaine de validité des règles ainsi définies, il sera remis en cause à chaque mise à jour de la règle, ou bien, lorsqu'un nouveau besoin apparaît.

Ainsi, pour faciliter la conception détaillée de chaque composant, tout en allégeant la structure du produit, nous avons appliqué une optimisation numérique avec la méthode des éléments finis sur le drone.

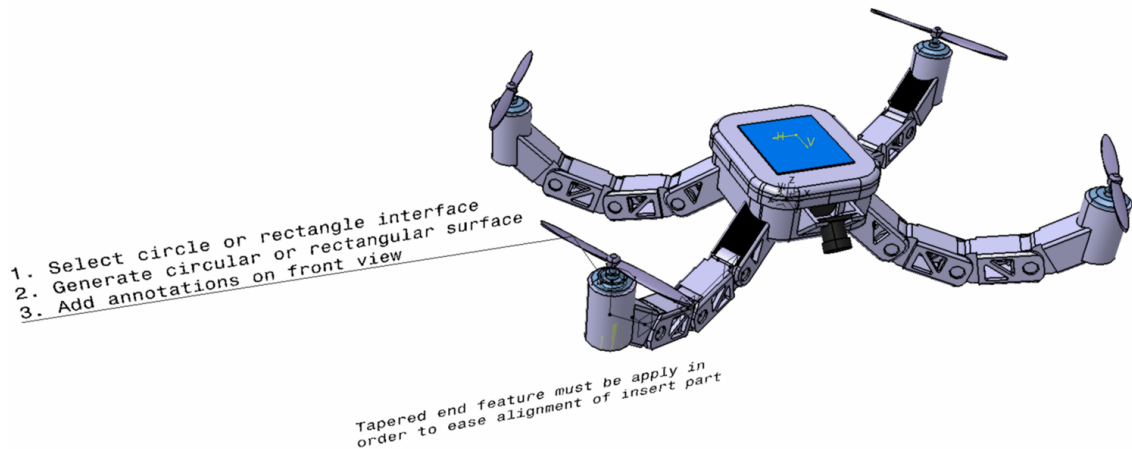


Figure 5.19 – Première définition du produit

5.3.3/ DÉMARCHE D'OPTIMISATION NUMÉRIQUE POUR AMÉLIORER L'EFFICACITÉ DE L'APPROCHE

Deux méthodes de modélisation et d'optimisation ont été appliquées dans le processus de conception du produit [Kvan, 2000], afin de trouver une formulation générique. La première méthode d'optimisation multi-objectifs consiste à trouver la meilleure solution de conception [Eggers *et al.*, 2003], tandis que la seconde est la prédiction du comportement du produit sous des charges multi-physiques spécifiques, avec l'exploitation de la méthode des éléments finis (**FEM - Finite Element Method**).

La boucle d'optimisation du produit peut être appliquée en utilisant des algorithmes d'optimisation multi-objectifs, afin de sélectionner les meilleures options disponibles, à partir d'un large éventail de choix possibles [Gomes *et al.*, 2009]. Une approche d'optimisation à objectif unique n'est pas suffisante pour faire face aux problèmes de la vie réelle. En fait, les ingénieurs sont souvent invités à résoudre des problèmes avec plusieurs fonctions objectifs contradictoires. L'optimisation multidisciplinaire et multi-objectifs consiste à trouver la conception optimale des systèmes dont l'ingénierie est complexe. L'optimisation nécessite une analyse, pour prendre en compte les interactions entre les disciplines (ou les parties du système), afin de mieux les exploiter de manière synergique.

D'un point de vue mathématique, la méthode pour un problème d'optimisation multi-objectifs se décrit comme suit (Equation 5.2, Equation 5.3, et Equation 5.4) :

$$\min[f_1(X), f_2(X), \dots, f_i(X)]; \quad (5.2)$$

$$G_j(X) \geq 0, \text{ and } G_1(X) = 0; \quad (5.3)$$

$$X = (x_1, \dots, x_{n+u}). \quad (5.4)$$

Où :

- f_1, \dots, f_i : **Fonctions objectif** pour les paramètres de réponse. Lorsque $i > 1$ et que les différentes exigences fonctionnelles sont contradictoires, il s'agit d'un problème d'optimisation multi-objectifs. Ce sont les quantités de fonctions que le concepteur souhaite maximiser ou minimiser ;
- $G_j(x_1, \dots, x_{n+u}) \geq 0$ et $G_1(x_1, \dots, x_{n+u}) = 0$: **Contraintes** d'égalité et d'inégalité sont imposées sur la conception du produit, directement liées aux exigences. Ces contraintes correspondent aux limites que le concepteur doit respecter en raison de la nécessité de se conformer aux normes, ou en raison des fonctions particulières de l'environnement, des fonctionnalités, des limitations physiques, etc. ;
- $X = (x_1, \dots, x_{n+u})$: **Vecteurs de variable** intégrant généralement les " n " exigences des clients et des " u " paramètres techniques de conception du produit. Ces variables sont considérées comme des données d'entrée du problème d'optimisation.

Nous pouvons à partir des espaces de conception, exploiter des connaissances plus détaillées sur les volumes-enveloppe, en choisissant la meilleure solution technique basée sur les contraintes et les objectifs de comportement.

5.3.4/ DESCRIPTION DU PROBLÈME D'OPTIMISATION

Exigences du drone-montre Les exigences sont utilisées comme des valeurs constantes avec des bornes minimum/maximum, lors des simulations, elles sont indiquées dans le Tableau 5.3. L'un des éléments importants concerne le temps de vol du drone, estimé à 10 minutes, celui-ci entraîne des modifications ultérieures, lorsque la batterie, les moteurs ou le poids sont impactés par l'un de ces paramètres.

Tableau 5.3 – Exigences du drone-montre DWARF

Exigence	Critère
Temps minimum de vol	10 minutes
Système électronique	Kit Crazyflie 2.0
Méthode de fabrication	Fabrication additive
Coût maximum	600 euros

La Figure 5.20 présente un diagramme simplifié des relations entre les principaux composants et les paramètres du temps de vol. Les moteurs, les hélices et la batterie sont des pièces modifiables en fonction de la performance et du temps de vol souhaités.

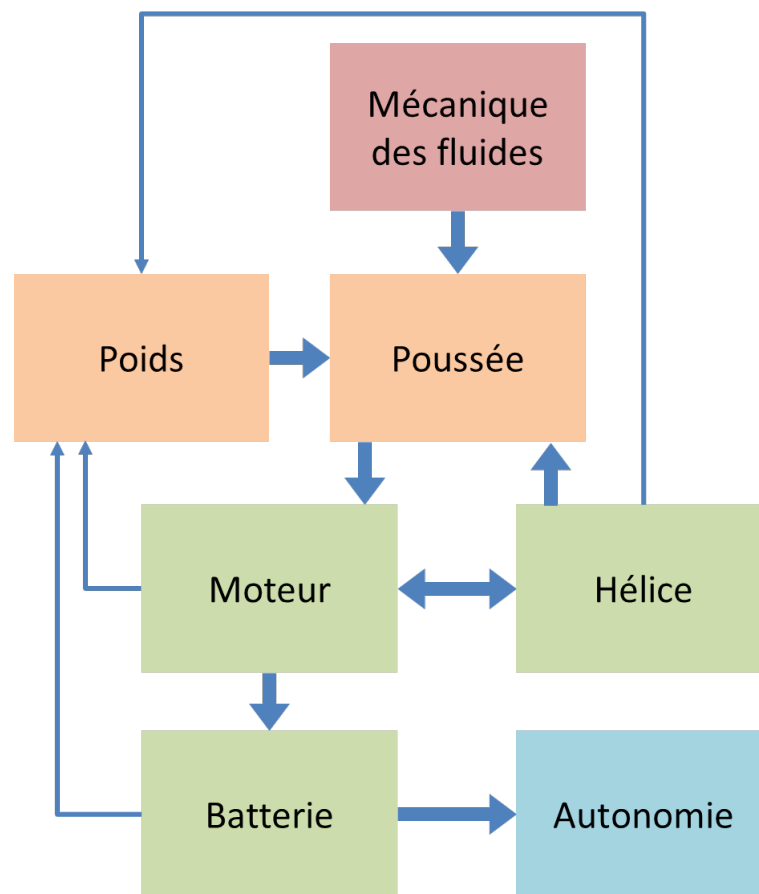


Figure 5.20 – Diagramme simplifié des relations entre les principaux composants avec les paramètres pour l'optimisation du temps de vol

Paramètres techniques Les paramètres disponibles pour la simulation sont présentés dans le Tableau 5.4 et le Tableau 5.5, notamment, les dimensions en question varient selon les limites présentées. Par exemple, pour la fabrication des composants, la plupart sont simulées et réalisées en fabrication additive avec la résine époxy. Les composants du drone nécessitent des formes géométriques précises pour l'assemblage et donc, lors de l'impression 3D, nous favorisons la résine époxy par rapport à l'ABS. Nous exploitons l'algorithme de recuit simulé (basé sur l'analogie de la solidification de fluide), car nous devons obtenir une solution rapide, en indiquant un faible nombre de mises à jour consécutives, car en exécutant de deux manières successives cet algorithme de recherche « *stochastique global* », il peut donner des résultats différents. Ainsi, l'application de cet algorithme de recherche globale va fournir après un laps de temps nécessaire, des recherches locales pour être généralement utilisé dans le cadre de fonctions non linéaires (objectifs et contraintes).

Tableau 5.4 – Exemple des paramètres techniques pour les variables continues

Variable continue	Critère
Épaisseur	$E \in [2mm; 4mm]$
Poids	$m \in [50g; 150g]$
Liens de largeur et de hauteur	$L \in [12mm; 16mm]$
Distance entre les axes	$L = 22,5mm$
Diamètre du moteur	8mm
Support moteur Ext. diamètre	$Dm \in [10mm; 16mm]$
Diamètre des broches	$D \in [4mm; 7mm]$

Tableau 5.5 – Exemple des paramètres techniques pour les variables discrètes

Variable discrète	Coût [e/g]	ρ [kg/m ³]	Re [MPa]
ABS	0.3	1180	50
Résine époxy	0.0036	1010	32

Fonctions objectifs et contraintes Les relations qui relient les paramètres aux résultats requis, comme les contraintes, sont présentées dans le Tableau 5.6 et le Tableau 5.7. Pour compléter les exigences existantes, des paramètres techniques ont été ajoutés en définissant les variables continues et discrètes. Par exemple, la description du temps de vol permet de trouver la meilleure solution, pour le choix de la capacité de la batterie.

Tableau 5.6 – Exemple des fonctions objectifs (la description des variables est détaillée dans la liste des abréviations et des symboles au début du document)

Fonction objectif	Critère
Temps de vol (à maximiser)	$\Delta t = \frac{4.419\alpha CT_b \sqrt{2\rho S}}{10^3 (Mg)^{\frac{3}{2}}}$
Poids global (à minimiser)	poids de départ : 114g
Capacité de batterie (à maximiser)	point de départ : 240mAh

Tableau 5.7 – Exemple des contraintes

Contrainte	Critère
Courant maximum autorisé	défini par le moteur
Contrainte d'épaisseur maximale	max. 4mm

Règles Les règles utilisées pour évaluer les résultats sont détaillées dans le Tableau 5.8. Par exemple, la première règle, concernant la vérification de la poussée générée par les hélices, doit être supérieure à la poussée requise par le drone. La deuxième règle vérifie la puissance générée par l'hélice comme étant plus grande que la puissance requise par le drone.

Tableau 5.8 – Exemple de règles, pour vérifier les critères d'optimisation (la description des variables est détaillée dans la liste des abréviations et des symboles au début du document)

Règle	Critère
Poussée générée par les hélices	$T_p = 28.35E^{-10}D^3\omega^2h$
Puissance fournie par les hélices	$W_p = 5.33E^{-5}D^4\omega^3h$
Puissance nécessaire à un vol stationnaire	$W_{st} = \frac{7.127(Mg)^{\frac{3}{2}}}{\sqrt{2\rho S}}$

Comparatif Il convient que tous les éléments soient renseignés dans le module de CATIA : Product Engineering Optimizer, à savoir les exigences, les paramètres techniques, les fonctions objectifs et les contraintes avec les règles nécessaires. Un comparatif a permis de prendre en compte tous les résultats issus des simulations, comme par exemple, le poids du drone avec les moteurs ayant une poussée supérieure au poids total. Les résultats des simulations ont estimé plusieurs batteries avec leurs prix respectifs selon trois paramètres : le temps de vol, le poids du drone et la capacité de la batterie. Les résultats générés sont présentés dans la Figure 5.21.

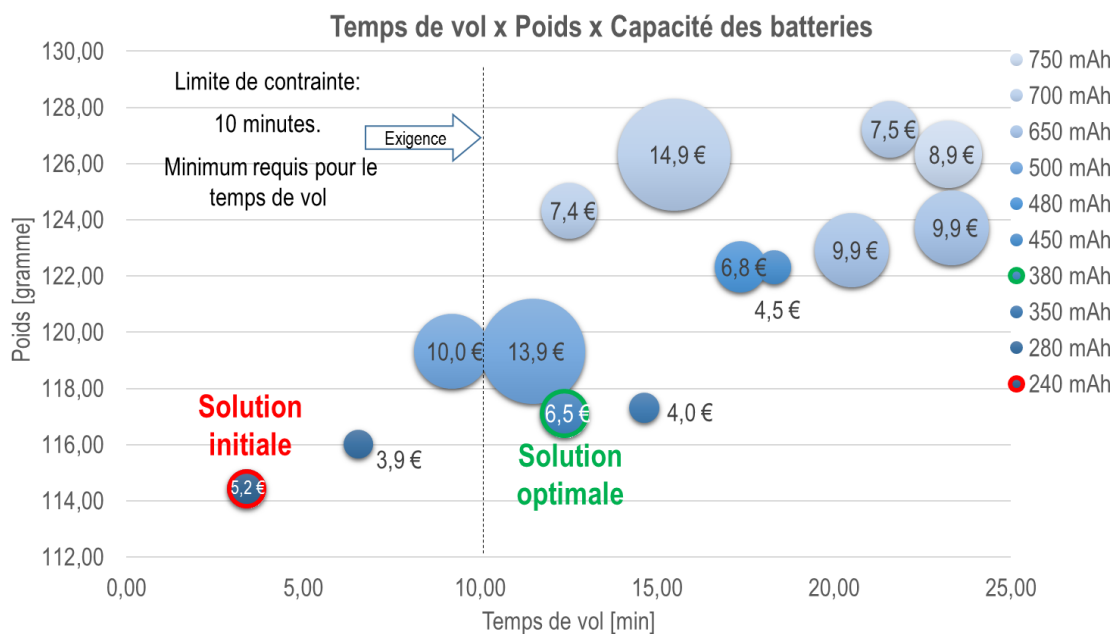


Figure 5.21 – Résultats d'optimisation entre le temps de vol, le poids du drone, le prix et la capacité de la batterie

Dans ce comparatif, nous observons que la capacité de la batterie fait varier les temps de vol. Ce facteur est considéré dans l'équation en complément du poids et du temps de vol, en sélectionnant la solution optimale. Le premier critère de choix s'est porté sur une batterie ayant une capacité de 380 mAh, car l'exigence correspond au temps minimum d'autonomie de vol (min. 10 minutes), tandis que le second critère est le prix de la batterie.

Méthode des éléments finis Une fois la solution retenue, la structure mécanique doit être conçue pour résister aux chutes et aux chocs mécaniques. Dans notre cas, un des cas critiques étudiés

correspond à une chute de 2 mètres jusqu'au sol, illustré à la Figure 5.22.

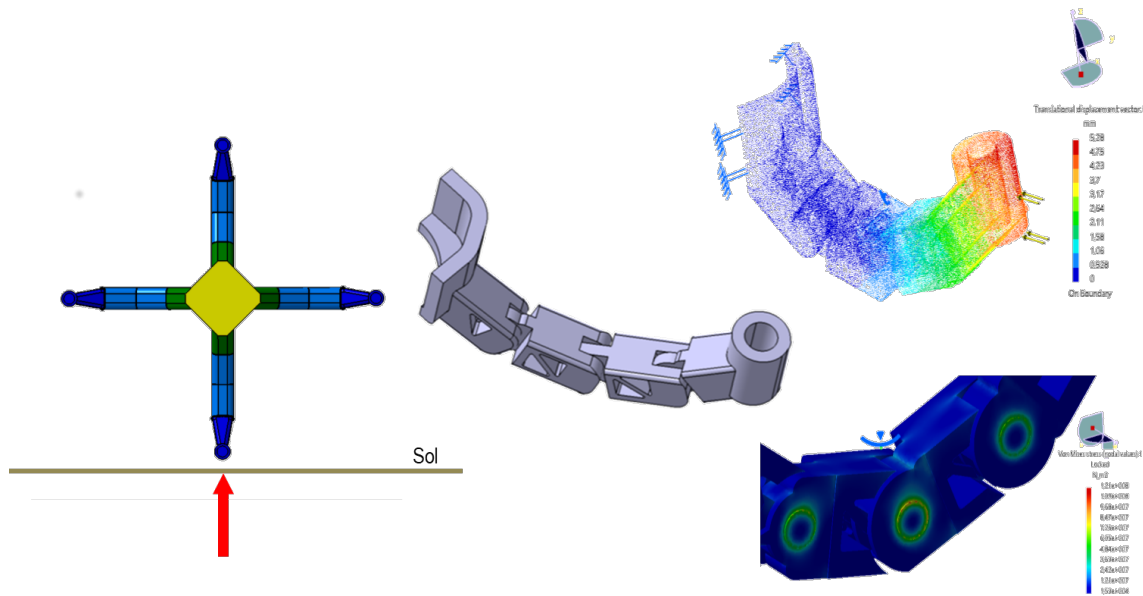


Figure 5.22 – Optimisation basée sur un cas critique - chute de deux mètres absorbée par un bras

La force équivalente à cette chute a été calculée compte tenu du fait que l'énergie cinétique correspond à une déformation acceptable égale à 0,5 mm. En utilisant comme référence l'axe des bras en x , il en résulte un choc d'une force importante. Pour simuler ce problème, une partie du bras a été considéré avec les volumes-enveloppe du drone créé précédemment, en tenant compte des deux symétries. Les paramètres décrits dans le Tableau 5.9 ont été créés et attribués à leurs dimensions respectives dans la géométrie. De cette façon, chaque dimension peut varier librement entre ses limites.

Tableau 5.9 – Les paramètres et leurs limites utilisés lors de l'optimisation dans CATIA

Paramètre	Minimum (Configuration A)	Maximum (Configuration B)
LK_Epaisseur_Laterale (mm)	2	4
LK_Epaisseur_Verticale (mm)	2	4
MM_Diamètre_Externe (mm)	10	16
MM_Epaisseur_Laterale (mm)	2	4
MM_Epaisseur_Verticale (mm)	2	4
CH_Epaisseur (mm)	2	4
Taille_Liens (mm)	12	16
Diametre_Broche (mm)	4	6

Le déplacement maximal du bras est évalué à 10,8 mm, ce qui représente 10% de la longueur totale du bras, soit à peu près 100 mm. Les contraintes moyennes atteignent 60% de la limite du matériau. Quelques points présentent des contraintes élevées, au-dessus de la résistance de la traction du matériau. Une configuration avec les dimensions minimales de chaque paramètre (configuration A) est supérieure à la limite de résistance mécanique, selon les exigences du produit. La configuration B présente un déplacement maximal de 5,326 mm, soit la moitié du déplacement de la configuration A. Les contraintes sont également plus faibles que dans la configuration A, mais elles sont encore dans la limite de résistance à la traction du matériau. Les régions du drone qui présentent des contraintes plus élevées peuvent également être générées selon des mauvaises conditions dans le maillage dans ces régions. Un chanfrein peut aider à diminuer les contraintes maximales, par exemple la même configuration de simulation a été effectuée avec un matériau différent : l'ABS (configuration D), toutefois, il ne présente pas de différences considérables. Ainsi, les résultats obtenus sont représentés dans le Tableau 5.10.

Tableau 5.10 – Résultat des cinq simulations dans les différentes configurations

Paramètre	Config. A	Config. B	Config. C	Config. D	Config. E
Matériaux	Résine époxy	Résine époxy	Résine époxy	ABS	ABS
Max. Von Mises (N/mm^2)	$1.987E + 8$	$1.243E + 8$	$1.303E + 8$	$1.987E + 8$	$1.243E + 8$
Max. déplacement (mm)	11.867	5.326	5.543	17.800	7.989
Volume (mesuré) (m^3)	$8.682E - 6$	$1.980E - 5$	$1.82E - 05$	$8.289E - 6$	$1.982E - 5$
Poids total (structure du drone) (g)	40.98	93.46	85.90	33.49	79.99
Coût total (€)	12.29	28.04	25.77	0.12	0.28

La première simulation a été effectuée en tenant compte de la résine époxy. Plusieurs paramètres des cinq configurations ont été simulés en détail, notamment, les configurations A, B et C qui ont le même matériau dans leurs compositions (résine époxy). La configuration A présente les paramètres les plus bas tandis que la configuration B présente les paramètres les plus élevés. Les configurations D et E sont respectivement égales à A et B, mais elles sont constituées d'ABS. La configuration C comporte des paramètres avec des valeurs intermédiaires. L'ABS présente un déplacement maximal plus élevé par rapport à la résine époxy. Les pièces ABS sont plus légères que les autres, mais la plus grande différence réside dans le coût du produit. La configuration la plus lourde avec un poids de 79,99 grammes en ABS, coûte 0,28 € (concernant la structure du drone complète), alors que la partie la plus légère en résine époxy pèse 40,98 grammes et ne coûte que 12,29 €. Les configurations A, B et C sont représentées sur la Figure 5.23, dont il est possible de voir la différence dans la largeur et l'épaisseur des pièces. Ainsi, la configuration A est celle qui a été sélectionnée pour travailler le détail des composants.

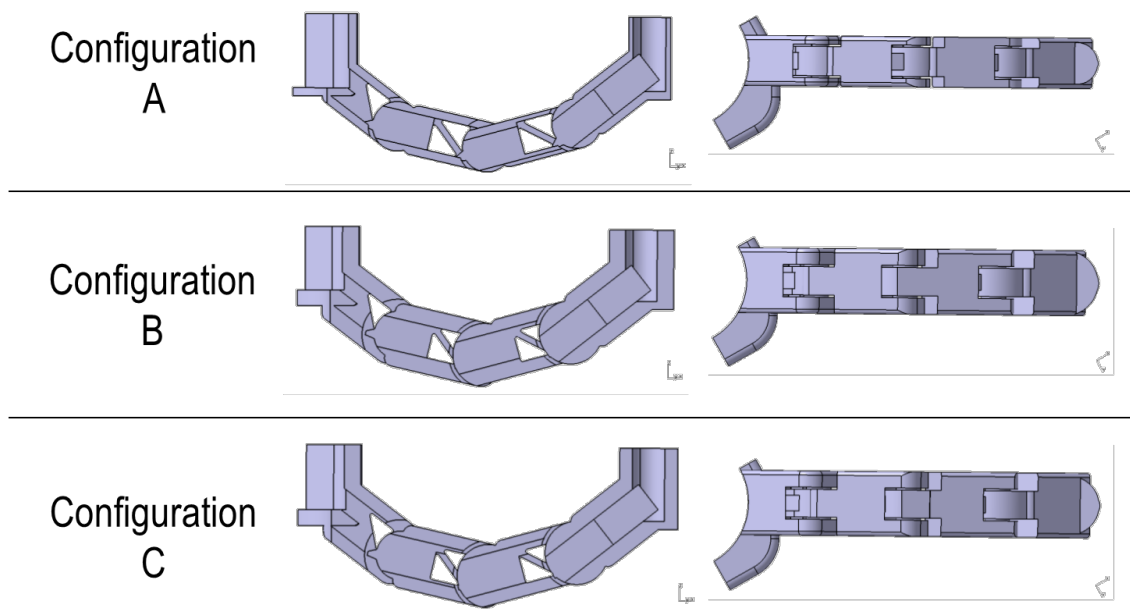


Figure 5.23 – Visualisation des configurations A, B et C

Conclusion L'approche MERCURY2 combinée aux méthodes d'optimisation, permettent de voir le gain potentiel de la méthode dans le cas d'étude avec PEGASUS. Si le processus est assez long pour générer les espaces de conception, nous avons vu que l'instanciation des connaissances orientée assemblage, a permis par exemple, de faire évoluer la géométrie des composants à la fois sur l'optimisation des épaisseurs, mais aussi sur leurs impacts concernant le poids. PEGASUS a facilité la diffusion des connaissances en lien, avec l'interface PEGASUS CAD Assistant, afin de détailler les composants. Sur la base de la CAO réalisée, nous avons pu élaborer un prototype du drone en six mois pour vérifier la qualité des résultats (Figure 5.24).

L'analyse du temps de vol du drone a été étudié en détail, ce qui a permis, à partir de la mécanique des fluides, d'atteindre la puissance électrique nécessaire pour établir des relations fiables entre les caractéristiques du drone. Toutes les relations ont été développées, afin d'estimer l'autonomie du drone. Nous avons pu démontrer que les relations présentent des résultats raisonnables, surtout en considérant la simplicité des relations proposées. Une analyse mécanique a été effectuée en évaluant le comportement d'un bracelet du drone lorsqu'une force équivalente à une chute de 2 mètres est appliquée. Le modèle utilisé est dans la limite de la rupture. Une évolution à la fois du HUB PEGASUS sur des liens à automatiser entre la génération automatique de fichier de relations .xml et du parseur sont nécessaires. Quant au prototype du drone, il est nécessaire de mettre en œuvre une amélioration du poids, pour alléger sa structure, il a depuis été développé d'autres itérations du prototype qui ont permis d'étoffer les informations de base, pour vérifier les calculs et les simulations effectués dans ce cas d'étude.

5.4/ CONCLUSION

Ce chapitre a mis en avant l'application du cadre méthodologique DAMON avec l'approche MERCURY2 sur les deux cas d'étude dans deux domaines différents (mécanique et mécatronique). Les



Figure 5.24 – Prototype du drone-montre DWARF

expérimentations ont permis de mettre en avant l'utilité du cadre DAMON pour la réutilisation efficace des connaissances sur la pince de désincarcération, avec un comparatif des connaissances réutilisées pour chaque itération, démontrant une diffusion plus pertinente dans la conception. En outre, l'utilisation du démonstrateur informatique PEGASUS, a mis en avant la centralisation des informations du « *contexte augmenté* », à la fois pour diffuser les connaissances orientées assemblage, mais aussi pour aller plus loin dans la conception, en détaillant chaque composant à l'aide de l'optimisation numérique et de la méthode des éléments finis.

6

CONCLUSION GÉNÉRALE ET PERSPECTIVES

« Là où se trouve une volonté, il existe un chemin. »

Winston Churchill

Sommaire

6.1 Conclusion générale	146
6.2 Perspectives de recherche	147
6.2.1 Perspectives à court terme	147
6.2.2 Perspectives à moyen terme	148
6.2.3 Perspectives à long terme	148

L'objectif de ce chapitre est de synthétiser notre contribution scientifique en introduisant les futurs travaux de recherche que nous envisageons. La première sous-section concerne la conclusion générale, tandis que la deuxième sous-section définit les enjeux de ces travaux pour proposer des pistes de réflexions et élargir le domaine du « *contexte augmenté* ».

6.1/ CONCLUSION GÉNÉRALE

Au travers des travaux présentés, nous avons vu l'intérêt que porte la communauté scientifique sur la réutilisation des connaissances avec les approches de génération de prédiction dans un contexte de conception proactive. Cela a permis l'émergence de notre problématique de recherche visant à décrire les associations de prédiction-connaissance afin d'améliorer le processus actuel de réutilisation des connaissances dans la conception de produits. Une proposition de décomposition des connaissances a été définie ainsi que leurs interrelations avec le contexte et les prédictions.

L'état actuel des travaux de la littérature scientifique sur les défis dans la gestion des connaissances, de la prédiction et du contexte de conception ont été abordés dans ces travaux de thèse. Grâce à ces fondations, un cadre méthodologique DAMON dédié à la conception orientée assemblage a été proposé. Celui-ci applique les règles du DFA, afin de faciliter le processus de décision dans le choix des instanciations des connaissances (représentées par des fonctions d'assemblage) dans le contexte de conception augmenté. Le chevauchement des relations d'assemblage avec la planification de la séquence d'assemblage a été défini comme la représentation du « *contexte augmenté* ». L'objectif de ces travaux est de fournir toutes les informations et connaissances nécessaires à la recherche et à leur instanciation, pour sensibiliser les concepteurs à l'impact de leurs décisions de conception sur le cycle de vie du futur produit. Nous avons abordé l'approche MERCURY2, une méthode élaborée dans le but de modéliser un système sensible au contexte, en utilisant et en instanciant les connaissances en trois catégories (conditionnelle, déclarative et procédurale), dans le but de faciliter aux systèmes d'information, l'inférence et l'envoi de requêtes dans la base de connaissances.

Le **premier chapitre** a présenté le contexte industriel et scientifique sur les problématiques liées à la réutilisation des connaissances avec le PLM dans le domaine de la conception mécanique. Il en est ressorti des enjeux inhérents à une réutilisation ciblée et efficace des connaissances.

Dans le **second chapitre** un état de l'art sur les travaux de recherche a été abordé dans les domaines de la conception proactive pour l'assemblage, de la réutilisation des connaissances, de la génération de prédiction et de la gestion du cycle de vie des produits. Cette revue de la littérature scientifique a permis de discerner au travers de la modélisation du contexte et du futur contexte, le « *contexte augmenté* » pour les approches concernant la réutilisation des connaissances exploitant les technologies sémantiques.

Dans le **troisième chapitre**, en se basant sur les problématiques présentées précédemment, nous avons proposé un cadre méthodologique appelé DAMON, qui amorce la présentation du contexte de conception augmenté en posant les bases de ce nouveau concept. Représenté par la théorie JANUS, le cadre en question, a été étendu aux notions spatiales et temporelles, afin d'avoir toutes les informations à la conception et à l'assemblage. De ces informations, découlent la recherche de connaissances candidates.

Le **quatrième chapitre** est l'application du cadre méthodologique DAMON, au travers d'une approche évoluée appelée MERCURY2, qui pose les fondements pour le développement d'un système sensible au contexte avec une application existante appelé PEGASUS. Considéré comme un HUB Incubateur, l'application est composée de trois couches. La première couche est la formalisa-

tion d'une ontologie existante, appelée PRONOA3. La deuxième couche est le raisonnement de l'architecture avec la modélisation de règles sémantiques SWRL et SQWRL appelées RUN, représentant des unités de raisonnement pour le parseur facilitant la conversion des données entre l'outil CAO et l'ontologie. La dernière couche concerne la définition de l'Interface Homme-Machine PEGASUS CAD Assistant pour fournir aux parties prenantes les interactions nécessaires permettant de capturer leurs intentions et diffuser les connaissances.

Le **cinquième chapitre** concerne l'application de DAMON et MERCURY2, via deux cas d'étude mécanique et mécatronique, dans lesquels une pince de désincarcération a servi de cas d'application pour le cadre méthodologique DAMON permettant de comparer le nombre de connaissances réutilisées selon trois itérations de la pince. Le deuxième cas d'étude, au travers du HUB PEGASUS, a été l'objet d'application de l'approche MERCURY2, sur un concept de drone-montre appelé DWARF. Le drone en question, a par la suite, été exploité pour réduire le nombre d'itérations, par l'ajout d'une boucle supplémentaire en optimisation numérique et en effectuant des calculs avec la méthode des éléments finis (FEM). L'optimisation numérique et multi-objective a permis d'appréhender la réduction du temps de développement en conception, et surtout de réduire le nombre d'itérations en proposant un prototype, afin d'évaluer les résultats issus de l'approche.

Pour résumer, le concept de « *contexte augmenté* » a permis de mettre en avant une méthodologie de conception proactive facilitant la prise de conscience des concepteurs dans une « *conception orientée assemblage* » des impacts de leurs décisions de conception sur le cycle de vie du produit, en leur permettant d'articuler les informations spatiales et temporelles pour comprendre et anticiper les intentions du concepteur. La proactivité de la démarche a permis de réduire le nombre d'itérations en phase amont du processus de conception de produits, en concevant de manière plus juste, les produits et leurs composants avec l'ajout de fonctions orientées DFA.

6.2/ PERSPECTIVES DE RECHERCHE

De multiples efforts peuvent être développés dans différents domaines pour faire évoluer le concept de « *contexte augmenté* » proposé. Ici nous nous focaliserons sur le domaine de l'ingénierie mécanique sur le développement de nouveaux travaux de recherche positionnés à court, moyen et long terme.

6.2.1/ PERSPECTIVES À COURT TERME

Un besoin émerge actuellement en ce qui concerne l'optimisation du HUB afin d'automatiser à 100% l'application en favorisant notamment son interopérabilité avec l'outil CAO, afin qu'il puisse instancier les connaissances directement sur les espaces de conception et soulager le concepteur de tâches fastidieuses, ce dernier gardant dans tous les cas la main sur le système et donc, le contrôle des opérations.

Ainsi, le HUB proposé, grâce à l'ontologie PRONOA3, pourrait générer les espaces de conception dès la phase de la description du contexte de conception par l'architecte produit, en vue d'interpréter directement les solutions géométriques du produit. Dans ce contexte, il faudra prédire instantanément les relations de précédence, l'intention du concepteur et celles des autres parties prenantes pour favoriser au plus tôt la génération des éléments nécessaires.

Par ailleurs, le HUB aiderait l'architecte produit pendant sa phase de définition de la nomenclature produit, pour l'aiguiller sur les composants favorables à telle ou telle technologie d'assemblage

avec tel ou tel matériau. Tout ceci favoriserait une présélection de connaissances à la fois pour la conception orientée assemblage et pour l'optimisation.

Une vision système de l'intégration du HUB s'impose et doit s'orienter vers un couplage ingénierie systèmes avec DAMON et MERCURY2. Pour faciliter l'interopérabilité entre les différents services et systèmes, il faudrait que le HUB s'adapte à l'environnement et au contexte. En modélisant les processus organisationnels métiers, il est possible d'adapter le HUB selon les différents types de contexte (production, service, maintenance, etc.).

6.2.2/ PERSPECTIVES À MOYEN TERME

Si la tendance de recherche consiste à encourager le système à prendre des décisions sur le choix des connaissances et des informations dans l'outil PEGASUS, le choix de la technologie requise s'impose : les Systèmes Multi-Agents (SMA) peuvent répondre à ces exigences. Dès lors, déléguer les décisions, aux agents virtuels et autonomes pourraient conduire vers davantage d'automatisation du système.

Dans cette perspective, plusieurs catégories d'agents existent et pourraient être mobilisés en tant que telles : des agents réactifs et des agents cognitifs. Les agents réactifs ont la particularité d'être le cœur du raisonnement d'un système pour aller chercher et distribuer aux agents cognitifs l'information, tandis que les agents cognitifs sont la représentation des parties prenantes physiques. Ils ont la tâche de capturer leur intention et de les diffuser au travers du HUB. Ces agents autonomes ont un cycle de vie qui va de la naissance à la mort, et dont l'intérêt réside sur les prises de décisions programmées, circonscrites à chacun de ces groupes d'agents.

L'approche DOCK de l'auteur [Monticolo, 2016], pourrait être appliquée pour concevoir un système multi-agents dédié à la gestion des connaissances. Cette approche spécifie comment modéliser l'organisation du système en utilisant la méthode KROM. En plus de préciser les connaissances sous forme d'ontologie, l'approche DOCK spécifie les organisations d'agents nécessaires (agent cognitif et agent réactif) afin de décrire les interactions entre les agents et les humains ainsi que les mécanismes d'exploitation des connaissances. Par exemple, chacun des agents réactifs assure une ou deux étapes du processus de gestion des connaissances comme la prise en charge d'identification et de sauvegarde des connaissances par un agent appelé **Agent Capitalisation des Connaissances - ACC**.

6.2.3/ PERSPECTIVES À LONG TERME

D'autres perspectives peuvent être formulées, en particulier, sur l'application du « *contexte augmenté* » sur l'un des cycles de vie du produit (fabrication, maintenance, service, etc.). Sur la conception orientée fabrication, les étapes peuvent s'inscrire dans la continuité de la conception orientée assemblage, car il est possible pour le concepteur de détailler son produit en augmentant le contexte de conception orienté fabrication en modélisant chaque composant avec une gamme de fabrication. En complément de la gamme d'assemblage déjà abordée dans nos travaux de recherche, cette gamme de fabrication pourra donner des informations sur les opérations d'usinage telles que l'ordre, les dimensions du brut, les dimensions finales de la pièce, ainsi que les matériaux sélectionnés. Nous pouvons identifier plusieurs situations de conception :

- **Situation de conception no.1** : identification des surfaces de matériaux bruts en fonction des contraintes d'usinage ;
- **Situation de conception no.2** : détermination des surfaces élémentaires ;

- **Situation de conception no.3** : attribution des géométries, des dimensions et de leurs surfaces technologiques ;
- **Situation de conception no.4** : détail et précision des surfaces de finitions.

Ainsi, des contraintes d'outillage, de maintien et de mise en position des pièces peuvent être complétées dans la gamme de fabrication afin d'instancier des connaissances en lien avec le Design For Machining.

Un autre exemple avec le domaine de la maintenance peut aussi être abordé, en particulier lorsque la gamme de maintenance est générée, dans la mesure où celle-ci exploite les connaissances basées sur des informations d'obsolescence du produit, de désassemblage, etc.

RÉFÉRENCES BIBLIOGRAPHIQUES

7.1/ VALORISATION DES TRAVAUX DE THÈSE

7.1.1/ REVUE INTERNATIONALE

[Marconnet *et al.*, 2017] Marconnet B., Demoly F., Monticolo D. and Gomes S. (2017). An Assembly Oriented Design and Optimization Approach for Mechatronic System Engineering. *International Journal for Simulation and Multidisciplinary Design Optimization* 8.

7.1.2/ CONFÉRENCES INTERNATIONALES

[Marconnet *et al.*, 2016] Marconnet B., Demoly F., Monticolo D. and Gomes S. (2016). Optimization and Design For Manufacturing and Assembly of mechatronic systems : Application to an innovative concept of "Follow Me Drone". *ASMDO16, Fifth International conference on multidisciplinary design optimization and applications*, Shenzhen, China, June 26-30.

[Marconnet *et al.*, 2015] Marconnet B., Demoly F., Monticolo D. and Gomes S. (2015). Towards an Approach to Link Knowledge and Prediction in Product Design. In *Product Lifecycle Management in the Era of Internet of Things*, 248–58. *12th IFIP WG 5.1 International Conference, PLM 2015*, Doha, Qatar, October 19-21.

[Marconnet *et al.*, 2014] Marconnet B., Demoly F. and Gomes S. (2014). A Semantic HUB within the Future of PLM. In *2014 Tenth International Conference on Signal-Image Technology and Internet-Based Systems*, Marrakech, Morocco, November 23-27, 709–14.

7.1.3/ COLLOQUES NATIONAUX

[Marconnet *et al.*, 2016] Marconnet B., Demoly F., Monticolo D. and Gomes S. (2016). Proposition d'un contexte de conception augmenté pour la réutilisation des connaissances métier. *STP GDR MACS, GT IS3C*, Colmar, France, 3-4 novembre.

[Marconnet *et al.*, 2014] Marconnet B., Demoly F., Monticolo D. and Gomes S. (2014). L'évolution de la gestion du cycle de vie produit (PLM), au sein de l'entreprise 3.0, pour une ingénierie proactive dans la relation produit/process. Séminaire CONFERE (Collège d'Etudes et de Recherches en Design et Conception de Produits) sur l'Innovation et la Conception, Sibenik, Croatie, 3-4 juillet.

7.2/ RÉFÉRENCES EXTERNES

[Abowd *et al.*, 1999] Abowd G. D., Dey A. K., Brown P. J., Davies N., Smith M. and Steggles P. (1999). Towards a Better Understanding of Context and Context-Awareness. In *Handheld*

and Ubiquitous Computing, edited by Hans-W. Gellersen, 304–7. Lecture Notes in Computer Science 1707. Springer Berlin Heidelberg.

[Abramovici, 2007] Abramovici M. (2007). Future Trends in Product Lifecycle Management (PLM). In *The Future of Product Development*, edited by Professor Dr-Ing Frank-Lothar Krause, 665–74. Springer Berlin Heidelberg.

[Ackoff, 1989] Ackoff RL (1989). From Data to Wisdom. *Journal of Applied Systems Analysis*.

[Ahmad *et al.*, 2005] Ahmad R., Yuqing F., Chaudhry I. and Hamidullah (2005). Collaborative Product Development in Extended Enterprise. In 9th International Multitopic Conference, IEEE INMIC 2005, 1–6.

[Alizon *et al.*, 2006] Alizon F., Shooter S. B. and Simpson T. W. (2006). Reuse of Manufacturing Knowledge to Facilitate Platform-Based Product Realization. *Journal of Computing and Information Science in Engineering* 6, no. 2 : 170.

[Allen & Sriram, 1998] Allen R. H. and Sriram R. D. (1998). Workshop on Knowledge-Based Systems Interoperability. *Journal of Research of the National Institute of Standards and Technology* ; Gaithersburg 103.5 : 535.

[Allweyer, 2010] Allweyer T. (2010) BPMN 2.0 : Introduction to the Standard for Business Process Modeling. Norderstedt : Books on Demand.

[Ameri & Dutta, 2005] Ameri F. and Dutta D. (2005). Product Lifecycle Management : Closing the Knowledge Loops. *Computer-Aided Design and Applications* 2, no. 5 : 577–90.

[Andolfatto, 2013] Andolfatto L. (2013). Assistance à l'élaboration de gammes d'assemblage innovantes de structures composites. Thèse, École normale supérieure de Cachan - ENS Cachan.

[Andolfatto *et al.*, 2014] Andolfatto L., Thiébaud F., Lartigue C. and Douilly M. (2014). Quality- and Cost-Driven Assembly Technique Selection and Geometrical Tolerance Allocation for Mechanical Structure Assembly. *Journal of Manufacturing Systems* 33 : 103–15.

[Andreasen and Mortensen, 1997] Andreasen M. M. and Mortensen N. H. (1997). Basic Thinking Patterns and Working Methods for Multiple DFX. Report on Beiträge Zum 7. Symposium "Fertigungsgerechtes Konstruieren," 1997.

[Andreasen, 2000] Andreasen M. M. (2000). Integrated Product Development. IPU.

[Antoniou & Van Harmelen, 2008] Antoniou G. and Van Harmelen F. (2008). A semantic Web primer. MIT Press, Cambridge, Mass., Second edition.

[Baader *et al.*, 2003] Baader, F., Calvanese D., McGuinness D. L., Nardi D. and Patel-Schneider P. F. (2003). *The Description Logic Handbook : Theory, Implementation, and Applications*. New

York, USA : Cambridge University Press.

- [Barnes, 1999] Barnes C. J. (1999). A Methodology for the Concurrent Design of Products and Their Assembly Sequence.
- [Barnes *et al.*, 2004] Barnes C. J., Jared G. E. M and Swift K. G (2004). Decision Support for Sequence Generation in an Assembly Oriented Design Environment. *Robotics and Computer-Integrated Manufacturing* 20, no. 4 : 289–300.
- [Baxter *et al.*, 2007] Baxter D., Gao J., Case K., Harding J., Young B., Cochrane S. and Dani S. (2007). An Engineering Design Knowledge Reuse Methodology Using Process Modelling. *Research in Engineering Design* 18, no. 1 : 37–48.
- [Baxter *et al.*, 2008] Baxter D., Gao X. and Roy R. (2008). Design Process Knowledge Reuse Challenges and Issues. *Computer-Aided Design and Applications*. Volume 5, 2008 - Issue 6.
- [Bazire & Brézillon, 2005] Bazire M. and Brézillon P. (2005). Understanding Context Before Using It. In *Modeling and Using Context*, edited by Anind Dey, Boicho Kokinov, David Leake, and Roy Turner, 29–40. *Lecture Notes in Computer Science* 3554. Springer Berlin Heidelberg.
- [Belkadi *et al.*, 2013] Belkadi F., Bonjour E., Camargo M., Troussier N. and Eynard B. (2013). A Situation Model to Support Awareness in Collaborative Design. *International Journal of Human-Computer Studies*, Special Issue on supporting shared representations in collaborative activities, 71, no. 1 : 110–29.
- [Bergson, 1919] Bergson H. (1919). *L'énergie spirituelle. Essais et conférences*.
- [Bernard & Perry, 2003] Bernard A. and Perry N. (2003). Fundamental Concepts of Product/Technology/Process Informational Integration for Process Modeling and Process Planning. *International Journal of Computer Integrated Manufacturing* 16, no. 7/8 : 557–65.
- [Berners-Lee *et al.*, 2001] Berners-Lee T., Hendler J. and Lassila O. (2001). The Semantic Web. *Scientific American*.
- [Bilgic & Rock, 1997] Bilgic T. and Rock D (1997). Product Data Management Systems : State-Of-The-Art and the Future. In *Proceedings of the 1997 Asme Design Engineering Technical Conferences and Computers in Engineering Conference*, 97–3720.
- [Boothroyd *et al.*, 1994] Boothroyd G., Dewhurst P. and Knight W. A. (1994). Product Design for Manufacture and Assembly. *Computer-Aided Design* 26, no. 7 : 505–20.
- [Boothroyd *et al.*, 2011] Boothroyd G., Dewhurst P. and Knight W. A. (2011). *Product Design for Manufacture and Assembly*, Third Edition. Taylor & Francis Group. CRC Press.
- [Boytsov & Zaslavsky, 2010] Boytsov A. and Zaslavsky A. (2010). Extending Context Spaces Theory by Proactive Adaptation. In *Smart Spaces and Next Generation Wired/Wireless Networking*,

edited by Sergey Balandin, Roman Dunaytsev, and Yevgeni Koucheryavy, 1–12. Lecture Notes in Computer Science 6294. Springer Berlin Heidelberg.

[Bralla, 1996] Bralla J. G. (1996). Design for Excellence. New York : McGraw-Hill.

[Breuker, 1994] Breuker V. d. v. (1994). Common KADS Library for Expertise Modelling. 1st ed. Amsterdam, The Netherlands, The Netherlands : IOS Press.

[Brimble & Sellini, 2000] Brimble R. and Sellini F. (2000). The MOKA Modelling Language. In Knowledge Engineering and Knowledge Management Methods, Models, and Tools, edited by Rose Dieng and Olivier Corby, 49–56. Lecture Notes in Computer Science 1937. Springer Berlin Heidelberg.

[Byun & Cheverst, 2004] Byun H. E. and Cheverst K. (2004). Utilizing context history to provide dynamic adaptations. Applied Artificial Intelligence, 18(6), 533-548.

[Cantamessa *et al.*, 2012] Cantamessa M., Montagna F. and Neirotti P. (2012). An Empirical Analysis of the PLM Implementation Effects in the Aerospace Industry. Computers in Industry 63, no. 3 : 243–51.

[Chandrasegaran *et al.*, 2013] Chandrasegaran S. K., Ramani K., Sriram R. D., Horváth I., Bernard A., Harik R. F. and Gao W. (2013). The Evolution, Challenges, and Future of Knowledge Representation in Product Design Systems. Computer-Aided Design 45, no. 2 : 204–28.

[Chen *et al.*, 2015a] Chen Y., Zhao M., Xie Y. and Zhang Z. (2015). A New Model of Conceptual Design Based on Scientific Ontology and Intentionality Theory. Part I : The Conceptual Foundation. Design Studies 37.

[Chen *et al.*, 2015b] Chen Y., Zhao M., Xie Y. and Zhang Z. (2015). A New Model of Conceptual Design Based on Scientific Ontology and Intentionality Theory. Part II : The Process Model. Design Studies 38.

[CIMDATA, 2014] CIMDATA (2014). Increasing Bid Success Through Integrated Knowledge Management. <http://www.cimdata.com>.

[Clarkson *et al.*, 2004] Clarkson P. J., Simons C. and Eckert C. (2004). Predicting Change Propagation in Complex Design. Journal of Mechanical Design 126, no. 5 : 788–97.

[Codina & Ceccaroni, 2010] Codina V. and Ceccaroni L. (2010). A Recommendation System for the Semantic Web, in : Carvalho A.P. de L.F. de, Rodriguez-Gonzalez S., Santana J.F.D.P., Rodriguez J.M.C. (Eds.), Distributed Computing and Artificial Intelligence, Advances in Intelligent and Soft Computing. Springer Berlin Heidelberg, pp. 45 ;V52.

[Colombo *et al.*, 2007] Colombo G., Mosca A. and Sartori F. (2007). Towards the Design of Intelligent CAD Systems : An Ontological Approach. Adv. Eng. Inform. 21, no. 2 : 153–168.

- [Curran *et al.*, 2010] Curran R., Verhagen W. J. C., van Tooren M. J. L. and van der Laan T. H. (2010). A Multidisciplinary Implementation Methodology for Knowledge Based Engineering : KNOMAD. *Expert Systems with Applications, Advances in Aligning Knowledge Systems, Improving Business Logistics, Driving Innovation and Adapting Customer Centric Services*, 37, no. 11.
- [Dalglish *et al.*, 2007] Dalglish G. F., Jared G. E. M. and Swift K. G. (2000). Design for Assembly : Influencing the Design Process. *Journal of Engineering Design* 11, no. 1 : 17–29.
- [Demoly, 2010] Demoly F. (2010). Conception intégrée et gestion d'informations techniques : application à l'ingénierie du produit et de sa séquence d'assemblage. Thèse, Université de Technologie Belfort-Montbéliard.
- [Demoly *et al.*, 2011] Demoly F., Yan X.-T., Eynard B., Rivest L. and Gomes S. (2011). An Assembly Oriented Design Framework for Product Structure Engineering and Assembly Sequence Planning. *Robotics and Computer-Integrated Manufacturing* 27, no. 1 : 33–46.
- [Demoly *et al.*, 2012] Demoly F., Matsokis A. and Kiritsis D. (2012). A Mereotopological Product Relationship Description Approach for Assembly Oriented Design. *Robot. Comput.-Integr. Manuf.* 28, no. 6 : 681–693.
- [Demoly *et al.*, 2013] Demoly F., Pels H. J. and Gomes S. (2013). Proactive Engineering and PLM : Current Status and Research Challenges. In *Product Lifecycle Management for Society*, edited by Alain Bernard, Louis Rivest, and Debasish Dutta, 170–81. *IFIP Advances in Information and Communication Technology* 409. Springer Berlin Heidelberg.
- [Demoly *et al.*, 2013] Demoly F., Dutartre O., Yan X.-T., Eynard B., Kiritsis D. and Gomes S. (2013). Product Relationships Management Enabler for Concurrent Engineering and Product Lifecycle Management. *Computers in Industry* 64, no. 7 : 833–48.
- [Dentler *et al.*, 2010] Dentler K., Cornet R., ten Teije A. and de Keizer N. (2010). Comparison of Reasoners for large Ontologies in the OWL 2 EL Profile. *Semantic Web – Interoperability, Usability, Applicability an IOS Press Journal*.
- [Dey, 2000] Dey A. K. (2000). Providing architectural support for building context-aware applications. Ph.D. Thesis, Georgia Institute of Technology.
- [Dey, 2001] Dey A. K. (2001). Understanding and Using Context. *Personal Ubiquitous Comput.* 5, no. 1 : 4–7.
- [Dhouieb, 2016] Dhouieb M. A. (2016). Structuration multi-échelle de la connaissance in-extenso d'entreprise. Thèse de l'Université de Nantes Angers Le Mans.
- [Dhuieb *et al.*, 2016] Dhuieb M. A., Florent L. and Alain B (2016). Context-Awareness : A Key Enabler for Ubiquitous Access to Manufacturing Knowledge. *Procedia CIRP, Research and Innovation in Manufacturing : Key Enabling Technologies for the Factories of the Future -*

Proceedings of the 48th CIRP Conference on Manufacturing Systems, 41, 484–89.

[Dubois, 1998] Dubois D. M. (1998). Computing Anticipatory Systems with Incursion and Hyperincursion. 437 :3–30.

[EADS & Mondon, 2008] EADS and Mondon J.– Y. (2008). Product Lifecycle Management (PLM) HARMONIZATION PROGRAM. (PHENIX Programme) – ENE.

[EADS, 2011] EADS (2011). PHENIX PLM Harmonization Program PHUSION Global Collaboration Project. PARIS Air Show, June 23rd 2011 – PHENIX & Extended Enterprise PLM Harmonization.

[Eggers *et al.*, 2003] Eggers J., Feillet D., Kehl S., Wagner M. O. and Yannou B. (2003). Optimization of the Keyboard Arrangement Problem Using an Ant Colony Algorithm. European Journal of Operational Research 148, no. 3 : 672–86.

[Ermine, 2003] Ermine J.-L. (2003). La Gestion Des Connaissances. Hermes Lavoisier.

[Främling *et al.*, 20013] Främling K., Holmström J., Loukkola J., Nyman J. and Kaustell A. (2013). Sustainable PLM through Intelligent Products. Eng. Appl. Artif. Intell. 26, no. 2 : 789–799.

[Ganeshan *et al.*, 1994] Ganeshan R., Garrett J. and Finger S. (1994). A Framework for Representing Design Intent. Design Studies 15, no. 1 : 59–84.

[Garetti *et al.*, 2005] Garetti M., Terzi S., Bertacci N. and Brianza M. (2005). Organisational Change and Knowledge Management in PLM Implementation. International Journal of Product Lifecycle Management 1, no. 1 : 43.

[Gero, 1990] Gero J. S. (1990). Design Prototypes : A Knowledge Representation Schema for Design. AI Magazine 11, no. 4 : 26.

[Gero & Kannengiesser, 2004] Gero J. S. and Kannengiesser U. (2004). The Situated Function–behaviour–structure Framework. Design Studies 25, no. 4 : 373–91.

[Glimm *et al.*, 2014] Glimm B., Horrocks I., Motik B., Stoilos G. and Wang Z. (2014). Hermit An OWL 2 reasoner. Journal of Automated Reasoning.

[Guindon, 1990] Guindon R. (1990). Knowledge exploited by experts during software system design. Int. J. Man-Mach. Stud. 33, 279-304.

[Gomes, 2008] Gomes S. (2008). Ingénierie à base de connaissances pour une conception productive, optimisée, collaborative et innovante du système Projet Produit Process Usage. Habilitation à diriger les recherches, Université de Technologie Belfort-Montbéliard.

[Gomes *et al.*, 2009] Gomes S., Varret A., Bluntzer J.B. and Sagot J.C. (2009). Functional Design and Optimisation of Parametric CAD Models in a Knowledge-Based PLM Environment.

International Journal of Product Development 9, no. 1/2/3 : 60.

- [Gómez *et al.*, 2013] Gómez A., Mas F., Menéndez J. L. and Ríos J. (2013). A Knowledge Based Application for Industrialization Design. *Procedia Engineering* 63 : 318–26.
- [Gregory *et al.*, 1999] Gregory T., Brunnermeier S. B. and Martin S. A. (1999). Interoperability Cost Analysis of the U.S. Automotive Supply Chain. Final report. Research Triangle Institute, Center for Economics Research : National Institute of Standards and Technology.
- [Gruber, 1993] Gruber T. R. (1993). A Translation Approach to Portable Ontology Specifications. *Knowledge acquisition* 5 : 199–220.
- [Gruhier *et al.*, 2014] Gruhier E., Demoly F., Abboudi S. and Gomes S. (2014). A Spatiotemporal Mereotopology-Based Theory For Qualitative Description in Assembly Design and Sequence Planning. *Design Computing and Cognition DCC'14* Springer.
- [Gruhier *et al.*, 2015] Gruhier E., Demoly F., Dutartre O., Abboudi S. and Gomes S. (2015). A Formal Ontology-Based Spatiotemporal Mereotopology for Integrated Product Design and Assembly Sequence Planning. *Advanced Engineering Informatics* 29, no. 3 : 495–512.
- [Gruhier *et al.*, 2017] Gruhier E., Demoly F. and Gomes S. (2017). A Spatiotemporal Information Management Framework for Product Design and Assembly Process Planning Reconciliation. *Computers in Industry* 90 : 7–41.
- [Gunpinar & Han, 2008] Gunpinar E. and Han S. (2008). Interfacing Heterogeneous PDM Systems Using the PLM Services. *Adv. Eng. Inform.* 22, no. 3 : 307–316.
- [Hackbarth, 1998] Hackbarth G. (1998). The Impact of Organizational Memory on IT Systems. *AMCIS 1998 Proceedings*.
- [Holt & Barnes, 2009] Holt R. and Barnes C. (2009). Towards an integrated approach to "Design for X" : an agenda for decision-based DFX research, *Research in Engineering Design* 21 : 123–36, Springer London.
- [Horrocks *et al.*, 2012] Horrocks I., Motik B. and Wang Z. (2012). The HerMiT OWL Reasoner.
- [Hossain *et al.*, 2012] Hossain S.A., Kadiri S.E. and Bouras A. (2012). Modeling Business Processes for PLM Services Integration in PLM System. In *15th International Conference on Computer and Information Technology (ICCIT)*, 369–75.
- [Huang, 1996] Huang G. Q. (1996). *Design for X : Concurrent Engineering Imperatives*. 1st ed. London : Chapman & Hall.
- [Huang *et al.*, 1999] Huang G.Q., Lee S.W. and Mak K.L. (1999). Web-based product and process data modeling in concurrent "design for X". *Robotics and Computer-Integrated Manufacturing*, 15, 53-63.

- [Jared *et al.*, 1994] Jared G. E. M., Limage M. G., Sherrin I. J. and Swift K. G. (1994). Geometric Reasoning and Design for Manufacture. *Computer-Aided Design* 26, no. 7 : 528–36.
- [Kim *et al.*, 2004] Kim K.-Y., Wang W., Muogboh O. S. and Nnaji B. O. (2004). Design Formalism for Collaborative Assembly Design. *CAD Computer Aided Design* 36, no. 9 : 849–71.
- [Kim *et al.*, 2006] Kim K.-Y., Manley D. G. and Hyungjeong Y. (2006). Ontology-Based Assembly Design and Information Sharing for Collaborative Product Development. *Computer-Aided Design* 38, no. 12 : 1233–50.
- [Kim *et al.*, 2008] Kim K.-Y., Yang H. and Kim D. W. (2008). Mereotopological Assembly Joint Information Representation for Collaborative Product Design. *Robotics and Computer-Integrated Manufacturing* 24, no. 6 : 744–54.
- [Kim & Kim, 2011] Kim K.-Y. and Kim Y.S. (2011). Causal Design Knowledge : Alternative Representation Method for Product Development Knowledge Management. *Computer-Aided Design* 43, no. 9 : 1137–53.
- [Kiritsis, 2013] Kiritsis D. (2013). Semantic Technologies for Engineering Asset Life Cycle Management. *International Journal of Production Research* 51, no. 23–24 : 7345–71.
- [Klein, 2000] Klein R. (2000). Knowledge Modeling In Design — The MOKA Framework. In *Artificial Intelligence in Design '00*, edited by John S. Gero, 77–102. Springer Netherlands.
- [Knight, 1975] Knight F. B. (1975). A Predictive View of Continuous Time Processes. *The Annals of Probability* 3, no. 4 : 573–96.
- [Knublauch *et al.*, 2004] Knublauch H., Ferguson R. W., Noy N. F. and Musen M. A. (2004). The Protégé OWL Plugin : An Open Development Environment for Semantic Web Applications. In *International Semantic Web Conference*, 229–243. Springer Berlin Heidelberg.
- [Kretschmer *et al.*, 2017] Kretschmer R., Pfouga A., Rulhoff S. and Stjepandić J. (2017). Knowledge-Based Design for Assembly in Agile Manufacturing by Using Data Mining Methods. *Advanced Engineering Informatics*.
- [Krogh, 1998] Krogh G. v. (1998). Care in Knowledge Creation. *California Management Review* 40, no. 3 : 133–53.
- [Kuhn, 2012] Kuhn O. (2012). Methodology for Knowledge-Based Engineering Template Update : Focus on Decision Support and Instances Update. Thèse de l'Université Claude Bernard - Lyon I.
- [Kuo *et al.*, 2001] Kuo T.-C., Huang S.H. and Zhang H.-C. (2001). Design for Manufacture and Design for 'X' : Concepts, Applications, and Perspectives. *Computers & Industrial Engineering* 41, no. 3 : 241–60.

- [Kusiak & Salustri, 2007] Kusiak A. and Salustri F. (2007). Computational Intelligence in Product Design Engineering : Review and Trends. IEEE Transactions on Systems, Man, and Cybernetics, Part C : Applications and Reviews 37, no. 5 : 766–78.
- [Kvan, 2000] Kvan T. (2000). Collaborative Design : What Is It?. Automation in Construction 9, no. 4 : 409–15.
- [Laroche, 2007] Laroche F. (2007). Contribution à la sauvegarde des objets techniques anciens par l'archéologie industrielle avancée. Proposition d'un modèle d'information de référence muséologique et d'une méthode inter-disciplinaire pour la capitalisation des connaissances du patrimoine technique et industriel. Thèse, Ecole Centrale de Nantes (ECN) ; Université de Nantes.
- [Lee & Lee, 2007] Lee J. S. and Lee J. C. (2007). Context Awareness by Case-Based Reasoning in a Music Recommendation System. In Ubiquitous Computing Systems, edited by Haruhisa Ichikawa, We-Duke Cho, Ichiro Satoh, and Hee Yong Youn, 45–58. Lecture Notes in Computer Science 4836. Springer Berlin Heidelberg.
- [Lehmann & Magidor, 2002] Lehmann D. and Magidor M. (2002). What Does a Conditional Knowledge Base Entail ?. arXiv :cs/020202.
- [Leibniz, 1976] Leibniz G W. (1976). Philosophical Papers and Letters. Edited by Leroy E. Loemker. Dordrecht : Springer Netherlands.
- [Lestocart, 2008] Lestocart L.-J. (2008). Théories de la « *prédiction* ». L'intelligible connaissance esthétique.
- [Letelier *et al.*, 2006] Letelier J.-C., Soto-Andrade J., Abarzúa F. G., Cornish-Bowden A. and Cárdenas M. L. (2006). Organizational Invariance and Metabolic Closure : Analysis in Terms of (M,R) Systems. Journal of Theoretical Biology 238, no. 4 : 949–61.
- [Lini, 2013] Lini S. (2013). L'anticipation et sa représentation dans les interfaces homme-système en aéronautique : L'anticipation et sa représentation dans les interfaces homme-système en aéronautique. Thèse de l'Université Sciences et Technologies-Bordeaux 1.
- [Liu *et al.*, 2009] Liu S., Duffy A. H. B., Whitfield R. I. and Boyle I.M. (2009). Integration of Decision Support Systems to Improve Decision Support Performance. Knowledge and Information Systems 22, no. 3 : 261–86.
- [Liu *et al.*, 2009] Liu W., Zeng Y., Maletz M. and Brisson D. (2009). Product Lifecycle Management : A Survey. Proceedings of the ASME 2009 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference.
- [Marguet, 2001] Marguet B. (2001). Contribution à l'analyse des variations géométriques dans les ensembles structuraux en aéronotiques : Démarches et Outils. Thèse, Ecole de Normale Supérieure de Cachan.

- [Markus M. L., 2001] Markus M. L. (2001). Toward a Theory of Knowledge Reuse : Types of Knowledge Reuse Situations and Factors in Reuse Success. *Journal of Management Information Systems* 18 : 57–93.
- [Maropoulos & Ceglarek, 2010] Maropoulos P.G. and Ceglarek D. (2010). Design verification and validation in product lifecycle. *CIRP Annals - Manufacturing Technology*, 59(2), 740-759.
- [Martin & Roucoules, 2009] Martin L. et Roucoules L. (2009). Identification produit-procédés pour une conception proactive : application à la fonderie. CFM'09 - XIXème Congrès Français de Mécanique 24-28 août, Marseille, France. pp.1-6.
- [Mathieu & Marguet, 2001] Mathieu L. and Marguet B. (2001). Integrated Design Method to Improve Producibility Based on Product Key Characteristics and Assembly Sequences. *CIRP Annals - Manufacturing Technology* 50, no. 1 (January 1, 2001) : 85–88.
- [Mayrhofer, 2004] Mayrhofer R. (2004). An Architecture for Context Prediction. In *Advances in Pervasive Computing*, Number 3-85403-176-9. Austrian Computer Society (OCG 176).
- [Medeiros, 2009] Medeiros C. B. (2009). Spatio-Temporal Information Systems. *Advanced Geographic Information Systems* 1, no. 51.
- [Merminod & Rowe, 2012] Merminod V. and Rowe F. (2012). How Does PLM Technology Support Knowledge Transfer and Translation in New Product Development? Transparency and Boundary Spanners in an International Context. *Information and Organization* 22, no. 4 : 295–322.
- [Monticolo, 2008] Monticolo D. (2008). Une approche organisationnelle pour la conception d'un système de gestion des connaissances fondé sur le paradigme agent. Thèse de l'Université de Technologie Belfort-Montbéliard.
- [Monticolo *et al.*, 2012a] Monticolo D., Lahoud I. and Bonjour E. (2012). Distributed Knowledge extracted by a MAS using ontology alignment methods. *IEEE International Conference on Computer and Information Sciences*, Kuala Lumpur, Malaysia.
- [Monticolo *et al.*, 2012b] Monticolo D., Lahoud I., Bonjour E. and Demoly F. (2012). SemKnow : A Multi-Agent Platform to Manage Distributed Knowledge by Using Ontologies. *IMECS 2012 Vol.I*.
- [Monticolo, 2016] Monticolo D. (2016). Approches organisationnelles pour la conception de systèmes multi-agents dédiés à la gestion des connaissances ; application aux projets d'ingénierie et d'innovation. Habilitation à Diriger des Recherches, Université de Lorraine.
- [Nadin, 1991] Nadin M (1991). *Mind-Anticipation and Chaos*. Dresden University Press.
- [Nadin, 2010] Nadin M (2010). Anticipation and Dynamics : Rosen's Anticipation in the Perspective of Time. *International Journal of General Systems*. Vol. 39. 1 vols. 3.

- [Nadoveza & Kiritsis, 2014] Nadoveza D. and Kiritsis D. (2014). *Ontology-Based Approach for Context Modeling in Enterprise Applications*. Computers in Industry.
- [Newell, 1982] Newell A. (1982). *The Knowledge Level*. Association for the Advancement of Artificial Intelligence.
- [Newton, 2008] Newton I. (2008). *The Mathematical Papers of Isaac Newton 1664-1666*. Vol. 1. Cambridge University Press. D. T. Whiteside.
- [Nonaka, 1994] Nonaka I. (1994). *A Dynamic Theory of Organizational Knowledge Creation*. Organization Science, Vol. 5, No. 1 p. 14–37.
- [Nonaka & Takeuchi, 1995] Nonaka I. and Takeuchi H. (1995). *The Knowledge-Creating Company*.
- [O'Connor & Das, 2009] O'Connor M. J. and Das A. K. (2009). *SQWRL : A Query Language for OWL*. Proceedings of OWL : Experience and Directions (OWLED 2009).
- [Pahl & Beitz, 1996] Pahl G. and Beitz W. (1996). *Engineering Design - A Systematic Approach*. Aufl. Herausgegeben von London Wallace K, Springer.
- [Pahl *et al.*, 2007] Pahl G., Beitz W., Feldhusen J. and Grote K.-H. (2007). *Engineering Design : A Systematic Approach*. Springer Science & Business Media.
- [Perry, 2007] Perry N. (2007). *Industrialisation Des Connaissances : Approches D'intégration Pour Une Utilisation Optimale En Ingénierie (Cas de L'évaluation Économique)*. Habilitation à diriger les recherches. Sciences de l'ingénieur. Université de Nantes.
- [Petrazoller *et al.*, 2014] Petrazoller N., Demoly F. and Gomes S. (2014). *Top-Down Definition of Design Spaces Based on Skeleton Modelling*. In *Advances in Production Management Systems. Innovative and Knowledge-Based Production Management in a Global-Local World*, edited by Bernard Grabot, Bruno Vallespir, Samuel Gomes, Abdelaziz Bouras, and Dimitris Kiritsis, 249–56. IFIP Advances in Information and Communication Technology 438. Springer Berlin Heidelberg.
- [Pezzulo *et al.*, 2008] Pezzulo G., Butz M. V. and Castelfranchi C. (2008). *The Anticipatory Approach : Definitions and Taxonomies*. In *The Challenge of Anticipation*, edited by Rino Falcone, 23–43. Lecture Notes in Computer Science 5225. Springer Berlin Heidelberg.
- [Rampersad, 1994] Rampersad H. K. (1994). *Integrated and Simultaneous Design for Robotic Assembly : Product Development, Planning...* New York, NY, USA : John Wiley & Sons, Inc.
- [Reymen *et al.*, 2002] Reymen I. M. M. J., Kroes P. and Basten T. (2002). *Modelling the Role of the Design Context in the Design Process : A Domain-Independent Approach*. In *Proceedings of Design Research Society International Conference*, 917–27. Stoke on Trent, UK : Staffordshire University Press.

- [Riedl & Young, 2004] Riedl M. O. and Young R. M. (2004). An Intent-Driven Planner for Multi-Agent Story Generation. In Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems - Volume 1, 186–193. AAMAS '04. Washington, DC, USA : IEEE Computer Society.
- [Roucoules, 2007] Roucoules L. (2007). Contribution À L'intégration Des Activités Collaboratives et Métier En Conception de Produit. Une Approche Au Juste Besoin : Des Spécifications Fonctionnelles Du Produit Aux Choix Des Procédés de Fabrication. Habilitation à diriger les recherches à l'Université de Technologie de Troyes.
- [Rosen, 1985] Rosen R. (1985). Anticipatory systems. Philosophical, mathematical and methodological foundations. New York : Pergamon.
- [Rosen, 2012] Rosen R. (2012). Anticipatory Systems. In Anticipatory Systems, 313–70. IFSR International Series on Systems Science and Engineering 1. Springer New York.
- [Schmidhuber, 1991] Schmidhuber J. (1991). Adaptive Confidence And Adaptive Curiosity. Institut fur Informatik, Technische Universität München, Arcisstr. 21, 800 München 2, Germany.
- [Schneider, 2015] Schneider D. (2015). Flying Selfie Bots. IEEE Spectrum 52, no. 1 : 49–51.
- [Schuh *et al.*, 2008] Schuh G., Rozenfeld H., Assmus D. and Zancul E. (2008). Process Oriented Framework to Support PLM Implementation. Comput. Ind. 59, no. 2–3 : 210–218.
- [Serrafero *et al.*, 2006] Serrafero P., Gomes S., Bonnivard D. and Jézéquel L. (2006). De la mémoire projet à la compétence métier : vers la synthèse de connaissances métier en ingénierie robuste des produits/process, 6th International conference on Integrated Design and Manufacturing in Mechanical Engineering (IDMME'06), Grenoble, 17-19 mai.
- [Shah, 1995] Shah J. J. (1995). Parametric and Feature-Based CAD/CAM : Concepts, Techniques, and Applications. John Wiley & Sons.
- [Sharma, 2005] Sharma A. (2005). Collaborative Product Innovation : Integrating Elements of CPI via PLM Framework. Computer-Aided Design 37, no. 13 : 1425–34.
- [Shih & Anderson, 1997] Shih C.-H. and Anderson B. (1997). A Design/Constraint Model to Capture Design Intent. In Proceedings of the Fourth ACM Symposium on Solid Modeling and Applications, 255–264. SMA '97. New York, NY, USA : ACM.
- [Sidawi & Hamza, 2012] Sidawi B. and Hamza N. (2012). Using Situated FBS Ontology to Explore Designers' Patterns of Behavior in Parametric Environments. Journal of Information Technology in Construction (ITcon) 17, no. 17 : 271–82.
- [Skarka, 2007] Skarka W. (2007). Application of MOKA Methodology in Generative Model Creation Using CATIA. Engineering Applications of Artificial Intelligence, Soft Computing Applications, 20, no. 5 : 677–90.

- [Sly, 2004] Sly D. (2004). Manufacturing Process Management (MPM). Technology Trends in PLM.
- [Sohn *et al.*, 2015] Sohn M., Jeong S., Kim J. and Lee H. J. (2015). Augmented Context-Based Recommendation Service Framework Using Knowledge over the Linked Open Data Cloud. Pervasive and Mobile Computing, Special Issue on Secure Ubiquitous Computing, 24 : 166–78.
- [Song *et al.*, 2005] Song M., Van Der Bij H. and Weggeman M. (2005). Determinants of the Level of Knowledge Application : A Knowledge-Based and Information-Processing Perspective. Journal of Product Innovation Management 22, no. 5 : 430–44.
- [Song *et al.*, 2006] Song M., Van Der Bij H. and Weggeman M. (2006). Factors for Improving the Level of Knowledge Generation in New Product Development. R&D Management 36, no. 2 : 173–87.
- [Sowa, 1984] Sowa J. F. (1984). Conceptual Structures : Information Processing in Mind and Machine. Boston, MA, USA : Addison-Wesley Longman Publishing Co., Inc.
- [Standard ISO 15926, 2003] Standard ISO 15926 (2003). Integration of lifecycle data for process plant including oil and gas production facilities.
- [Tăndăreanu & Ghindeanu, 2003] Tăndăreanu N. and Ghindeanu M. (2003). Towards a Mathematical Modelling of Conditional Knowledge.
- [Terzi *et al.*, 2010] Terzi S., Bouras A., Dutta D., Garetti M. and Kiritsis D. (2010). Product Lifecycle Management – from Its History to Its New Role. International Journal of Product Lifecycle Management 4, no. 4 : 360–89.
- [Ullman, 2009] Ullman D. (2009). The Mechanical Design Process. 4 edition. Boston : McGraw-Hill Science/Engineering/Math.
- [Vernier, 2014] Vernier C. (2014). Ingénierie à base de connaissances pour une conception semi-automatisée de systèmes mécaniques et des procédés de fabrication associés - Application à la ligne d'échappement automobile. Thèse de l'Université de Technologie Belfort-Montbéliard.
- [Viera *et al.*, 2011] Vieira V., Tedesco P. and Salgado A. C. (2011). Designing Context-Sensitive Systems : An Integrated Approach. Expert Systems with Applications, Intelligent Collaboration and Design, 38, no. 2 : 1119–38.
- [Vogel, 1991] Vogel C. (1991). Expression langagière de l'expertise : Problèmes d'analyse, intellectuelle 1991/2, n. 12 : Expertise et Sciences Cognitives, n.d., Pages 65-100.
- [Von Glasersfeld, 1998] Von Glasersfeld E. (1998). Anticipation in the Constructivist Theory of Cognition. 437 : 38–48.

- [Vijaykumar & Chakrabarti, 2008] Vijaykumar G. and Chakrabarti A. (2008). Understanding the Knowledge Needs of Designers During Design Process in Industry. *Journal of Computing and Information Science in Engineering* 8, no. 1.
- [W3C, 2013] W3C Recommendation (2013). SPARQL 1.1 Overview. <http://www.w3.org/TR/sparql11-overview/>.
- [Weber, 2003] Weber C., Werner H. and Deubel T. (2003). A Different View on Product Data Management/Product Life-Cycle Management and Its Future Potentials. *Journal of Engineering Design* 14, no. 4 : 447–64.
- [Wieggers & Vergeest, 2001] Wieggers T. and Vergeest J. S. M. (2001). Extraction of Cad Tool Requirements from Industry and from Experimental Design Projects. In *Proceedings of ASME*.
- [Wiener, 1964] Wiener N (1964). *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. The MIT Press.
- [Wray *et al.*, 2004] Wray R. E., Lisse S. A. and Beard J. T. (2004). Ontology Infrastructure for Execution-Oriented Autonomous Agents. *Robotics and Autonomous Systems, Knowledge Engineering and Ontologies for Autonomous Systems AAAI Spring Symposium*, 49, no. 1–2 : 113–22.
- [Yan *et al.*, 2001] Yan X.-T, Borg J. and Juster N. P. (2001). Concurrent modelling of components and realisation systems to support proactive design for manufacture/assembly. *J.Eng. Manuf., In :ProInstMechEngPartB*, 215,1135-1141.
- [Zamenopoulos & Alexious, 2007] Zamenopoulos T. and Alexious K. (2007). Towards an Anticipatory View of Design. *Design Studies* 28, no. 4 : 411–36.
- [Zamenopoulos, 2012] Zamenopoulos T. (2012). A Complexity Theory of Design Intentionality. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 26, no. 01 : 63–83.
- [Zhang & Luo, 2010] Zhang Y. and Luo X. (2010). Semantic Query and Reasoning for Design Meta-Intent Information. *3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT)*. 9 : 672–76.
- [Zhu *et al.*, 2010] Zhu L., Jayaram U., Jayaram S. and Kim O. (2010). Querying and Reasoning With Product Engineering Ontologies : Moving Past Modeling. 549–61.
- [Zhu *et al.*, 2011] Zhu L., Jayaram U. and Kim O. (2011). Semantic Applications Enabling Reasoning in Product Assembly Ontologies—Moving Past Modeling. *Journal of Computing and Information Science in Engineering* 12, no. 1.
- [Zhu *et al.*, 2012] Zhu L., Jayaram U. and Kim O. (2012). Semantic Applications Enabling Reasoning in Product Assembly Ontologies; Moving Past Modeling. *J. Comput. Inf. Sci. Eng.* 12, 011009-011009.

- [Zimmermann *et al.*, 2007] Zimmermann A., Andreas L. and Reinhard O. (2007). An Operational Definition of Context. In *Modeling and Using Context*, edited by Boicho Kokinov, Daniel C. Richardson, Thomas R. Roth-Berghofer, and Laure Vieu, 558–71. *Lecture Notes in Computer Science* 4635. Springer Berlin Heidelberg.
- [Zins, 2007] Zins C. (2007). Conceptual Approaches for Defining Data, Information, and Knowledge : Research Articles. *J. Am. Soc. Inf. Sci. Technol.* 58, no. 4 : 479–493.

Document réalisé avec L^AT_EX :

Le style L^AT_EX pour Thèse de Doctorat créé par S.Galland - <http://www.multiagent.fr/ThesisStyle>
La collection de paquets `tex-upmethodology` — <http://www.arakhne.org/tex-upmethodology/>
Images libre de droit — www.freepik.com, <https://pixabay.com>, www.shutterstock.com

Résumé :

De nos jours, l'intégration et la réutilisation des connaissances dans la gestion du cycle de vie des produits restent un enjeu, surtout dès les premières phases de conception. En effet, au cours des dernières décennies, les efforts de recherche ont été concentrés sur l'acquisition de connaissances métier de manière à les réutiliser facilement pendant le processus de conception du produit. Cela a conduit à l'approche réussie du Design For X (DFX), mais les mécanismes de réutilisation des connaissances lors de la conception du produit restent actuellement encore limités, surtout lorsqu'il s'agit de pouvoir les réactiver efficacement. Depuis, les produits deviennent de plus en plus complexes et les principes de l'ingénierie concurrente ont introduit de nombreuses contraintes, pour un meilleur équilibre entre chacune des phases du cycle de vie du produit. Les concepteurs ont besoin d'avoir une vision claire et être conscients des impacts de leurs décisions, en particulier au cours du processus de réutilisation des connaissances dans le développement de nouveaux produits. Pour surmonter ce problème, nous proposons une démarche ambitieuse s'appuyant sur la notion de prédiction dans la conception proactive du couple produit-process, afin de fournir au concepteur une vision du futur en combinaison avec les retours d'expérience du passé (connaissance) tout en assurant la sensibilisation des concepteurs (présent) à l'impact de leurs décisions de conception. Ainsi, l'objectif de ces travaux est de mettre en avant un nouveau concept appelé « *contexte augmenté* » qui consiste à superposer, de manière adéquate, les informations du contexte présent avec le futur. Le contexte en question s'est limité au domaine de la conception pour l'assemblage (DFA), avec l'exploitation de la gamme de montage en tant que futur contexte. Par conséquent, le fait de générer des prédictions, qui sont en lien avec la planification des différentes étapes du cycle de vie, permettra l'activation/la mise à jour de connaissances appropriées au cours des étapes de conception de produits, et donc de réduire considérablement les erreurs de conception.

Mots-clés : Gestion du cycle de vie des produits, Gestion des connaissances, Prédiction, Contexte augmenté

Abstract:

Nowadays, the knowledge integration and reuse in product lifecycle management is still a challenging issue, especially in the early stages of product design. Indeed, over the last decades, research efforts have been made on knowledge capture in a way to be easily reusable during the product design process. This has led to successful Design For X (DFX) approaches, but it is still limited to reactive ones, since products become more and more complex and concurrent engineering principles have introduced numerous constraints to be well-balanced regarding the whole product life cycle, designers need to have a clear vision and be aware of the impact of decisions, especially during the knowledge reuse process in new product development. To overcome this issue, we propose to go beyond by introducing the notion of prediction in product design so as to provide a future vision in combination with a past one (knowledge), therefore ensuring awareness of designers (present). Thus the objective of this work is to put forward, a concept called "*augmented context*" which consist to increase the design context, for reuse in an appropriate manner adequate knowledge. The context in question is oriented towards Design For Assembly (DFA), with the exploitation of the assembly sequence planning as the future context. In such a context, the fact of generating predictions (related to lifecycle planning) will enable appropriate knowledge activation/update during product design stages and therefore reducing considerably design mistakes.

Keywords: Product Lifecycle Management, Knowledge Management, Prediction, Augmented context

The logo for SPIM (École doctorale SPIM) features a stylized white 'S' on a blue background, followed by the letters 'PIM' in a large, white, sans-serif font.

■ École doctorale SPIM - Université de Technologie Belfort-Montbéliard

F - 90010 Belfort Cedex ■ tél. +33 [0]3 84 58 31 39

■ ed-spim@univ-fcomte.fr ■ www.ed-spim.univ-fcomte.fr

