



**HAL**  
open science

# Mise en œuvre d'une approche d'estimation de besoin de maintenance par système à base de connaissances

Alexandre Sarazin

## ► To cite this version:

Alexandre Sarazin. Mise en œuvre d'une approche d'estimation de besoin de maintenance par système à base de connaissances. Autre [cs.OH]. Ecole des Mines d'Albi-Carmaux, 2021. Français. NNT : 2021EMAC0013 . tel-03575193

**HAL Id: tel-03575193**

**<https://theses.hal.science/tel-03575193>**

Submitted on 15 Feb 2022

**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.

Université Fédérale



Toulouse Midi-Pyrénées

# THÈSE

en vue de l'obtention du

## DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

délivré par

IMT – École Nationale Supérieure des Mines d'Albi-Carmaux

---

présentée et soutenue par  
Alexandre SARAZIN

le 26 octobre 2021

Mise en œuvre d'une approche d'estimation de  
besoin de maintenance par système à base de  
connaissances

---

**École doctorale et discipline ou spécialité :**

EDSYS : Informatique et Génie Industriel

**Unité de recherche :**

Centre Génie Industriel, IMT Mines Albi

**Directeur de thèse :**

Xavier LORCA, Professeur, IMT Mines Albi

**Autres membres du jury :**

Anne LAURENT, Professeure, LIRMM (*Rapporteure*)

Nejib MOALLA, Professeur, Université Lumière Lyon 2 (*Rapporteur*)

Jean-Pierre BOUREY, Professeur, Ecole Centrale de Lille (*Examinateur*)

Chihab HANACHI, Professeur, Université Toulouse 1 (*Président*)

Sébastien TRUPTIL, Ingénieur-chercheur, CEA Tech (*Co-encadrant*)

Aurélie MONTARNAL, Maître-assistante, IMT Mines Albi (*Invitée*)

Jérémy BASCANS, HO Data IA, APSYS (*Invité*)



# Remerciements

Il paraît que ça fait bien de commencer des remerciements par une citation donc je vais choisir celle-ci : « le plus important, ce n'est pas la destination, mais les mésaventures et les souvenirs que l'on crée le long du chemin ». Le moins que l'on puisse dire, c'est que le chemin en question a été incroyablement riche, avec beaucoup de hauts mais aussi des bas, des moments de joie et des envies d'abandon. Je ne serais pas arrivé au bout de cette aventure seul, et j'aimerais remercier celles et ceux qui ont contribué à la richesse et l'aboutissement de ce voyage. Préparez-vous, ils sont nombreux !

Tout d'abord, je tiens à remercier mon encadrement aux Mines d'Albi : *Sébastien Truptil, Aurélie Montarnal, Xavier Lorca et Jacques Lamothe*.

*Sébastien*, tu es la première personne que à laquelle je pense au moment d'écrire ces remerciements. Sans toi, cette aventure n'aurait sûrement jamais commencé et se serait probablement finie trop vite. Après tout, je fais partie de ceux qui ont dit un jour, qu'ils ne se lanceraient jamais dans une thèse. Tu m'as fait mentir et je t'en remercie, tes enseignements et ton soutien m'ont fait grandir en tant que chercheur mais aussi, et surtout, en tant que personne. *Aurélie*, comme Sébastien je te connais depuis mon arrivée à l'école en 2014 ! Et d'aussi loin que je me souviens tu as toujours apporté avec toi une bonne humeur incroyable que j'ai d'autant plus appréciée durant mon quotidien de thésard. Je garde en mémoire les nombreuses heures passées à discuter de nos modèles et ton oreille toujours à l'écoute quand j'en avais besoin. *Xavier*, tu as pris la direction d'une thèse en marche, à la suite de Jacques. Cet exercice n'a pas dû être simple pourtant cela a paru facile et naturel. Je me souviendrai longtemps des moments passés avec Aurélie et toi à débattre sur les modèles et je te revois encore faire les cent pas dans ton salon, en plein confinement, lors de la rédaction du manuscrit. Pour tout ces souvenirs et ces enseignements, merci ! Bien sûr, je n'oublie pas *Jacques*, mon premier directeur de thèse. Merci de m'avoir accompagné dès les premiers pas de cette aventure et de m'avoir lancé pour mes premiers articles.

Une raison majeure qui m'a poussé à faire cette thèse, c'est la possibilité de la réaliser en entreprise, au sein d'APSYS. Mes premiers remerciements vont à *Jérémy Bascans*. De l'avis général, ton arrivée a marqué un tournant dans cette aventure. Merci pour ton soutien, tes idées et le combat rudement mené pour obtenir les données nécessaires à l'expérimentation. Nous n'avons pas fini de nous voir et je n'ai

---

certainement pas fini d'apprendre de nos échanges. Un grand merci également à *Julien Commanay* pour m'avoir donné l'opportunité de faire cette thèse et accompagné dans les moments difficiles qui ont jalonné son lancement. Le chemin n'était pas sans embûches mais le résultat en vaut aujourd'hui clairement la peine. Merci à *Jean-Baptiste Sciau* pour ta contribution aux expérimentations. J'ai hâte de pouvoir t'accompagner pour ta propre thèse à venir. Merci également à *Laurent Sagaspe* pour être venu à de nombreuses reprises à Albi (depuis Bordeaux!) apporter ta vision et tes nombreuses idées. Enfin merci à *Emmanuel Arbaretier* sans qui cette thèse n'aurait jamais vu le jour.

Je remercie également les membres du jury pour l'attention apportée à mon travail et les nombreux conseils qui me donnent aujourd'hui de nouvelles perspectives pour la suite de mes recherches.

J'ai également une pensée pour mes collègues, à APSYS et à Albi, qui ont ponctué cette aventure de moments inoubliables. Merci à l'équipe Outils et Modélisations qui ne disparaîtra jamais vraiment. Merci au pôle Data et IA pour qui l'histoire ne fait que commencer. Merci à tous les doctorants et les permanents du centre de génie industriel des Mines d'Albi pour tous les bons moments partagés ensemble. Je ne vous citerai pas tous au risque d'en oublier et de rendre ces remerciements plus long qu'un des chapitres. Je ferai juste une entorse pour Manon. Je te souhaite le meilleur pour la fin de ta thèse et la suite de ton parcours et te remercie pour le chemin parcouru ensemble.

Merci à mes amis de m'avoir supporté, en particulier durant la rédaction. Merci à Rémi, Alexis, José, Ventu, Marion, Fabien, Yvan, Bubuche, Rico, Colantonio, Titi, Tigrou, Antoine, Claire, Erwan et tous ceux dont le nom n'est pas écrit mais à qui je paierai volontiers un verre en dédommagement s'ils ont le courage de lire mon manuscrit jusqu'à ce point.

Enfin, je dédie cette thèse à ma petite soeur *Aurélie* pour lui prouver que même l'improbable n'est pas impossible. Durant ces nombreuses années d'étude j'ai toujours pu compter sur le soutien inconditionnel de ma famille. Je profite donc de ce préambule pour remercier ma mère *Cathy*, mon père *Eric*, ma tante *Josiane*, *Bruno*, ainsi que mes oncles *Pascal* et *Jean-René* pour m'avoir accompagné, soutenu et poussé durant ces études interminables.

Malgré la taille déjà conséquente de ces remerciements, je sais qu'il me reste encore beaucoup de monde à citer, en particulier côté famille, mais vous savez que je suis étourdi et je sais que vous ne m'en voudrez pas. Même sans vous citer, je vous le dit : MERCI!

# Sommaire

Remerciements . . . . .	iii
Sommaire . . . . .	v
<hr/>	
1 Introduction générale . . . . .	1
Partie I État de l'art et positionnement . . . . .	5
2 État de l'art . . . . .	7
2.1 La notion de maintenance . . . . .	7
2.2 Le pronostic et la gestion de la santé - PHM . . . . .	11
3 Verrous scientifiques et positionnement . . . . .	37
3.1 Identification des verrous scientifiques . . . . .	37
3.2 Présentation de l'approche proposée . . . . .	38
Partie II Présentation de la solution de diagnostic . . . . .	41
4 Principe d'un système expert . . . . .	45
5 Base de connaissances . . . . .	49
5.1 Stockage des données . . . . .	49
5.2 Structure des connaissances . . . . .	54
6 Moteur d'inférences . . . . .	71
Partie III Transformation de modèle . . . . .	75
7 Architecture dirigée par les évènements . . . . .	77
8 Architecture dirigée par les modèles . . . . .	81
8.1 Principe . . . . .	81
8.2 Transformation des règles de surveillance vers un modèle de règles génériques . . . . .	84
8.3 Transformation des règles génériques en EPL . . . . .	88
9 Illustrations . . . . .	91
9.1 Description de système . . . . .	91

---

9.2	Présentation des CIM	92
9.3	Présentation des PIM	94
9.4	Présentation des PSM	96
Partie IV	Cas d'étude	101
10	Les enjeux de la maintenance dans le domaine aéronautique	103
11	Protocole expérimental	107
11.1	Présentation des données	107
11.2	Identification d'un composant défaillant	108
11.3	Le modèle statistique : la loi de Weibull	110
12	Analyse des résultats	115
13	Conclusion et perspectives	119
13.1	Bilan des travaux	119
13.2	Limites et perspectives	121
	Table des figures	125
	Liste des tableaux	129
	Acronymes	131
	Bibliographie	133
	Table des matières	145

# Chapitre 1

## Introduction générale

### IMT

Les travaux présentés dans ce manuscrit sont les fruits d'une collaboration entre la société APSYS et le Centre de Génie Industriel des Mines d'Albi dans le cadre d'une convention CIFRE.

APSYS est une société fondée en 1985 et spécialisée dans le domaine de la maîtrise des risques. Elle propose un ensemble d'offres de service visant à maintenir la continuité de service des systèmes et installations de ses clients dans un objectif de durabilité d'activité.

Le chiffre d'affaires de la société en 2020 s'élève à 65 828 000 et ses effectifs fin 2020 s'approchent des 600 salariés principalement répartis sur trois agences en France. L'activité commerciale d'APSYS est principalement tournée vers les secteurs du transport, de la cybersécurité et des durabilités urbaine et industrielle. Le premier secteur se spécialise plus particulièrement sur l'étude de la sûreté de fonctionnement de systèmes aéronautiques, spatiaux, navals, automobiles et ferroviaires.

Depuis sa création, APSYS entretient un lien particulièrement fort avec les domaines de l'aéronautique et du spatial, renforcé par son statut de filiale du groupe Airbus. La sûreté de fonctionnement et la maintenance s'inscrivent fondamentalement dans le cadre de ces activités. Cette thèse a ainsi été initiée par le département DTO (Digital Transformation Office) chargé du support informatique et du développement de solutions digitales. Ces solutions ont pour but de soutenir l'activité commerciale d'APSYS par une action conjointe et collaborative avec les départements Safe Mobile Systems (SMS), Security et Industrial and Urban Sustainability (IUS).

L'objectif des travaux présentés est, plus particulièrement, de proposer des solutions face aux défis et mutations touchant la maintenance des systèmes complexes. En effet, le domaine de la maintenance fait aujourd'hui face à de nouveaux enjeux du fait de l'émergence du numérique et de la multiplicité des données. De plus, la complexité croissante des systèmes maintenus engendre une augmentation des risques et de la diversité des pannes pouvant les affecter.

Bien que toutes les documentations techniques existent, leur volume sur un tel système complexe devient de plus en plus difficile à exploiter sans avoir recours à une approche

---

automatique. Cette difficulté survient par ailleurs dans un contexte d'accroissement des exigences relatives à la fiabilité et à la rentabilité des systèmes dans le but de réduire les impacts humains et économiques causés par leurs défaillances.

Face à ces challenges, les capacités nouvelles de collecte et de gestion des flux de données représentent une opportunité pour compléter ou remplacer les stratégies de maintenances correctives et systématiques avec une approche conditionnelle, prévisionnelle ou prescriptive.

Ces enjeux se traduisent par des défis techniques comme :

- La capacité à ingérer, agréger et exploiter des données de surveillance volumineuses, véloces et hétérogènes
- L'identification des pannes et des composants défaillants dans un système complexe
- L'estimation fiable et précise de la durée de vie résiduelle des pièces.
- La valorisation des analyses réalisées pour accompagner l'utilisateur final dans sa prise de décision.

Dans le cadre de cette thèse, deux problématiques sont particulièrement abordées :

1. La valorisation des connaissances expertes capitalisées pour l'analyse des données de surveillance
2. La gestion du traitement en temps réel de données volumineuses, variées et véloces dans une perspective de détection d'anomalies

Ces problématiques seront abordées en tenant compte de certaines hypothèses liées aux caractéristiques des systèmes considérés :

- les systèmes étudiés sont suffisamment complexes pour justifier une gestion automatique du processus de maintenance,
- ils sont équipés de systèmes de collecte de données renvoyant les informations nécessaires à l'évaluation de son état de santé. Du fait de la complexité du système, ces données sont caractéristiques d'un contexte "big data",
- l'application de modèles d'apprentissage n'est pas toujours possible du fait d'un faible volume d'occurrences de pannes. Ces données relèvent d'un problème de type "small data" (KITCHIN et al. 2015).
- une connaissance experte capitalisée et structurée est disponible au travers d'une documentation technique. Cette documentation permet d'exploiter des informations sur la composition et la fiabilité du système surveillé.

Afin de répondre aux contraintes et problématiques précédemment décrites, une approche a été imaginée sur la base du concept de *pronostic et gestion de la santé* (PHM : *Prognostics and Health Management*). L'architecture fonctionnelle de l'approche proposée est décrite dans la figure 1.1.

Cette architecture fonctionnelle se compose de quatre grandes étapes. La première étape consiste à collecter les données provenant de sources diverses comme des capteurs, ou des systèmes d'information de maintenance.

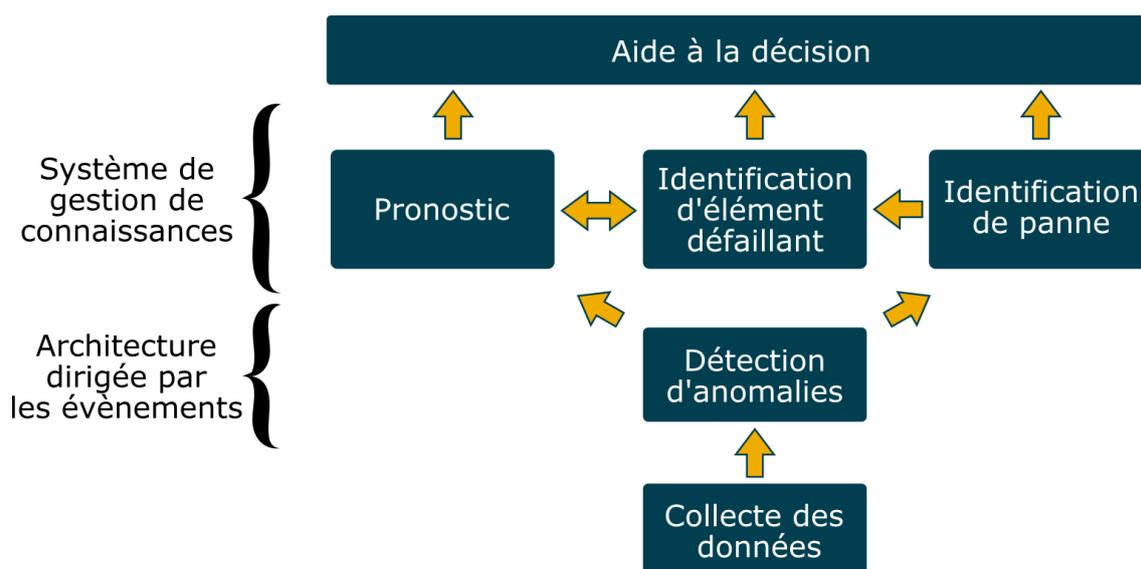


Figure 1.1 – Présentation de l'architecture fonctionnelle

Les données collectées, volumineuses, véloces et hétérogènes, sont ensuite surveillées par une architecture dirigée par les événements pour détecter les anomalies.

Les anomalies détectées sont ensuite analysées par un système de gestion de connaissances pour identifier les défaillances et éléments défaillants actuels ou à venir. Dans le cas où la défaillance ne s'est pas encore produite, un pronostic de la durée de vie résiduelle du système peut être réalisé.

Les analyses produites sont alors transmises à un module d'aide à la décision chargé d'accompagner l'utilisateur final dans l'exploitation des informations générées.

Bien que le domaine aéronautique soit un axe d'application privilégié, de par l'activité d'APSYS, une attention toute particulière a été portée à la généricité et l'adaptabilité de l'approche proposée à des systèmes et secteurs d'activité variés.

La partie I de ce manuscrit de thèse présente un état de l'art introduisant les grandes stratégies de maintenance et les principales techniques utilisées dans chacune des étapes d'une approche PHM.

Dans la partie II, la valorisation des connaissances expertes au moyen d'un système de gestion de connaissances sera présentée. Cette partie constituera la première contribution de cette thèse qui a donné lieu à la publication d'un article dans le journal Expert System With Applications (SARAZIN et al. 2021).

La partie III décrira l'implémentation d'une architecture dirigée par les événements permettant de détecter des anomalies dans des flux de données volumineux, véloces et hétérogènes. Cette seconde contribution a également donné lieu à la publication d'un article dans la conférence MODELSWARD qui a ensuite été sélectionné et étendu pour publication (SARAZIN et al. 2020a).

Enfin, un cas d'étude sera présenté dans la partie IV pour illustrer la mise en oeuvre du système de gestion de connaissances dans le cadre de l'identification de pannes dans un sous-système d'aéronef.



# **Partie I**

## **État de l'art et positionnement**



# Chapitre 2

## État de l'art

### 2.1 La notion de maintenance

À l'origine, le mot « *maintenance* », dérivé du verbe « *maintenir* », est issu des termes latins « *manu* » qui signifie main et « *tenere* » tenir. Il désigne « *l'ensemble des opérations d'entretien permettant de maintenir un appareil ou un ensemble d'appareils, un véhicule, un navire en état de fonctionner normalement* » (ACADÉMIE FRANÇAISE 2021). Par définition, la maintenance est donc fortement liée à la notion d'outil car il n'y a pas d'outil sans maintenance et réciproquement. La maintenance serait donc apparue avec les premiers outils. Bien qu'elle se soit longtemps restreinte aux seuls biens tangibles, la dématérialisation des outils de travail au 20ème siècle a provoqué une altération profonde de son champ d'application. Suite à ces évolutions, ce périmètre englobe aujourd'hui notamment le maintien de processus, de programmes informatiques et, plus généralement de systèmes d'information.

Au fil du temps, la maintenance s'est déclinée en diverses stratégies qu'il convient de définir et catégoriser. A cette fin, ce manuscrit s'appuiera sur les concepts normalisés par l'AFNOR et l'ISO (AFNOR 2018). Par conséquent, la maintenance sera à présent considérée comme l'ensemble de toutes les actions techniques, administratives et de management durant le cycle de vie d'un bien, destinées à le maintenir ou à le rétablir dans un état dans lequel il peut accomplir la fonction requise. Un bien désigne ici tout « *élément, composant, mécanisme, sous-système, unité fonctionnelle, équipement ou système qui peut être considéré individuellement* ». On appelle défaillance, la perte de l'aptitude d'un bien à accomplir une fonction requise. La défaillance correspond ainsi à un événement ou une succession d'événements amenant le bien dans un état de panne. Une panne résulte généralement d'une défaillance mais elle peut également être liée à la conception, la spécification, la construction ou la maintenance du système. On parle alors de panne latente.

La figure 2.1 présente une classification des typologies de maintenances selon AFNOR (2018). D'après cette classification, la maintenance peut se décomposer en trois stratégies principales : l'amélioration, qui consiste à modifier des caractéristiques techniques du bien pour le rendre plus fiable, la maintenance corrective qui intervient

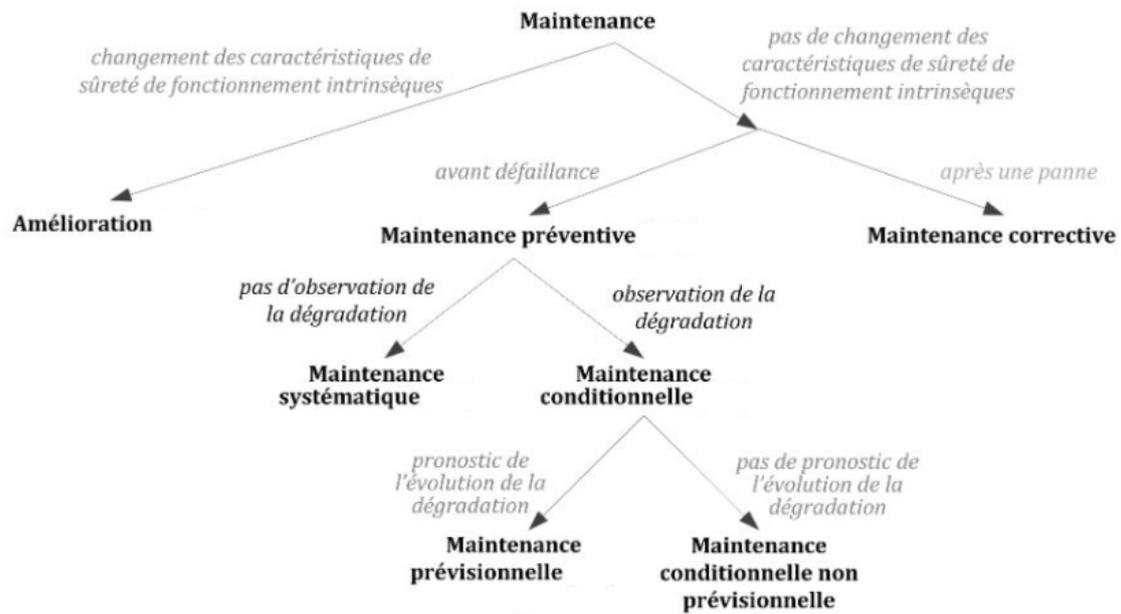


Figure 2.1 – Classification des typologies de maintenance (AFNOR 2018)

après l'apparition d'une défaillance et la maintenance préventive qui vise à prévenir l'émergence des défaillances.

### 2.1.1 La maintenance corrective

Historiquement, la première stratégie adoptée pour gérer des biens consiste simplement à attendre l'apparition d'une défaillance avant d'effectuer une réparation ou un remplacement. Cette pratique, connue sous le nom de maintenance corrective, est exécutée après détection d'une défaillance et destinée à rétablir un bien dans un état dans lequel il peut accomplir une fonction requise. Bien que rudimentaire, cette stratégie possède de nombreux avantages et est toujours largement utilisée. Elle concerne entre autre les produits dit consommables mais ne s'y limite pas.

Cette stratégie tend à réduire la gravité d'une panne par la mise en place de procédures permettant de rétablir l'équipement dans un état fonctionnel le plus rapidement possible et/ou en réduisant l'impact de l'indisponibilité sur la production (SHEUT et al. 1994). Par ailleurs, aucune mesure de prévention ne saurait garantir à 100% l'absence de défaillance. La mise en place de procédures correctives ne devrait ainsi pas être négligée dans l'élaboration d'un programme de maintenance (FINCH et al. 1986 ; KELLY 1989 ; PEELE et al. 1984). Parmi les solutions de maintenance correctives se trouvent la préparation de machines de rechange et/ou d'un inventaire pour les pièces, la redondance de composants critiques ou encore la gestion optimisée d'une équipe de maintenance (ELSAIED et al. 1979 ; SHEUT et al. 1994 ; SIMPSON 1978).

Cependant, ce type de traitement n'est pas toujours satisfaisant. Il est des systèmes dont la fiabilité est primordiale comme c'est le cas des moyens de transport (avion,

train, voiture), des équipements médicaux (systèmes d'assistance respiratoire) ou encore des systèmes de distribution d'énergie. Les défaillances de ces systèmes pouvant mettre en jeu la vie de personnes, des actions visant à prévenir ces risques doivent être mises en place.

Sur le plan économique, les pannes imprévues peuvent également générer des coûts importants à la fois directs et indirects. Dans le domaine aéronautique par exemple, l'immobilisation d'un avion au sol entraîne, d'une part, des coûts directs liés à l'achat, à l'acheminement imprévu des pièces et à la main d'oeuvre mais également des coûts indirects plus importants encore liés à l'annulation ou au retard potentiel d'un ou plusieurs vols, au dédommagement des passagers, à la gestion des membres d'équipages, à la distribution des bagages ou encore liés aux taxes aéroportuaires (BADKOOK 2016).

Dans le secteur industriel, la défaillance d'une machine goulot dans une chaîne de production peut engendrer des pertes importantes liées à la réparation même du bien mais aussi et surtout du fait du ralentissement voire de l'arrêt de la chaîne logistique (TSANG 1995).

## 2.1.2 La maintenance préventive

Afin de répondre à ces contraintes, il est nécessaire d'anticiper et de réduire le risque d'occurrence des défaillances en programmant des opérations de maintenance avant leur apparition. Les stratégies développées dans cette optique intègrent le domaine de la maintenance préventive autrement dit des maintenances destinées à évaluer et/ou atténuer la dégradation et réduire la probabilité de défaillance d'un bien.

D'après la figure 2.1, la maintenance préventive se divise en deux catégories, la maintenance systématique et la maintenance conditionnelle.

La maintenance systématique est exécutée à intervalles de temps préétablis ou selon un nombre défini d'unités d'usage mais sans contrôle préalable de l'état du bien. En pratique, cela revient à programmer une opération de maintenance selon une périodicité calendaire ou en unités d'usage comme les heures de vol ou le nombre de cycles décollage/atterrissage pour un avion. Cette pratique s'accompagne généralement d'une évaluation théorique de la durée de vie du bien considéré. Cette estimation peut être réalisée par des méthodes statistiques ou au moyen de modèles de dégradation théoriques en reproduisant artificiellement les conditions d'utilisation. On pourrait prendre en exemple la réalisation d'un test de fatigue sur banc d'essais. Une fois cette durée de vie établie, on peut anticiper le remplacement du bien en fin de vie en prenant une marge de sécurité. Ce type d'approche permet ainsi de réduire le risque de défaillance par un entretien régulier. Toutefois, cette stratégie a un coût.

Bien que le risque d'indisponibilité imprévue diminue, la maintenance peut nécessiter un arrêt programmé du bien, ce qui réduit sa disponibilité globale. Il faut également mobiliser des ressources, humaines et matérielles, pour effectuer ces opérations de maintenances programmées ce qui peut représenter un coût non négligeable. Enfin, la marge de sécurité prise dans l'évaluation de la durée de vie du bien amène à se

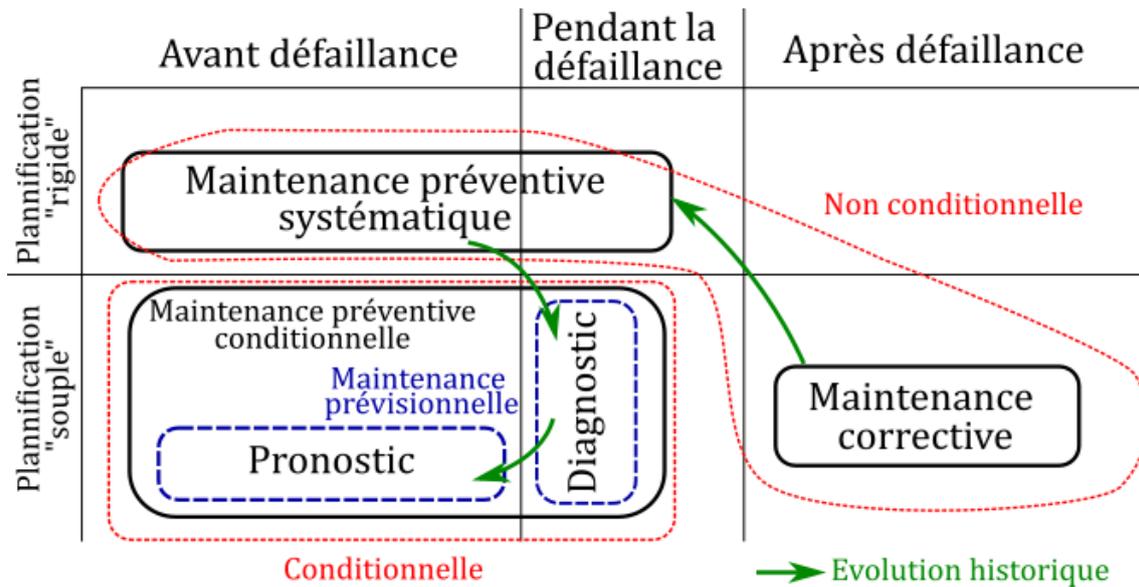
débarrasser d'un équipement fonctionnel pour réduire le risque de panne, ce qui diminue son rendement.

De nos jours, l'activité de maintenance est soumise à des exigences de plus en plus fortes. Ces exigences ont à la fois trait à la fiabilité de systèmes toujours plus complexes et donc difficiles à maintenir, mais également à la réduction des coûts qui est devenue un enjeu majeur sur toutes les activités non porteuses de valeur ajoutée. Le besoin d'une maintenance conditionnelle a vu le jour afin de réduire le coût de la maintenance tout en conservant ou améliorant la fiabilité des biens. Elle inclut les stratégies de maintenance préventives relatives à l'évaluation des conditions physiques, leur analyse et la mise en place des éventuelles actions de maintenance qui en découlent.

L'intérêt de telles approches réside dans le suivi en temps réel du fonctionnement d'un bien dans ses conditions réelles d'utilisation. Ce type de solutions permet ainsi d'aller plus loin dans l'amélioration de la disponibilité et de la fiabilité par la détection d'anomalies de fonctionnement. La corrélation entre les anomalies détectées et les défaillances connues pouvant affecter le bien permet de prédire une future panne et d'alerter les équipes de maintenance. Cette surveillance continue permet ainsi de réduire ou supprimer les actions de maintenance inutiles tout en optimisant la durée de vie de l'équipement. De plus, l'observation des données de fonctionnement en temps réel permet de remonter à la cause originelle d'une défaillance (LEE et al. 2014).

En s'appuyant sur la figure 2.1, on remarque que cette démarche peut encore se diviser en deux types d'approches selon leur capacité, ou incapacité, à évaluer la dégradation du bien considéré et à estimer sa durée de vie résiduelle, communément appelée RUL (Remaining Useful Life). Cette estimation du RUL correspond ainsi à la durée entre la détection de l'anomalie et l'occurrence de la défaillance. Ces méthodes de maintenance dites prévisionnelles permettent ainsi d'identifier la fenêtre temporelle optimale pour planifier l'intervention tout en maximisant la durée de vie du bien. L'essor de ces approches basées sur les données a ainsi été rendu possible à la fin du 20ème siècle par la prolifération des sources de données, comme les capteurs, et la multiplication de la puissance de calcul des ordinateurs entraînant par ricochet un fort développement des techniques de traitement de l'information. Elles font actuellement l'objet d'un intérêt grandissant à la fois de la part des acteurs industriels et de la communauté scientifique ce qui aboutit à l'émergence de nombreux travaux de recherche (LEI et al. 2018).

Ainsi, on distingue trois grandes familles de maintenance : corrective, préventive systématique et préventive conditionnelle. La figure 2.2 reprend la chronologie d'apparition de ces différentes familles tout en distinguant leurs phases d'application (i.e : avant pendant ou après la défaillance). La maintenance corrective intervient, en effet, après la défaillance de manière imprévue, d'où une planification « *souple* ». Elle est historiquement la plus ancienne. Ensuite, la maintenance préventive systématique a fait son apparition. Son objectif est de prévenir les défaillances par la mise en place d'actions de maintenance périodiques. La périodicité des actions de maintenance traduit une planification rigide de cette stratégie. Pour pallier aux limites de la maintenance préventive systématique, la maintenance préventive conditionnelle a



**Figure 2.2** – Classification et évolution historique de la maintenance (HANACHI et al. 2018)

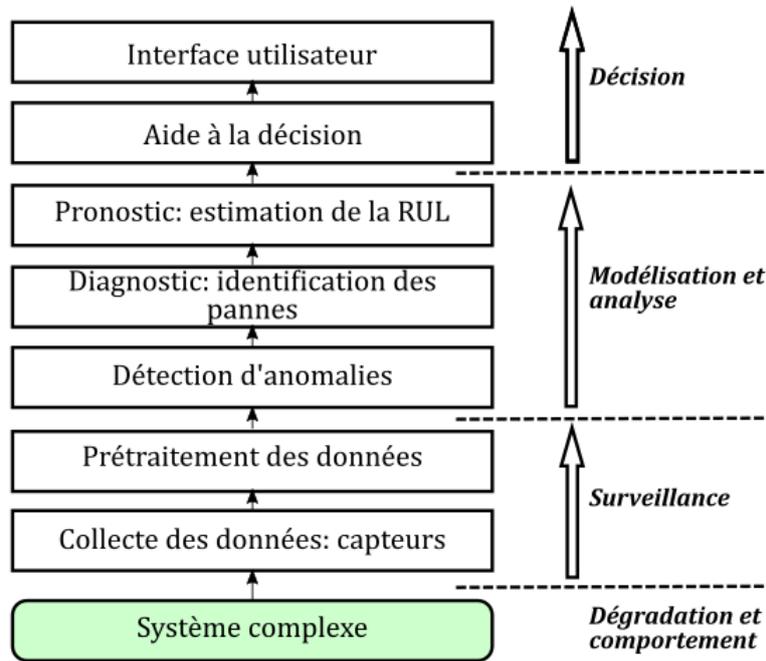
été développée. La première étape est le diagnostic, qui vise à identifier une panne à partir d'anomalies, donc après les premiers signes de défaillance. Enfin, le pronostic estime la RUL du bien surveillé. Cette estimation est donc réalisée avant la défaillance. Les stratégies qui intègrent à la fois le diagnostic et le pronostic font partie de la catégorie des maintenances prévisionnelles.

Afin de répondre à des exigences grandissantes de fiabilité et de réduction des coûts, une approche de maintenance prévisionnelle est proposée dans ce manuscrit. Cette approche a pour but d'analyser des données de surveillance afin de produire une identification des défaillances passées ou futures et un pronostic de la durée de vie restante du système surveillé en temps réel. Ce type d'approche fait partie de ce que l'on appelle le pronostic et la gestion de la santé, ou PHM (Pronostic and Health Management) VICHARE et al. (2006).

## 2.2 Le pronostic et la gestion de la santé - PHM

### 2.2.1 Présentation du PHM

Le pronostic et la gestion de la santé (PHM : Prognostic and Health Management) est une méthode permettant d'évaluer la fiabilité d'un bien dans les conditions réelles de son cycle de vie afin de prévenir l'occurrence de défaillances et de limiter les risques auxquels il est soumis (VICHARE et al. 2006). Elle se base sur une surveillance directe du système pour fournir une vision complète, en temps réel, de son état de santé afin d'anticiper sa dégradation et de cibler les besoins de maintenance (BLANCHARD



**Figure 2.3** – Les 7 étapes du PHM (JOUIN et al. 2013)

et al. 1995 ; LEE et al. 2014). Cette approche se décline en sept étapes principales comme décrit dans la figure 2.3 (JOUIN et al. 2013) :

1. l'acquisition des données par l'intermédiaire de capteurs
2. le prétraitement des données
3. la détection d'anomalie(s) à partir de ces données prétraitées
4. l'identification de la panne à partir d'une ou plusieurs anomalies détectées
5. l'estimation de la durée de vie utile restante (i.e. RUL) en lien avec la panne identifiée
6. le transfert des informations générées vers un outil d'aide à la décision
7. l'affichage des résultats au travers d'une Interface Homme-Machine (IHM)

On peut remarquer que cette démarche est similaire à celle d'un diagnostic médical. Ainsi, une analogie entre les deux démarches sera détaillée dans la description des 7 étapes.

Afin de maintenir un système en temps réel, la première étape consiste à identifier et collecter les variables à surveiller. Cette étape est particulièrement cruciale car elle impacte fortement la qualité de toute analyse ultérieure. Le choix des paramètres observés doit être pertinent au regard de l'étude réalisée. La fréquence et la précision des mesures sont également identifiées comme des facteurs limitant la performance de la détection d'anomalie et par conséquent, de l'identification des pannes. Si l'on reprend l'analogie médicale, cette étape correspondrait à la collecte des constantes du patient.

La seconde étape est le prétraitement des données dont l'objectif est de trier et transformer les données collectées pour générer des variables pertinentes pour l'analyse à effectuer. Cette étape permet en particulier de détecter et supprimer les mesures manquantes ou aberrantes pour ne pas fausser l'analyse. Des formules mathématiques peuvent également être appliquées sur ces données en préparation de l'analyse. On peut par exemple calculer une moyenne ou un différentiel de pression qui correspondrait à une données d'entrée de l'analyse à effectuer. Lors d'un diagnostic médical, ce processus est réalisé par le médecin, qui va juger de la pertinence et de la fiabilité des valeurs mesurées.

La troisième étape consiste à analyser l'état de santé du bien maintenu. Cette analyse se base sur la détection d'anomalies observées parmi les données collectées et prétraitées. Ces anomalies peuvent être assimilées à des symptômes présentés par le bien.

Une fois ces symptômes détectés, un diagnostic doit être réalisé afin d'identifier la panne affectant le bien. Comme pour la détection d'anomalies, deux approches principales peuvent être utilisées pour effectuer ce diagnostic. La première consiste à se baser sur des règles fournies par des experts ou renseignées dans la documentation technique. On parle alors de système à base de règles. La deuxième approche revient à utiliser des algorithmes d'apprentissage automatique.

Lorsque la panne est identifiée, une estimation de la durée de vie restante du bien peut être effectuée. Dans certains cas, cette estimation peut être réalisée tout au long de la durée de vie du bien, sans attendre la détection d'une anomalie. Toutefois, la connaissance préalable de la défaillance à venir est une information importante pour l'estimation de la RUL. Cette étape est connue sous le nom de pronostic.

Quand la panne et la RUL sont identifiées, ces informations doivent alors être transmises à un outil d'aide à la décision. Cet outil peut remplir plusieurs fonctions. Il peut par exemple estimer la gravité de la situation, prévoir des scénarios ou encore lancer des commandes de matériel par anticipation. Toutefois, il est considéré ici que la décision finale revient à l'utilisateur. L'outil d'aide à la décision n'a donc ici qu'un rôle informatif. Par analogie avec le diagnostic médical, l'outil d'aide à la décision est assimilable au guide national de prescription de médicaments. Bien qu'il puisse apporter des informations utiles, le médecin est en droit de le suivre ou non pour faire sa prescription.

Le résultat de son analyse est alors transmis au décideur à travers une IHM.

D'après la figure 2.3, ces sept étapes peuvent se regrouper en 3 phases :

- la surveillance qui sera présentée en 2.2.2
- l'analyse qui peut être distinguée en 2 parties :
  - la détection d'anomalies et l'identification et des pannes détaillé dans la section 2.2.3
  - le pronostic détaillé dans la section 2.2.4.
- l'aide à la décision discutée en 2.2.5

## 2.2.2 Surveillance

### Nature des données collectées

La première partie d'une approche PHM, comme pour toute maintenance conditionnelle, est la phase de collecte des données. De manière générale, l'alimentation d'un système d'information (SI) en données peut se faire soit manuellement, soit par l'intermédiaire de ce que (AALST 2014) définit comme l'internet des événements. Celui-ci est composé de :

- L'internet des objets, constitué d'un réseau de capteurs qui génèrent en continu un volume important de données (PERERA et al. 2014)
- L'internet des personnes faisant référence aux données générées par les réseaux sociaux en ligne sous des formats divers comme le texte, l'image ou la vidéo et connectant des centaines de personnes à travers le monde (GUILLE et al. 2013)
- L'internet des lieux composé de données ou informations de géolocalisation
- L'internet du contenu qui contient tout le reste des données accessibles.

Dans le cadre de la maintenance conditionnelle, l'essentiel des données recueillies provient de l'internet des objets. Dans tout système de maintenance conditionnelle, la collecte des données est une étape cruciale car la performance du diagnostic repose sur leur analyse et leur exploitation (CAMPOS 2009). Les données collectées se présentent sous la forme d'événements ou de données de surveillance (JARDINE et al. 2006).

Les données de surveillance correspondent aux mesures de paramètres physiques effectuées sur le système afin d'évaluer son état. Elles sont habituellement générées au moyen de réseaux de capteurs et peuvent se présenter sous des formes diverses. On peut cependant les regrouper en trois types (JARDINE et al. 2006) :

- Ondulatoire : sur une période définie, les données collectées sont des séries temporelles comme des vibrations ou des signaux acoustiques.
- Multidimensionnel : sur une période définie, les données collectées sont des variables multidimensionnelles. Les exemples les plus courants sont les images à rayons X, infrarouges ou visuelles.
- Valeur : sur une période définie, les données collectées se présentent sous la forme d'une valeur simple. Les informations de températures, pression ou d'humidité sont des exemples appartenant à cette catégorie.

Les données issues de cette dernière catégorie sont généralement plus simple à analyser et à interpréter que les deux autres. En particulier, les données ondulatoires sont souvent accompagnées de bruits susceptibles de fausser leur traitement et nécessitent une attention particulière.

La collecte des données de surveillance se traduit ainsi par la mise en place de réseaux de capteurs de plus en plus complexes. La multiplication des données émises engendre des problématiques généralement classées suivant cinq caractéristiques (SICULAR 2013) :

- Volume : la taille conséquente des jeux de données nécessite un processus de stockage et des traitements adaptés.
- Vitesse : la rapidité de transfert et de mise à jour des données, issues de sources multiples, requiert une intégration adaptée des nouvelles entrées dans les collections existantes.
- Variété : la nature hétérogène des données récoltées contribue à la complexité des modèles. Elles peuvent, en effet, se présenter sous la forme de textes, images ou encore vidéos.
- Véracité : la fiabilité des sources de données impacte directement celle de l'analyse. Cette dimension est particulièrement déterminante dans le traitement d'informations fournies par l'homme comme dans le cas des réseaux sociaux. Cependant les capteurs peuvent également se montrer défaillants. Il est donc nécessaire d'être critique vis-à-vis des données collectées.
- Valeur : la valeur fait référence à l'identification et à l'extraction d'informations pertinentes dans des jeux de données larges et complexes.

Lorsque plusieurs de ces problématiques apparaissent, on dit alors qu'on évolue dans un contexte Big Data.

Dans le secteur aéronautique, la complexité des systèmes présents sur les aéronefs requiert un réseau de capteurs conséquent pour s'assurer de leur bon fonctionnement. Ces capteurs génèrent un flux constant de données hétérogènes qu'il convient de collecter et d'analyser.

Ce contexte répond ainsi aux cinq caractéristiques des problèmes de type Big Data.

La mise en place d'une architecture de type PHM doit donc intégrer des composants répondant à ces problématiques. FERTIER (2018) propose une classification des méthodes de collecte dédiées à la détection d'évènements en temps réel. Cette classification a pour but de décrire une situation courante à partir de données provenant de l'internet de personnes (réseaux sociaux), l'internet des objets ou l'internet des lieux en présentes dans la littérature. Ces méthodes de collecte se distinguent en quatre catégories selon les types de données reçues et d'évènements à détecter :

- Les méthodes 1 à 1, où un type particulier de données connues est analysé pour détecter des types d'anomalies (Q. CHEN et al. 2014 ; HU et al. 2018), ou isoler des groupes de données qui présentent des caractéristiques particulières (CARRINO et al. 2016 ; LAYLAVI et al. 2017 ; Z. XU et al. 2017)
- Les méthodes x à 1, où des données provenant des plusieurs sources sont analysées pour détecter un évènement particulier, comme des bouchons (S. WANG et al. 2017) ou des glissement de terrain (MUSAEV et al. 2015)
- Les méthodes 1 à x, où un type de données connues est analysé pour détecter plusieurs types d'évènements. L'exemple donné est celui de l'analyse des tweets pour la détection d'un tremblement de terre ou d'un incendie (MAZHAR RATHORE et al. 2017)

- Les méthodes x à y, où des données provenant de plusieurs sources permettent de détecter plusieurs types d'évènements.

Dans le cas de l'approche PHM, la collecte de données est de type x à y puisque plusieurs sources de données peuvent être nécessaires à la création d'évènements d'anomalies. De plus, une même source peut entraîner la génération de plusieurs évènements d'anomalies distincts.

## Prétraitement des données

Une fois les données brutes collectées, il est souvent nécessaire d'effectuer un premier traitement. Comme l'indique FAMILI et al. (1997), un prétraitement des données permet de répondre à deux challenges.

Le premier challenge est lié à l'existence de défauts dans les données collectées. Ces défauts peuvent être classés en trois types : l'excès de données, le manque de données et la dissociation des données. La figure 2.4 présente reprend cette classification en proposant des exemples pour chacune des catégories.

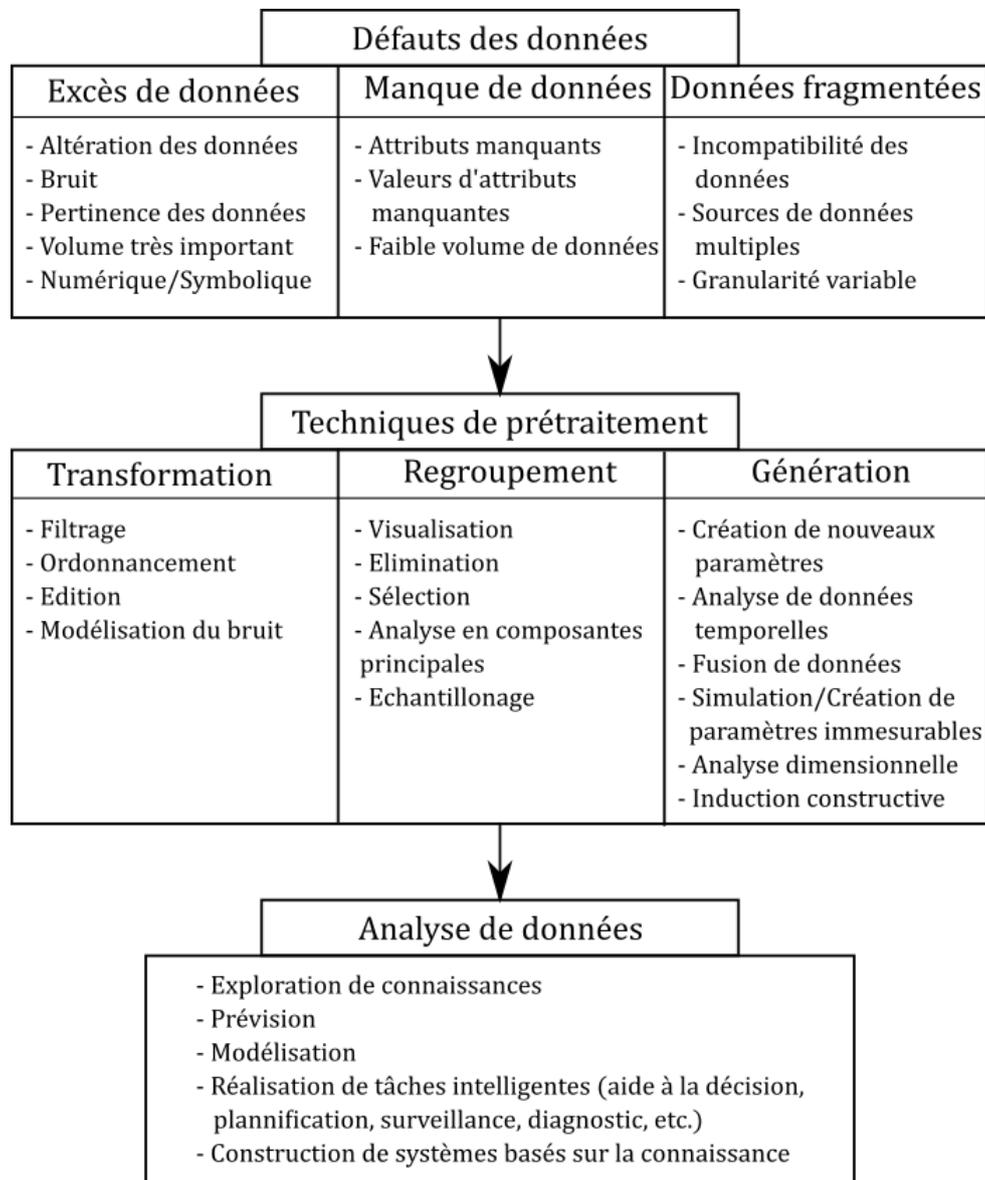
L'excès de données, en particulier quand elles proviennent de capteurs, présente des risques importants d'altération liés au bruit ou à une mauvaise captation ou transmission des données. Un grand volume de données brutes rend également plus difficile la recherche et l'extraction des paramètres ou caractéristiques portant une valeur utile pour l'analyse (J. F. DAVIS et al. 1999). Le prétraitement peut alors jouer un rôle de filtre afin de ne conserver que la donnée intéressante au regard de l'utilisation recherchée.

Au contraire, un volume de données trop faible peut générer un manque de ressources pour l'analyse et ainsi affecter sa performance. Ce cas de figure est particulièrement gênant lorsque les données sont utilisées pour l'entraînement d'algorithmes d'apprentissage car la performance de prédiction augmente avec le nombre d'exemples et de paramètres pertinents disponibles. Un faible volume de données peut également être dû à l'absence ponctuelle de valeur dans les relevés. Dans ce cas, la perte d'un nombre important de valeurs, contenant potentiellement des informations à forte valeur ajoutée, rend l'interprétation difficile (FAMILI et al. 1997).

Le problème de dissociation des données est le plus souvent lié à la diversité des sources. Cette diversité peut générer des obstacles au croisement et à la combinaison des données. Parmi ces obstacles, on peut citer en exemple le manque de données, comme une date ou un identifiant commun, ou encore gestion de niveaux de granularité d'information différents. Cela crée un manque de cohérence entre les jeux de données qui pénalise fortement l'analyse (FAMILI et al. 1997).

D'après la figure 2.4, trois types de solutions peuvent être utilisés pour corriger ces défauts : les techniques de transformation de données, les techniques de regroupement permettant de mieux appréhender leur nature et les techniques de génération permettant d'en créer de nouvelles données.

Le second challenge est lié à la compréhension de la nature des variables collectées et à la mise en place de transformations permettant de préparer et d'améliorer l'efficacité



**Figure 2.4** – Défauts et techniques de prétraitement des données (FAMILI et al. 1997)

de l'analyse à réaliser. Dans le cas de l'apprentissage artificiel, le prétraitement peut ainsi permettre de définir le mode de représentation des données en modifiant la structure des données reçues, et par extension, les concepts à enseigner. Les techniques de prétraitement peuvent également être utilisées pour présenter le résultat de l'analyse et faciliter sa compréhension et son application dans le monde réel (FAMILI et al. 1997). Dans le cas de la modélisation, elle peut empêcher la prise en compte de données parasites, apparaissant lors de la captation et/ou la transmission de la donnée, dans la définition du modèle. Les modèles ainsi créés sont alors plus robustes, même si cela n'affecte pas forcément la qualité de l'analyse.

Le prétraitement des données est ainsi défini comme l'ensemble des actions menées avant le démarrage de l'analyse et qui s'apparente à une transformation  $T$  des données réelles, sous forme de vecteur  $X_{ik}$ , en un nouveau jeu de données  $Y_{ij}$  tel que :

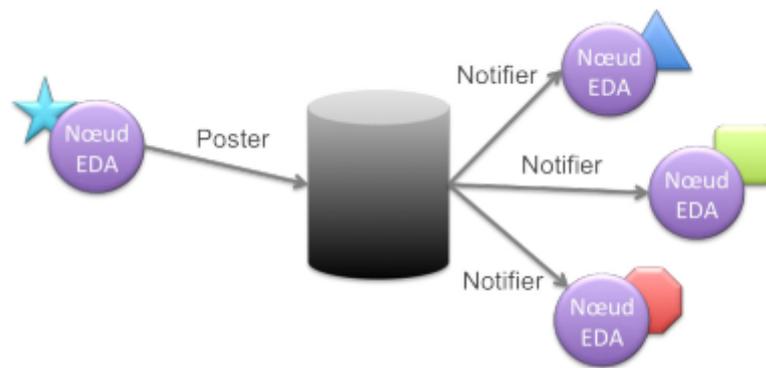
$Y_{ij} = T(X_{ik})$  avec  $i = 1, \dots, n$  où  $n =$  nombre d'objets,  $j = 1, \dots, m$  où  $m =$  nombre de paramètres à la fin du prétraitement,  $k = 1, \dots, I$  où  $I =$  nombre de paramètres avant le prétraitement, et en général,  $m \neq I$ .

$T$  a pour objectif de conserver l'information utile présente dans  $X_{ik}$  et de rendre  $Y_{ij}$  plus exploitable que  $X_{ik}$  en éliminant au moins un défaut de  $X_{ik}$  (FAMILI et al. 1997).

Les techniques présentées dans la figure 2.4 ne présentent cependant qu'une partie des techniques permettant de faire du prétraitement de données. Parmi les outils les plus courants, on trouve notamment les techniques d'analyse exploratoire de données définies par TUKEY (1977). Ces techniques servent par ailleurs d'outils à un grand nombre de techniques comme la visualisation de données, le filtrage ou la création de nouveaux paramètres. Le principe de base de l'analyse exploratoire de données est de caractériser une population à partir de l'observation d'un échantillon et d'évaluer la précision de cette généralisation. (J. SELTMAN 2018) indique que parmi les utilisations principales figurent :

- La détection de valeurs aberrantes (appelées par la suite outliers) dans le jeu de données
- La vérification d'hypothèses
- Une première sélection de modèles pertinents
- la recherche de corrélations entre des variables
- l'évaluation de la nature et du niveau des corrélations entre ces variables

Ainsi, le prétraitement des données est une étape indispensable pour assurer la qualité des données utilisées dans l'étape d'analyse. Afin de réaliser ce prétraitement tout en répondant aux problématiques de volumétrie, vitesse et variété des données, une architecture dirigée par les événements semble appropriée car ce type de solution est conçue pour gérer et traiter des flux de données sous forme d'évènements simples ou complexes.



**Figure 2.5** – Principe du mécanisme de Publication/Souscription (MARÉCHAUX 2006)

## Architecture dirigée par les évènements

D'après TAYLOR et al. (2009), un évènement désigne un changement d'état méritant l'attention et la surveillance d'un système. Il peut provenir d'une source interne ou externe au système et induire un problème, un risque, une opportunité, un seuil ou une déviation (MICHELSON 2006). Un évènement se décompose en deux parties :

- L'entête : ensemble des éléments décrivant les caractéristiques de l'occurrence comme son identifiant, son type, son nom, son horodatage ou encore sa source.
- Le corps : ensemble des informations décrivant la situation suite à laquelle l'évènement a été créé.

On dit qu'une architecture est dirigée par les évènements (Event-Driven Architecture : EDA) lorsqu'elle est capable de les détecter et d'y réagir (TAYLOR et al. 2009). Cette réaction peut se traduire par l'appel à un service, le déclenchement d'un processus métier ou encore la recherche d'informations supplémentaires. Dans ces architectures, certains composants communiquent et sont dirigés par les évènements. Les architectures EDA ont été popularisées avec l'apparition de standards pour les places de marchés et les systèmes de vente aux enchères. De par sa nature, une architecture dirigée par les évènements est couplée de façon extrêmement lâche et elle est hautement distribuée (JOSUTTIS 2007). La communication entre les processus est réalisée par échange de messages. Cette communication est initiée par un évènement. Ce déclencheur correspond typiquement à une occurrence métier quelconque. Tous les abonnés au type d'évènement correspondant à l'instance émise sont alors notifiés et donc activés. C'est ce que l'on appelle le mécanisme Publish/Subscribe (Publication/Souscription) de l'EDA, représenté dans la Figure 2.5.

Il existe trois principaux types de traitements au sein d'une EDA : le traitement d'évènements simples, complexes ou de flux (MICHELSON 2006). Il est important de noter que ces différents traitements peuvent être utilisés conjointement dans une même architecture.

Le traitement d'évènements simples est principalement utilisé pour déclencher une action directement après réception d'un évènement.

**Table 2.1** – Principales caractéristiques de l'EDA (BARTHE 2013)

Propriété	Description
Couplage lâche	Les producteurs (sources) d'évènements ne sont pas au courant de l'existence de leurs abonnés.
Communication n-m	Le mécanisme Publication/Souscription permet qu'un seul évènement spécifique puisse avoir un impact sur plusieurs abonnés
Déclencheur basé sur les évènements asynchrones	Le flux de contrôle est déterminé par le destinataire, sur la base de l'évènement reçu.

Le traitement d'évènements complexes consiste à évaluer une convergence d'évènements avant de déclencher une action. Ces évènements peuvent être de types différents et apparaître sur des fenêtres temporelles.

Dans le traitement de flux, des évènements significatifs et ordinaires sont observés. Le flux est alors "filtré" : les évènements significatifs sont détectés et traités tandis que les évènements ordinaires sont ignorés.

Ce type d'architecture présente un couplage faible entre ses composants. Le créateur d'un évènement n'est pas nécessairement informé de son utilisation, des traitements dont il est l'objet et de ses consommateurs finaux (MICHELSON 2006). Les consommateurs de l'évènement s'abonnent aux types d'évènements qui les intéressent et peuvent être des évènements dérivés. Les consommateurs n'ont aucun moyen de savoir si l'évènement qu'ils consomment est un évènement « original » simplement filtré ou s'il est dérivé. La traçabilité d'un évènement à travers un réseau multi évènements dynamique peut être difficile dans ce contexte. Ainsi, les architectures dirigées par les évènements sont mieux utilisées pour les flux asynchrones de travail et d'information. Elles ont cependant l'avantage de faciliter la séparation entre communication et calcul tout en permettant une intégration simplifiée de composants autonomes et hétérogènes dans des systèmes complexes, évolutifs et extensibles (FIEGE et al. 2003). Cette faculté, couplée aux trois modes de traitement des évènements, rend les architectures EDA particulièrement adaptées à l'observation de flux de données de surveillance pour la détection d'anomalies.

Les propriétés fondamentales de l'EDA sont résumées dans la Table 2.1.

Une fois les anomalies détectées, l'étape suivante consiste à les analyser pour identifier une défaillance potentielle et/ou estimer la durée du vie résiduelle du système surveillé.

### 2.2.3 Détection d'anomalies et identification des pannes

L'un des enjeux dans un contexte de type Big Data est d'identifier et d'apporter de la valeur aux données collectées. D'après BLOCH et al. (1983), la plupart des

défaillances d'un système sont précédées de signes, conditions ou indications avant coureurs. Dans le cadre du PHM, une première valeur consiste ainsi à détecter les anomalies de fonctionnement des systèmes surveillés dans les conditions réelles de leur cycle de vie. Les anomalies détectées permettent alors de prévoir l'apparition d'une défaillance ou d'identifier une panne existante. Dans ce dernier cas, on parle alors de diagnostic. L'identification d'une panne existante ou à venir constitue ainsi une deuxième valeur.

Les approches permettant d'apporter ces deux valeurs aux données collectées reposent principalement sur la notion d'intelligence artificielle (IA). Cette notion ne possède pas de définition faisant l'unanimité scientifique. D'après FEIGENBAUM et al. (1981), l'IA désigne la partie de l'informatique dédiée à la conception de programmes intelligents, c'est-à-dire présentant les caractéristiques associées à l'intelligence humaine comme la compréhension du langage, l'apprentissage, le raisonnement ou encore la capacité à résoudre des problèmes.

D'après RUSSELL et al. (1995), l'IA désigne quatre types de systèmes :

- les systèmes qui pensent comme les humains dont le but est d'automatiser les activités associées à la pensée humaine comme la prise de décision, la résolution de problèmes ou l'apprentissage (BELLMAN 1978)
- les systèmes qui pensent rationnellement et appartiennent au domaine d'étude des calculs permettant la perception, le raisonnement et l'action (WINSTON 1992)
- les systèmes qui agissent comme des humains et possèdent ainsi la faculté d'effectuer des tâches que, jusqu'alors, l'homme était le seul ou le plus apte à réaliser (RICH et al. 1991)
- les systèmes qui agissent rationnellement afin d'expliquer et d'imiter un comportement intelligent au moyen de processus informatiques (SCHALKOFF 1990)

Aujourd'hui, on peut considérer que l'IA repose sur trois domaines : la représentation des connaissances, le raisonnement à partir de connaissances capitalisées et l'apprentissage à partir d'exemples passés.

Concernant la représentation des connaissances, R. DAVIS et al. (1993) décrit ce domaine à travers cinq caractéristiques. Premièrement, une représentation de connaissances est un substitut du système réel contenant, par essence, des approximations. Deuxièmement, cette représentation est toujours réalisée suivant un point de vue donné. Elle est également supportée, explicitement ou implicitement, par une logique déductive régissant la nature des entités et relations de la représentation. Enfin, la représentation de connaissances remplit le rôle de support à l'expression humaine mais aussi à l'implémentation informatique.

Le domaine du raisonnement a pour but de reproduire le raisonnement logique d'un expert humain. Les approches de raisonnement principales sont les systèmes à base de règles et, en particuliers les systèmes experts. Elles reposent sur la formalisation et l'exploitation de connaissances expertes capitalisées.

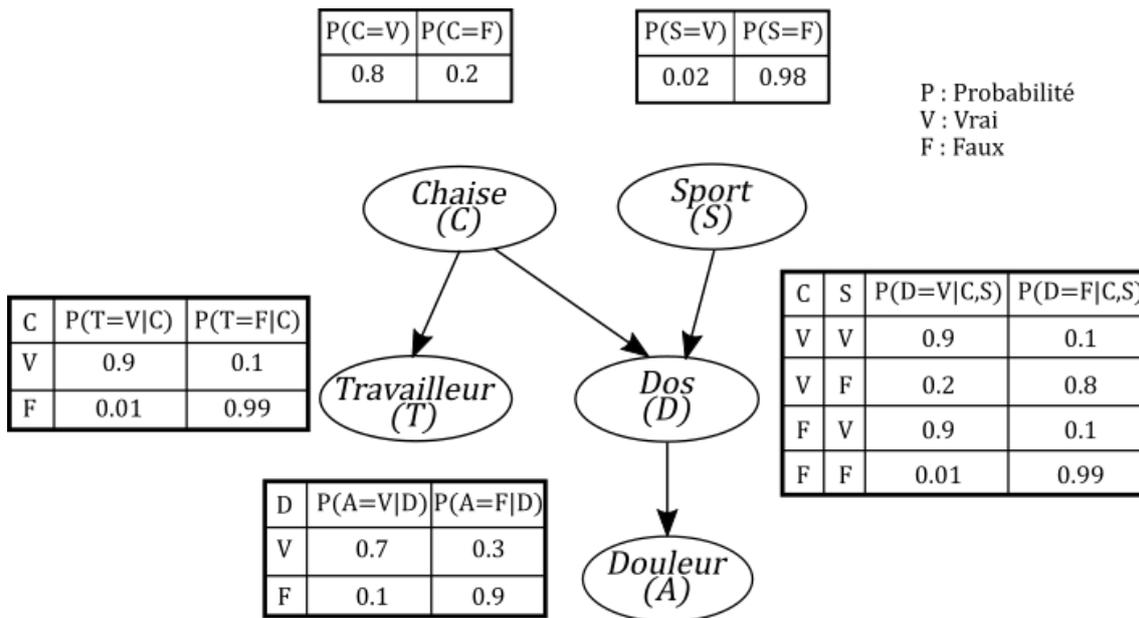


Figure 2.6 – Exemple de réseau bayésien (BEN-GAL 2008)

Le domaine de l'apprentissage vise à rendre un programme capable de résoudre un problème pour lequel il n'a pas été spécifiquement conçu. Les algorithmes d'apprentissage se basent sur des méthodes d'analyse statistiques afin d'imiter le mécanisme d'apprentissage humain basé sur l'exemple. Sur la base d'une collection d'exemples historiques, ces algorithmes permettent ainsi de prédire une valeur de sortie avec une certaine probabilité.

Bien que ces trois domaines offrent une classification globale des algorithmes d'IA, ils ne sont pas hermétiques en eux. Les réseaux bayésiens en sont un parfait exemple. Ils appartiennent à la famille des modèles graphiques probabilistes. Ces structures graphiques permettent de représenter de la connaissance dans un domaine défini. Chaque noeud du graphe désigne un fait ou une variable tandis que les liens entre ces noeuds constituent des relations probabilistes. Ces relations probabilistes sont généralement calculées au moyen de méthodes statistiques et peuvent être améliorées au fil du temps (BEN-GAL 2008). Les réseaux bayésiens font ainsi appel aux trois domaines de l'IA : la représentation des connaissances dans un graphe, le raisonnement sur la base de ces connaissances pour arriver à une conclusion et l'apprentissage permettant d'améliorer les performances à partir d'exemples.

La figure 2.6 illustre un exemple de réseau bayésien extrait de (BEN-GAL 2008).

Cet exemple considère le cas d'une personne pouvant souffrir d'une blessure au dos, évènement représenté par la variable D. Cette blessure peut provoquer une douleur (évènement A). La blessure au dos (D) peut être causée par une activité sportive (évènement S) ou suite à l'installation de nouvelles chaises de bureau inconfortables (évènement C). Dans ce dernier cas, il est probable qu'un collègue souffre également d'une douleur au dos et le signale (évènement T). Toutes les variables sont binaires,

autrement dit vraies (V) ou fausses (F). Les probabilités conditionnelles de chaque évènement sont présentées dans les tableaux situés à côté de chaque noeud.

A partir de ce graphe, on pourrait par exemple déterminer la cause probable d'un mal de dos ou l'influence de l'installation des chaises sur la santé des employés.

La représentation des connaissances est visible à travers le graphe. L'apprentissage est effectué par la mise à jour des tableaux de probabilité au fil du temps. Le raisonnement regroupe l'ensemble des règles et processus permettant d'aboutir à une conclusion sur la base des informations disponibles.

Il convient de remarquer que le raisonnement est fortement lié à la représentation des connaissances. Dans les faits, la majorité des solutions de raisonnement s'appuie sur une représentation et une structuration préliminaire du domaine considéré.

Dans le cadre de la détection d'anomalies et de la détection des pannes, les variables que l'on cherche à déterminer sont discrètes. On essaie ainsi de résoudre un problème de classification. Pour cela, deux approches possibles sont généralement utilisées : une approche basée sur l'exploitation de données historiques d'évènements passés et une approche basée sur l'utilisation d'une expertise humaine capitalisée et formalisée (ISO 2012).

## Approche d'apprentissage basée sur les données

L'approche d'apprentissage basée sur les données consiste à exploiter l'historique des évènements passés afin d'identifier des tendances ou profils et d'en déduire des règles. Ces règles peuvent alors être appliquées à de nouveaux cas pour détecter des anomalies ou des pannes.

Cette approche est réalisée au moyen de techniques d'apprentissage artificiel. La notion d'apprentissage est définie dans le dictionnaire comme un ensemble des processus de mémorisation mis en œuvre par l'animal ou l'homme pour élaborer ou modifier les schémas comportementaux spécifiques sous l'influence de son environnement et de son expérience. Par extension, l'apprentissage artificiel regroupe "toutes les techniques par lesquelles un programme est capable d'évoluer au cours du temps pour résoudre des problèmes complexes" (MAZAC 2015).

Les algorithmes d'apprentissage ont ainsi pour particularité d'améliorer leurs performances de résolution au fil du temps. Ils appartiennent au domaine d'étude permettant à un ordinateur d'apprendre sans être explicitement programmé en reproduisant le processus d'apprentissage de l'homme pour résoudre des problèmes de classification ou de régression (SIMON 2013).

Mathématiquement, un algorithme d'apprentissage répond à la problématique suivante : soient  $x$  un vecteur de paramètres fourni en entrée et  $\hat{y}$  la variable à prédire, on cherche à définir la fonction  $f$  telle que  $f(x)=\hat{y}$ .

Les algorithmes de classification se distinguent des algorithmes de régression par la nature de la variable de sortie,  $\hat{y}$ , à déterminer. Si cette variable ne peut prendre qu'un nombre fini de valeurs, autrement dit, si  $\hat{y}$  est discrète, on parle d'un problème

de classification. Dans le cas contraire, si  $\hat{y}$  est continue, on parle alors d'un problème de régression.

Dans le cas de la détection d'anomalies et d'identification de pannes, il n'existe qu'un nombre fini de valeurs possibles appelées classes. C'est pourquoi on adresse un problème de classification.

HANACHI et al. (2018) rappelle que deux types d'apprentissages peuvent être utilisés dans le cadre de la maintenance : le supervisé et le non supervisé. L'apprentissage supervisé repose sur la présence d'un jeu d'entraînement contenant un grand nombre d'exemples avec, pour chacun, la valeur de sortie  $y$  associée. Il est important de distinguer ici  $y$  et  $\hat{y}$ . Tandis que  $y$  est la valeur réelle connue et fournie par l'utilisateur,  $\hat{y}$  est la valeur prédite par le modèle. La performance d'un apprentissage supervisé peut ainsi être évaluée à travers l'écart entre  $y$  et  $\hat{y}$ . Plus cette distance est faible, plus le modèle est performant. Un autre critère de performance concerne la faculté du modèle à s'adapter à de nouveaux exemples ne faisant pas partie du jeu d'entraînement.

Dans le contexte de la maintenance, les exemples sont généralement des relevés de mesures contenant plusieurs paramètres (e.g. pression, température) associés à une anomalie ou panne détectée.

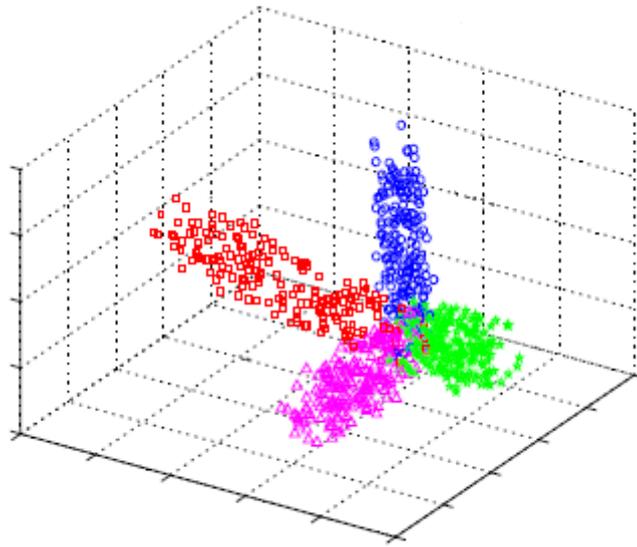
Dans l'apprentissage non supervisé, la valeur de sortie  $y$  n'est pas fournie dans le jeu d'entraînement. L'algorithme a alors pour objectif d'identifier automatiquement des groupes, formes ou tendances dans le jeu d'entraînement. Cette identification se base généralement sur l'analyse de la similarité géométrique entre les variables dans un espace multidimensionnel (PRESTON et al. 1972).

La figure 2.7 illustre un résultat de classification par apprentissage non supervisé extrait de HANACHI et al. (2018). Cette figure présente une classification de défaillances potentielles à partir de relevés de mesures effectués sur des turbines à gaz. Chaque couleur représente un type de défaillance.

Pour résoudre un problème de détection d'anomalies ou d'identification des pannes, il est préférable d'utiliser un historique d'anomalies avérées afin d'entraîner un algorithme d'apprentissage de classification supervisé. Le problème principal d'une approche non supervisée est que l'on ne sait pas à quelle anomalie ou panne correspondent les groupes identifiés. Par ailleurs, la connaissance des anomalies ou pannes dans le jeu d'entraînement permet une meilleure identification des tendances associées à chaque classe.

LEE et al. (2014) présente une revue comparative des algorithmes d'apprentissage utilisés dans divers travaux sur le PHM. En particulier, les algorithmes de réseaux de neurones (MACMANN et al. 2016), logiques floues (YAN et al. 2018), modèles de Markov cachés (ADITIYA et al. 2017) et SVM (Z. WANG et al. 2012) font partie des plus courants (B. SUN et al. 2010).

Ce type d'approche présente l'avantage de ne pas nécessiter une connaissance experte du système et de permettre une amélioration de la capacité de détection au fil du temps en intégrant de nouveaux cas au jeu d'entraînement. Toutefois, la principale limite réside dans la taille du jeu d'entraînement à disposition et la pertinence des paramètres le constituant. Ces facteurs sont, en effet, déterminants dans la qualité



**Figure 2.7** – Illustration de détection de pattern par apprentissage non supervisé extrait de (HANACHI et al. 2018)

de l'apprentissage. De plus, dans le cas des réseaux de neurones, il est très difficile de déterminer les phénomènes physiques pris en compte par le modèle entraîné pour réaliser la détection (JARDINE et al. 2006).

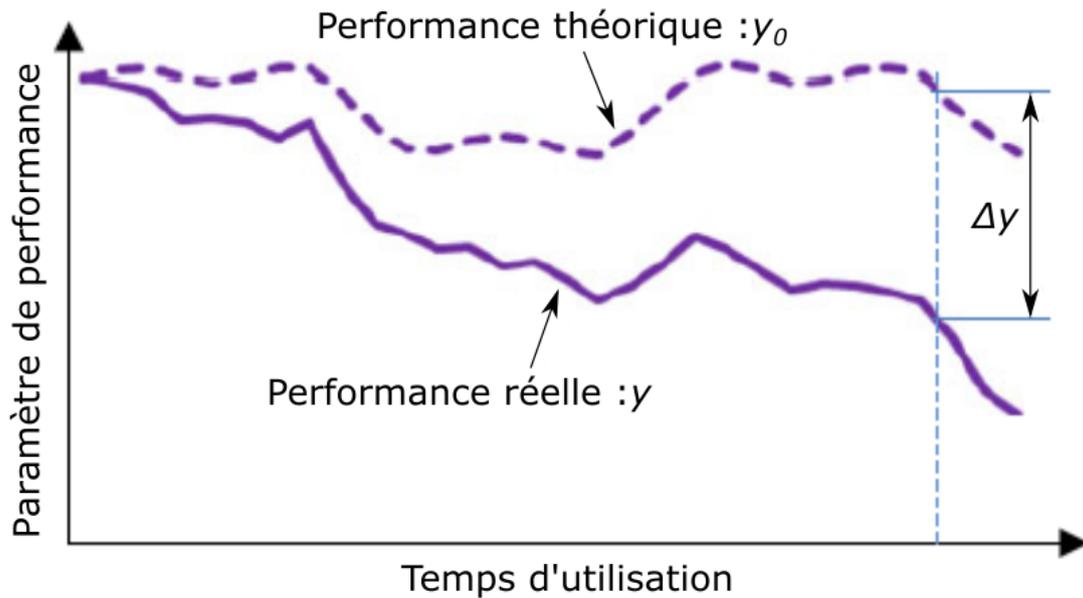
Dans le secteur aéronautique, la fiabilité des systèmes est un enjeu primordial dès leur conception. Les efforts fournis dans ce domaine diminuent le risque de répétition de pannes similaires sur un bien. La constitution d'un historique de défaillances similaires suffisamment volumineux pour entraîner un algorithme d'apprentissage peut alors s'avérer difficile car la période d'observation nécessaire s'étend de plusieurs années à plusieurs décennies. De plus, des modifications du bien peuvent être réalisées entre-temps et impacter son fonctionnement. L'apprentissage en est alors affecté.

En revanche, une documentation technique et des analyses de fiabilité fournies sont disponibles. L'utilisation d'une approche de raisonnement basée sur la connaissance capitalisée semble donc plus adaptée.

### **Approche de raisonnement basée sur la connaissance capitalisée**

L'approche de raisonnement basé sur la connaissance capitalisée repose sur l'utilisation de représentations mathématiques pour incorporer une compréhension physique du système en intégrant la modélisation du système et la "physique des pannes" (PECHT et al. 2010).

Ces techniques de détection consistent en pratique à simuler le comportement d'un système à partir de lois théoriques. La détection se base alors sur l'observation de l'écart entre les fonctionnements réels et théoriques du système considéré. Le modèle de comportement peut se présenter sous la forme de fonctions mathématiques de



**Figure 2.8** – Exemple de détection d'anomalie par les modèles (HANACHI et al. 2018)

complexités variables. Le modèle le plus simple revient à appliquer une valeur de seuil sur une variable unidimensionnelle de type “valeur”. Le dépassement du seuil indique alors une anomalie de fonctionnement du système surveillé (ISERMANN 1984).

Il est également possible de comparer les performances réelles d'un système avec un modèle de comportement théorique. Dans ce cas, l'apparition d'un écart significatif entre les performances réelles et attendues est révélatrice d'une anomalie. La figure 2.8 donne une illustration de ce type de détection.

Dans des systèmes complexes, aboutir à une conclusion nécessite l'application d'un raisonnement plus subtil. Ce raisonnement peut être connu par des experts humains et/ou renseigné dans la documentation du système. Il faut alors structurer et exploiter cette connaissance au moyen de règles. Les systèmes appliquant ce type de processus sont appelés systèmes experts (SE).

Un SE est un programme informatique dans lequel il est possible d'intégrer de la connaissance experte afin reproduire un raisonnement logique et de résoudre un problème ou donner une recommandation à partir d'une situation donnée (JACKSON 1998).

L'avantage des SE sur les approches d'apprentissage vient du fait qu'aucun jeu de données historique n'est nécessaire. De plus, le raisonnement sur la base de règles et de modèles mathématiques offre une analyse fiable, systématique et explicable. En comparaison, garantir l'explication d'un résultat généré par un réseau de neurones demeure un véritable challenge scientifique (JARDINE et al. 2006).

En revanche, ces systèmes requièrent une connaissance experte et approfondie des biens considérés et il est moins facile d'intégrer de nouveaux cas au raisonnement.

**Table 2.2** – Répartition des publications par domaine

Domaine	Nombre de publications
Comptabilité	34
Aérospatial	13
Agriculture	6
Automobile	13
Bancaire	30
Chimie	5
Construction et exploitation minière	7
Enseignement	14
Finance	28
Administration	10
Santé	4
Technologies de l'information	14
Assurance	12
Droit	2
Logistique	8
Production	31
Médical	24
Militaire	14
Pétrole et gaz	11
Publication	1
Immobilier	1
Vente au détail	6
Recherche scientifique	3
Télécommunications	11
Transport	2
Services publics	7

Ils sont donc moins adaptables à des situations inconnues apparaissant par exemple suite à des modifications du bien ou à la découverte de nouvelles pannes et anomalies. De plus, les SE actuels présentent des limites communes en termes d'efficacité, de scalabilité et d'applicabilité (XIAOXUE et al. 2019).

Dans la littérature, WAGNER (2017) propose une revue des applications des systèmes dans divers domaines de 1984 à 2016, dont la liste est fournie dans le tableau 2.2. Le nombre de publications par type de problème est, elle, fournie en table 2.3. On remarque que sur 311 études menées, la principale application concerne le diagnostic avec 73 articles (23%). La surveillance arrive également en troisième position avec 33 publications (10%) derrière les problématiques de planification (13%). Ainsi, plus d'un tiers des applications des systèmes experts se sont portées sur les questions de surveillance et de diagnostics sur cette période.

Parmi ces travaux, LONG et al. (2008) démontre l'intérêt des systèmes experts pour le diagnostic des défaillances affectant le système de carburant des avions. ŽARKOVIĆ et al. (2017) propose la mise en place d'un système expert pour la détection et

**Table 2.3** – Nombre de publications par application des systèmes experts

Type de problème	Nombre de publications
Classification	26
Correction de bugs	16
Diagnostic	73
Interprétation	27
Contrôle	10
Instruction	10
Surveillance	33
Prédiction	15
Correction/réparation	7
Configuration	14
Conception	28
Planning	41
Planification	11

la classification des pannes affectant les transformateurs de puissance à partir de raisonnements intégrant des logiques floues.

Les techniques précédemment présentées répondent aux besoins de détection des pannes et des anomalies. Cependant, les systèmes PHM peuvent aller plus loin en estimant, en temps réel, la durée de vie utile restante (RUL) du système surveillé à partir des anomalies détectées. Le pronostic de la RUL adresse alors un problème de régression pour lequel diverses approches peuvent être employées.

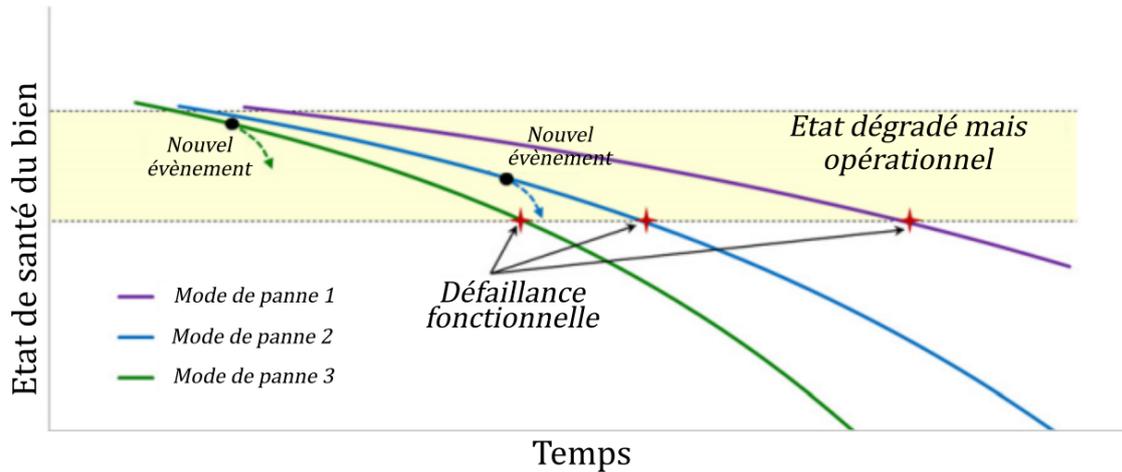
## 2.2.4 Pronostic

En maintenance prévisionnelle systématique, la durée de vie moyenne théorique totale d'un bien est généralement déterminée par son fabricant à la suite de tests sur bancs d'essais. Une marge de sécurité est alors déduite de cette durée de vie. Une fois cette nouvelle durée atteinte, le bien est remplacé, quel que soit son usure réelle, afin de limiter le risque de défaillance.

Bien que cette approche améliore la fiabilité d'un bien par rapport à une approche curative, elle possède deux grandes limites.

Tout d'abord, la durée de vie du bien surveillé n'est pas optimisée. Le remplacement d'un équipement quel que soit son usure réelle engendre une forme de gaspillage et représente un coût pour l'utilisateur.

De plus, cette durée est calculée dans des conditions de fonctionnement artificiellement recréées. Dans le cas d'une utilisation réelle, la dégradation du bien peut cependant diverger de celle simulée sur un banc d'essai. L'environnement dans lequel évolue le bien ainsi que son utilisation peuvent, en effet, accélérer sa dégradation et engendrer une défaillance précoce.



**Figure 2.9** – Courbes de dégradation de santé d'un bien (SIKORSKA et al. 2011)

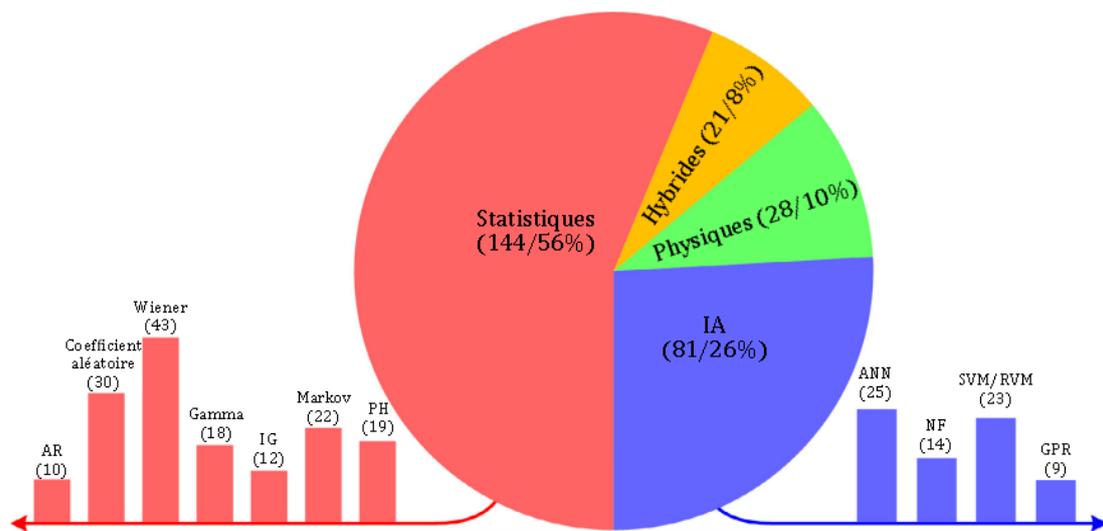
Afin de répondre à ces limites, il est nécessaire de prendre en compte les conditions réelles de fonctionnement du bien maintenu. La démarche visant à estimer la RUL du bien à partir de ces données est appelée pronostic.

Le pronostic se base sur le principe, énoncé par BLOCH et al. (1983), selon lequel toute défaillance est précédée de conditions, indications ou signes avant-coureurs. La détection de ces signes et leur analyse doit permettre d'évaluer la dégradation du système et d'estimer sa RUL. Ainsi, d'après ENGEL et al. (2000), le pronostic est la capacité de fournir une détection des signes avant-coureurs de défaillance et/ou des fautes naissantes sur un système, et de prédire le suivi de son état de santé jusqu'à la panne. Dans ISO (2016), le pronostic est défini comme "l'analyse des symptômes des défauts dans l'intention de prédire l'état futur et la survie en fonction des paramètres de conception". L'estimation de la RUL s'accompagne toujours d'un niveau de précision qu'il convient d'évaluer.

Bien que le diagnostic puisse fournir des informations utiles par lui-même, le pronostic dépend beaucoup des données fournies par le diagnostic pour fonctionner (e.g : anomalies détectées, taux de dégradation) et ne peut pas être effectué indépendamment (SIKORSKA et al. 2011). Comme l'illustre la figure 2.9, différents modes de pannes peuvent entraîner des profils de dégradations divers et, par extension, des RUL variées.

LEI et al. (2018) présente une revue de littérature récente des travaux sur le pronostic. D'après cette étude, on distingue quatre grandes familles de méthodologies : les modèles physiques, les modèles statistiques, les modèles d'intelligence artificielle, ici au sens de l'apprentissage, et les modèles hybrides. Bien que cette étude ne couvre pas toute les publications sur le pronostic, la figure 2.10 donne un aperçu de la répartition de ces méthodologies dans la littérature.

Ces quatre catégories sont rapidement présentées dans les parties suivantes.



**Figure 2.10** – Répartition des méthodologies de pronostic dans la littérature (LEI et al. 2018)

## Les modèles physiques

Les approches basées sur les modèles physiques décrivent le processus de dégradation à partir de modèles mathématiques. Ces modèles mathématiques sont construits sur la base des mécanismes de défaillance connus.

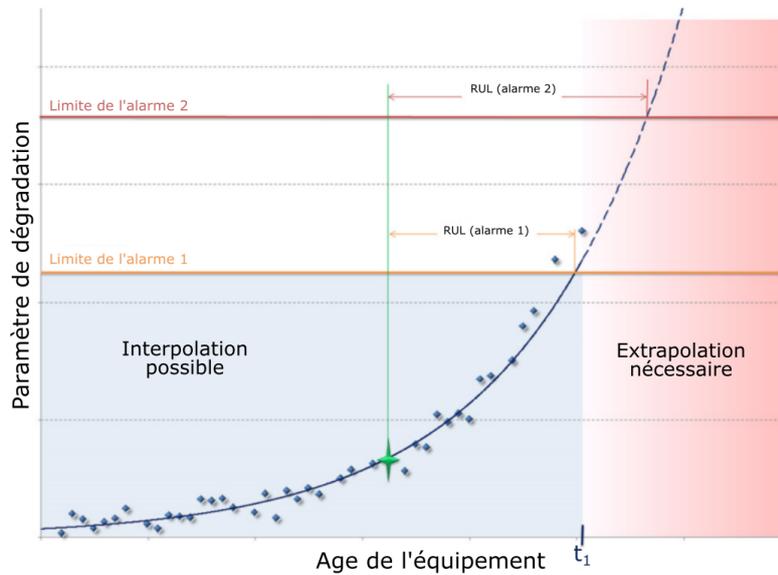
L'approche la plus simple consiste à considérer qu'il existe un paramètre permettant de déterminer à lui seul l'apparition d'une défaillance. Ce paramètre peut provenir d'un seul capteur ou être calculé à partir d'une multitude de signaux de capteurs agrégés en une seule variable.

La figure 2.11, extraite de SIKORSKA et al. (2011), illustre une démarche possible à adopter dans un tel cas. A partir des données collectées en temps réel, il est possible d'interpoler une loi reflétant la dégradation du bien surveillé. Cette loi s'ajuste automatiquement avec les nouvelles données reçues. Elle peut alors servir à déduire le profil de dégradation du bien par extrapolation afin de déduire la durée restante avant la défaillance. Dans la figure 2.11, deux seuils sont prédéfinis afin d'alerter sur la condition du bien. La régression permet ainsi de déterminer le temps restant avant que chacun des seuils d'alerte soit atteint.

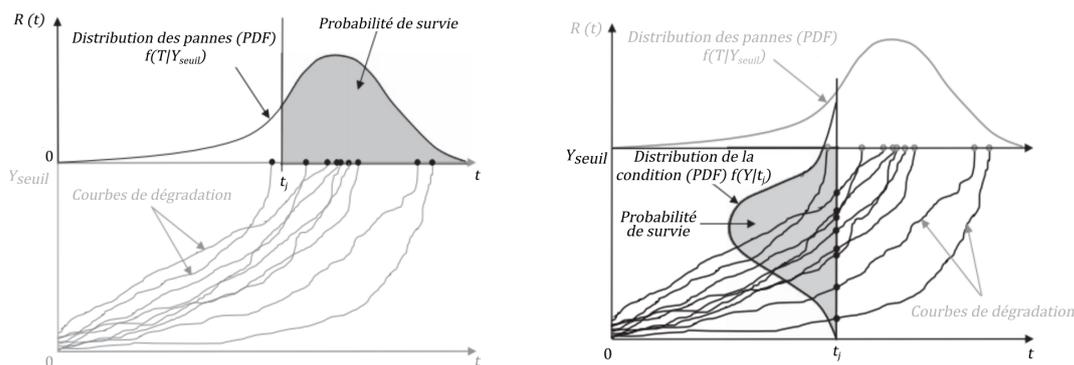
Lorsque les mécanismes de défaillance sont parfaitement compris et intégrés au modèle physique, ce type d'approche permet d'estimer très précisément la RUL. Cependant, dans le cas de systèmes mécaniques complexes, il est difficile de comprendre et modéliser les phénomènes de dégradation. Cette limite est une contrainte forte à l'utilisation de ces approches (LEI et al. 2018).

## Les modèles statistiques

Les approches statistiques estiment la RUL d'un bien à partir de modèles statistiques basés sur des connaissances empiriques. La prédiction de la RUL est alors souvent



**Figure 2.11** – Illustration de pronostic par extrapolation (SIKORSKA et al. 2011)



**Figure 2.12** – Exemple de pronostic par approche statistique (HENG et al. 2009)

présentée sous la forme d'une fonction de densité de probabilité (PDF) construite à partir d'observations.

Dans l'exemple présenté en figure 2.12, extrait de HENG et al. (2009), la fonction de fiabilité correspond à la distribution de probabilité d'une variable aléatoire  $T$  représentant la durée précédant la défaillance. Une distribution normale est choisie pour estimer la probabilité de survie. Celle-ci est définie à partir de courbes de dégradations réelles connues. Ces courbes de dégradations permettent de connaître les durées au bout desquelles le seuil de défaillance  $Y_{seuil}$  est atteint. Ces durées permettent de paramétrer la fonction de distribution normale. Cette fonction peut alors être utilisée, comme illustré en figure 2.12, pour estimer la probabilité de survie à partir d'un instant  $t_j$ .

Toutefois, cette méthode repose sur l'existence de courbes de dégradation et donc sur la capacité à connaître précisément la dégradation d'un bien à tout moment. Cette hypothèse est souvent difficile à vérifier dans la réalité.

En absence de modèles physiques de dégradation existants, il est parfois préférable de faire appel à des approches d'apprentissage artificiel par régression pour estimer la RUL du bien surveillé.

## Les modèles d'apprentissage artificiel par régression

Les approches par apprentissage artificiel visent à identifier les profils de dégradation à partir d'exemples disponibles. Cela évite la conception de modèles statistiques ou physiques souvent complexes et peu adaptables. Les approches par apprentissage permettent, en effet, de gérer des systèmes complexes dont les mécanismes de dégradations sont difficiles à représenter par des modèles statistiques ou physiques.

Pour cette raison, ces approches sont de plus en plus utilisées et forment la deuxième catégorie la plus représentée dans la littérature d'après la figure 2.10 après les approches statistiques.

Un exemple d'application est donnée par ZHANG et al. (2017). Dans cet étude, douze algorithmes d'apprentissages artificiels de régression sont appliqués pour estimer la RUL de turboréacteurs. Parmi les algorithmes utilisés figurent certains des plus courants dans le domaine du pronostic. Le jeu de données utilisé provient du simulateur C-MAPSS fourni par la NASA. Il est divisé en quatre parties notées FD001, FD002, FD003 et FD004. Les algorithmes utilisés dans ZHANG et al. (2017) sont :

- le réseau de croyances profondes multi-objectif (Multi-Objective Deep Belief Network Ensemble : MODBNE)
- le réseau de croyances profondes (DBN)
- le filtre de Karman commutant (Switching Karman Filter : SKF)
- le perceptron multi-couches (Multi-Layer Perceptron : MLP)
- l'apprentissage artificiel extrême (Extreme Learning Machine : ELM)
- l'apprentissage artificiel extrême hiérarchique (HELM)
- les machines à vecteurs de support (SVM)
- la régression Lasso
- l'arbre de régression accru (Extra Tree Regression : ETR)
- la régression par les K voisins (K-Neighbors regression : KNR)
- la stimulation de gradient (Gradient Boosting GB)
- les forêts aléatoires (Random Forest : RF)

Les résultats des apprentissages pour les quatre jeux de données sont comparés par RMSE (Root Mean Square Error) et fournis dans le tableau présenté en figure 2.13. Cette étude démontre la performance de l'algorithme MODBNE présenté et défendu dans l'article.

TW=1	Average	MODBNE	DBN	SKF	MLP	ELM	HELM	SVM	LASSO	ETR	KNR	GB	RF
FD001	Score	640.27	1001.44	762.85	959.63	740.52	835.53	852.07	894.21	1359.38	604.26	<b>575.04</b>	802.23
	RMSE	<b>17.96</b>	18.48	19.24	18.48	19.40	20.02	20.58	22.43	22.05	19.73	18.80	20.23
FD002	Score	<b>10851</b>	15633	23834	13018	53634	46691	521461	231995	231030	76373	44213	84068
	RMSE	28.06	30.05	<b>27.45</b>	29.78	28.89	28.79	36.27	39.43	33.01	29.69	28.67	30.01
FD003	Score	<b>683.41</b>	2074.57	925.78	1442.70	809.84	897.77	1108.68	1144.55	1757.60	1129.82	863.66	1000.51
	RMSE	19.41	20.99	<b>18.29</b>	19.64	21.48	21.83	23.30	23.72	24.52	23.38	22.25	22.34
FD004	Score	<b>7210</b>	8411	7544	30717	24853	25084	46611	100321	69771	26847	20538	22250
	RMSE	29.45	30.02	<b>26.76</b>	34.41	30.04	30.02	40.77	43.71	32.42	33.42	29.46	29.62

**Figure 2.13** – Comparaison des performances de 11 algorithmes d'apprentissage sur 4 jeux de données issus du simulateur C-MAPPS (ZHANG et al. 2017)

La limite principale des approches d'apprentissage relève de la disponibilité d'un jeu de données suffisamment volumineux pour l'entraînement. De plus, les résultats obtenus sont difficilement explicables, en particuliers lorsque des réseaux de neurones sont utilisés. Enfin, le paramétrage des algorithmes et la définition du degré de confiance des résultats sont également des challenges à surmonter (SIKORSKA et al. 2011).

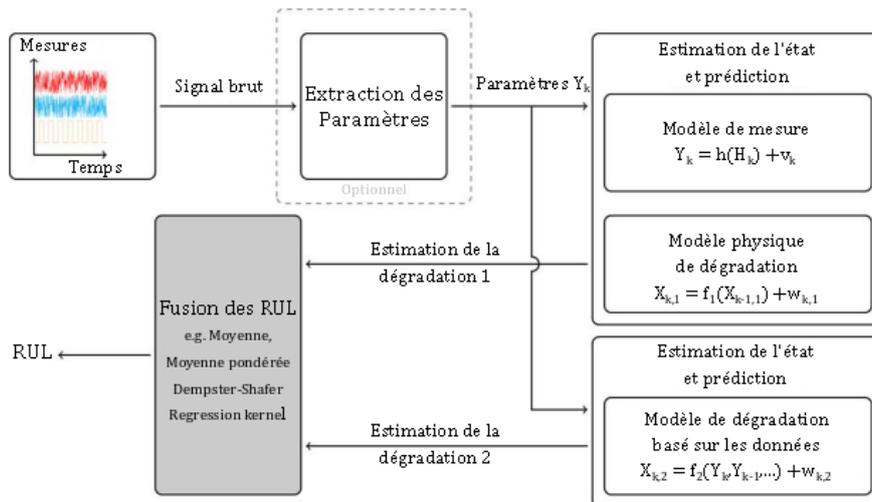
## Pronostic par fusion

L'approche par modèle physique présente de très bonnes performances à condition de comprendre et modéliser précisément les mécanismes de défaillance. Il est donc peu adapté aux systèmes complexes. Les approches statistiques se basent sur des courbes de dégradation historiques. Elles s'adaptent plus facilement à des systèmes complexes que les modèles physiques mais l'existence de ces courbes de dégradation est nécessaire. Enfin, l'approche par apprentissage est la mieux adaptée à des systèmes complexes car elle ne repose ni sur des modèles physiques, ni sur des courbes de dégradation mais sur un historique conséquent d'exemples connus. Cependant, une explication des résultats obtenus ne peut être assurée.

Ces trois solutions possèdent donc chacune leurs avantages et leurs limites. Toutefois, elles ne sont pas incompatibles entre elles. Les approches hybrides consistent à combiner les résultats de méthodologies différentes pour profiter des avantages de chacune.

Un exemple de pronostic hybride est proposé par LIAO et al. (2014) et présenté en figure 2.14. Dans cet exemple, un modèle de mesure  $h()$  permet de connaître l'état de santé  $X_k$  du système à partir des paramètres mesurés  $Y_k$ . Un état de santé dégradé permet de déclencher le pronostic physique du système par le modèle  $f_1$ . Un pronostic à partir du modèle  $f_2$  basé sur les données est également effectué avant que les résultats soient fusionnés avec ceux du modèle physique.

Toutefois, si les approches par fusion permettent d'obtenir les avantages de plusieurs méthodes, elles en partagent aussi les inconvénients. Développer une approche contenant un modèle physique et un modèle par apprentissage signifie par exemple qu'il faut à la fois un jeu de données suffisant pour l'entraînement et un modèle physique représentatif de la dégradation du bien. Ces contraintes expliquent que ce type de méthodologie soit le moins représenté dans la littérature (cf figure 2.10).



**Figure 2.14** – Procédure de pronostic par fusion de modèles physiques et orienté données d’après (LIAO et al. 2014)

Diverses solutions existent donc pour collecter les données, détecter les anomalies et les pannes et estimer la RUL d’un bien. Réaliser une architecture PHM revient donc à faire un choix entre les solutions existantes et à les connecter entre elles.

## 2.2.5 Aide à la décision

Les solutions PHM sont principalement utilisées dans la maintenance de systèmes critiques pour lesquels la disponibilité est un enjeu majeur (HADDAD et al. 2012). Pour ces systèmes, le PHM permet de fournir à l’utilisateur des diagnostics et des pronostics de pannes sur un système donné. Par ailleurs, cette approche permet d’optimiser la durée de vie des systèmes suivis ce qui peut engendrer une baisse des coûts (SANDBORN 2005). Ces informations doivent alors être prises en compte et intégrées dans le processus de maintenance. Afin de faciliter cette intégration, un outil d’aide à la décision peut être utilisé.

Tout d’abord, l’anticipation de la défaillance d’un bien impacte le processus logistique. Des stratégies de gestion de pièces de rechange peuvent être mises en place pour optimiser la disponibilité du bien à partir des prévisions de pannes et des contraintes budgétaires. Contrairement à une stratégie de maintenance systématique, la commande des pièces est plus variable car les pièces ne sont pas changées à la fin de leur durée de vie théorique mais juste avant la panne. Cela permet d’optimiser la durée d’utilisation des pièces mais implique une gestion plus flexible de l’inventaire et de son approvisionnement (TU et al. 2007).

La planification des tâches de maintenance est également impactée par la mise en place d’une approche PHM. Connaître la RUL d’un bien permet, en effet, de déterminer le meilleur moment pour effectuer une intervention. Par ailleurs, lorsque plusieurs composants présentent des niveaux de dégradation importants, un outil

d'aide à la décision rend possible le regroupement des opérations à réaliser pour réduire la dégradation totale du bien à moindre coût (FERRI et al. 2013). Il est également possible d'adapter le niveau de dégradation toléré selon les contraintes économiques, de fiabilité, de disponibilité des ressources ou d'utilisation du bien (HADDAD et al. 2012).

Toutefois, l'intégration d'une démarche PHM dans un processus de maintenance présente également plusieurs challenges.

En effet, la plupart des solutions de PHM ne s'appliquent pas au bien complet mais uniquement à des composants ou des sous-systèmes. Dans ce contexte, le décideur doit alors faire face à une multitude de distributions de RUL qui peuvent rendre complexe l'analyse et la prise de décision. Il faut également considérer la possibilité que certains composants soient redondants. Dans ce cas, la panne n'impacte pas la disponibilité du bien et le décideur pourrait choisir de laisser la défaillance se produire en attendant une opportunité d'intervention adaptée (FERRI et al. 2013). Il est donc nécessaire d'intégrer les solutions de pronostic de composants individuels dans une architecture globale avant d'envisager tout impact sur le processus de maintenance. Pour cela, FERRI et al. (2013) propose de définir la RUL du bien à partir de celles de ses composants au moyen d'un arbre de défaillances. Cette RUL globale permet ainsi de savoir quand le bien cessera réellement de fonctionner.

De plus, la gestion de l'inventaire peut s'avérer complexe dans certains secteurs comme l'aéronautique. (TU et al. 2007) identifie notamment quatre défis :

- La valeur d'immobilisation importante : les pièces de rechange présentent dans l'inventaire représente un coût non négligeable et leur valeur décroît au fil du temps.
- La diversité des fournisseurs : la sous-traitance d'activité à des fournisseurs est une pratique courante. Une planification collaborative, une visibilité globale et une coordination des activités entre plusieurs fournisseurs sont des éléments essentiels.
- L'objectif de disponibilité des biens
- Une demande irrégulière : les conditions et les horaires d'exploitation génèrent une variabilité de la demande.
- Variabilité du délai d'approvisionnement : cette variabilité est une contrainte non négligeable qui complique l'optimisation de la gestion des pièces de rechange.

Enfin, l'outil d'aide à la décision doit tenir compte de l'incertitude des prévisions fournies par les algorithmes de diagnostic et pronostic. SANDBORN (2005) propose un modèle permettant de déterminer la marge de sécurité optimale des prévisions de diminution de la RUL et l'incertitude liée au pronostic.



# Chapitre 3

## Verrous scientifiques et positionnement

### 3.1 Identification des verrous scientifiques

Le secteur de la maintenance est soumis à des exigences de fiabilité des systèmes et de diminution des coûts générant de nouveaux besoins. En effet, pour des systèmes critiques par leur impact sur la santé ou leur environnement industriel, il est nécessaire de surveiller, analyser et préconiser pour prévenir et gérer les défaillances. Un suivi de l'état de santé des biens surveillés en temps réel à partir de données collectées pendant son fonctionnement devient nécessaire en place ou en complément des approches systématiques. Les maintenances conditionnelles prévisionnelles, et le PHM en particulier, connaissent un intérêt croissant de la part des acteurs industriels et de la communauté scientifique.

Le PHM s'articule autour de trois axes principaux que sont : la surveillance, l'analyse et l'aide à la décision.

Pour chacun de ces axes, une revue des enjeux et des solutions existantes a été présentée. Parmi les challenges identifiés, deux problématiques sont adressées dans cette thèse.

Tout d'abord, la complexité des biens considérés rend toute analyse manuelle fastidieuse. Il est nécessaire de mettre en place une approche automatique basée sur les capteurs présents sur l'appareil. De plus le volume, la vélocité et la variété des données collectées relèvent d'un environnement Big Data. L'ingestion et le traitement de ces données par une approche automatique constitue ainsi un premier challenge.

De plus, la détection des anomalies et des pannes peut être réalisée par des approches de raisonnement ou d'apprentissage. Si l'apprentissage a l'avantage de ne nécessiter qu'une faible connaissance métier et s'adapte facilement à de nouvelles situations, elle requiert également un jeu de données d'entraînement comprenant suffisamment d'exemples et de paramètres pour permettre une caractérisation performante du phénomène à déterminer. Pour des systèmes complexes et pour lesquels la fiabilité est un enjeu majeur, comme les avions, ces jeux de données peuvent s'avérer difficiles à

constituer pour tous les systèmes et toutes les défaillances. Une approche uniquement basée sur l'apprentissage artificiel n'est donc pas envisageable à l'échelle du système entier. Par ailleurs, l'explicabilité des résultats obtenus est un besoin crucial dans un domaine comme la maintenance et pour lequel les algorithmes d'apprentissage profond, comme les réseaux de neurones, ne parviennent pas encore à répondre. Cependant, une documentation et des analyses de fiabilité complètes sont disponibles ce qui facilite l'implémentation d'une approche par raisonnement. Un second challenge consiste donc à concevoir une méthode permettant de profiter des avantages des approches par raisonnement et par apprentissage.

Ainsi, dans le cadre de cette thèse, les deux problématiques abordées sont :

1. Comment analyser les données de surveillance pour identifier et prévenir les défaillances en s'appuyant sur les connaissances capitalisées au travers de la documentation technique ?
2. Comment gérer le traitement en temps réel de données volumineuses, variées et véloces dans une perspective de détection d'anomalies ?

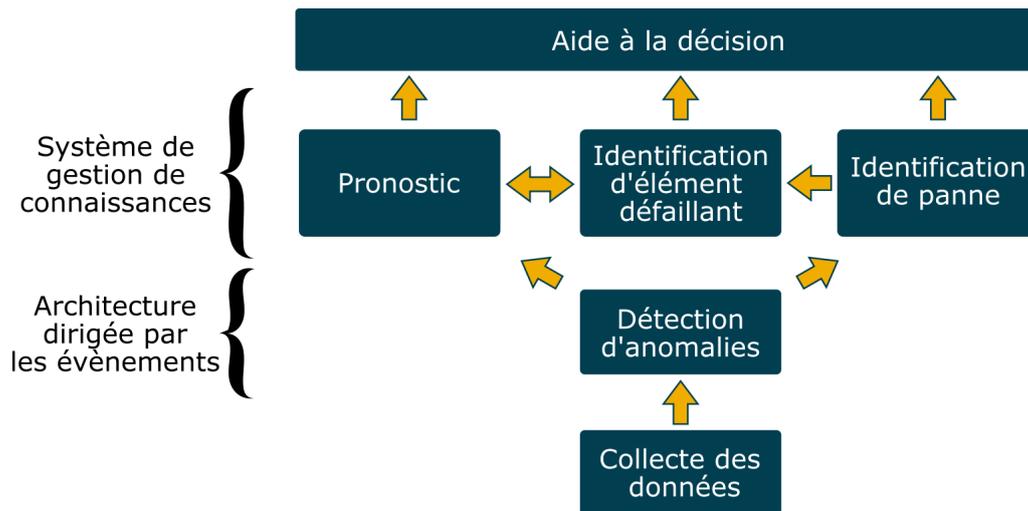
Afin de mener cette étude, certaines conditions et hypothèses ont été définies :

- les systèmes étudiés doivent être suffisamment complexes pour justifier une gestion automatique du processus de maintenance,
- ils doivent être équipés de systèmes de collecte de données renvoyant les informations nécessaires à l'évaluation de leurs états de santé. Du fait de la complexité des systèmes, traduite par des sources de données variées et nombreuses générant des données à hautes fréquences, le traitement de ces données relève d'un contexte "big data",
- l'application de modèles d'apprentissage n'est pas toujours possible du fait d'un faible volume d'occurrences de pannes. Ces données relèvent d'un problème de type "small data" (KITCHIN et al. 2015).
- une connaissance experte capitalisée et structurée doit être disponible au travers d'une documentation technique. Cette documentation doit permettre l'exploitation d'informations sur la composition et la fiabilité du système surveillé.

## 3.2 Présentation de l'approche proposée

Afin d'adresser ces problématiques, nous nous appuyerons sur l'architecture fonctionnelle décrite dans l'introduction générale. Un rappel de cette architecture, présentée en introduction générale, est disponible en Figure 3.1.

Pour répondre à la question de l'exploitation des compétences capitalisées pour l'identification de défaillances, d'éléments défaillants et le pronostic, une première contribution réside dans la définition d'un système de gestion de connaissances. Ce système a pour but de structurer, stocker et exploiter les connaissances pertinentes et nécessaires contenues dans la documentation technique. L'approche utilisée a été volontairement définie sur la base de concepts génériques pour permettre une adaptation et une application à des systèmes divers. Bien que le cas d'étude choisi



**Figure 3.1** – Rappel de l'architecture fonctionnelle de la solution proposée

soit un sous-système d'avion, l'approche proposée n'est pas spécifique. Elle supporte des systèmes issus d'autres domaines d'application, dans la limite des conditions et hypothèses précédemment définies. L'exploitation de ces données est réalisée au travers d'une approche par raisonnement incluant un système à base de règles adapté qu'il a fallu définir.

Cette contribution a fait l'objet d'un article de journal publié dans *Expert Systems With Applications* (SARAZIN et al. 2021).

La seconde contribution répond aux problèmes d'ingestion et de traitement des données de surveillance. La solution adoptée repose sur une architecture dirigée par les événements (EDA : Event-Driven Architecture) ayant pour but de filtrer les données d'entrée sur la base de règles de surveillance. Ces règles de surveillance, contenues dans la base de connaissances, sont adaptées grâce à une approche dirigée par les modèles afin d'être exploitables par l'EDA. Cette solution permet ainsi de réduire le flux des données d'entrée en détectant les anomalies.

Cette contribution a également donné lieu à une publication scientifique sous la forme d'un article dans la conférence MODELSWARD qui a ensuite été sélectionné et étendu pour publication (SARAZIN et al. 2020a).

La présentation de ces contributions dans la suite de ce manuscrit s'organise de la manière suivante :

- **Partie II** : présentation de l'approche par raisonnement sur la base des connaissances expertes capitalisées.
- **Partie III** : description de l'architecture dirigée par les événement et transformation des règles de surveillance
- **Partie IV** : réalisation d'un cas d'étude sur l'identification d'éléments défaillants pour un système aéronautique.



# Partie II

## Présentation de la solution de diagnostic

Dans la partie I, une présentation des stratégies de maintenance existantes a permis d'identifier l'approche PHM comme une solution permettant de répondre aux besoins actuels de réduction des coûts et d'amélioration de la disponibilité des biens. Cette approche se structure en trois grandes phases : la surveillance des données, leur analyse et l'aide à la décision. La phase d'analyse se décompose en trois parties :

- La détection d'anomalies dans les données recueillies
- L'identification des pannes sur la base des anomalies détectées. Cette identification peut concerner des défaillances passées ou à venir. Dans ce dernier cas, il est possible et pertinent de réaliser un pronostic.
- Le pronostic de la RUL du système affecté par la défaillance à venir.

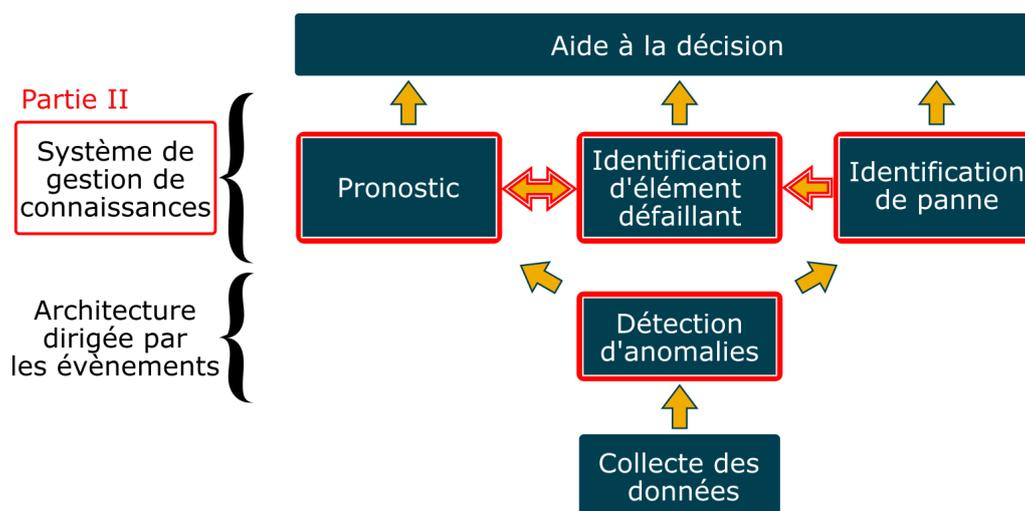
Comme illustré en Figure 3.2, la partie II a pour but de valoriser les connaissances expertes capitalisées au moyen d'un système de gestion de connaissances pour supporter la phase d'analyse du PHM. Dans le cadre de ce manuscrit, nous nous intéressons à des systèmes complexes contenant plusieurs sous-systèmes constitués eux-mêmes de plusieurs composants. De plus, ces composants peuvent influencer les uns sur les autres. La défaillance d'un composant peut impacter le fonctionnement d'un autre. Ces systèmes peuvent, par ailleurs, être modifiés. La solution retenue ne doit pas se restreindre à un bien ou modèle de bien en particulier et doit être capable de s'adapter à des modifications.

Deux hypothèses seront également considérées dans cette partie :

- *Hypothèse 1* : une documentation complète et exploitable du système surveillé est disponible. Cette documentation doit contenir des informations sur la composition du système et les défaillances pouvant affecter ses composants.
- *Hypothèse 2* : un système de collecte de données renvoie les informations nécessaires à l'évaluation de l'état de santé du système surveillé.

Dans la section 2.2.3, nous avons vu que la phase d'analyse peut être réalisée au moyen d'approches :

- Par apprentissage artificiel
- Par raisonnement
- Par combinaison des deux approches précédentes



**Figure 3.2** – Positionnement de la partie II dans l'architecture fonctionnelle

L'apprentissage artificiel possède l'avantage de ne nécessiter aucune connaissance métier du système considéré. De plus, il est possible d'améliorer ses performances en alimentant le jeu de données d'entraînement servant à construire le modèle. Cependant, plusieurs problèmes peuvent se poser dans l'établissement de ce jeu de données :

- Le nombre d'occurrences de pannes observées sur des systèmes similaires est trop faible pour construire un modèle de prédiction fiable.
- La qualité et la diversité des paramètres surveillés sont insuffisantes pour produire une analyse performante.
- Les données réelles d'exploitation sont inaccessibles du fait de leur sensibilité et des restrictions liées aux politiques de protection des données.
- Le constructeur n'a pas réalisé ou partagé de jeu de données à partir de tests sur banc d'essai.
- Les jeux de données du constructeur ne sont pas représentatifs du fonctionnement du système en conditions réelles.

Du fait de ces limites, l'approche par apprentissage peut s'avérer insuffisante pour détecter les pannes à l'échelle d'un système complexe. Afin de combler ces lacunes, une solution consiste à remplacer l'approche par apprentissage avec une approche par raisonnement s'appuyant sur la documentation technique et les analyses de fiabilité disponibles. Cette solution n'est cependant possible que si l'hypothèse 1 est validée. Lorsque cela est possible, les deux approches peuvent également être combinées pour améliorer la qualité de l'analyse.

Pour combiner ces approches, la proposition faite dans ce manuscrit est de concevoir une approche par raisonnement adaptée à la maintenance prévisionnelle. Celle-ci devra être capable d'intégrer des modèles d'apprentissage déjà entraînés.

Dans cette partie, le principe du système expert sera présenté dans le chapitre 4. Ensuite, les principaux composants de la solution développée seront présentés, à

---

savoir la base de connaissance dans le chapitre 5 et le moteur d'inférences dans le chapitre 6.



# Chapitre 4

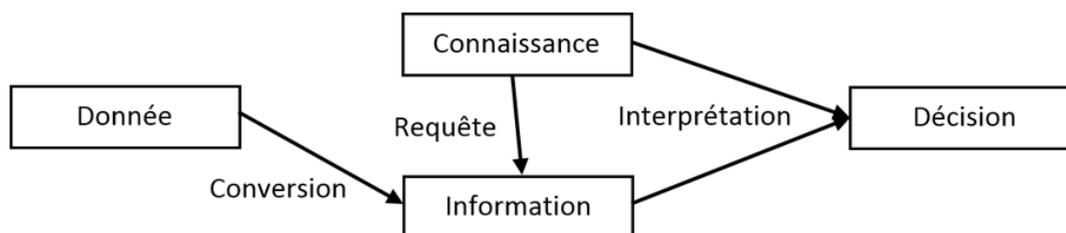
## Principe d'un système expert

Avant de présenter la notion de système expert, il convient de définir et distinguer trois notions fondamentales dans la gestion de la connaissance : la donnée, l'information et la connaissance.

Une donnée résulte d'une procédure d'acquisition. Elle peut être de nature qualitative ou quantitative et ne porte pas d'interprétation. Une donnée est, ainsi fondamentalement objective. Elle est ainsi définie dans les normes comme un fait représenté sous une forme conventionnelle convenant à un traitement, soit par l'homme, soit par des moyens automatiques.

L'information est une donnée ou une agrégation de données organisées pour construire un message ayant du sens. Le mode d'organisation des données résulte d'une intention. Une information est donc partiellement subjective. La notion d'information est, par ailleurs, étroitement liée à la notion de communication. Elle implique en effet un émetteur, un récepteur et un média. L'information est un ensemble de données replacées dans un contexte et porteuse d'un sens particulier.

La connaissance peut se définir comme une information interprétée voire traduite par un récepteur. Elle permet à un individu d'interpréter, d'intégrer et de contextualiser une information. WATSON (2008) propose de distinguer données, information et connaissance en s'appuyant sur les relations décrites en Figure 4.1.



**Figure 4.1** – Relations entre donnée, information et connaissance (WATSON 2008)

Les données sont définies comme des faits qui n'ont pas été analysés, ni résumés (données brutes). L'information est traitée dans une forme significative dépendant du contexte. Enfin, la connaissance est expliquée comme la capacité à utiliser l'information, c'est-à-dire, savoir quelles informations sont requises et quelle est la signification de cette information.

Un système de gestion de connaissance peut alors être défini comme « *un système dynamique qui fournit des fonctions pour soutenir l'identification, l'acquisition, la rétention, l'entretien, la recherche, la récupération, la distribution, la vente et la logistique des connaissances* » (LEHNER et al. 2000).

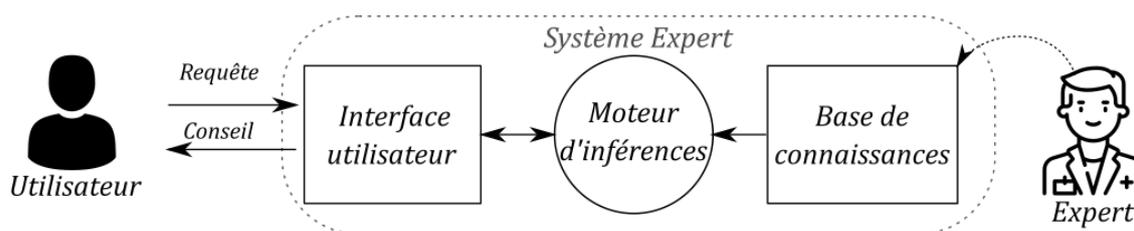
D'après JACKSON (1998), un système expert (SE) est un programme informatique dans lequel il est possible d'intégrer de la connaissance experte afin de reproduire un raisonnement logique pour résoudre un problème ou donner une recommandation à partir d'une situation donnée. Cette technologie fait ainsi partie des solutions d'intelligence artificielle par raisonnement. Ces systèmes englobent des connaissances organisées relatives à un domaine d'expertise humaine spécifique afin de réaliser des analyses performantes à moindre coût (MICHIE et al. 1982). Ces analyses, basées sur un raisonnement logique appliqué aux données reçues, permettent ainsi de prendre une décision parmi plusieurs choix possibles. Cette démarche fait appel à des inférences à partir de règles définies au préalable pour traiter les données sur la situation rencontrée (KALOGIROU 2003).

Les SE se distinguent en particulier par les caractéristiques suivantes (KALOGIROU 2003 ; PHAM et al. 1988) :

- Ils sont conçus pour résoudre des problèmes complexes nécessitant habituellement une expertise humaine.
- Ils intègrent des connaissances expertes et des techniques d'inférence logiques. Les connaissances sont structurées dans un langage symbolique déclaratif. Les règles d'inférence se présentent sous la forme de procédures de raisonnement exploitant les informations répertoriées.
- Ils présentent une performance élevée dans des domaines techniques très spécifiques, le traitement de données incomplètes ou incertaines, la gestion de situations imprévues et la justification des résultats obtenus.
- Ils se limitent à un domaine précis d'expertise humaine.
- Ils peuvent être conçus pour évoluer et s'améliorer avec le développement de l'expertise.
- Ils sont capables d'exploiter des techniques de représentation de connaissances pour manipuler le savoir difficile à exprimer sous forme de règles.

Un SE est généralement constitué de trois composants principaux présentés en figure 4.2 : une base de connaissances, un moteur d'inférences et une interface utilisateur (LIEBOWITZ 1995).

La base de connaissances structure un ensemble de faits et de règles qui décrivent le système considéré. Un fait désigne ici un élément atomique de connaissance disponible avant le raisonnement (LEVINE et al. 1990). Appliqué à la maintenance, la base de



**Figure 4.2** – Éléments constitutifs d'un système expert

connaissances répertorie et décrit le système et sa composition ainsi que les anomalies et pannes pouvant affecter l'affecter. Cette base de connaissances est le composant principal du SE. Cette connaissance ne se résume pas à un ensemble de données mais comprend également les éléments permettant d'appréhender leur structure et leur compréhension. Ces éléments proviennent directement du mode de représentation des connaissances choisi (PHAM et al. 1988).

Au sein de la base de connaissances, il est important de distinguer deux types de données : les données statiques et les données dynamiques (KALOGIROU 2003).

Les données statiques correspondent au savoir expert du domaine technique. Cela comprend la description du système physique, l'ensemble des pannes et anomalies connues ainsi que les règles permettant de les détecter. Il est primordial que ces données soient complètes, cohérentes et précises afin de supporter une analyse performante de la condition du système surveillé. Bien que ces données soient considérées statiques et stables, il est évidemment possible de les mettre à jour au fil des évolutions du système et de l'accumulation des connaissances le concernant. Toutefois, la vitesse reste faible.

Les données dynamiques regroupent les informations fournies par l'utilisateur sur l'état du système, comme les relevés de capteurs, et les conclusions intermédiaires générées lors du raisonnement. Par nature, ces données ne sont donc pas persistantes mais possèdent une forte vitesse.

Le rôle du moteur d'inférences réside dans l'exploitation de la base de connaissances afin de déduire un résultat. Bien que la base de connaissances soit une partie importante du SE, son utilité dépend en grande partie des mécanismes d'inférence appliqués aux informations répertoriées (PHAM et al. 1988). Le moteur d'inférences est notamment responsable de l'application des règles et de la gestion des conflits potentiels. En effet, si les conditions d'activation de plusieurs règles sont remplies, le moteur d'inférences est chargé d'assurer la cohérence des règles à appliquer. Il permet également de tester différentes stratégies pour arriver à une conclusion (LIEBOWITZ 1995). Dans le cadre de la détection d'anomalies et de l'identification des pannes, il est important de noter que les règles considérées répondent à une logique propositionnelle du premier ordre.

L'interface utilisateur a pour but de permettre à l'utilisateur d'interagir avec le SE pour consulter les résultats, les raisonnements appliqués ou mettre à jour la base de connaissance.

L'avantage des SE sur les approches d'apprentissage vient du fait qu'aucun jeu de données historique n'est nécessaire. De plus, le raisonnement sur la base de règles et de modèles mathématiques offre une analyse fiable, systématique et explicable. En comparaison, garantir l'explication d'un résultat généré par un réseau de neurones demeure un véritable challenge scientifique (YANG et al. 2020).

En revanche, ces systèmes requièrent une connaissance experte et approfondie des systèmes considérés ce qui rend plus délicate l'intégration de nouveaux cas au raisonnement. Ils sont donc moins adaptables à des situations inconnues apparaissant par exemple suite à des modifications du bien ou à la découverte de nouvelles pannes et anomalies.

De plus, le critère d'évaluation principal d'un SE relève de la gestion de la connaissance qu'il contient. Cependant, d'après XIAOXUE et al. (2019), les solutions de gestion de connaissances actuelles souffrent de trois limites principales :

- L'efficacité : la complexité d'un système, impliquant une base de connaissances contenant de nombreux faits et règles, peut fortement impacter la qualité du raisonnement et le temps de traitement.
- La scalabilité : dans le cas de systèmes complexes, la gestion du volume de données stockées et l'intégration de nouveaux éléments peut s'avérer difficile.
- L'adaptabilité : un SE est lié à un domaine d'expertise très précis. De fait, il est généralement peu adaptable à des systèmes différents.

Afin de concevoir une solution adaptée aux besoins d'un système complexe et adressant les limites présentées ci-dessus, un SE permettant d'intégrer des modèles d'apprentissage entraînés a été imaginé. Dans le chapitre 5, la conception de la base de connaissances sera détaillée. Dans le chapitre 6 le moteur d'inférence adapté à l'exploitation de cette base sera décrit.

# Chapitre 5

## Base de connaissances

La base de connaissances est un composant central du SE sur lequel s'appuie le raisonnement. Elle a pour rôle de stocker et structurer les données capitalisées et d'intégrer les changements de situations du système. Dans ce chapitre, l'objectif est de définir une base de connaissances capable de répondre aux limites d'efficacité, de scalabilité et d'adaptabilité identifiées dans le chapitre précédent.

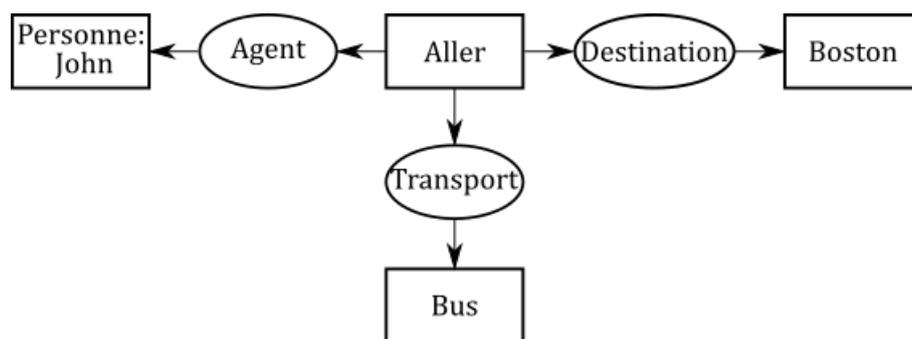
Sa conception doit répondre à deux questions fondamentales : comment stocker les données et structurer la connaissance ?

Dans un premier temps, le mode de stockage des données permettra de répondre aux limites d'efficacité et de scalabilité. Ensuite, le problème d'adaptabilité de l'architecture à différents systèmes sera traité par une représentation des connaissances générique adaptée à la maintenance conditionnelle.

### 5.1 Stockage des données

D'après XIAOXUE et al. (2019), les problèmes d'efficacité et de scalabilité peuvent être traités par l'utilisation de graphes pour supporter la base de connaissances. Ces graphes peuvent être définis comme des bases de connaissances sémantiques structurées permettant de décrire des concepts et leurs relations dans le monde physique de manière symbolique. Grâce à cela, la représentation et la gestion de la connaissance permettent de résoudre des problèmes d'association de connaissances comme l'extraction d'informations et la réponse à des questions sémantiques.

Les graphes de connaissances sont ainsi capables de collecter, organiser et exploiter efficacement de la connaissance provenant de jeux de données volumineux pour améliorer la qualité des services d'information. Cette exploitation repose sur des approches de raisonnement sur le graphe (X. CHEN et al. 2020). En s'appuyant sur les travaux de BASCANS (2017), on s'intéressera particulièrement à trois solutions de graphes de connaissances : les réseaux bayésiens, les graphes conceptuels et les réseaux d'informations hétérogènes (HIN). D'après AGRAWAL et al. (2000), les classificateurs naïfs bayésiens sont parmi les meilleurs classificateurs textuels. A ce titre, ils peuvent être utilisés pour créer, traiter et maintenir une structure



**Figure 5.1** – Exemple de graphe conceptuel (FU et al. 2017)

hiérarchique de documents textuels au moyen d’opérations d’extraction interactives. Les réseaux bayésiens peuvent également être employés dans les systèmes de diagnostic et d’aide à la décision pour la maintenance. Le raisonnement se base sur l’attribution de probabilités pouvant être mises à jour, a posteriori, pour valoriser le retour d’expérience sur le système considéré. Cela a pour effet d’enrichir le réseau afin d’améliorer sa performance (CORSET et al. 2003). Un exemple de réseau bayésien est fourni dans la section 2.2.3 (voir Figure 2.6).

Les graphes conceptuels sont des modèles représentations de connaissances exploitées au moyen d’opérateurs de raisonnement définis par SOWA (1983). Ils ont été conçus pour répondre à des problématiques de traitement de données textuelles en langage naturel, d’inférence de bases de données et d’ingénierie des connaissances. Un graphe conceptuel est composé de noeuds reliés entre eux par des arcs typés. Ils permettent de représenter des connaissances : les noeuds représentant des concepts, objets ou faits, et les arcs représentant des relations ou associations sémantiques entre ces noeuds. Un concept est formé d’un type et d’un référent qui peut être générique ou individuel (BASCANS 2017).

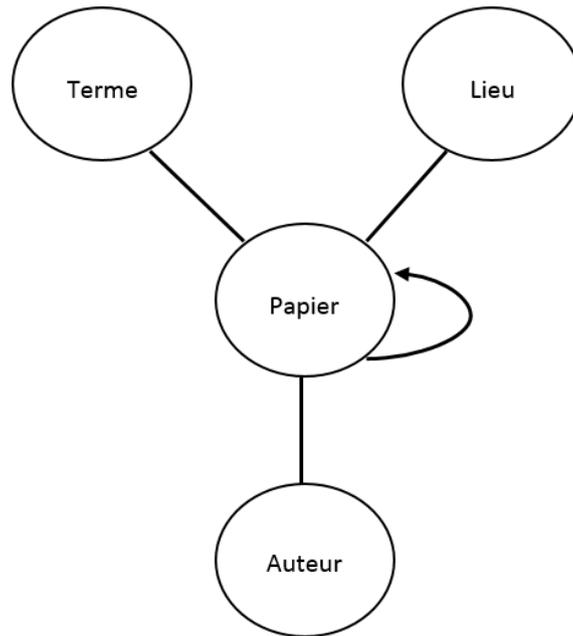
Un exemple de graphe conceptuel est donné en Figure 5.1.

Dans cet exemple, le graphe décrit le fait qu’une personne, John, se déplace en utilisant un moyen de transport, le bus, vers une destination, Boston. Les concepts manipulés sont représentés dans des rectangles tandis que les relations conceptuelles sont matérialisées dans des ellipses.

Par la suite, quatre règles fondamentales ont été définies pour construire de nouveaux graphes (CHEVALLET 1992) :

- la recopie d’un graphe,
- la restriction d’un concept d’un graphe (remplacer ce concept par une de ses spécialisations, sous-types ou référents),
- la jointure de graphes,
- la simplification d’un graphe.

Dans un réseau logique, le graphe est interprété comme un énoncé logique de propriétés des différentes entités. Les primitives de base d’un réseau logique sont les



**Figure 5.2** – Exemple de HIN au travers d’un réseau bibliographique (BASCANS 2017)

prédicats définis pour les entités. Des propositions distinctes peuvent être combinées en utilisant les opérateurs logiques.

Les HIN sont définis comme des représentations abstraites du monde réel au moyen d’objets connectés entre eux par des relations (Y. SUN et al. 2012). D’après BASCANS (2017), les médias sociaux sont le plus souvent des réseaux hétérogènes complexes avec des noeuds et des relations entre noeuds de types différents, correspondant à différents objets, concepts et relations. Le réseau supportant un HIN lui permet de gérer des systèmes complexes grâce à une intelligibilité et un traitement de la connaissance adaptés à une scalabilité variable.

BASCANS (2017) propose une illustration de réseau d’information hétérogène en utilisant l’exemple d’un réseau bibliographique (Figure 5.2). Un réseau bibliographique, comme DBLP (Digital Bibliography & Library Project), contient 4 types d’objets, à savoir le papier, l’auteur, le terme et le lieu de publication (par exemple les conférences et les revues). Des liens existent entre les auteurs et les documents par la relation de « écrire » et « écrit par », entre les documents et les termes par « mention » et « mentionné par », et entre les lieux et les documents par « publier » et « publié par ». La relation « Citation » entre les documents peut être ajoutée en utilisant d’autres sources de données, telles que Google Scholar.

Les HIN sont également utilisés pour la prédiction supervisée et non supervisée de relations dans un environnement multi-relationnel (R. DAVIS et al. 1993) et pour le traitement de problèmes de classification, où les noeuds peuvent être associés à différents types ayant chacun leur propre ensemble de classes (ANGELOVA et al. 2012).

Graphe de connaissances	Généricité	Hétérogénéité	Intelligibilité	Adaptabilité	Extensibilité	Volumétrie	Vélocité
Réseaux bayésiens	✓	✗	✗	✓	✓	✗	✗
Graphes conceptuels	✓	✗	✓	✓	✓	✗	✗
HIN	✓	✓	✓	✓	✓	✓	✓

**Table 5.1** – Tableau comparatif des solutions de graphes de connaissances

Afin de choisir entre ces trois solutions, (BASCANS 2017) propose une comparaison basée sur sept critères :

- Généricité : capacité de représenter tout type de système.
- Hétérogénéité : capacité à supporter la variété des données et des informations.
- Intelligibilité : capacité à structurer des données et des informations de manière facilement compréhensible par l’homme.
- Adaptabilité : capacité à interagir avec d’autres composants, notamment le moteur d’inférences.
- Extensibilité : capacité d’intégration de nouvelles données et règles.
- Volumétrie : capacité à gérer des masses de données et d’informations, ainsi que les relations entre elles.
- Vélocité : capacité à traiter des données et des informations selon des coûts et temps maîtrisables et raisonnables.

La comparaison des trois solutions selon ces critères est présentée dans la table 5.1. Le HIN présente des avantages en terme d’hétérogénéité, d’intelligibilité, de volumétrie et de vélocité par rapport aux autres solutions. Pour ces raisons, cette solution a été choisie pour supporter la base de connaissances du SE.

La structure d’un HIN est définie par Y. SUN et al. (2012) comme un graphe orienté  $G = (V, E)$  où  $V$  correspond à l’ensemble des noeuds et  $E$ , l’ensemble des arcs. Ce graphe est doté d’une fonction de parcours d’objets et d’une fonction de parcours d’arcs où chaque objet  $v \in V$  possède un type  $\tau(v) \in T$  et chaque arc  $e \in E$  possède un type  $\phi(e) \in M$ . De plus, si deux arcs sont du même type, alors les types d’objets en entrée sont identiques et il en va de même pour les types d’évènements de sortie. On considère qu’un réseau d’informations est hétérogène s’il possède plusieurs types d’arcs ou d’objets.

Le graphe sera par la suite décrit avec les conventions suivantes :

- Les ensembles sont en lettres latines majuscules.
- Les fonctions sont en lettres latines minuscules.
- Les valeurs sont en lettres grecques minuscules.

Ainsi,  $G$  est un graphe avec :  $G = (V, E)$ , où  $V = \{v_1, \dots, v_u\}$  est l’ensemble des objets, aussi appelés noeuds, et  $E = \{\epsilon_1, \dots, \epsilon_e\}$  l’ensemble des arcs avec

$$E = \{\epsilon_k = (v_i, v_j) \in V^2, (i, j) \in [1; u]^2, i \neq j\} \quad (5.1)$$

- Chaque noeud possède un type  $\tau$  appartenant à l'ensemble  $T = \{\tau_1, \dots, \tau_t\}$ . La fonction  $f_v : v \rightarrow \tau$  permet de retourner le type  $\tau$  du noeud  $v$ .
- Chaque arc possède un type  $\mu$  appartenant à l'ensemble  $M = \{\mu_1, \dots, \mu_m\}$ . La fonction  $f_\epsilon : \epsilon \rightarrow \mu$  permet de retourner le type  $\mu$  de l'arc  $\epsilon$ .

Cette formalisation est utilisée pour définir les caractéristiques des noeuds et des arcs à travers leurs types. En effet, chaque type possède un ensemble d'attributs appartenant à l'ensemble  $X = \{\chi_1, \dots, \chi_x\}$ . Ces attributs sont des propriétés caractéristiques des types de noeuds et des arcs qui sont personnalisables selon les informations nécessaires à leur implémentation et leur exploitation.

Les attributs d'un noeud de type  $\tau$  peuvent être retournés par la fonction

$$f_\tau : \tau \rightarrow X_j \text{ avec } X_j \subset X. \quad (5.2)$$

Pour les arcs de type  $\mu$ , cela donne respectivement

$$f_\mu : \mu \rightarrow X_j \text{ with } X_j \subset X. \quad (5.3)$$

Pour deux noeuds  $v_p$  et  $v_n$  et un arc de type  $\mu_l$ , la fonction  $g$  retourne, s'il existe, l'arc  $\epsilon_x$  les connectant et l'ensemble  $\{\epsilon_1, \dots, \epsilon_a\}$  contenant tous les arcs existants entre les noeuds  $v_p$  et  $v_n$  avec  $\{\epsilon_1, \dots, \epsilon_a\} \subset E$  sans contrainte sur le type d'arc.

$$g : v_p * v_n * \mu_l \rightarrow \epsilon_x \quad (5.4)$$

$$g : v_p * v_n \rightarrow \{\epsilon_1, \dots, \epsilon_a\} \quad (5.5)$$

La valeur  $\alpha$  d'un attribut  $\chi$  pour un noeud  $v$  est fourni par la fonction :

$$h_v : v * \chi \rightarrow \alpha \quad (5.6)$$

Respectivement, la valeur  $\beta$  de l'attribut  $\chi$  de l'arc  $\epsilon$  est fourni par la fonction :

$$h_\epsilon : \epsilon * \chi \rightarrow \beta \quad (5.7)$$

Ces valeurs sont modifiables grâce à l'utilisation de la fonction  $l$  :

$$l_v : \chi * v * \alpha \rightarrow v \text{ pour un noeud } v \quad (5.8)$$

$$l_\epsilon : \chi * \epsilon * \alpha \rightarrow \epsilon \text{ pour un arc } \epsilon \quad (5.9)$$

Le tableau 5.2 récapitule la liste des notations utilisées dans le graphe

**Table 5.2** – Tableau récapitulatif des notations de la base de connaissances

Notation	Signification
$V$	Ensemble des noeuds
$E$	Ensemble des arcs
$R$	Ensemble des règles
$X$	Ensemble des attributs
$T$	Ensemble des types de noeuds
$M$	Ensemble des types d'arcs
$D$	Ensemble des types de règles
$v$	Noeud
$\epsilon$	Arc
$\gamma$	Règle
$\xi$	Attribut
$\tau$	Type de noeud
$\mu$	Type d'arc
$\delta$	Type de règle
$\alpha$	Valeur de l'attribut $\xi$ d'un noeud $v$
$\beta$	Valeur de l'attribut $\xi$ d'un arc $\epsilon$

Bien que ce HIN soit adaptable à tout système, ce critère dépend également de la structure des données qu'elle contient. Pour cela, il est nécessaire de définir une structure de données générique permettant de représenter n'importe quel système pour une application dans le cadre d'un PHM.

## 5.2 Structure des connaissances

Dans le chapitre précédent, une réponse aux limites de performance et de scalabilité a été apportée par l'implémentation de la base de connaissances sous la forme d'un HIN. Une troisième limite à prendre en compte dans la conception d'un SE est son adaptabilité, autrement dit sa capacité à se transposer à différents systèmes.

En effet, la base de connaissances a pour rôle de structurer les informations relatives au domaine d'application du SE. Par conséquent, le champs d'application du SE est limité à la liste des concepts et des relations définies dans cette base.

Pour améliorer l'adaptabilité d'un SE, il est nécessaire de définir et structurer des concepts et des relations génériques, c'est-à-dire partagés par tous les systèmes, dans le domaine considéré, ici la maintenance.

Une telle conception rentre dans le domaine de la représentation de connaissances évoqué dans la section 2.2.3. et fait particulièrement référence aux notions d'ontologie et de modélisation.

En se basant sur les définitions de T. R. GRUBER (1993) et BORST (1999), STUDER et al. (1998) considère une ontologie comme une spécification formelle et explicite

d'une conceptualisation commune. La conceptualisation désigne ici un modèle abstrait d'un phénomène impliquant une identification des concepts pertinents pour celui-ci. Cette conceptualisation doit être commune car elle reflète la vision consensuelle d'un groupe de personnes et non le résultat de la perception d'un seul individu. La spécification doit être explicite dans le sens où les types des concepts manipulés et les contraintes inhérentes à leur usage doivent être définis explicitement. Elle doit également être formelle pour être interprétable par une machine.

D'après GÓMEZ-PÉREZ et al. (2006), l'ingénierie des ontologies se base sur quatre principaux types de composants :

- les *classes*, qui représentent les concepts manipulés dans le domaine considéré
- les *relations*, qui formalisent les liens entre les classes
- les *axiomes formels*, qui sont des assertions formalisant une connaissance que d'autres composants ne permettraient pas de comprendre et permettant de limiter les interprétations possibles de cette connaissance (T. GRUBER 1993).
- les *instances*, qui sont les individus appartenant à une ou plusieurs classes.

La notion d'ontologie est assez proche de la notion de modèle. Un modèle est défini par BELAUNDE et al. (2003) comme une spécification formelle d'une fonction, d'une structure et/ou du comportement d'une application ou système. D'après MOF (2002), un modèle est une description ou une spécification d'un système selon un certain point de vue.

TRUPTIL (2011) souligne l'importance du point de vue adopté par le modèle au travers d'un exemple tiré de BÉZIVIN et al. (2004) illustré en figure 5.3.

Dans cet exemple, le système considéré est la "France métropolitaine". Ce système est représenté par trois modèles selon des perspectives différentes : fluviale, routière et administrative. Bien que ces trois modèles s'appliquent au même système, les représentations sont distinctes car les concepts manipulés et le langage de représentation diffèrent selon le point de vue adopté.

Si l'on représente une ontologie par un modèle, on peut alors considérer que le point de vue du modèle est lié au domaine d'application de l'ontologie. Sa structure peut ainsi être décrite au moyen d'un méta-modèle. En effet, tout modèle se conforme à une structure constituée de concepts et des relations entre eux. Cette structure peut également se présenter sous la forme d'un modèle appelé méta-modèle. On considère ainsi qu'un modèle est une instance d'un méta-modèle et que tous les modèles doivent être construits conformément à leur propre méta-modèle (J. BEZIVIN et al. 2001). Cette hiérarchie est décrite dans la figure 5.4.

Le méta-modèle étant lui-même un modèle, il doit se conformer à son propre méta-modèle appelé méta-méta-modèle. On peut alors remonter cette hiérarchie jusqu'à arriver au modèle du langage UML appelé MOF (Meta-Object Facility) défini par l'OMG (Objet Modelling Group) (MOF 2016).

Dans le domaine de la maintenance, divers méta-modèles et ontologies ont été développés pour définir et structurer les concepts et relations en jeu.

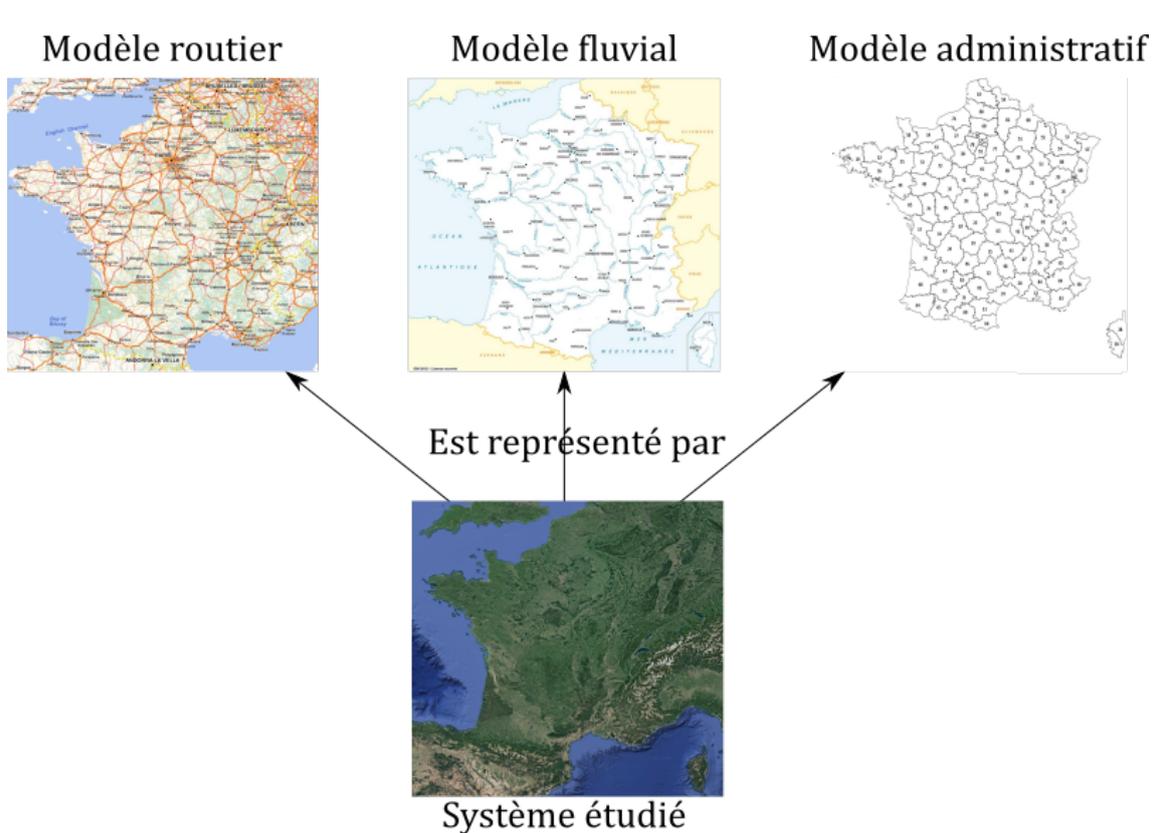


Figure 5.3 – Illustration de la notion de modèle (BÉZIVIN et al. 2004)

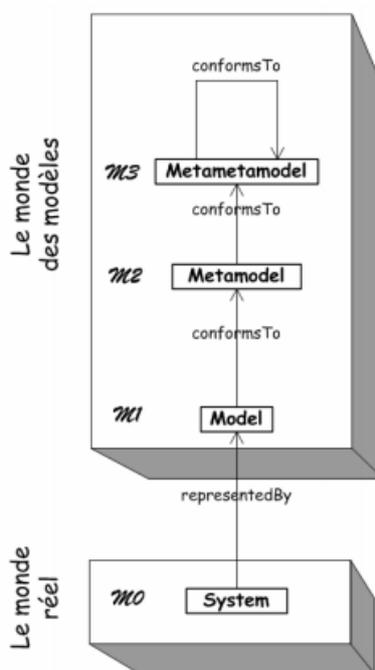


Figure 5.4 – Relation entre système, modèle et méta-modèle (BÉZIVIN et al. 2004)

## 5.2.1 Ontologie OntoProg

La figure 5.5 décrit la structure de l'ontologie OntoProg définie par NUNEZ et al. (2017). Cette structure est composée de trois concepts de haut niveau : “ManufacturingItems”, “DataCollectionFeatures” et “FailureAnalysisFeatures”.

“ManufacturingItems” regroupe les concepts liés à la structure du système physique considéré. Il se décline ainsi par héritage en systèmes, sous-systèmes et composants respectivement désignés sur la figure par “System”, “SubSystem” et “Component”.

“FailureAnalysisFeature” intègre les classes liées à l'analyse des paramètres du système pour détecter les symptômes et les pannes. Les concepts définis à partir de cette classe sont :

- Les paramètres surveillés “EvaluationParameter”
- Les techniques de mesures utilisées “MeasurementTechnique”
- Les causes de défaillance “Cause”
- Les effets produits “Effect”
- Les symptômes identifiés “Symptom”
- Les pannes pouvant affecter le système “FailureMode”

“DataCollectionFeatures” comprend les éléments de collecte et de surveillance des données. La classe “MeasurementFeature” désigne les sources de données, comme les capteurs. Les éléments de cette classe sont, de fait, liés à la composition du système. Cette connexion est matérialisée par une relation “isPartOf”. “ZoneLimit” correspond à un seuil de dégradation défini pour un composant. Lorsque ce seuil est atteint, une alerte, de classe “WarningMessage”, est émise.

Cette ontologie aborde donc la maintenance en tenant compte de la structure du système considéré et des éléments nécessaires à la détection des symptômes, des pannes et des causes de défaillance d'un composant.

## 5.2.2 Ontologie de documentation technique

KOUKIAS et al. (2015) propose une ontologie structurant la documentation technique. Cette structure est illustrée sur la figure 5.6.

Selon cette ontologie, un composant, de classe “Component” est décrit par une classe “Description” et remplit une fonction “Function”. Pour s'assurer de la réalisation de cette fonction, des actions de maintenance, de classe “Activity”, doivent être exécutées. Ces actions peuvent prendre plusieurs formes comme la réalisation de mesures se modélisant par les concepts “MeasurementActivity” et “Measure”. Ces actions permettent de prévenir ou corriger des pannes. Le protocole de réalisation de la tâche est défini dans une procédure. Celle-ci peut nécessiter l'usage d'accessoires et de ressources. On distingue ici la ressource, qui est un consommable, de l'accessoire qui est un élément essentiel ou supplémentaire d'un bien. Par exemple, un embout de visseuse serait considéré comme un accessoire. Enfin, les actions de maintenance

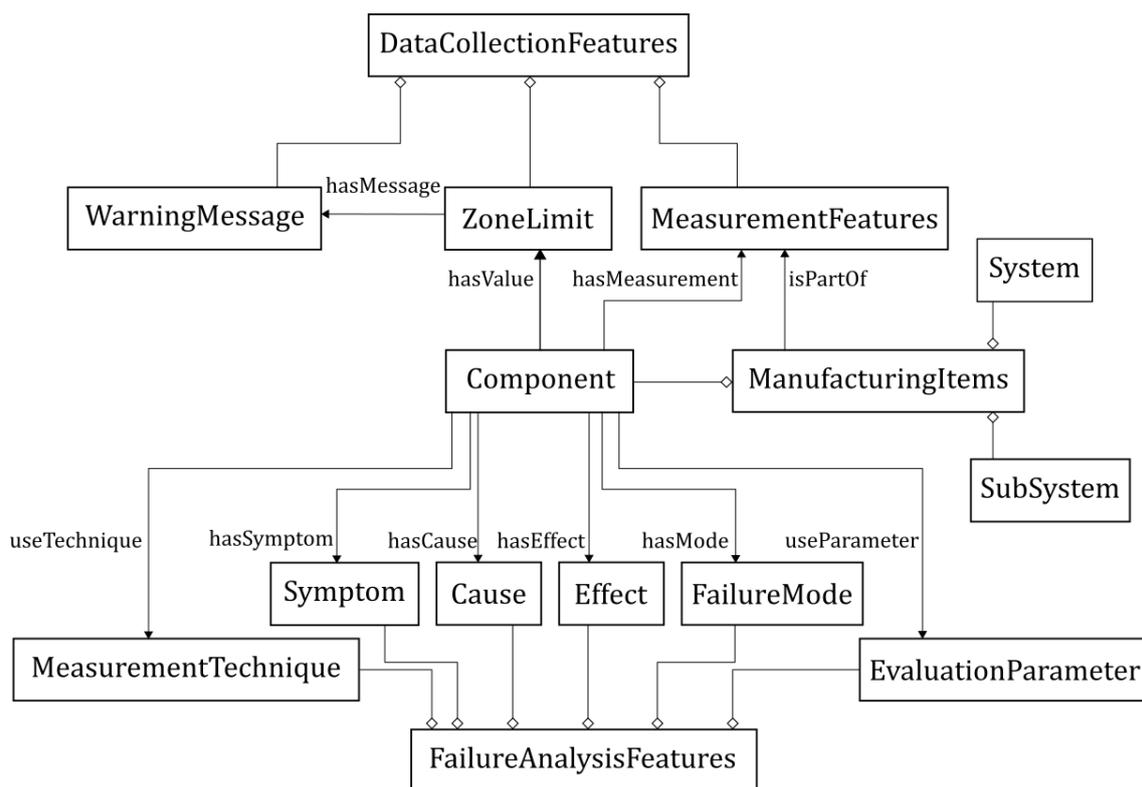


Figure 5.5 – Ontologie OntoProg (NUNEZ et al. 2017)

peuvent être effectuées dans un certain ordre suivant une liste d’instructions. Cette liste peut être systématique ou dépendre de critères comme le résultat d’un test.

Cette ontologie présente une vision complète du processus de réalisation des tâches de maintenance. Toutefois, elle ne tient pas compte des problématiques de surveillance, de détection d’anomalies, d’identification des pannes et de pronostic. Cependant, une fois la panne identifiée, cette ontologie peut servir de support pour guider les agents de maintenance dans la réalisation de leurs tâches.

### 5.2.3 Ontologie de diagnostic SemDia

Dans le domaine du diagnostic, l’ontologie SemDia, détaillée en figure 5.7, présente une structure des concepts de détection d’anomalies et d’identification des pannes (MEHDI et al. 2017).

Dans cette approche, on considère trois modules :

- Le module *équipement* décrit la structure et l’environnement d’un système industriel. Ce système, désigné par la classe “Equipment”, fait partie d’un site industriel, “DeploymentSite”, appartient à un groupe fonctionnel, possède une configuration et doit être constitué d’au moins un composant physique.
- Le module de *surveillance* s’applique aux composants physiques au travers des capteurs, de classe “SensingDevice”. Ceux-ci observent des propriétés comme

Figure 1 Technical documentation ontology (see online version for colours)

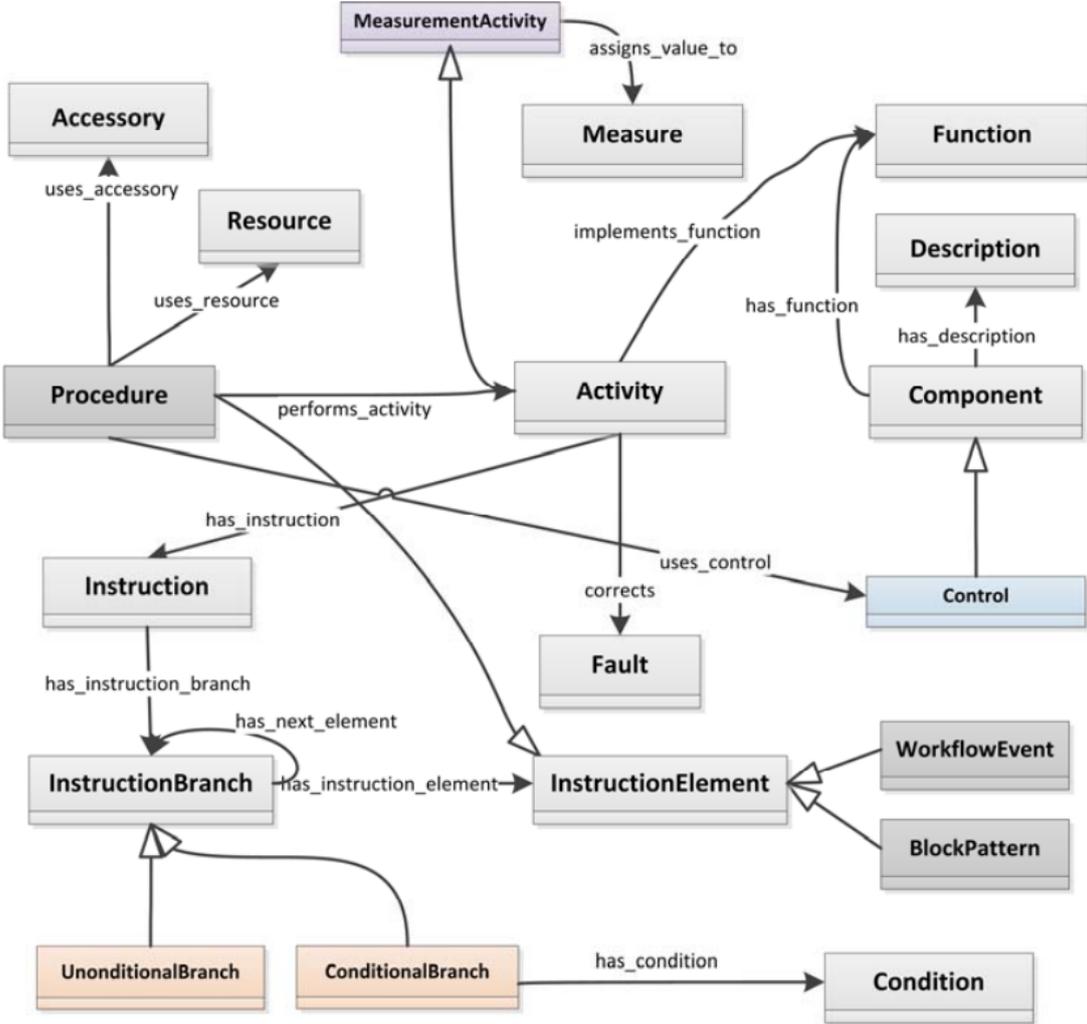


Figure 5.6 – Ontologie de documentation technique (KOUKIAS et al. 2015)

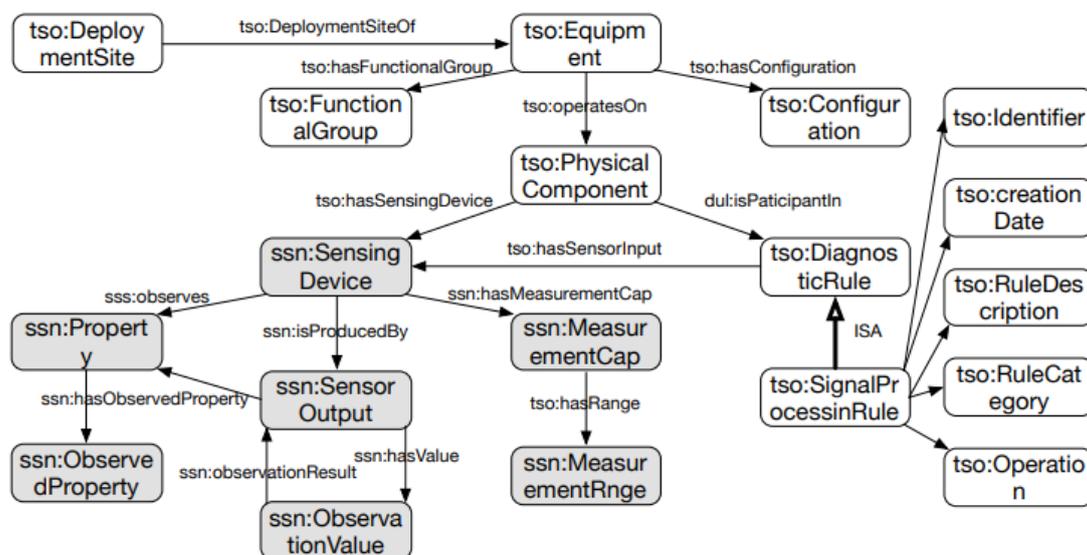


Figure 5.7 – Ontologie de diagnostic SemDia (MEHDI et al. 2017)

la température ou la pression et renvoient les valeurs mesurées, de classe “SensorOutput”. Ces capteurs sont caractérisés par leur capacité, désignée par “MeasurementCap”, qui inclut la précision ou encore la fenêtre d’observation.

- Le module des *règles de diagnostic* regroupe les règles permettant d’identifier les pannes et les relie au composant physique concerné. Ces règles peuvent être de différents types. Dans la figure 5.7, l’exemple fourni est celui des règles de traitement de signal. Ces règles peuvent posséder un certain nombre de propriétés comme un identifiant, une date de création, une description, une catégorie ou encore une opération devant être appliquée aux données des capteurs pour la détection d’anomalies.

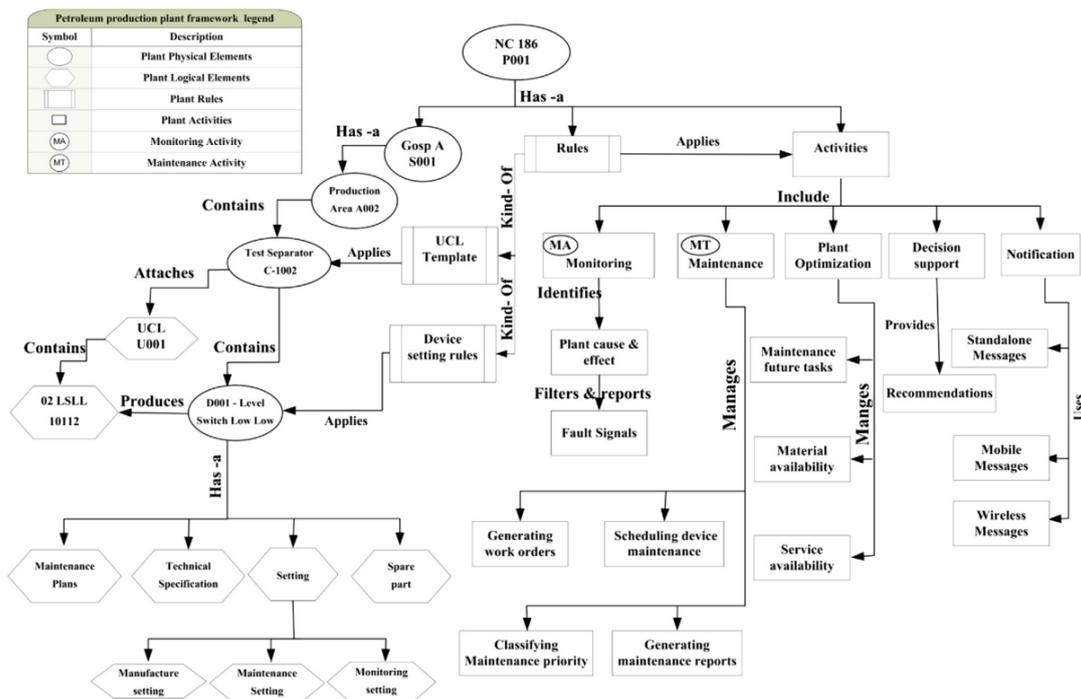
L’une des limites de cette approche réside dans la faible complexité des variables de surveillance ainsi que l’absence de distinction entre la règle de diagnostic et la panne identifiée. De plus, le pronostic ne fait pas partie du périmètre de cette ontologie.

## 5.2.4 Ontologie de surveillance et de maintenance de processus industriel

(ELHDAD et al. 2013) propose une ontologie de maintenance et de surveillance à l’échelle d’un site industriel appliqué au domaine de l’industrie pétrolière. La structure de cette ontologie est présentée en figure 5.8.

Dans cette ontologie, quatre types de concepts sont manipulés :

- *les éléments physiques* qui décrivent la composition d’un système en cinq niveaux de granularité : l’usine, l’unité de séparation du pétrole et du gaz, la zone de production, l’équipement et le composant. Ces éléments sont respectivement



**Figure 5.8** – Ontologie de surveillance et de maintenance de processus industriel (ELHDAD et al. 2013)

désignés par NC 186 P001, GOSP A S001, A002, C-1002 et D001 dans le modèle de la figure 5.8.

- les éléments logiques sont constitués d'un ensemble d'unités de contrôle logiques (UCL U001) réalisant le diagnostic d'un équipement sur la base des signaux issus de la surveillance de ses composants (02 LSSL 10112).
- les règles permettant de paramétrer le diagnostic réalisé par l'unité de contrôle logique et les signaux à observer pour chaque composants.
- les activités qui se décomposent en quatre types : les activités de surveillance, de maintenance, d'aide à la décision et d'alerte.

Bien que cette ontologie soit appliquée à l'industrie pétrolière, les concepts identifiés et la classification présentée sont transposables à d'autres activités industrielles.

## 5.2.5 Ontologie de diagnostic de pannes

Un autre exemple d'ontologie de diagnostic de panne est proposé par F. XU et al. (2018) en figure 5.9.

La structure de cette ontologie est centrée autour de la notion de panne. Une "Fault" est liée à un équipement défaillant qui peut être un équipement, "Device", un sous-système d'équipement, "System", un ensemble de composants, "Assembly" ou un composant, "Part". L'opération de maintenance à réaliser dépend de la panne et de

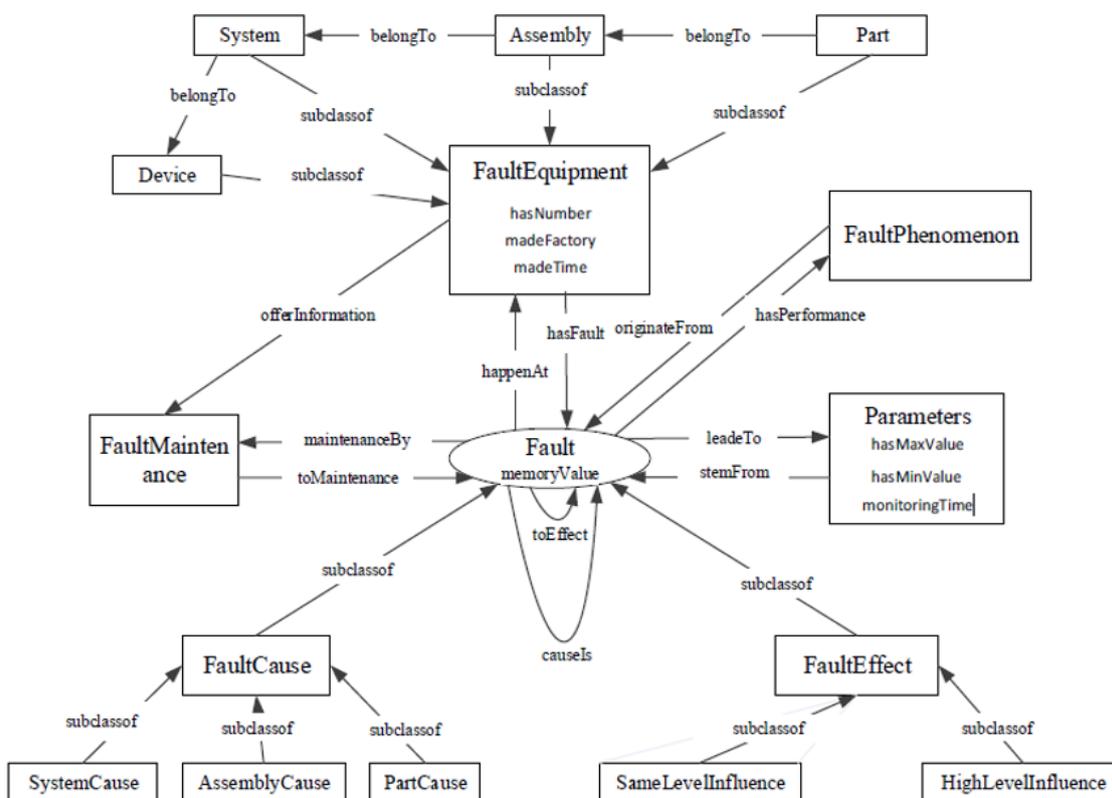


Figure 5.9 – Ontologie de diagnostic de pannes (F. XU et al. 2018)

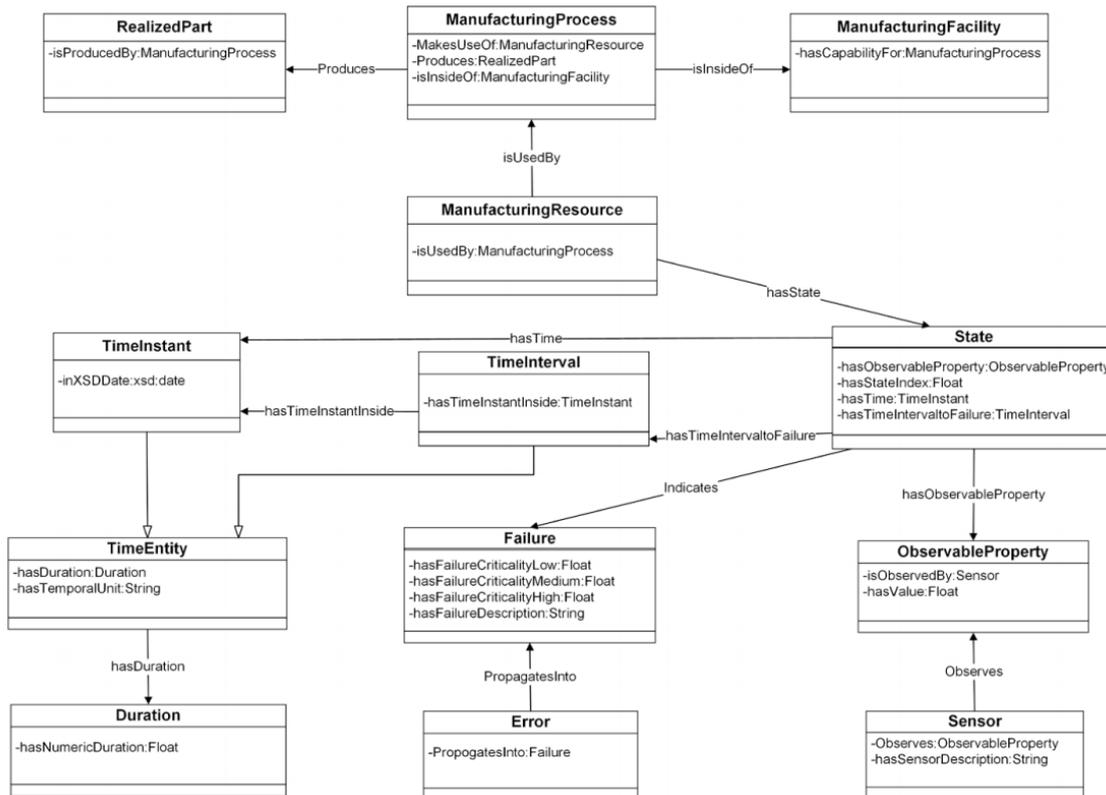


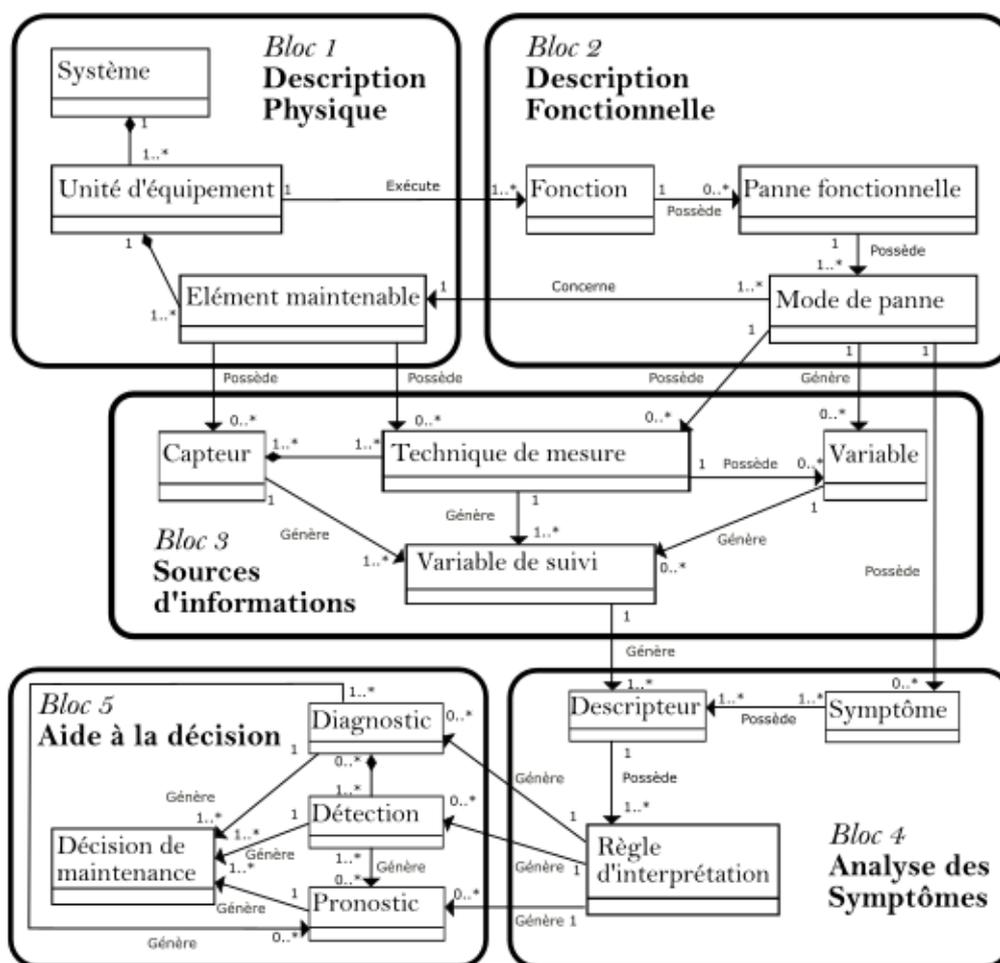
Figure 5.10 – Structure de l'ontologie MFPO (CAO et al. 2019)

l'équipement défaillant. La panne est générée par une cause, "FaultCause", produit des effets, "FaultEffect", et se caractérise par des symptômes représentés en figure 5.9 par la classe "FaultPhenomenom". Une classe "Parameters" identifie les signaux et seuils nécessaires à l'identification de la panne.

## 5.2.6 Ontologie de prévision de pannes industrielles

(CAO et al. 2019) présente une ontologie de prévision de pannes industrielles nommée MFPO (Manufacturing Failure Prediction Ontology). La structure de cette ontologie est détaillée en figure 5.10.

Dans cette ontologie, l'état de santé d'un composant ou système, désigné par une entité de type "State", est déterminé à partir de propriétés observables surveillées par capteurs. Cet état de santé est défini à un instant précis, de type "TimeInstant". Il peut indiquer la présence d'une panne résultant d'une ou plusieurs erreurs. La durée estimée avant l'apparition d'une panne est matérialisée par une entité de type "TimeInterval". L'état de santé s'applique à un bien de type "ManufacturingResource". Ce bien fait partie d'un processus de production intégré dans une infrastructure et réalisant un produit.



**Figure 5.11** – Méta-modèle de maintenance conditionnelle (GUILLÉN et al. 2016)

## 5.2.7 Méta-modèle de maintenance conditionnelle

(GUILLÉN et al. 2016) propose un méta-modèle générique permettant de supporter une maintenance conditionnelle. Ce méta-modèle est présenté en figure 5.11.

Les classes de ce méta-modèle peuvent être regroupées en cinq catégories, ou blocs :

- **Description physique** : ce bloc décrit la composition du système avec un point de vue semblable à la nomenclature. Un système, correspondant ici au bien complet, est composé d'au moins une unité d'équipement. L'unité d'équipement est, de fait, assimilable à un sous-système. Elle est elle-même composée d'au moins un élément maintenable qui est le plus petit composant maintenu. Un élément maintenable peut éventuellement être décomposé dans la nomenclature mais en pratique, il est remplacé ou réparé dans son ensemble lors de la maintenance.
- **Description fonctionnelle** : ce bloc présente le système selon une perspective fonctionnelle. Les fonctions remplies par les unités d'équipement sont réper-

torisées. Elles peuvent être associées à des pannes fonctionnelles, elles-mêmes liées à au moins un mode de panne. Un mode de panne correspond ici à la défaillance d'un élément maintenable.

- **Sources d'informations** : cette catégorie regroupe les éléments permettant de générer les variables de suivi de la maintenance conditionnelle et définit les classes nécessaires à la phase de surveillance, comme présentée en section 2.2.2. Elle contient les capteurs associés aux éléments maintenables et les variables issus d'autres sources comme le système d'information de maintenance. Ces variables peuvent, par exemple, se présenter sous la forme d'indicateurs statistiques calculés à partir d'un historique d'interventions. Contrairement aux capteurs, elles ne sont pas liées aux éléments maintenables mais directement aux modes de pannes. Les variables de suivi peuvent être générées directement à partir des capteur et variable ou par une transformation de ces données au moyen de techniques de mesure.
- **Analyse des symptômes** : cette partie détaille les éléments nécessaires à la détection d'anomalies. Les variables de suivi générées dans le bloc 3 sont transformées en descripteurs. Un descripteur est défini comme une caractéristique, ou une variable, générée à partir d'observations externes ou de paramètres bruts ou prétraités. La différence entre descripteur et variable de suivi est due au fait qu'un descripteur peut être généré à partir de plusieurs variables de suivi, caractérise des symptômes et modes de pannes et produit au moins une règle d'interprétation. Les symptômes sont les effets qualitatifs entraînés par un mode de panne. Ce dernier peut posséder plusieurs symptômes qui sont utilisés pour leur identification au moyen de descripteurs. Les règles d'interprétation ont pour but d'analyser précisément les valeurs des descripteurs afin de détecter les anomalies et d'identifier les symptômes et modes de panne associés. Les règles d'interprétation doivent être les plus précises possible et notamment tenir compte du contexte d'utilisation du système. Si l'on prend l'exemple d'un avion, un comportement normal en phase de décollage peut, en effet, s'avérer anormal en phase de vol. Tout descripteur possède au moins une règle d'interprétation et potentiellement plusieurs.
- **Aide à la décision** : ce bloc détaille les processus exploitant les données des blocs précédant. Un bloc de détection permet d'exploiter les règles d'interprétation afin d'identifier les symptômes et modes de défaillance qui affectent ou vont affecter le système. Cette détection peut alimenter un processus de diagnostic et/ou de pronostic. Le diagnostic a pour but de déterminer l'élément maintenable affecté par le mode de panne. Il se base sur le résultat du processus de détection et/ou de l'exploitation des règles d'interprétation. Le pronostic consiste à déterminer la durée restante avant la défaillance. Ce processus est alimenté par le diagnostic, la détection et les règles d'interprétation. Les informations ainsi générées sont transmises à un module de décision de maintenance chargé d'exploiter ces données pour faciliter la prise de décision de l'utilisateur. Cette étape peut, par exemple, générer des scénarios de maintenance en tenant compte des contraintes logistiques et économiques.

Ce méta-modèle, défini à partir de termes normalisés, donne une vision assez complète des éléments nécessaires à la mise en place d'une solution PHM. Il servira par la suite de base à la mise en place de la structure de la base de connaissances.

## 5.2.8 Méta-modèle retenu pour le SE

A partir des exemples présentés précédemment, un méta-modèle générique, adapté à tous types de systèmes pour le PHM, doit être défini.

En particulier, le méta-modèle de GUILLÉN et al. (2016), basé sur des concepts normatifs, a servi de référence principale. Cependant, cette solution possède quelques limites. Premièrement, les notions de fonction et de pannes fonctionnelles n'apportent pas de plus value à la réalisation des activités d'une architecture PHM et ne devrait donc pas être conservées dans la base de connaissances. De plus, les règles liées aux processus de détection, d'identification de pannes et de pronostic ne sont pas clairement reliées aux éléments de la base de fait dans le méta-modèle de Guillen. Enfin, la structure présentée ne permet pas d'assurer un suivi des pièces interchangeables et de leur usure.

Pour ces raisons, une adaptation du méta-modèle de GUILLÉN et al. (2016) a été réalisée et présentée en figure 5.12.

Ce modèle contient trois types de concepts : les noeuds, les arcs et les règles. Les noeuds et les arcs sont les éléments atomiques de la base de faits sur laquelle viennent s'appliquer les règles, contenues dans la base de règles. Les types de règles s'appliquent sur des types d'arcs, eux-mêmes définis entre deux types de noeuds. Par exemple, les règles de détection s'exécutent sur les arcs de type "Peut déclencher". Ceux-ci sont, par ailleurs, définis entre des noeuds de type "Descripteur" et "Symptôme".

Dans ce modèle, on retrouve les éléments du bloc "Source d'information" de GUILLÉN et al. (2016) : les capteurs, les variables, les techniques de mesures et les variables de suivi. Ces variables de suivi permettent d'alimenter un descripteur qui peut déclencher un symptôme si la règle de détection est validée. Suivant le même processus, un ou plusieurs symptômes actifs peuvent permettre de déterminer un mode de panne avec une règle d'identification de panne. Les modes de pannes activés permettent alors de connaître l'élément maintenable concerné. Une panne pouvant provenir de la défaillance de plusieurs éléments maintenables potentiels, des règles d'identification d'élément défaillant, liés aux arcs de type "Affecte", peuvent être nécessaires. Dans le cas d'une défaillance anticipée, une règle de pronostic, appliquée sur un arc de type "Risque d'affecter", peut être utilisé pour estimer la RUL de la pièce concernée.

La distinction entre l'entité "Pièce" et l'entité "Élément maintenable" provient du fait que l'élément maintenable correspond à l'emplacement de la pièce du point de vue de la nomenclature du système. En revanche, la pièce est le composant physique situé à cet emplacement. La pièce possède donc un numéro de série, une marque, un modèle et une durée de vie pouvant être recalculée par l'application de modèles de pronostic. De plus, plusieurs pièces peuvent se succéder sur un même appareil et pour un même élément maintenable. L'élément maintenable est considéré comme le niveau de nomenclature le plus bas. Ils s'assemblent ensuite en systèmes s'agréant

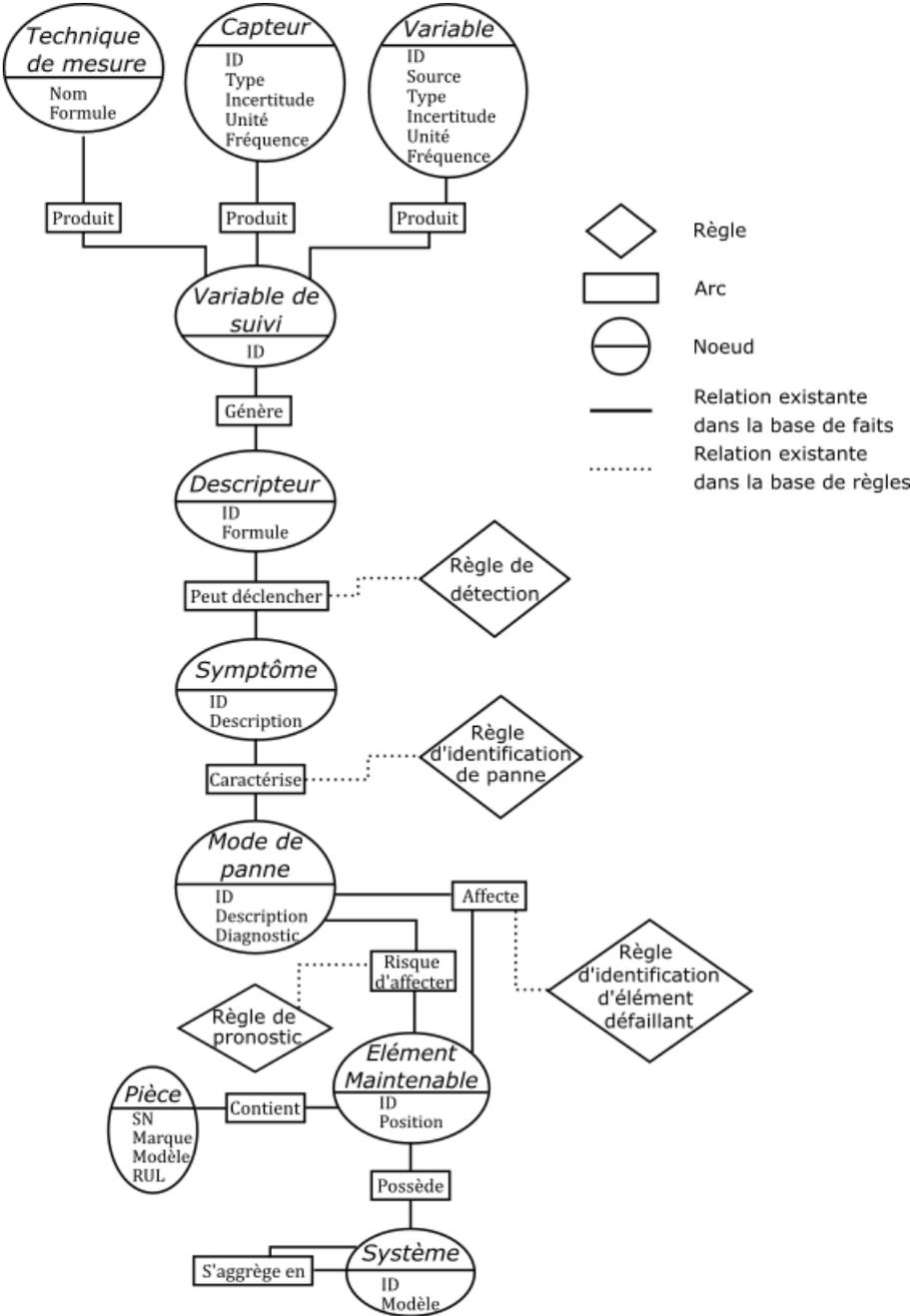


Figure 5.12 – Modèle de structure de la base de connaissances

entre eux pour former des systèmes plus complexes. On remarque que ce modèle de base de connaissances intègre à la fois des éléments statiques conceptuels issus de la documentation et les pièces, qui sont données d'exploitation.

Les éléments statiques conceptuels proviennent de différentes sources :

- La composition du système jusqu'aux éléments maintenables est générée à partir de sa nomenclature.
- Les modes de panne, symptômes et descripteurs proviennent des analyses de fiabilité.
- Les capteurs, variables, techniques de mesure et variables de suivi sont issus d'un système de collecte de données.

Afin d'illustrer l'implémentation de ce méta-modèle, un exemple, tiré de GUILLÉN et al. (2016), est présenté en figure 5.13.

Dans cet exemple, le système considéré est une porte constituée d'un panneau et de deux actionneur qui assurent son ouverture et sa fermeture. L'actionneur 2, noté A2, sert de doublon à l'actionneur 1, noté A1. La position de chaque actionneur est détectée par un capteur d'ouverture et de fermeture. Ces capteurs produisent des variables de suivi. Les valeurs de ces variables sont booléennes notées VS\_A1\_FO, VS\_A1\_FC, VS\_A2\_FO et VS\_A2\_FC. Ces variables de suivi sont alors transformées en descripteurs. Dans cet exemple, les valeurs des descripteurs sont égales à celles des variables de suivi dont ils sont issus. Cependant, ce n'est pas toujours le cas. Un descripteur peut être généré par transformation d'une ou plusieurs variables de suivi.

Les descripteurs générés sont alors analysés par une règle de détection RD1 située sur la relation de type "Peut déclencher" avec le symptôme S1. Dans cet exemple, on cherche à détecter la situation présentant des valeurs différentes entre A1\_FO et A2\_FO et entre A1\_FC et A2\_FC. On notera que la règle d'interprétation peut faire appel à des attributs situés sur d'autres noeuds ou arcs. Lorsque les conditions d'application de la règles sont remplies, le symptôme S1 est alors actif. Ce symptôme est lié au mode de panne d'un des actionneurs, notée P1, qu'il active lorsque les conditions de la règle d'identification de panne RIP1 sont remplies. P1 peut concerner les éléments maintenables (EM) des actionneurs 1 ou 2 qui constituent la porte avec le panneau. Pour déterminer l'élément maintenable défectueux, une règle d'identification d'élément défectueux RIED1 est définie sur les relations entre P1 et les EM des actionneurs. En cas de panne, cette règle peut mettre à jour la RUL de la pièce liée à l'EM accusé sur l'appareil concerné.

Dans ce chapitre, la conception de la base de connaissances a été abordée selon deux points de vue. Tout d'abord, suite à une revue des solutions de graphes de connaissances, le HIN a été choisi comme mode de stockage des données. Ce choix permet une meilleure gestion des systèmes complexes impliquant une volumétrie et une hétérogénéité des données. Ensuite, une structure de données permettant de manipuler les concepts nécessaires au support d'une solution PHM a été définie. Le choix de cette structure s'appuie sur une revue des ontologies et méta-modèles de maintenance identifiés dans la littérature. Le méta-modèle obtenu ne dépend pas

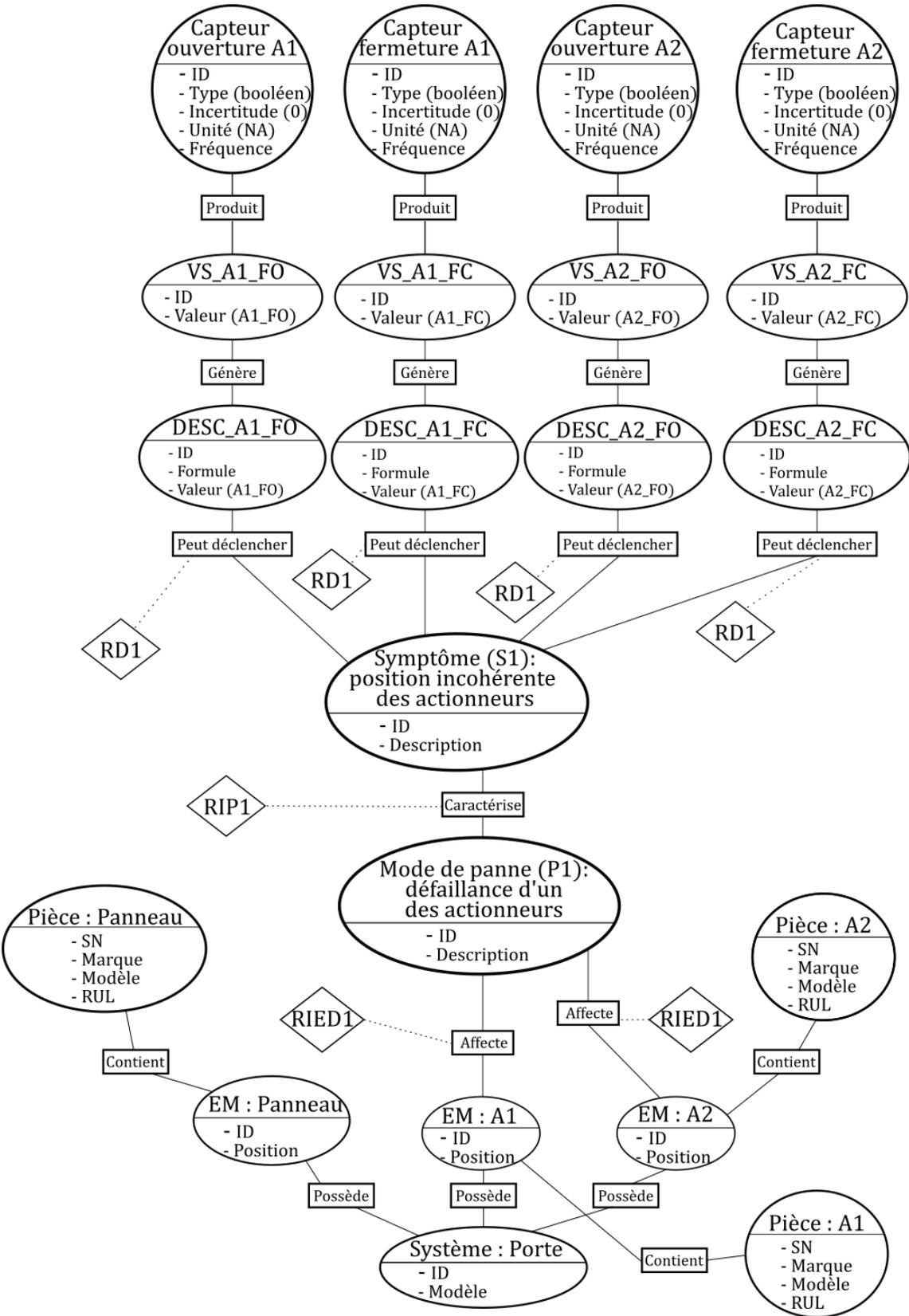


Figure 5.13 – Exemple de base de connaissances

d'un système ou d'un type de systèmes particulier. A ce titre, il répond à l'objectif de généralité de la solution proposée.

Bien que la base de connaissances soit au coeur du SE, son utilité dépend principalement du moteur d'inférences choisi pour l'exploiter. Dans le prochain chapitre, un moteur d'inférences adapté à l'exploitation d'un HIN sera présenté et les mécanismes d'application des règles seront détaillés.

# Chapitre 6

## Moteur d'inférences

Une fois la base de connaissances définie sous forme de HIN, il est nécessaire de développer un moteur d'inférences adapté pour son exploitation. Celui-ci doit répondre à deux objectifs :

- Appliquer les règles métier aux éléments de la base de fait
- Prévenir les conflits entre ces règles

L'ensemble des règles est défini par  $R = \{\gamma_1, \dots, \gamma_n\}$  où  $n$  correspond au nombre de règles. Chaque règle  $\gamma_i$  possède un type  $\delta$  et est représentée et intégrée grâce à une combinaison d'opérateurs algébriques prédéfinis permettant :

- d'intégrer de nouvelles connaissances : ajout de noeuds
- de lier des objets existants : ajout d'arcs
- de parcourir le graphe : chercher des chemins dans le graphe selon certaines règles

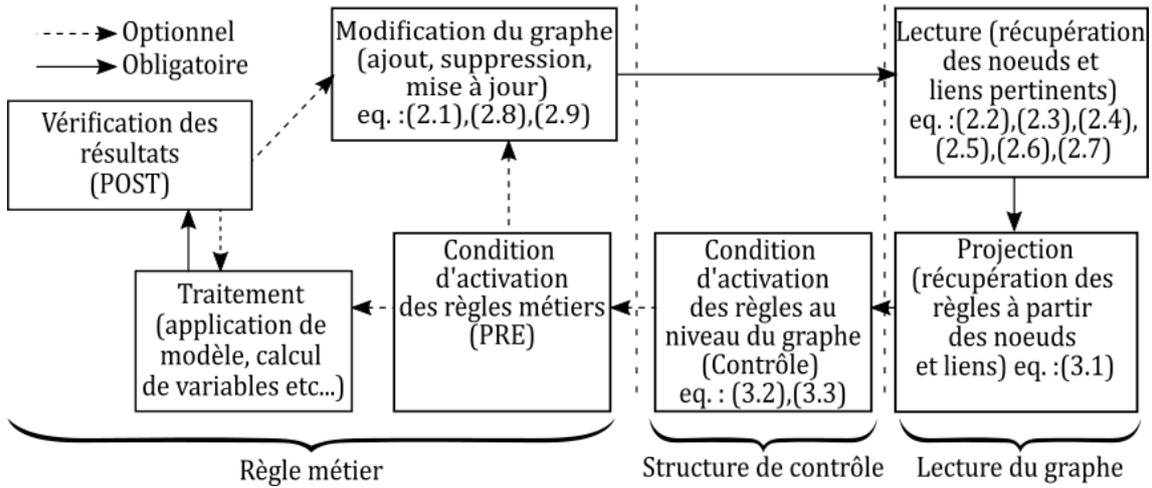
Le processus d'application des règles du moteur d'inférences est détaillé sur la figure 6.1.

Le déclenchement du processus provient d'une modification du graphe. Cette modification peut prendre trois formes :

- *L'ajout* de nouveaux noeuds et/ou d'arcs
- *La mise à jour* de noeuds et/ou d'arcs existants
- *La suppression* de noeuds et/ou d'arcs existants

La réalisation de ces opérations se base sur l'exploitation du graphe et des fonctions 2.1, 2.8 et 2.9 définies précédemment.

Une fois la modification est réalisée, une phase de lecture permet d'extraire un sous-graphe de noeuds et arcs connectés à l'élément altéré. Selon le type d'élément modifié, la profondeur et les types de noeuds et arcs sélectionnés peuvent varier. La sélection de ce sous-graphe a pour but de diminuer les temps de traitement en limitant le périmètre d'action aux éléments pertinents au lieu du graphe entier.



**Figure 6.1** – Moteur d'inférences : processus d'application des règles

En se basant sur le résultat de la phase de lecture, une fonction de projection est alors appliquée pour récupérer les règles présentes dans le sous-graphe. Cette fonction renvoie les règles situées sur les arcs de type  $\mu$  reliant deux noeuds de types  $\tau_1$  et  $\tau_2$ . La formule générale est exprimée par :

$$Projection : \tau_1 * \tau_2 * \mu \rightarrow R_j \text{ avec } R_j \subset R \quad (6.1)$$

Une fois les règles du sous-graphe récupérées, une structure de contrôle doit intervenir pour vérifier leur applicabilité. En effet, une modification du graphe peut déclencher l'application de plusieurs règles. De plus, cette étape permet, en complément de la structure acyclique du graphe, d'éviter d'initier une boucle infinie de traitements. Pour cela, la structure de contrôle s'appuie sur la fonction "Contrôle" qui applique une contrainte aux attributs des noeuds et arcs. Le résultat de cette fonction est une liste des règles applicables ordonnée ou non. La formule générale de cette fonction est donnée par :

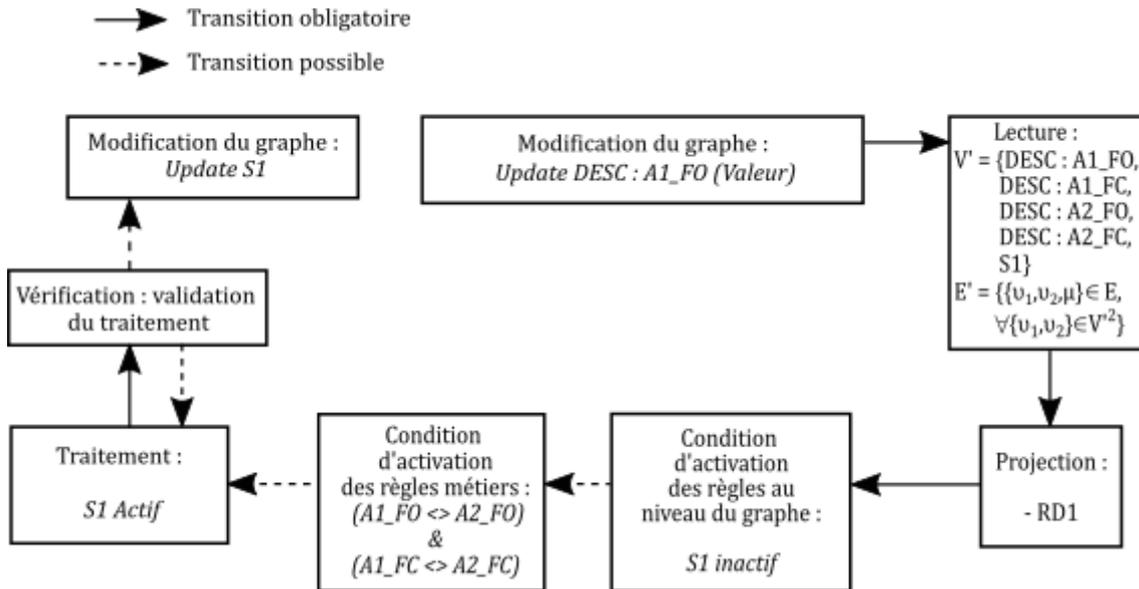
$$Contrôle_V : X_i * V_j * R_k \rightarrow R'_k \text{ avec } X_i \subset X, V_j \subset V, R'_k \subset R_k \subset R \quad (6.2)$$

$$Contrôle_E : X_i * E_j * R_k \rightarrow R'_k \text{ avec } X_i \subset X, E_j \subset E, R'_k \subset R_k \subset R \quad (6.3)$$

L'application de ces fonctions de contrôle permet ainsi d'identifier les règles applicables. On considère qu'une règle est généralement composée de deux parties principales : la condition d'activation et le traitement.

Le traitement n'est effectué que si la condition d'activation est remplie. Dans le cadre de la maintenance, cette dernière serait, par exemple, le dépassement d'une valeur de seuil pour la détection d'anomalies. Une fois le seuil atteint et la condition, validée, un traitement est alors réalisé.

Celui-ci se décompose en deux étapes. Tout d'abord, un premier traitement peut consister à calculer des variables à partir de modèles ou d'algorithmes complexes



**Figure 6.2** – Illustration du processus d'application des règles

comme des modèles d'identification de panne, de composant défaillant ou de pronostic. Il peut aussi altérer temporairement le graphe en créant, supprimant ou modifiant des noeuds et/ou des arcs.

Dans un second temps, ce résultat est vérifié dans une étape de post traitement afin de valider les résultats précédemment obtenus et/ou d'entériner d'éventuelles modifications du graphe. Ces modifications peuvent être des ajouts ou suppressions de noeuds et/ou d'arcs. Des valeurs d'attributs peuvent également être mises à jour. Ces modifications peuvent par la suite déclencher de nouvelles règles, démarrant ainsi un nouveau cycle. La vérification des résultats permet ainsi d'éviter une altération erronée du graphe et sert de rempart supplémentaire au déclenchement d'une boucle infinie.

A partir de ce processus, les règles du SE peuvent être appliquées à une base de données de type HIN en s'assurant de l'applicabilité des règles. Du fait de la nature cyclique du processus, la structure du HIN doit être acyclique pour éviter les boucles infinies.

La conception de ce SE a fait l'objet d'une publication dans le journal Expert System With Applications (SARAZIN et al. 2021).

La figure 6.2 illustre ce processus d'application des règles sur la base de l'exemple présenté dans le chapitre précédent.

En particulier, cet exemple concerne l'activation d'un symptôme sur un appareil suite à la modification de la valeur d'un descripteur. Suite à cette modification, la fonction de lecture permet d'identifier et de lister les noeuds et arcs pertinents au traitement de ce type d'altération. La liste des noeuds comprend le descripteur modifié, le symptôme S1 ainsi que les autres descripteurs auxquels ce dernier est

relié. La liste des arcs contient toutes les relations entre les noeuds précédemment identifiés.

Cet ensemble de noeuds et de arcs constitue un sous-graphe sur lequel est réalisée la phase de projection. Dans cet exemple, la seule règle identifiée est la règle de détection d'anomalie RD1. Puisqu'il n'y a qu'une seule règle, la prévention des conflits est assez limitée dans cet exemple.. En revanche, on peut considérer que l'applicabilité de la règle dépend de l'absence préalable de détection du symptôme S1. Si cette condition est vérifiée, la règle métier est alors déclenchée. Si plusieurs règles étaient applicables en même temps, une fonction de contrôle, définie suivant les équations 6.2 et 6.3, indiquerait leur applicabilité et/ou leur ordonnancement.

La première partie de la règle métier RD1 consiste à détecter une anomalie de position des actionneurs au travers des variables *A1\_FO*, *A2\_FO*, *A1\_FC*, *A2\_FC*. Cette anomalie se traduit par la vérification de la condition :

$$(A1\_FO \langle \rangle A2\_FO) \& (A1\_FC \langle \rangle A2\_FC)$$

.

Si cette condition est remplie, le symptôme S1 est alors activé. Cette activation est ensuite validée dans pendant la phase de vérification avant d'être rendue effective par la modification du symptôme S1 dans le graphe.

Cet exemple simple permet d'illustrer un cycle d'application des règles métiers sur le HIN. La modification engendrée peut alors enclencher un nouveau cycle pour identifier une panne avérée ou à venir.

Le SE ainsi défini permet de stocker et d'exploiter une connaissance structurée pour un système de maintenance conditionnelle tout en répondant aux limites d'efficacité, de scalabilité et d'adaptabilité.

Dans l'architecture présentée dans la section 3.1, ce SE est connecté aux CEP afin de réaliser la détection des anomalies. Il est, de fait, nécessaire d'extraire les règles de détection présentes dans la base et de les transformer pour les rendre compréhensibles par les CEP.

## Partie III

# Transformation de modèle

Dans la partie II, une solution de gestion de connaissances basée sur un SE a été présentée. Comme illustré dans la figure 6.3, le périmètre fonctionnel considéré dans la partie III correspond à la détection d'anomalie de l'architecture PHM. L'objectif est de proposer une solution répondant à certaines limites actuelles des SE et au besoin de traitement de données en temps réel en particulier. Le contenu de ce chapitre a donné lieu à la publication d'un article dans la conférence Modelsward (SARAZIN et al. 2020b). Suite à la conférence, l'article a été sélectionné et une version étendue a été rédigée (SARAZIN et al. 2020a).

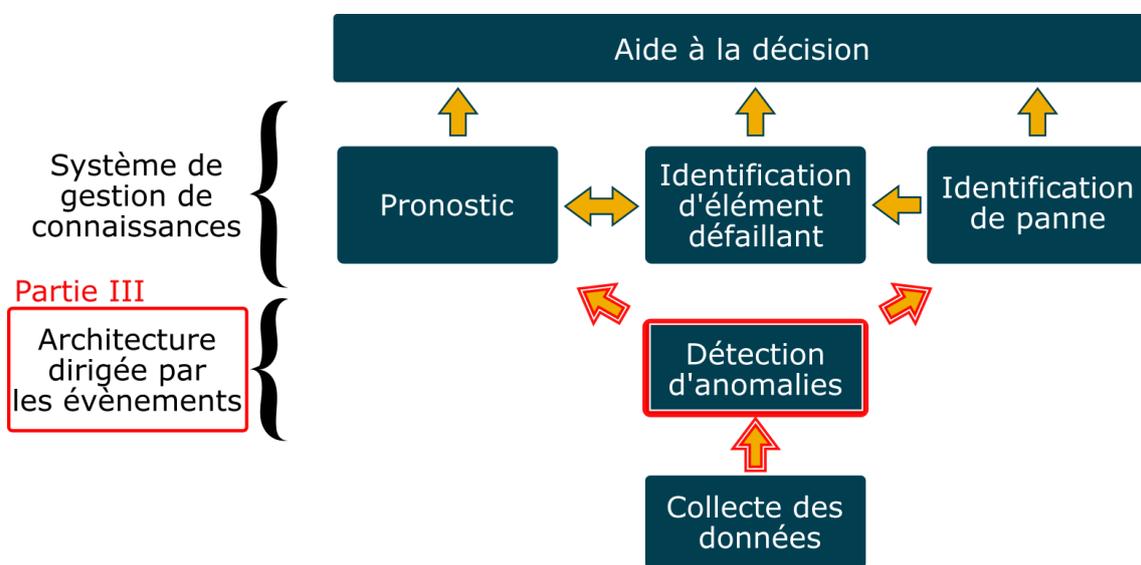


Figure 6.3 – Positionnement de la partie III dans l'architecture fonctionnelle

# Chapitre 7

## Architecture dirigée par les évènements

Dans la partie précédente, un SE permettant de formaliser, de structurer et d'exploiter la connaissance accumulée d'un système a été présenté. Ce SE s'appuie sur un HIN structuré grâce à un méta-modèle générique adapté au PHM décrit en section 5.2.8.

Cependant, la surveillance d'un système complexe en temps réel implique des problématiques de l'ordre du Big Data notamment les aspects suivants (SICULAR 2013) :

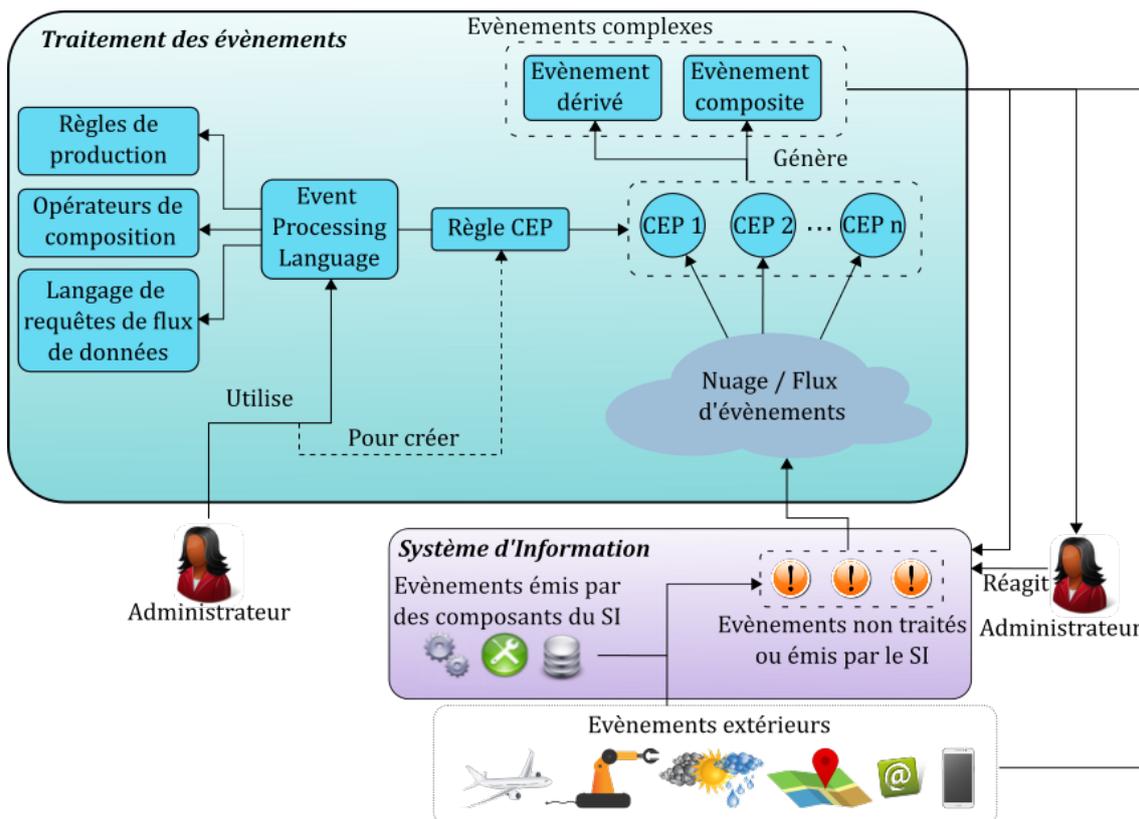
- Un volume de données d'entrée conséquent dû à un réseau de capteurs dont la complexité croît avec celle du système surveillé.
- Une vitesse de données élevée dans le cadre d'une surveillance en temps réel.
- Une variété importante des paramètres d'entrée à manipuler et analyser.

De plus, dans le cadre de la détection d'anomalies, une analyse instantanée des paramètres peut s'avérer insuffisante. En effet, en regroupant un nombre significatif d'observations dans un intervalle temporel prédéfini, on peut évaluer l'évolution de ces paramètres. Les fenêtres temporelles ainsi étudiées fournissent alors des informations précieuses et parfois nécessaires pour estimer l'état de santé du système surveillé.

Du fait de ce contexte d'utilisation, la surveillance du bien pour la détection d'anomalies ne peut pas être réalisée par l'implémentation d'un HIN avec à une base graphe. En effet, une base graphe n'est pas conçue pour gérer des paramètres d'entrée sous forme de séries temporelles. De plus, les données surveillées n'ont pas toutes vocation à être conservées dans la base. Seuls les symptômes détectés sont stockés pour alimenter le processus d'identification de panne.

Compte tenu des caractéristiques du HIN, il est nécessaire de gérer en amont la volumétrie et la vitesse des données afin d'en extraire uniquement les éléments pertinents ou déduits devant être traités par le HIN. Ainsi une solution complémentaire doit être utilisée. Au delà du traitement de la volumétrie et de la vitesse, cette solution doit également être en mesure de gérer des fenêtres temporelles ou d'occurrences pour générer des symptômes.

Les architectures dirigées par les évènements (EDA), présentées en section 2.2.2, permettent de répondre à ces besoins par :



**Figure 7.1** – Principe de fonctionnement d'une EDA basée sur des CEP (BARTHE 2013; FÜLÖP et al. 2012)

- leur capacité à gérer des événements de simples, complexes ou de flux.
- leur faculté à agir comme un filtre sur les données d'entrée pour réduire la vélocité
- la possibilité de générer des événements complexes en se basant sur des règles
- une conception adaptée à la gestion des flux avec un fort potentiel de parallélisation

Au sein d'une architecture EDA, l'utilisation de CEP (Complex Event Processing pour Traitement d'Evènements Complexes) peut répondre aux besoins identifiés. En effet, le CEP est une technologie de réseau pouvant s'abonner à des sources d'évènements diverses, comme des bases de données ou des applications en temps réel, pour déclencher des actions et créer de la connaissance sur une situation donnée. Il est chargé de l'évaluation d'une confluence d'évènements et d'une prise de mesures, suite à un nombre d'occurrences ou sur des périodes temporelles (ou fenêtres) de durées variables. Cette évaluation est réalisée au moyen de règles d'interprétation d'évènements sophistiquées, de définition de motifs et de correspondance d'évènements.

Ainsi, le CEP peut être utilisé pour la détection d'anomalies, de menaces et d'opportunités dans des environnements complexes et hétérogènes. La Figure 7.1 résume les principaux concepts du CEP (FÜLÖP et al. 2012).

Le traitement des événements s'applique sur un SI capable de recevoir des événements. Ceux-ci peuvent être générés par le SI lui-même ou provenir de sources externes comme des capteurs ou des applications (ERP, CRM, etc.). Ils sont alors récupérés et filtrés par les CEP grâce à des règles créées par l'administrateur. Lorsque les conditions d'activation des règles sont remplies, un événement complexe, signalant par exemple une anomalie détectée, est alors généré. La mise en place d'une architecture EDA contenant des CEP est donc particulièrement adaptée à la détection d'événements significatifs dans le nuage ou le flux d'événements émis par le SI et le monde extérieur. Cette détection s'appuie sur des règles formalisées dans un EPL (Event Processing language). Il est possible de classer les EPL selon quatre grands types (BARTHE 2013 ; ETZION et al. 2011) :

- Les langages de règles, qui peuvent eux-mêmes se décomposer en trois grandes familles :
  - Les règles de production, ou d'inférence, qui sont utilisées pour représenter les comportements de type SI condition ALORS action. Plusieurs règles de ce type peuvent se succéder, le résultat d'une règle  $n$  pouvant faire partie de la condition d'une règle  $n + 1$  (VIANU 1997).
  - Les règles actives, ou règles ECA (Event Condition Action). Les moteurs de règles actives, comme IBM WebSphere Business Events ou RAPIDE, détectent et réagissent aux événements entrants et exécutent des motifs d'événements.
  - Les règles basées sur la programmation logique, qui s'exécutent à partir d'assertions logiques. Le logiciel Prolog fait partie des solutions les plus connues fonctionnant sur ce principe.
- Les langages de programmation impératifs, dans lesquels les opérations sont formalisées par des séquences d'instructions élémentaires exécutables par le processeur pour modifier l'état du programme. De nombreux langages utilisent ce type de programmation comme le FORTRAN, COBOL, BASIC, Pascal, C, Ada, Perl, Python, PHP, ou encore Java.
- Les langages orientés flux qui se rapprochent du SQL et permettent de requêter des flux d'événements. Parmi les solutions adoptant ce type de langage se trouvent Esper EPL, Oracle CQL (Continuous Query Language) ou encore CCL (Continuous Computation Language).
- D'autres langages orientés flux mais non typés SQL comme SPADE (GEDIK et al. 2008).

Dans le cadre d'une démarche PHM, la mise en place d'une architecture EDA contenant des CEP permet une surveillance en temps réel d'un système intégrant la gestion des fenêtres temporelles. Cette solution est complémentaire du SE qui permet de structurer et stocker les informations connues sur la composition et la fiabilité du système et à les exploiter par raisonnement pour le diagnostic, le pronostic et l'identification des équipements défaillants.

Toutefois, l'implémentation de cette architecture requiert un paramétrage des CEP avec les règles de surveillance contenues dans la base de connaissances du SE. Cette

connexion doit s'accompagner de transformations car le formalisme des règles dans la base de connaissance est un formalisme métier qui n'est pas directement exploitable par un CEP. Afin de réaliser cette transformation, une architecture dirigée par les modèles a été choisie pour transformer les règles métier de détection d'anomalies en langage EPL (Event Processing Language).

# Chapitre 8

## Architecture dirigée par les modèles

### 8.1 Principe

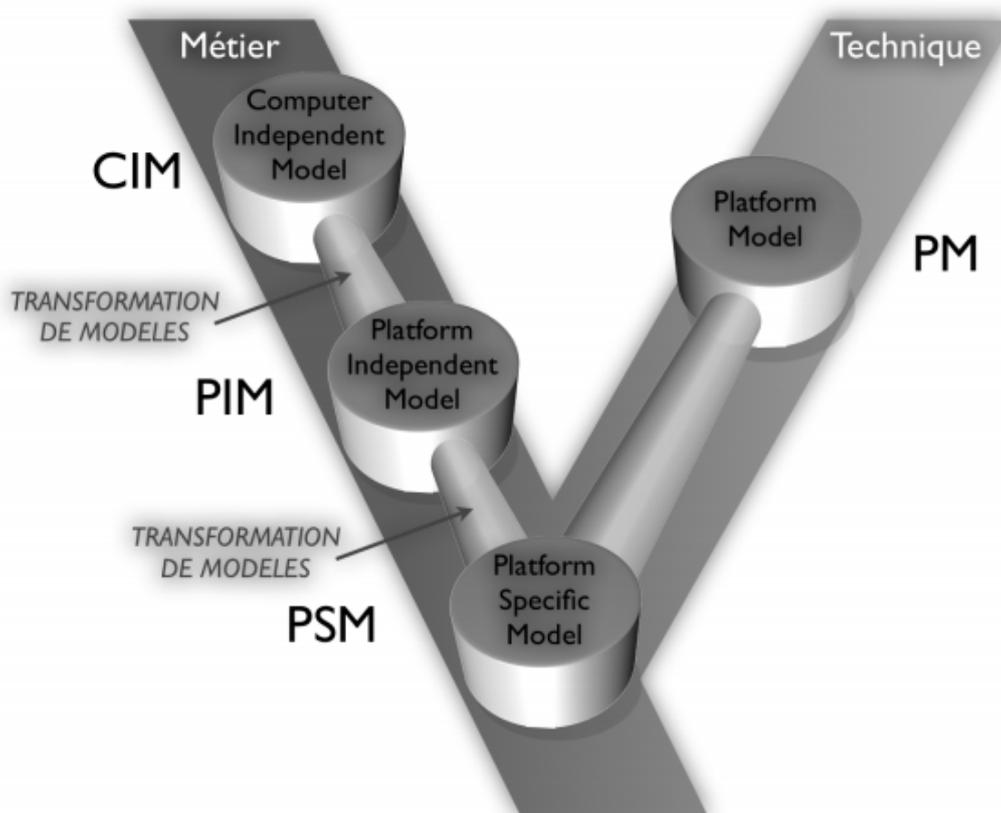
L'utilisation d'un ensemble de CEP pour la détection d'anomalies repose sur l'application de règles de surveillance sur les flux d'évènements suivis. Ces règles de surveillance peuvent provenir d'une documentation technique ou d'une expertise métier formalisée et structurée dans la base de connaissances du SE. Cette connaissance n'est cependant pas directement exploitable par un CEP. Chaque moteur CEP possède un formalisme de règle qui lui est propre.

Dans le cadre de ces travaux de thèse, la solution choisie est le moteur Esper. Le choix de cette solution a été motivé par la grande variété des opérateurs disponibles, notamment les fenêtres d'occurrence et temporelle, et une performance permettant de traiter plus de 500 000 évènements par seconde. De plus, sa nature open-source adaptée à un projet de prototypage à visée académique, (BOUBETA-PUIG et al. 2014).

La méthodologie MDA (Model Driven Architecture) a été adoptée pour effectuer la transformation des règles depuis la base de connaissances vers l'EPL. Cette méthodologie a été conçue afin de différencier les concepts métiers manipulés dans un système des solutions techniques mises en oeuvre pour les implémenter (SIEGEL 2014). Basée sur les modèles, elle est utilisée en développement logiciel pour distinguer les étapes de conception, formalisant la logique métier, des étapes d'implémentation. D'après SIEGEL (2014), elle a pour effet d'améliorer la portabilité, l'interopérabilité et la réutilisation de l'application développée.

Comme décrit dans la Figure 8.1, le MDA repose sur quatre types de modèles :

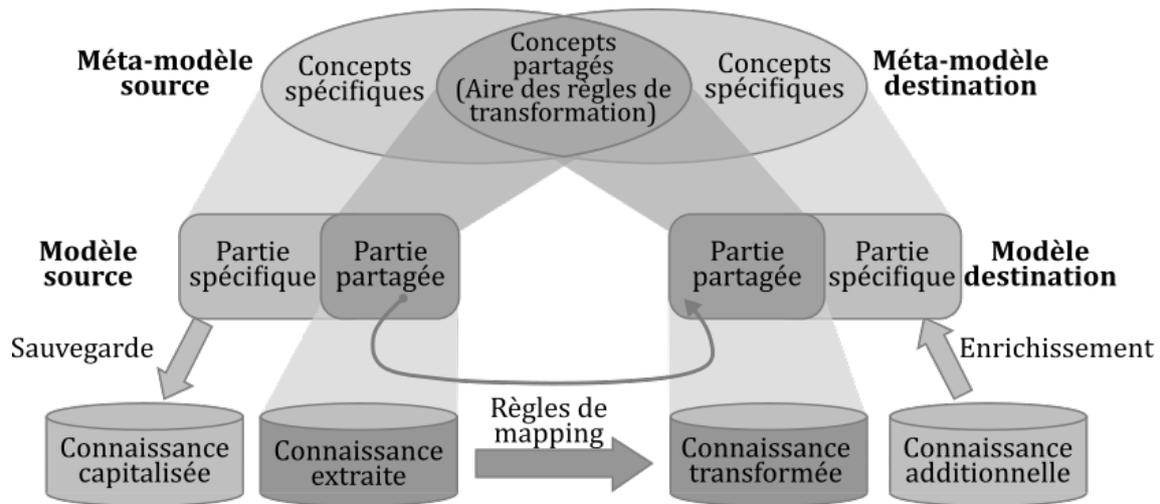
- Le modèle indépendant du calcul (Computation Independent Model CIM)
- Le modèle indépendant de la plateforme (Platform Independent Model PIM)
- Le modèle de la plateforme (Platform Model PM)
- Le modèle spécifique à la plateforme (Platform Specific Model PSM)



**Figure 8.1** – Schéma de principe de la vision MDA (BENABEN 2012)

Le CIM est conçu à partir de la formalisation de la logique métier par un expert du domaine utilisant son propre langage. D'après la méthodologie MDA, une fois le CIM défini, le PIM peut être réalisé pour inclure un premier niveau de spécification. Son objectif est de rendre possible l'adaptation du CIM à différentes plateformes du même type. Le PM décrit la plateforme utilisée pour implémenter le modèle. Il présente les concepts manipulés et la structure reliant ces concepts entre eux. Enfin, le PSM est défini comme une vision d'un système du point de vue spécifique de la plateforme. C'est l'implémentation du PIM sur la plateforme modélisée par le PM (BELAUNDE et al. 2003). La méthodologie MDA consiste ainsi à effectuer des transformations du CIM au PIM puis du PIM au PSM.

Une transformation de modèle se définit comme une opération de transformation  $M_t$  partant d'un modèle source  $M_a$  et produisant un modèle cible  $M_b$  (Jean BEZIVIN et al. 2006). Dans ces transformations, tous les concepts du modèle source ne sont pas toujours partagés avec le modèle cible. De fait, la première étape consiste à identifier les concepts partagés par ces modèles, constituant le domaine de transformation, et les concepts spécifiques à chacun. La figure 8.2 illustre les principes de la transformation de modèle avec les domaines des concepts partagés et spécifiques. Dans le domaine de transformation, les concepts sont convertis du modèle source au modèle cible au moyen de règles de transformation, aussi appelées règles de mapping. La partie spécifique du modèle source est alors considérée comme de la connaissance capitalisée tandis que la partie spécifique du modèle cible est constituée de concepts supplémentaires



**Figure 8.2** – Principes de transformation de modèles (TRUPTIL et al. 2010)

provenant de sources de données extérieures à la transformation (TRUPTIL et al. 2010).

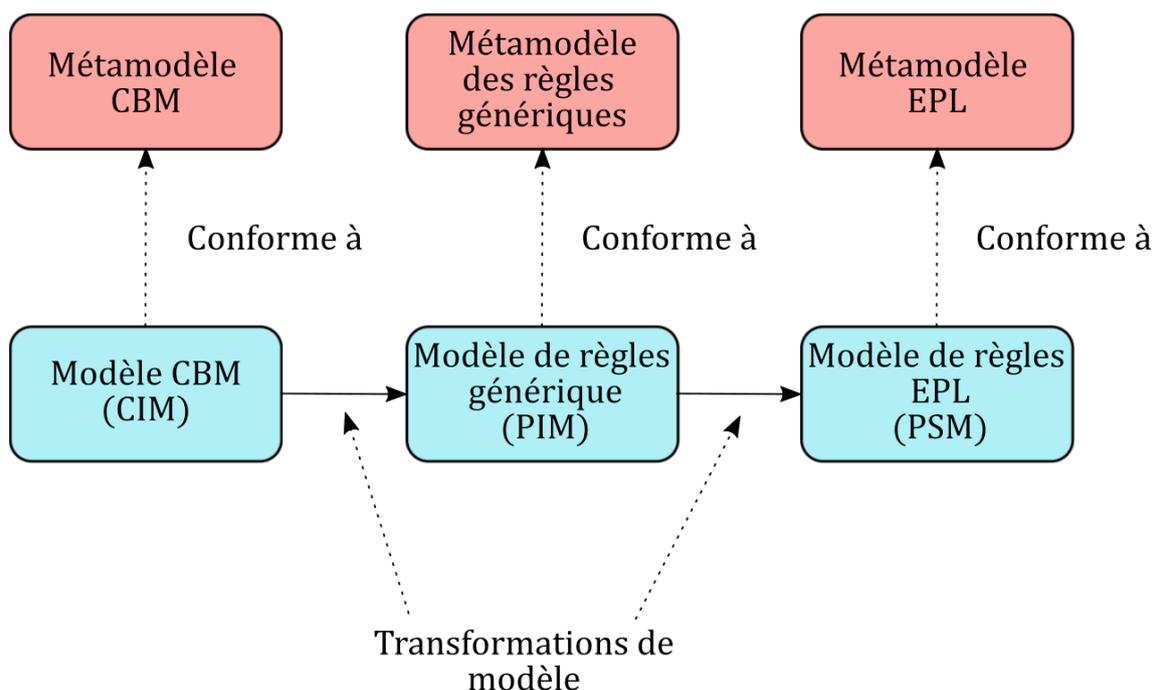
Dans le cadre de la transformation des règles de la base de connaissances en règles EPL, cette méthodologie se formalise par le processus détaillé en figure 8.3.

Les règles de la base de connaissances sont des règles de maintenance conditionnelle (CBM pour Condition Based Maintenance) appartenant au modèle précédemment défini dans le chapitre 5. Le modèle décrivant les règles dans un formalisme métier, indépendamment de la solution d'implémentation, correspond au CIM.

Une première transformation va consister à extraire et formaliser ces règles dans une structure générique. Le modèle cible, obtenu suite à la transformation, comprend un premier niveau de spécification sans dépendre d'une solution technique. Ce modèle correspond au PIM de l'architecture. Il sert à définir la structure d'une règle indépendamment de la solution technique utilisée pour permettre une meilleure adaptation à des évolutions technologiques entraînant des modifications du PSM.

Les règles obtenues sont alors transformées pour correspondre au formalisme EPL. Ce formalisme regroupe les concepts communs à tous les langages EPLs semblables au SQL. BOUBETA-PUIG et al. (2014) propose une solution permettant de convertir les règles d'un modèle EPL vers une implémentation de ces règles adaptée à Esper. Dans le cadre de ces travaux de thèse, le périmètre de recherche s'est donc porté sur la transformation des règles de la base de connaissances vers le formalisme EPL.

Dans la section 8.2, la transformation des règles CBM en modèle de règles générique sera décrit. La section 8.3 portera ensuite sur la transformation de ce modèle en modèle de règles EPL.



**Figure 8.3** – Transformation de modèles de la base de connaissances vers le langage EPL

## 8.2 Transformation des règles de surveillance vers un modèle de règles génériques

La première étape du processus de transformation des règles métier de la base de connaissances, appelées par la suite règles CBM (pour Condition Based Maintenance), dans un formalisme EPL passe par une première transformation vers un PIM. Dans ce chapitre, cette première transformation est présentée, en commençant par la définition du méta-modèle du PIM nommé méta-modèle des règles génériques. Ensuite, les règles de transformation de modèles sur la base des concepts partagés entre le CIM et le PIM seront détaillées.

### 8.2.1 Métamodèle des règles génériques

La première transformation de modèle a pour objectif de transformer les règles CBM en règles indépendantes de toute solution d'implémentation, autrement dit génériques. Le méta-modèle de la base de connaissances ayant été défini dans la section 5.2.8, celui des règles génériques doit maintenant être présenté. Son objectif est de définir la structure d'une règle pour améliorer l'adaptabilité de la transformation face à une modification ou évolution de la plate-forme technique utilisée, en l'occurrence de l'EPL. Cette structure doit donc être conçue indépendamment de toute solution technique en restant à un niveau conceptuel.

Ce méta-modèle est présenté en Figure 8.4.

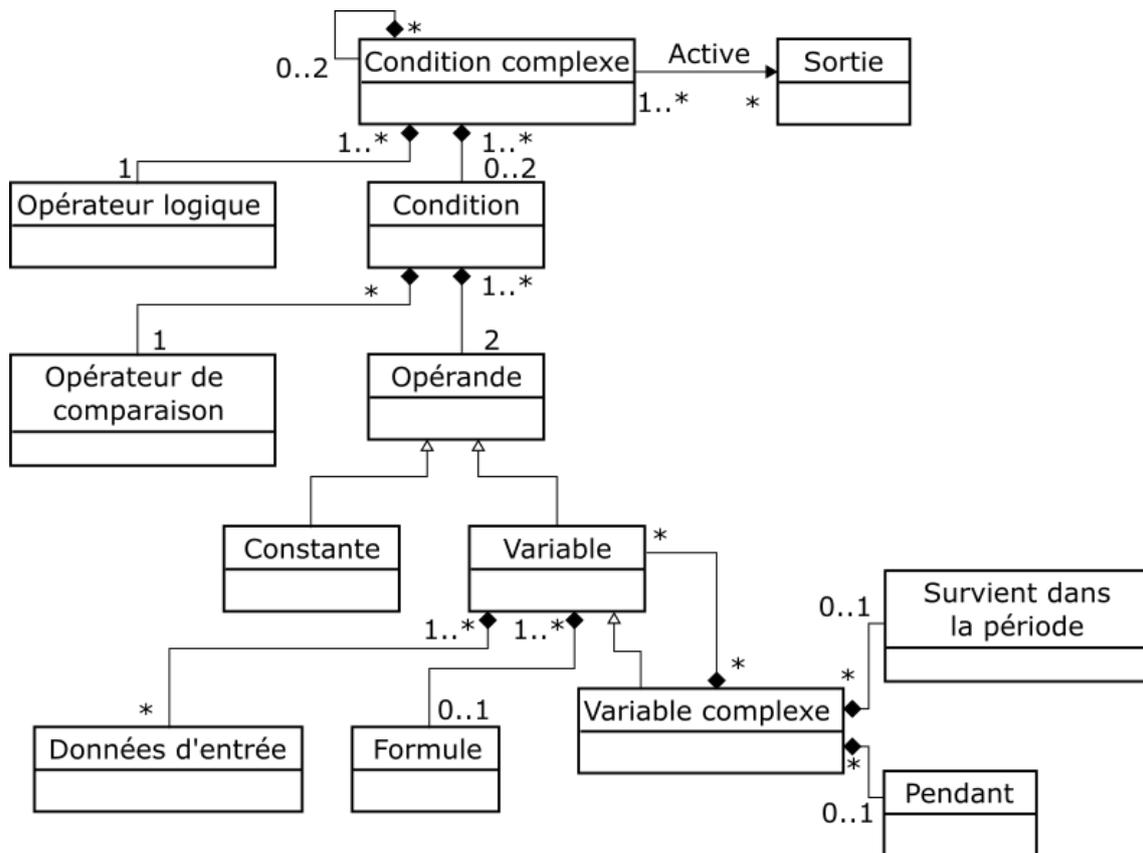


Figure 8.4 – Méta-modèle des règles génériques (SARAZIN et al. 2020b)

D'après ce méta-modèle, les données d'entrée génèrent des variables à l'aide de formules. Des variables complexes peuvent également être créées en appliquant une formule sur plusieurs autres variables et/ou en utilisant des opérateurs temporels. Les variables complexes sont des variables particulières généralement plus coûteuses à créer en temps et en calcul. Deux Opérateurs temporels sont présentés dans ce modèle : "Pendant" qui mesure la durée d'un évènement ou d'une situation et "Survient dans la période" qui compte le nombre d'occurrences d'un évènements dans un intervalle de temps prédéfini.

On considère ici qu'une condition est le résultat de la comparaison de deux opérandes au moyen d'un opérateur de comparaison (Exemples :  $\geq$ ,  $\leq$ ,  $=$ ). Les opérandes peuvent être des variables, simples ou complexes, ou des constantes. Plusieurs conditions peuvent s'agréger en conditions complexes avec des opérateurs logiques comme "ET" et "OU". Par exemple, la condition " $(a \geq b)OU(c \leq d)$ " est une condition complexe composée de deux conditions " $(a \geq b)$ " et " $(c \leq d)$ " liées par l'opérateur logique "OU". La condition ayant le plus haut niveau de complexité déclenche une règle qui génère une sortie. Cette sortie peut potentiellement être utilisée comme donnée d'entrée ou variable d'une autre règle. Une règle CBM peut ainsi générer plusieurs règles génériques.

Ce méta-modèle de règles génériques définit la structure du modèle cible de cette première transformation. Il est maintenant nécessaire de détailler les règles de mapping permettant de lier les modèles source et cible.

## 8.2.2 Transformation du CIM vers le PIM

Une fois les méta-modèles source et cible définis, la première étape de la transformation consiste à identifier les concepts partagés.

Dans le modèle source, ces concepts sont :

- les capteurs et variables pouvant être considérés comme données d'entrée
- les techniques de mesure qui forment une première étape de traitement
- les variables de suivi résultant de la transformation des données d'entrée
- les descripteurs définissant le traitement des variables de suivi
- les symptômes qui sont nécessaires dans la définition des actions à réaliser
- les règles de détection permettant de définir les conditions d'activation, incluant les opérateurs et les constantes, comme les valeurs de seuil.

Afin de lier ces éléments aux concepts du modèle cible, les règles de mapping détaillées dans la Table 8.1 doivent alors être appliquées.

D'après ces règles, les concepts de capteur et de variable du modèle source correspondent aux données d'entrée du modèle cible. Dans le modèle cible, ces données d'entrée sont alors transformées en variable au moyen d'une formule. Ce processus est similaire à la génération d'une variable de suivi à partir de techniques de mesures appliquées à un capteur ou une variable dans le modèle source. Par conséquent, les

**Table 8.1** – Règles de mapping du modèle CIM vers le PIM

Concepts du CIM	Concepts du PIM
Capteur	Données d'entrée
Variable	Données d'entrée
Technique de mesure	Formule Pendant
Descripteur	Survient dans la période Formule Variable
Variable de suivi	Variable complexe Variable Constante
Règle de détection	Opérateur de comparaison Opérateur logique Condition Condition complexe
Symptôme	Sortie Sortie

techniques de mesure sont identifiables à des formules et les variables de suivi aux variables du modèle cible.

Un descripteur est créé à partir de la transformation d'une ou plusieurs variables de suivi au moyen d'une formule et/ou des opérateurs temporels. Il peut donc être associé à plusieurs concepts du méta-modèle cible. Selon le contenu du descripteur, des formules et des variables peuvent être créées dont des variables complexes incluant des opérateurs temporels.

Les règles de détection définissent les conditions appliquées à un descripteur déclenchant une action. A l'instar des descripteurs, elles peuvent être liées à plusieurs concepts du méta-modèle cible. Elles sont en effet identifiables aux conditions complexes et ainsi générer les conditions et les opérateurs logiques et de comparaison qui les composent. Si une règle de détection est trop complexe, une sortie peut être générée et intégrée comme variable dans une nouvelle règle. Un tel exemple sera présenté dans le chapitre 9. L'activation d'une règle de détection entraîne l'activation d'un symptôme qui constitue alors une sortie.

Ces règles de mapping permettent de générer automatiquement un modèle de règle générique à partir du modèle de la base de connaissances précédemment défini. Cette première transformation d'un CIM vers un PIM doit ensuite être suivie d'une seconde transformation du PIM vers le PSM, qui correspond au modèle d'une règle EPL.

## 8.3 Transformation des règles génériques en EPL

Afin de transformer le PIM en PSM, il faut commencer par définir le méta-modèle cible. Dans le cadre de ces travaux de thèse, la solution développée par BOUBETA-PUIG et al. (2014) servira de référence pour la définition des règles en EPL et leur implémentation dans le CEP Esper par une approche model-to-code. Après avoir présenté le PSM, les règles de mapping liant le PIM au PSM seront détaillées.

### 8.3.1 Présentation du méta-modèle EPL

Selon BOUBETA-PUIG et al. (2014), l'un des inconvénients des systèmes CEP est leur complexité apparente. Afin de faciliter le travail des experts du domaine dans l'implémentation de solutions CEP, un méta-modèle décrivant l'EPL et une solution model-to-code ont été conçus par BOUBETA-PUIG et al. (2014). Ces travaux ont pour objet de simplifier la génération de règles CEP dans des solutions commerciales à partir de modèles EPL. Le méta-modèle EPL proposé par BOUBETA-PUIG et al. (2014) est détaillé dans la Figure 8.5.

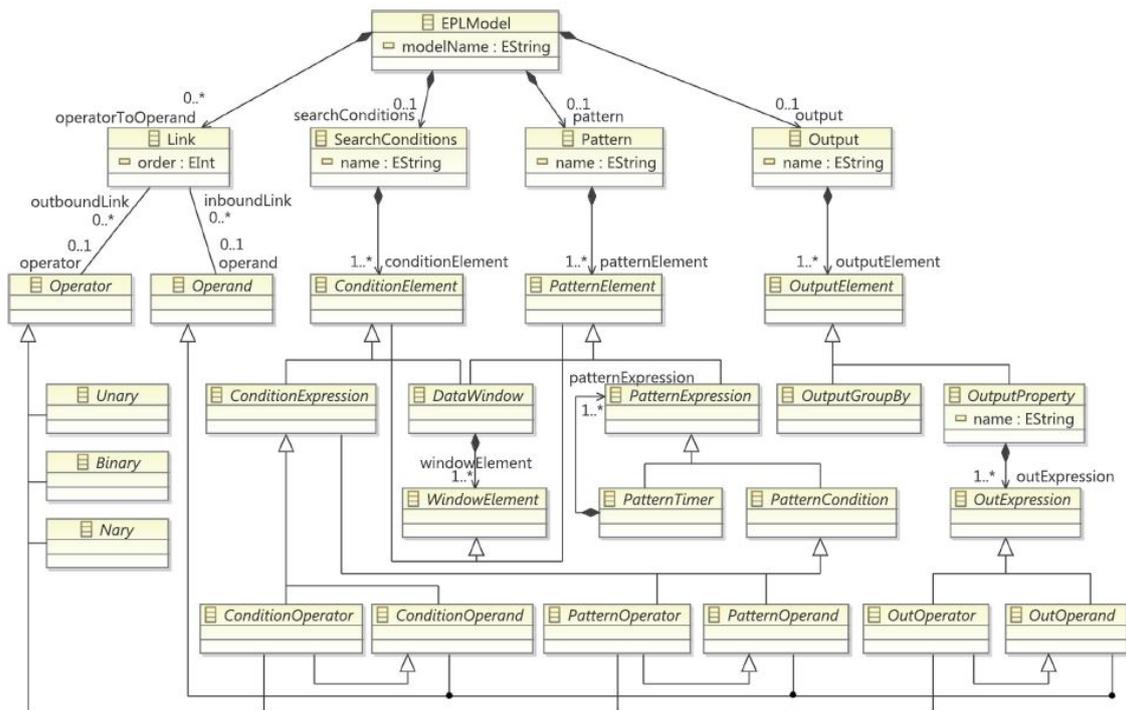


Figure 8.5 – Méta-modèle EPL (BOUBETA-PUIG et al. 2014)

Il se décompose en quatre principaux types de composants : “SearchConditions”, “Pattern”, “Output” et “Link”. L'exemple de règle suivant est utilisé pour illustrer ces types de composants : “SI  $x > y$  ET Moyenne( $k,l,m$ )  $> y$  ALORS  $z$ ”. Dans

cet exemple,  $x$ ,  $k$ ,  $l$  et  $m$  sont des données d'entrée assimilable à des observations de variables issues de capteurs tandis que  $y$  est une valeur de seuil. La variable  $z$  correspond à la sortie générée par la règle. Elle peut représenter l'activation d'un symptôme

Les éléments "Link" sont conçus pour relier les éléments de types "SearchConditions", "Pattern" et "Output" entre eux. L'ensemble des éléments "Link" se subdivise en opérands (comme  $x$ ,  $k$ ,  $l$ ,  $m$ ,  $y$  et  $\text{Moyenne}(k,l,m)$ ) et en opérateurs (comme  $>$ ). Les opérateurs s'appliquent sur une ou plusieurs opérands. Ils peuvent être Unaire, Binaire ou N-aire selon le nombre d'opérands concernées.

La composante "SearchConditions" est un ensemble d'éléments de type "ConditionElement". Ces derniers sont des conditions composées de "ConditionExpression" (comme " $x > y$ " et " $\text{Moyenne}(k,l,m) > y$ ") liées par des opérateurs logiques. Les éléments de type "ConditionExpression" sont construits à partir de "DataWindow" (comme  $x$ ,  $y$  et  $\text{Moyenne}(k,l,m)$ ), agissant comme opérands associées au moyen d'opérateurs.

Les "DataWindow" sont des transformations de "WindowElement" (comme  $k$ ,  $l$ ,  $m$  et  $x$ ) qui sont des données d'entrée. Le composant "Pattern" est défini comme "un modèle spécifiant les conditions qui peuvent correspondre à des ensembles d'événements connexes" (BOUBETA-PUIG et al. 2014). Il est utilisé pour gérer des événements se produisant dans un délai donné. Par exemple, un "PatternCondition" peut compter les occurrences d'un type particulier d'événement tandis que le "PatternTimer" précise la durée d'une période de surveillance.

Enfin, le composant "Output" (comme  $z$ ) spécifie les caractéristiques de l'événement complexe généré par l'activation des règles. Il peut être composé de plusieurs événements possédant chacun des propriétés et générés en utilisant des expressions.

Plus de détails sur les éléments de ce modèle sont disponibles dans BOUBETA-PUIG et al. (2014).

### 8.3.2 Transformation du PIM vers le PSM

Une fois le méta-modèle du PSM défini, les règles de mapping permettant de le générer à partir du PIM doivent être détaillées. Ces règles sont présentées dans la table 8.2.

Dans ce tableau, les concepts du PIM de la Table reftabTM1 ont été regroupés pour simplifier la connexion avec les concepts du PSM.

Pour rappel, le CEP est conçu pour surveiller de grandes quantités de données en temps réel et détecter des comportements grâce à des règles qui respectent le méta-modèle EPL. Pour atteindre cet objectif, un CEP reçoit des données entrantes, appelées "WindowElement", qui doivent être combinées avec d'autres pour créer un "DataElement". La combinaison des données entrantes peut être fondée sur une "ExpressionPattern", comme le nombre d'occurrences au cours d'une période donnée.

**Table 8.2** – Règles de mapping du PIM vers le PSM

Concepts du PIM	Concepts du PSM
Donnée d'entrée	WindowElement
Variable	DataWindow or WindowElement
Formule	PatternExpression
Survient dans la période	PatternCondition
Pendant	PatternTimer
Variable complexe	DataWindow
Constante	Operand
Opérateur de comparaison	Operator
Opérateur logique	Operator
Condition	ConditionElement
Condition complexe	SearchConditions
Sortie	OutputElement

Une fois qu'un "DataElement" est généré, un CEP vise à identifier un comportement défini au travers d'un élément de type "SearchCondition" et qui est une combinaison de "ConditionElement".

Selon ces règles de cartographie, les données d'entrée du modèle source correspondent au "WindowElement" du modèle cible. Un "WindowElement" peut être transformé par une "PatternExpression", qui correspond à une formule, pour générer un "DataElement" identifiable à une variable dans le modèle source.

Concernant les variables complexes, elles sont similaires aux éléments de type "DataWindow" car elles sont générées sur la base d'une combinaison de variables grâce à des opérateurs d'agrégation. Cela signifie qu'une variable pourrait être un "WindowElement". Ainsi, les variables pourraient se référer à "DataWindow" ou à un "WindowElement".

L'opérateur "Survient dans la période" correspond à un élément de type "PatternCondition" tandis que l'opérateur "Pendant" est associé à un "PatternTimer". Une constante correspond à un "Operand", tandis que les opérateurs logiques et de comparaison sont liés aux éléments de classe "Operator". Une condition, composée d'opérandes et d'opérateurs, correspond à un "ConditionElement". Une condition complexe, composée de plusieurs conditions peut être assimilée à un "SearchCondition". Enfin, l'activation d'une condition complexe peut générer une sortie assimilable à un "OutputElement".

En outre, un "OutputElement" pourrait être utilisé comme un nouvel événement entrant dans le CEP sous la forme d'un "WindowElement" dans une autre règle.

Ces règles de mapping permettent de transformer les règles génériques en règles EPL. Dans le chapitre suivant, des exemples permettant d'illustrer ces transformations seront présentés.

# Chapitre 9

## Illustrations

Dans le chapitre précédent, une démarche d'ingénierie dirigée par les modèles, reposant sur les méta-modèles et les règles de transformations, a été proposée pour transformer des règles de détection de la base de connaissances en EPL. Cette transformation fait intervenir trois modèles (CIM, PIM et PSM) et deux ensembles de règles de mapping. Le premier permet de transformer le CIM en PIM tandis que le second définit le passage du PIM vers le PSM.

Le présent chapitre a pour but d'illustrer la démarche globale de la transformation du CIM vers le PSM en s'appuyant sur des exemples extraits de GUILLÉN et al. (2016). Dans un premier temps, le système considéré sera présenté. Ensuite les exemples seront représentés dans le formalisme du modèle CIM, ici celui de la base de connaissances. L'application des règles de mapping permettra alors de générer les PIM et PSM associés afin d'obtenir les règles en EPL.

### 9.1 Description de système

Dans le cas d'étude choisi, on cherche à détecter des anomalies sur un système chargé de la régulation de l'air d'une pièce. Pour assurer cette fonction, le système considéré est constitué de deux actionneurs A1 et A2 responsables de l'ouverture d'un panneau. Notons que A2 est utilisé comme doublon pour palier une défaillance éventuelle de A1. En fonctionnement normal, les deux actionneurs doivent donc être ouverts et fermés en même temps. Une situation anormale fait alors référence à (1) la défaillance d'un actionneur ou (2) à l'apparition d'une surpression dans la pièce. Pour détecter ces situations, des capteurs sont utilisés pour capter des données et les transmettre à l'architecture PHM.

Les variables suivies sont :

- deux variables booléennes indiquant la position des actionneurs :
  - FO : vaut 0 si l'actionneur n'est pas ouvert et 1 si ouvert
  - FC : vaut 0 si l'actionneur n'est pas fermé et 1 si fermé
- P : la pression dans la pièce.

Il est important de noter l'existence de période transitoires pendant lesquelles un actionneur, en cours d'ouverture ou de fermeture, n'est ni fermé, ni ouvert. Cet état transitoire explique la présence de deux variables pour traduire la fermeture ou l'ouverture d'un actionneur.

Les variables disponibles en entrée pour détecter les anomalies sont donc A1\_FO, A1\_FC, A2\_FO, A2\_FC et P. Sur la base de ces données, les règles suivantes peuvent être utilisées pour la détection des défaillances.

1. **un des deux actionneurs est défaillant** : cette situation se produit lorsque A1 et A2 sont dans des positions différentes ou en cas d'une perte de signal empêchant la mise à jour d'un booléen. La condition permettant d'identifier cette situation s'exprime par :

$$SI (A1\_FO \neq A2\_FO) ET (A1\_FC \neq A2\_FC)$$

2. **Augmentation du risque de surpression dans la pièce** : Cette situation correspond à une augmentation de la pression dans la pièce de 0.14psid sur une période de 500ms en continu pendant au moins 1s ou si cette augmentation se répète trois fois en 10s. La Figure 9.1 illustre le mécanisme de détection d'une surpression dans une pièce. On simule la réception d'une valeur de pression P toutes les 500ms. A partir de ces données, on compte le nombre de surpressions sur des périodes de 1s à 10s. La première ligne indique qu'une surpression survient lorsque la valeur de  $\Delta P$  dépasse 0.14psi. La deuxième ligne indique la continuité du dépassement pendant 1s. La troisième ligne surveille l'apparition d'une troisième répétition du dépassement dans un intervalle de 10s. Lors d'une surpression, les compteurs pour chacune des périodes augmentent de 1. On observe par ailleurs que la valeur du compteur de la deuxième ligne diminue lorsque  $\Delta P \leq 0.14psi$ . Si la valeur d'un compteur dépasse le seuil autorisé, 2 pour la deuxième ligne et 3 pour la troisième, une alerte doit être émise.

Ce cas d'étude est pertinent car la première règle permettra d'illustrer la transformation d'une règle faisant intervenir plusieurs variables de surveillance tandis que la seconde détaille la gestion des fenêtres temporelles.

## 9.2 Présentation des CIM

Les modèles CIM des règles précédemment présentées, définis à partir du méta-modèle décrit dans la section 5.2.8, sont illustrées dans la Figure 9.2.

Dans le premier exemple, A1\_FO, A2\_FO, A1\_FC et A2\_FC sont surveillés et comparés par la règles de détection  $(A1\_FO \langle \rangle A2\_FO) ET (A1\_FC \langle \rangle A2\_FC)$ . Le symptôme concerné, noté "S1", est activé lorsque la condition de la règle de détection est remplie. Cet exemple permet de présenter la gestion de plusieurs variables d'entrée dans le processus de transformation.

Dans le second exemple, seule la pression est surveillée. Les données générées par le capteur sont transformées par une technique de mesure  $\Delta$  en variable de surveillance  $\Delta P$ . Cette variable produit un descripteur  $\Delta P$  de même valeur qui est traité par la

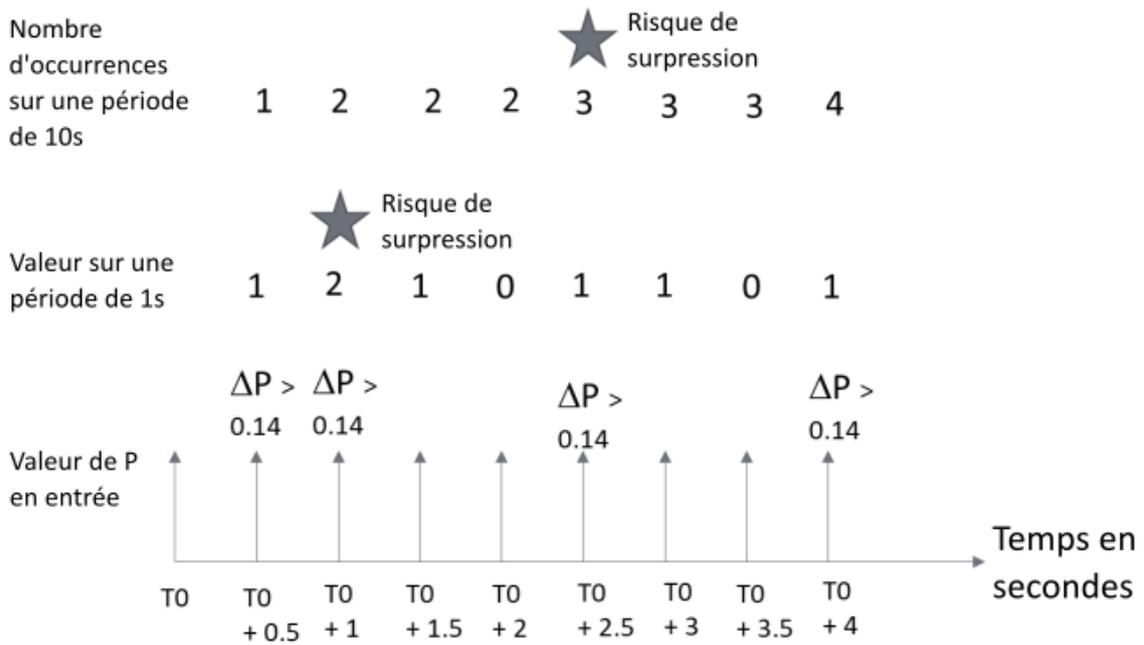


Figure 9.1 – Illustration de la détection d’une surpression dans la pièce

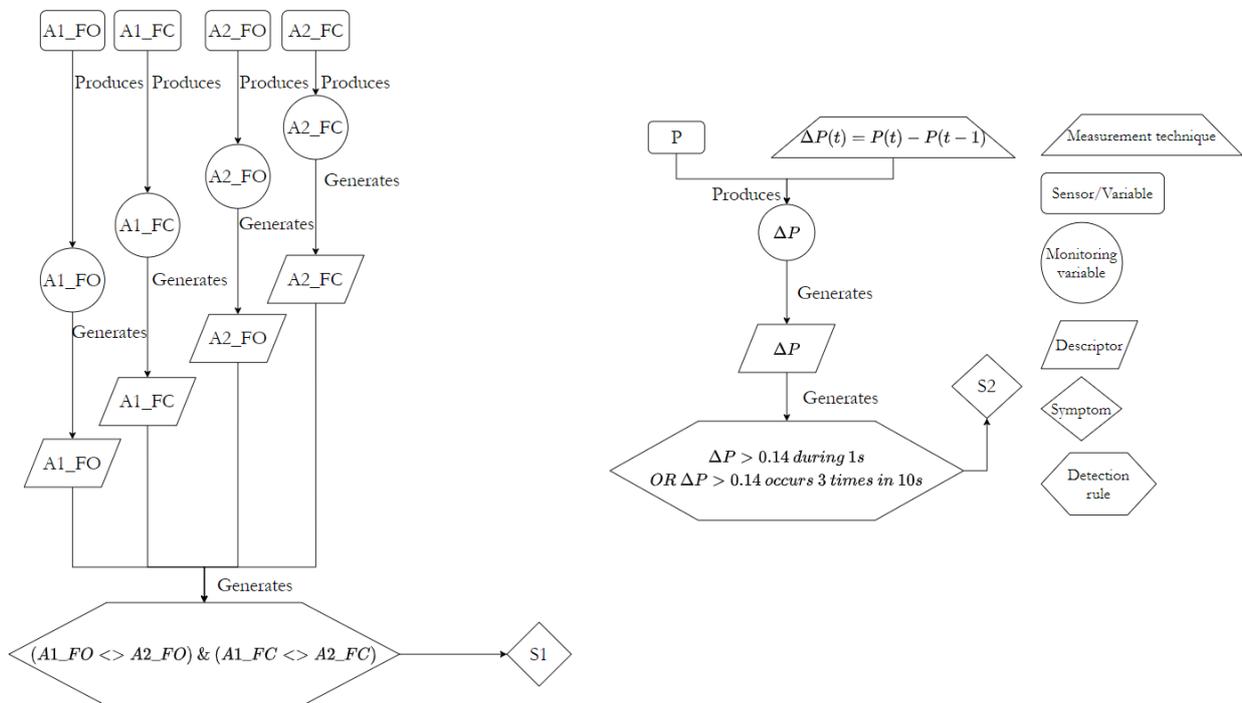


Figure 9.2 – Modèles CIM des règles de détection

**Table 9.1** – Application des règles de transformation du CIM au PIM pour la première règle

Concept CIM	Elément CIM	Elément PIM	Concept PIM
Variable de suivi	A1_FO	A1_FO	Variable
Variable de suivi	A1_FC	A1_FC	Variable
Variable de suivi	A2_FO	A2_FO	Variable
Variable de suivi	A2_FC	A2_FC	Variable
Règle de détection	$A1\_FO \neq$	&	Opérateur Logique
	$A2\_FO$ AND	$\neq$	Opérateur de comparaison
	$A1\_FC \neq$	$A1\_FO \neq A2\_FO$	Condition (C1)
	$A2\_FC$	$A1\_FC \neq A2\_FC$	Condition (C2)
		C1 & C2	Condition complexe
Symptôme	S1	S1	Output

règle de détection ( $\Delta P > 0.14$  pendant 1s) OU ( $\Delta P > 0.14$  Survient 3 fois en 10s). Le symptôme concerné par cette règle est noté “S2”. Cet exemple a pour but de détailler le traitement des opérateurs temporels “pendant” et “survient dans la période” dans le processus de transformation.

## 9.3 Présentation des PIM

A partir des CIM précédemment décrits et des règles de mapping présentées dans le chapitre 8, les modèles PIM des règles peuvent être définis.

Concernant la première règle, visant à détecter la défaillance de l’un des actionneurs, l’application des règles de mapping permet de définir les concepts du PIM comme détaillé dans la Table 9.1. Le modèle résultant de cette première transformation est présenté en Figure 9.3. Dans ce modèle, la règle de détection du modèle source est divisée en plusieurs éléments dans le modèle cible. Elle correspond en effet à une condition complexe composée de deux conditions liées par un opérateur logique. Chaque condition est composée de deux variables et d’un opérateur de comparaison.

La seconde règle a pour but de détecter le risque de surpression dans une pièce. Cette règle se base sur le nombre de dépassements d’un seuil critique. Si, pour une périodicité des mesures de 0.5s, ce nombre d’occurrences est supérieur ou égal à 2 en 1s ou supérieur ou égal à 3 en 10s, alors le risque doit être confirmé. Dans ce type de règle, il est nécessaire de définir une variable complexe et de scinder la règle métier en deux règles au niveau du PIM. La première règle générique doit générer une variable complexe lorsque la variation de pression est supérieure à 0.14psi. La deuxième règle doit surveiller le nombre d’occurrences sur une période prédéfinie pour détecter les risques de surpression. La Table 9.2 détaille la correspondance entre les éléments du modèle source et du modèle cible permettant la détection des dépassement de seuil. Le modèle résultant de cette correspondance se situe à gauche de la Figure 9.4.

Une fois la variable complexe  $\Delta P > 0.14$  psi est générée, la deuxième partie de la seconde règle peut être traitée. La Table 9.3 présente la correspondance entre les éléments du modèle source et ceux du modèle cible pour cette transformation. Le modèle généré est décrit dans la partie droite de la Figure 9.4

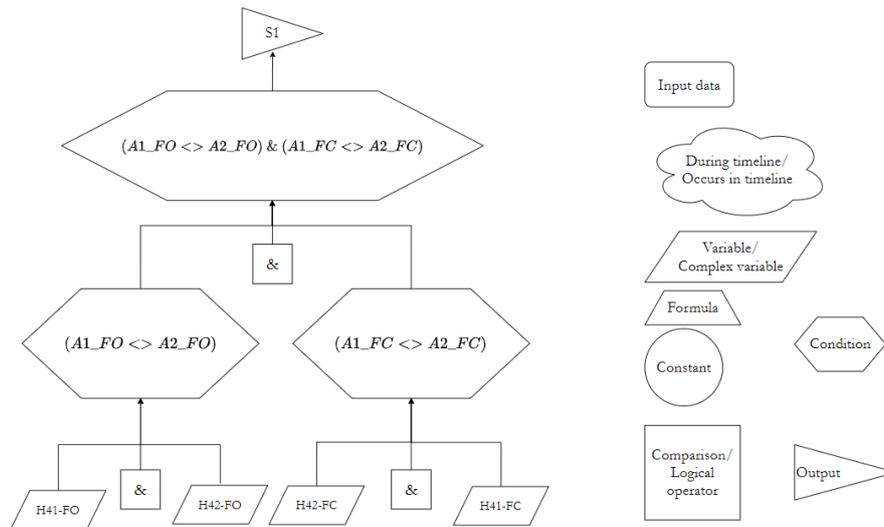


Figure 9.3 – PIM de la première règle

Table 9.2 – Application des règles de mapping du CIM au PIM pour la seconde règle (model a)

Concept CIM	Élément CIM	Élément PIM	Concept PIM
Variable	P	P	Donnée d'entrée
Technique de mesure	$\Delta$	$\Delta$	Formule
Variable de suivi	$\Delta P$	$\Delta P$	Variable
Règle de détection	$\Delta P > 0.14$ pendant 1s	0.14	Constante
	OU $\Delta P > 0.14$	>	Opérateur de comparaison
	survient 3 fois en 10s	$\Delta P > 0.14$	Condition
		$\Delta P > 0.14$	Variable complexe

Table 9.3 – Application des règles de mapping du CIM au PIM pour la seconde règle (model b)

Concept CIM	Élément CIM	Élément PIM	Concept PIM
Règle de détection	$\Delta P > 0.14$	$\Delta P > 0.14$	Variable complexe
	pendant 1s	or	Opérateur logique
	OU $\Delta P > 0.14$	1s	pendant
	survient 3 fois en 10s	3 times in 10s	Survient sur la période
Symptôme		C1 or C2	Condition complexe
		$\Delta P > 0.14$ pendant 1s	Condition (C1)
		$\Delta P > 0.14$ survient 3 fois en 10s	Condition (C2)
	S2	S2	Sortie

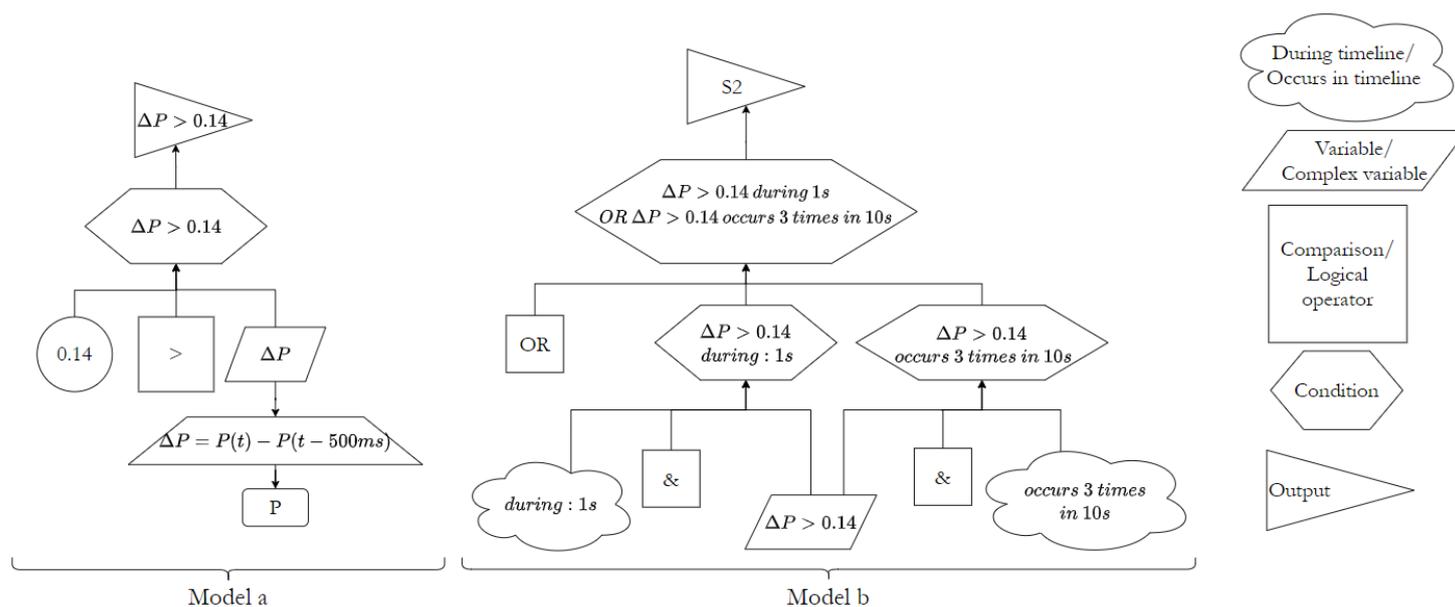


Figure 9.4 – Modèles PIM décrivant la seconde règle

## 9.4 Présentation des PSM

Pour la deuxième transformation, on cherche à transformer les modèles PIM, définis dans la section précédente, en PSM. Ces transformations sont basées sur les règles de mapping définies dans la Table 8.2.

L'application de ces règles au premier exemple donne la table de correspondance 9.4

Cette table de correspondance permet de définir le modèle EPL pour la première règle. Ce modèle est présenté en Figure 9.5.

Le second exemple, dont l'objectif est de détecter l'apparition d'une surpression dans la pièce, a généré deux modèles PIM dans la section précédente. Le premier permet de détecter le dépassement d'un seuil de pression, le second confirme la présence surpression en comptant le nombre de dépassements sur des périodes prédéfinies.

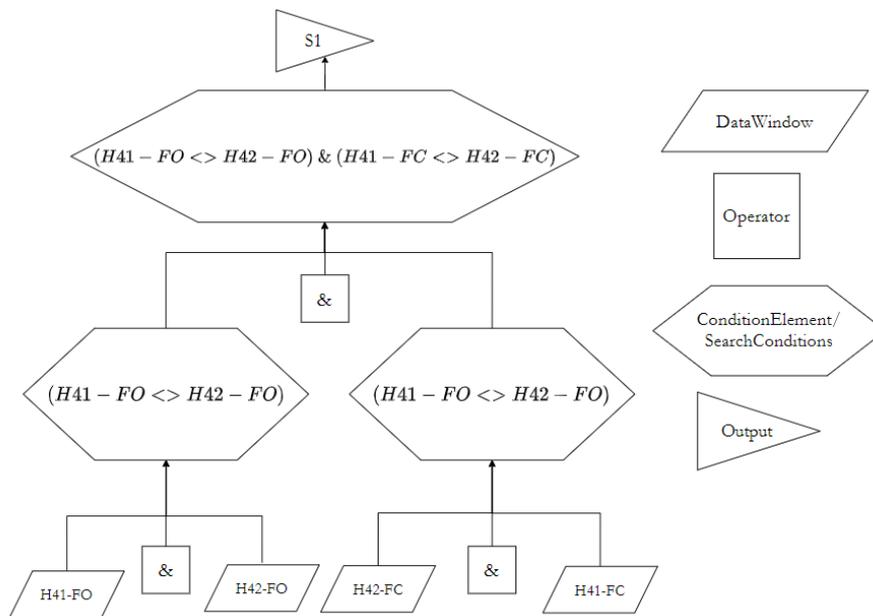
L'application des règles de mapping à ce premier modèle permet d'obtenir la correspondance décrite dans la Table 9.5. Le modèle issu de la transformation est présenté dans la partie gauche de la Figure 9.6.

Enfin, les règles de mapping peuvent être utilisées sur la deuxième partie de la règle. Le modèle obtenu est décrit dans la partie droite de la Figure 9.6 d'après les correspondances détaillées dans la Table 9.6.

Dans une architecture PHM, la phase de collecte requiert une solution capable d'analyser des données sur des fenêtres temporelles tout en s'adaptant aux contraintes de volume, de vitesse et de variété. Dans la partie III, une approche d'architecture basée sur les événements intégrant un ensemble de CEP a été utilisée pour répondre à ces exigences. Pour intégrer cette architecture dans la solution PHM, une transformation de modèle, suivant l'approche MDA, a été utilisée pour traduire les règles de détection

**Table 9.4** – Application des règles de mapping du PIM au PSM pour la première règle

Concept PIM	Elément PIM	Elément PSM	Concept PSM
Variable	A1_FO	A1_FO	WindowElement
Variable	A1_FC	A1_FC	WindowElement
Variable	A2_FO	A2_FO	WindowElement
Variable	A2_FC	A2_FC	WindowElement
Opérateur logique	&	&	Operator
Condition (C1)	$A1\_FO \neq A2\_FO$	SearchCondition	
Condition (C2)	$A1\_FC \neq A2\_FC$	SearchCondition	
Condition complexe	C1 & C2	C1 & C2	SearchElement
Sortie	S1	S2	Output



**Figure 9.5** – Règle EPL pour le premier exemple

**Table 9.5** – Application des règles de mapping du PIM au PSM pour la 2ème règle (modèle a)

Concept PIM	Elément PIM	Elément PSM	Concept PSM
Donnée d'entrée	P	P	WindowElement
Formule	$\Delta$	$\Delta$	PatternExpression
Variable	$\Delta P$	$\Delta P$	DataWindow
Condition	$\Delta P > 0.14$	$\Delta P > 0.14$	SearchCondition
Sortie	$\Delta P > 0.14$	$\Delta P > 0.14$	OutputElement

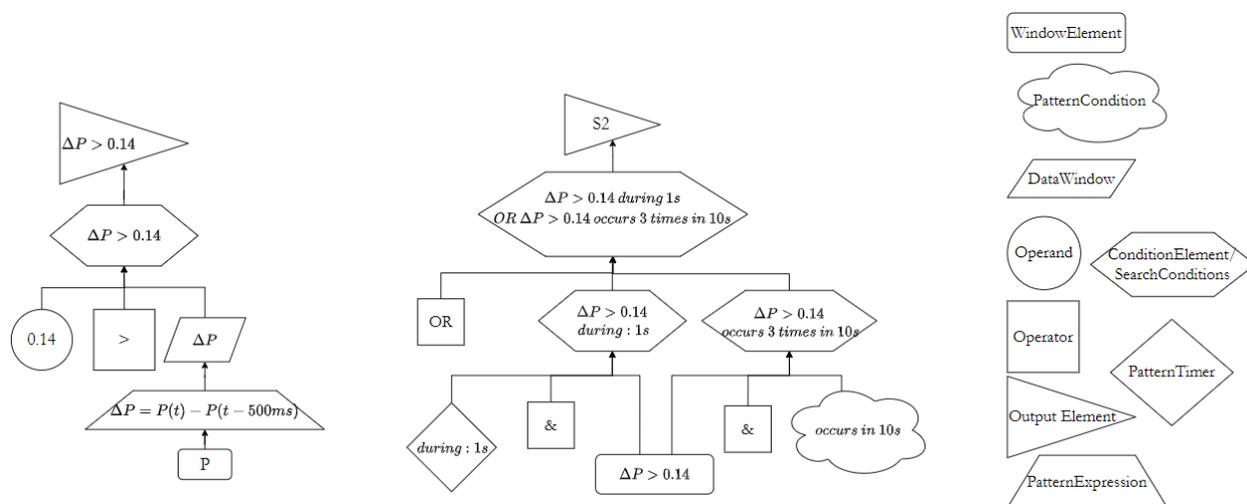


Figure 9.6 – Modèle de règle EPL pour la 2ème règle

Table 9.6 – Application des règles de mapping du PIM au PSM pour la 2ème règle (modèle b)

Concept PIM	Elément PIM	Elément PSM	Concept PIM
Variable complexe	$\Delta P > 0.14$	$\Delta P > 0.14$	WindowElement
Opérateur logique	OR	OR	Opérateur
Pendant	1s	during : 1s	PatternTimer
Survient dans la période	3 times in 10s	occurs in 10s	PatternCondition
Condition (C1)	$\Delta P > 0.14$ during 1s	$\Delta P > 0.14$ during 1s	ConditionElement
Condition (C2)	$\Delta P > 0.14$ occurs 3 times in 10s	$\Delta P > 0.14$ occurs 3 times	ConditionElement
Condition complexe	C1 or C2	C1 or C2	SearchConditions
Sortie	S2	S2	Output

contenues dans la base de connaissances dans un langage exploitable par un CEP. Des exemples concrets sont venus illustrer la mise en place de cette transformation avec, notamment, la gestion de multiples variables et de règles basées sur des fenêtres temporelles.

Dans la partie [IV](#), un cas d'étude basé sur un système aéronautique complexe sera utilisé pour illustrer l'implémentation de cette solution PHM.



# Partie IV

## Cas d'étude

Comme détaillé en partie 2.2.1, une approche PHM est une stratégie de maintenance prévisionnelle qui se base sur la surveillance d'un système en temps réel pour détecter les anomalies et identifier les défaillances survenues, on parle alors de diagnostic, ou à venir, on parle alors de pronostic. Cette surveillance est réalisée au moyen de capteurs, placés sur l'appareil, générant des flux de données qui sont alors analysés. Les résultats sont alors transmis à l'utilisateur via un outil d'aide à la décision afin de l'aider dans leur interprétation et simplifier la mise en oeuvre de procédures opérationnelles.

Un approche PHM se divise ainsi en trois phases : une phase de surveillance, une phase d'analyse et une phase d'aide à la décision. Dans le cadre de cette thèse, deux problématiques ont été posées :

1. Comment analyser les données de surveillance pour identifier et prévenir les défaillances en s'appuyant sur les connaissances capitalisées au travers de la documentation technique ?
2. Comment gérer le traitement en temps réel de données volumineuses, variées et véloces dans une perspective de détection d'anomalies ?

Afin d'adresser la première problématique, la partie II décrit un système de gestion de connaissance de type système expert permettant de stocker, structurer et exploiter des connaissances capitalisées pour analyser l'état de santé d'un système complexe.

La partie III répond à la deuxième problématique en complétant cette solution avec une architecture dirigée par les événements capable de surveiller et détecter des patterns dans les données d'entrée, y compris lorsque l'usage de fenêtres temporelles est nécessaire.

Du fait de contraintes expérimentales liées à la nature des données disponibles, le niveau de maturité de ces deux contributions diffère.

Dans le cas d'étude présenté, l'accent est mis sur la contribution de la partie 3. Il présente l'implémentation du SE pour un sous-système complexe d'un avion. L'expérimentation a pour but d'identifier un élément maintenable défaillant à partir de pannes détectées en utilisant un modèle statistique.

Dans un premier chapitre, une présentation du contexte aéronautique détaillera les enjeux et contraintes inhérentes à ce domaine.

Dans un deuxième chapitre, le protocole expérimental sera présenté. Après avoir décrit les jeux de données utilisés, le processus global d'identification d'élément défaillant sera décrit. Les modèles statistiques utilisés pour évaluer la probabilité de défaillance d'un équipement seront ensuite détaillés.

Enfin, dans un troisième chapitre, les résultats expérimentaux seront présentés et analysés.

# Chapitre 10

## Les enjeux de la maintenance dans le domaine aéronautique

Les avions sont des systèmes complexes, divisibles en sous-systèmes fonctionnels chargés, par exemple, de la propulsion, des commandes de vol ou encore du conditionnement de l'air. Cette complexité implique un nombre de pannes potentielles élevé dont certaines peuvent entraîner des conséquences humaines graves.

D'après l'Organisation de l'Aviation Civile Internationale (ICAO), le trafic aérien mondial en 2019 comptait 31023 avions civils pour un total de 38,3 millions de vols effectués et près de 4,5 milliards de passagers transportés (ICAO 2019b). Jusqu'en 2019, le nombre de passagers était par ailleurs en constante augmentation comme en atteste la figure 10.1. En 2019, le nombre d'accidents s'élève à 155 dont 6 mortels ce qui donne un taux de 3 accidents par million de départ de vols réguliers (ICAO 2019a). Cette fiabilité traduit l'importance primordiale de la sûreté des appareils pour les concepteurs et exploitants aéronautiques.

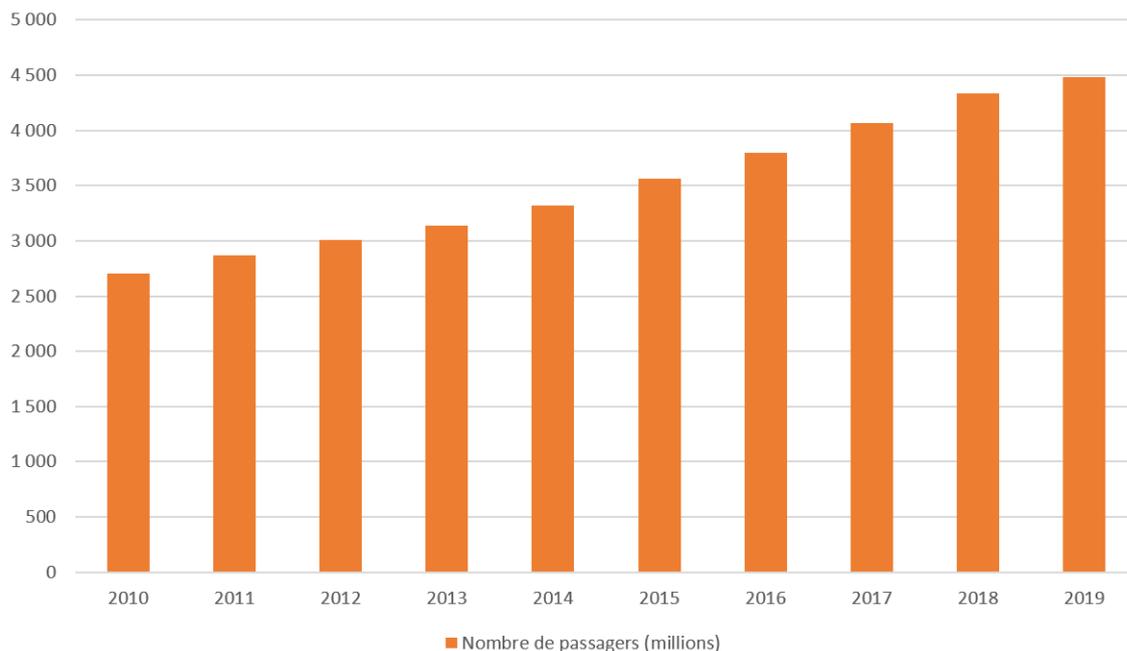
Ce niveau de fiabilité requiert cependant aujourd'hui des périodes d'indisponibilités régulières de l'appareil pour la réalisation d'actions de maintenance préventives systématiques. A ces indisponibilités s'ajoutent les immobilisations dues à des pannes. Ces immobilisations représentent également des coûts non négligeables pour les compagnies aériennes qui se répartissent en deux grandes catégories (BADKOOK 2016) :

— **les coûts directs :**

- frais de remplacement et de réparation des pièces défectueuses
- approvisionnement
- main d'oeuvre

— **les coûts indirects :**

- taxes aéroportuaires
- gestion des équipages
- impact sur la navigabilité de la flotte dont le planning des vols
- hébergement et repas des passagers



**Figure 10.1** – Évolution du trafic aéronautique payant international et intérieur de 2010 à 2019 (ICAO 2019b)

— gestion des bagages

Dans ce contexte de grande complexité des systèmes aéronautiques avec des exigences de fiabilité et de réduction des coûts élevées, l'accès à un volume croissant de données représente une opportunité d'évolution des stratégies de maintenance historiquement correctives et systématiques vers des approches prévisionnelles comme le PHM.

La fiabilité des systèmes aéronautiques rend cependant difficile la collecte d'un volume d'occurrences de pannes suffisants pour une détection par apprentissage. Ce contexte s'apparente à ce que KITCHIN et al. (2015) définit comme le *small data*, désignant ainsi les études basées sur des données produites de manière très contrôlée par des méthodes d'échantillonnage limitant leur portée, leur temporalité, leur taille et leur variété tout en évaluant leur niveau d'erreur, d'incertitude et de subjectivité. D'après eux, les *small data* se caractérisent ainsi par un volume limité de données discontinues et peu hétérogènes destiné à répondre à une question ou un usage précis.

Il s'oppose ainsi au *big data*, qui se caractérise par un volume massif de données produite en continu et naturellement hétérogènes.

Si la collecte des données issues des capteurs des avions rentre pleinement dans le domaine du *big data* du fait de la complexité du système surveillé, les relevés de pannes avérées appartiennent bel et bien au *small data* de par la fiabilité des systèmes considérés.

Afin de traiter ce type de données, de nombreuses connaissances experts structurées sont toutefois disponibles dans des analyses de fiabilité et des documentations techniques. Ces informations constituent une base favorable à la mise en oeuvre d'une approche par raisonnement.

L'approche PHM proposée répond aux besoins identifiés par sa capacité à ingérer et traiter des données de surveillance provenant d'un système complexe sur la base de règles métier et/ou de modèles obtenus par apprentissage. Le contexte aéronautique est ainsi particulièrement pertinent comme cas d'étude bien que l'architecture soit générique et adaptable à d'autres secteurs comme le ferroviaire ou la maintenance de machines industrielles.

Pour appliquer l'approche proposée à ce cas d'étude, un protocole expérimental a été défini sur la base de trois jeux de données collectés pour implémenter le SE et appliquer une règle statistique d'identification d'élément défaillant à des modes de pannes identifiés.



# Chapitre 11

## Protocole expérimental

Le cas d'étude proposé consiste à identifier un équipement défaillant à partir de modes de pannes et de modèles statistiques. Une même panne peut, dans certains cas, être provoquée par la défaillance, simultanée ou non, de plusieurs composants. Le protocole présenté dans ce chapitre a pour but d'illustrer l'utilisation de modèles statistiques basés sur des occurrences passées pour identifier le composant responsable de la panne. L'application des modèles se fera au moyen du SE défini dans la partie II.

Dans un premier temps, les données utilisées pour la réalisation du cas d'étude seront présentées. Dans un deuxième temps, le protocole d'application des règles sera décrit en se basant sur l'approche présentée dans le chapitre 6. Enfin, le modèle statistique choisi pour l'évaluation des probabilités de défaillance sera détaillé.

### 11.1 Présentation des données

Pour réaliser le cas d'étude, trois jeux de données ont été nécessaires pour construire la base de faits, définir les modèles statistiques d'identification et tester leur performance face à des modes de pannes réels. Pour des raisons de confidentialité, aucune information permettant d'identifier une compagnie aérienne, un modèle d'avion ou un équipement n'est divulguée dans cette partie.

Le premier jeu de données est issu de documentations techniques d'un sous-système d'avion. Cette documentation permet de définir la composition du système jusqu'à l'équipement maintenable qui est, pour rappel, le plus petit composant maintenu. Elle fournit également une liste des pannes pouvant affecter le système et des symptômes permettant de les détecter. Bien que restreint à un sous-système, le graphe généré à partir du méta-modèle défini dans la figure 5.12 contient plus de 4000 noeuds et 12000 arcs. Il est également à noter que seuls les composants physiques sont concernés. Les logiciels répondent à des logiques de maintenance différentes et ne sont pas traités dans le cadre des travaux présentés. Les nombres de noeuds et d'arcs traduisent la complexité du système considéré et la nécessité d'une approche automatique de diagnostic et pronostic.

Le second jeu de données a pour but d'alimenter un modèle statistique pour le diagnostic. Il est composé d'un ensemble de déposes de pièces pour panne observées

sur une période de cinq ans sur 70 avions du même modèle appartenant à plusieurs compagnies. Pour chaque pièce, on connaît son modèle, sa position sur l'avion ainsi que sa durée de vie au moment de la panne. Dans le domaine aéronautique, la durée de vie peut se mesurer selon trois unités : l'unité calendaire, le nombre d'heures de vol, le nombre de cycles, un cycle correspondant à la succession d'un décollage et d'un atterrissage. Dans le cas du système étudié, le nombre d'heures de vol est l'unité la plus significative vis-à-vis de la dégradation de la pièce.

Le troisième jeu de données sert à tester et évaluer la performance des modèles statistiques pour l'identification d'un composant défaillant à partir d'une alerte de panne. Pour ce faire un ensemble de 683 modes de pannes détectés sur une durée d'un an auprès de 4 compagnies aériennes a été collecté. Pour chacune, l'élément défaillant est connu et sera comparé au résultat de l'analyse.

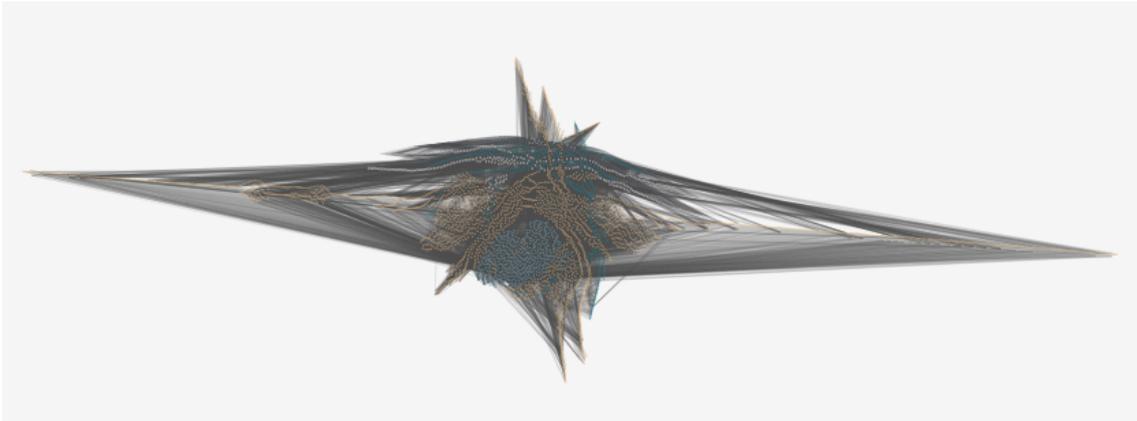
Grâce à ces données d'entrée, il est possible de générer la base de connaissances du SE, d'implémenter et de tester l'identification d'un composant défaillant à partir d'un mode de panne détecté.

## 11.2 Identification d'un composant défaillant

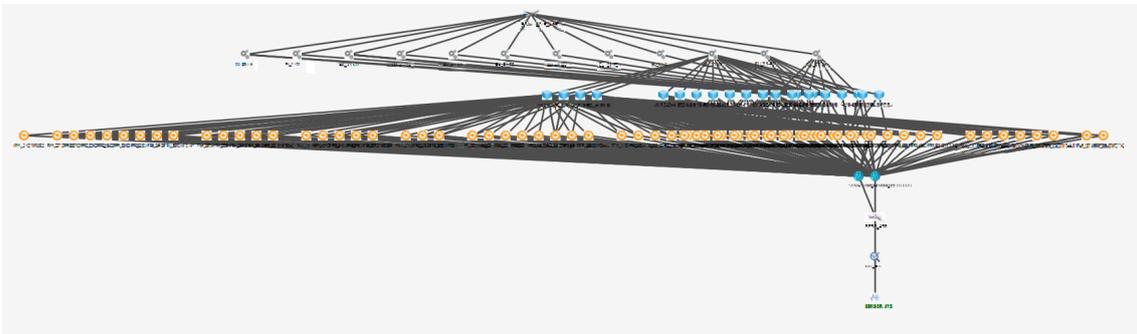
La première étape de l'expérimentation consiste à réaliser la base de connaissances à partir des données issues du premier jeu de données. Cette étape est réalisée au moyen d'une base graphe OrientDB. Le choix de cette solution est motivé par l'hétérogénéité des ressources intégrables et exploitables et en particulier sa capacité à gérer des documents. L'outil DigiBrain a été choisi pour gérer la base de règles et leur processus d'application du fait de sa compatibilité avec la gestion des règles définie en partie III (ACTIVUS 2020).

L'exploitation de la documentation technique fournie par le premier jeu de données permet de générer le graphe de la base de faits selon la structure définie dans la partie III. Les figures 11.1 et 11.2 illustrent le résultat obtenu. La figure 11.1 présente une vision globale du graphe tandis que la figure 11.2 détaille la structure d'une branche d'un point de vue hiérarchique en partant du capteur, en bas, puis en passant par les variables de suivi, les descripteurs, les symptômes, les pannes et les éléments maintenables qui s'agrègent en sous-systèmes d'un système global.

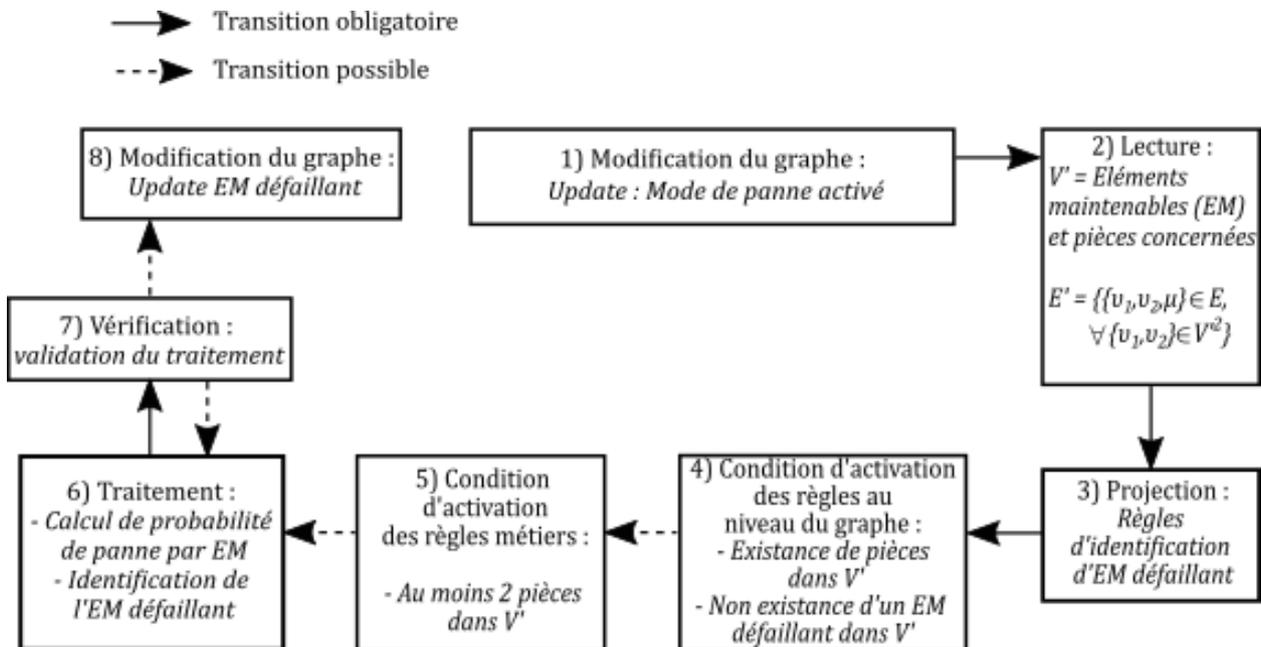
La base de règle est implémentée selon le processus défini dans le chapitre 5. Pour notre cas d'étude, on s'intéresse particulièrement à l'identification d'un élément maintenable défaillant à partir d'un mode de panne détecté. Cette identification s'appuie sur l'attribution d'une probabilité de défaillance pour chaque élément maintenable dans le temps selon sa durée de vie exprimée en heures de vol. La probabilité de défaillance sera ici déterminée par un modèle statistique tenant compte de la durée de vie de l'élément considéré. Nous considérons que la durée de vie correspond à la durée écoulée depuis la pose de l'élément sur l'appareil. Nous faisons ainsi l'hypothèse que la pièce n'est pas dégradée au moment de la pose. Dans le cas du sous-système étudié, la durée de vie s'exprime en heures de vol. Sur d'autres sous-systèmes, la dégradation peut davantage dépendre du nombre de cycles ou d'une durée calendaire. Un cycle



**Figure 11.1** – Illustration du graphe obtenu par l'implémentation de la base de faits d'un sous-système d'avion



**Figure 11.2** – Illustration d'une branche de la base de faits d'un point de hiérarchique



**Figure 11.3** – Processus d'identification d'un élément maintenable défaillant à partir d'un mode de panne

correspond à un couple décollage/atterrissage indépendamment de la durée du vol. La Figure 11.3 détaille le processus d'application de la règle d'identification.

Dans une première étape, la détection d'un ou plusieurs symptômes provoque l'activation d'un mode de panne résultant en une modification du graphe.

Dans une deuxième étape, une lecture du graphe permet d'identifier les éléments maintenables concernés par le mode de panne actif. Ceux-ci représentent les positions des composants sur l'avion d'après la nomenclature. Cependant, il est possible qu'aucune pièce ne soit présente sur l'une de ces positions au moment de la défaillance ce qui écarte l'élément maintenable du processus d'identification. Les pièces sont donc incluses dans la phase de lecture.

La troisième étape consiste à récupérer les règles s'appliquant aux noeuds et arcs sélectionnés. Dans le cas présent, les règles sélectionnées sont des règles d'identification d'éléments défaillants.

La quatrième étape est un premier contrôle réalisé au niveau du graphe pour vérifier qu'il existe bien une pièce sur les éléments maintenables concernés et que l'un d'entre eux n'est pas déjà identifié comme défaillant.

La cinquième étape est un second contrôle correspondant à la condition d'activation de la règle. Il s'agit ici de vérifier qu'il existe au moins deux possibilités avant de réaliser le traitement. Une analyse rapide du graphe indique que 33% des modes de pannes ne sont liés qu'à un seul élément maintenable. Dans ce cas, l'identification de l'élément défaillant est immédiate, il n'est pas nécessaire d'appliquer une règle d'identification quelconque. En revanche, pour les 66% restants, un traitement plus approfondi doit être réalisé.

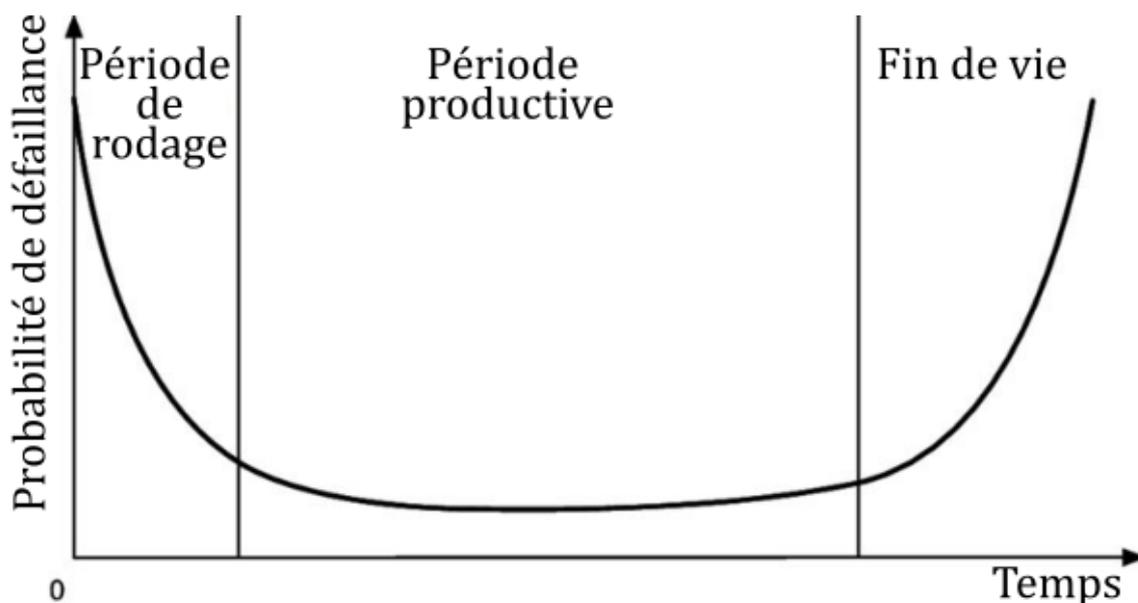
La sixième étape est le traitement. Il consiste ici à évaluer la probabilité de défaillance pour chaque élément maintenable et à générer une liste par ordre de valeur décroissante afin de déterminer celui étant le plus vraisemblablement responsable de la défaillance.

Une septième étape de vérification sert à valider la cohérence du résultat obtenu avant qu'il ne soit répercuté dans le graphe par une modification du noeud concerné. En cas d'échec de la vérification, aucune modification n'est apportée au graphe. Le traitement peut être relancé ou un message d'erreur peut être envoyé à l'utilisateur.

Ce processus a pour but d'appliquer les règles d'identification d'éléments défaillants à partir de mode de pannes identifiés. Dans le cas d'étude proposé, ces règles d'identification reposent sur des modèles statistiques permettant d'attribuer les probabilités de défaillance aux éléments maintenables.

## 11.3 Le modèle statistique : la loi de Weibull

A ce jour, les opérateurs de maintenance des compagnies aériennes usent de leurs connaissances du système et de leur expérience de cas passés pour identifier un élément maintenable défaillant. En suivant cette logique, une règle d'identification a été créée dans la base de règle en adoptant une approche probabiliste basée sur les



**Figure 11.4** – Illustration des trois phases d'une courbe en baignoire

occurrences de pannes passées. L'objectif est de déterminer, pour chaque élément maintenable concerné, sa probabilité de défaillance puis de les classer par probabilité décroissante. Celui ayant la probabilité la plus élevée sera donc à vérifier en premier pour les opérateurs.

Pour définir cette règle probabiliste, une analyse de fiabilité est menée pour modéliser l'évolution de la probabilité de défaillance d'un élément maintenable au cours de sa vie. Celle-ci dépend en partie du modèle de l'élément considéré. On considère généralement que la probabilité de défaillance d'une pièce évolue suivant une courbe dite "en baignoire" comprenant en trois phases comme illustré dans la Figure 11.4 (EBELING 1997).

La première phase s'appelle la période de rodage. Elle se situe au début du cycle de vie et présente une probabilité de défaillances élevée due à des défauts de fabrication ou d'installation. La seconde phase est la phase de production. Durant cette période, le risque de défaillance est globalement constant et faible. Les défaillances se produisant durant cette phase sont généralement liées à des phénomènes aléatoires indépendant de la durée de vie de la pièce. C'est la phase principale de son cycle de vie. Enfin, la troisième phase est la fin de vie. Durant cette période, le risque de panne augmente du fait de l'usure.

Il est à noter que ces trois phases ne sont pas distinctement présentes pour tous les types de pièces. Par exemple, les pièces mécaniques ont un profil généralement différent de celui des pièces électroniques (BELEULMI et al. 2014). Les pièces mécaniques ont tendance à avoir des phases de rodage et de vieillissement marquées avec une phase de production plus difficilement identifiables tandis que les pièces électroniques présentent généralement trois phases bien distinctes. Pour améliorer la sûreté de fonctionnement les systèmes critiques, notamment électronique sont souvent rodés en atelier.

Le second jeu de données, présenté dans la section 11.1, nous fournit pour chaque élément maintenable, les durées de vie auxquelles se sont produites les défaillances. La durée d'observation s'étend sur une période de 5 ans pour 70 appareils. L'objectif est ici de déterminer un modèle statistique permettant d'approcher les courbes expérimentales d'évolution cumulatives du nombre de défaillances générées à partir de ce jeu de données.

Dans la littérature, il existe de nombreuses distributions probabilistes utilisées pour modéliser la probabilité de défaillance comme la distribution exponentielle, la distribution Log-Normale ou la distribution de Weibull (COUALLIER V. 2016). Cependant, BELEULMI et al. (2014) considère la distribution de Weibull comme étant une distribution universelle pour la représentation des trois phases de la courbe en baignoire.

La distribution de Weibull peut toutefois se présenter sous plusieurs formes. Il est possible d'utiliser une loi simple ou une combinaison de deux distributions de Weibull. Cette dernière approche peut, dans certains cas, permettre de mieux représenter les trois phases du cycle de vie (LAI 2014; XIE et al. 1996).

Une loi de Weibull simple est définie par une densité de distribution  $f$  et une distribution cumulative  $F$  respectivement exprimées par les relations (LAI 2014) :

$$f(x) = \left(\frac{k}{\lambda}\right) \left(\frac{x}{\lambda}\right)^{(k-1)} e^{-\left(\frac{x}{\lambda}\right)^k} \quad (11.1)$$

$$F(x) = 1 - e^{-\left(\frac{x}{\lambda}\right)^k} \quad (11.2)$$

avec  $k > 0$ , et  $\lambda > 0$  désignant respectivement les paramètres de forme et d'échelle de la distribution de Weibull.

Dans le cas d'une distribution double, ces fonctions sont définies par relations :

$$f(x) = \left( \left(\frac{k_1}{\lambda_1}\right) \left(\frac{x}{\lambda_1}\right)^{(k_1-1)} + \left(\frac{k_2}{\lambda_2}\right) \left(\frac{x}{\lambda_2}\right)^{(k_2-1)} \right) \left( e^{-\left(\frac{x}{\lambda_1}\right)^{k_1}} e^{-\left(\frac{x}{\lambda_2}\right)^{k_2}} \right) \quad (11.3)$$

$$F(x) = 1 - e^{-\left(\frac{x}{\lambda_1}\right)^{k_1}} e^{-\left(\frac{x}{\lambda_2}\right)^{k_2}} \quad (11.4)$$

avec  $k_1 > 0$ ,  $\lambda_1 > 0$ ,  $k_2 > 0$ ,  $\lambda_2 > 0$  désignant respectivement les paramètres de forme et d'échelle des deux distributions de Weibull.

Pour estimer les paramètres d'une distribution de Weibull, une régression est réalisée en utilisant un estimateur des moindres carrés ordinaire sur la courbe cumulative des défaillances empiriques comme préconisé par ZHANG L. (2008). Dans le cas présent, aucun poids n'est cependant attribué aux défaillances.

Le critère d'information d'Akaike corrigé (AICc) est utilisé pour évaluer l'apport du modèle à double Weibull par rapport à une distribution de Weibull simple. Le choix de l'AICc est motivé par le faible nombre d'occurrences par défaillance et son

efficacité avec les modèles de régression (HURVICH et al. 1989). Sa formule est définie par :

$$AICc = 2p - \ln(L) + \frac{2p(p+1)}{n-p-1} \quad (11.5)$$

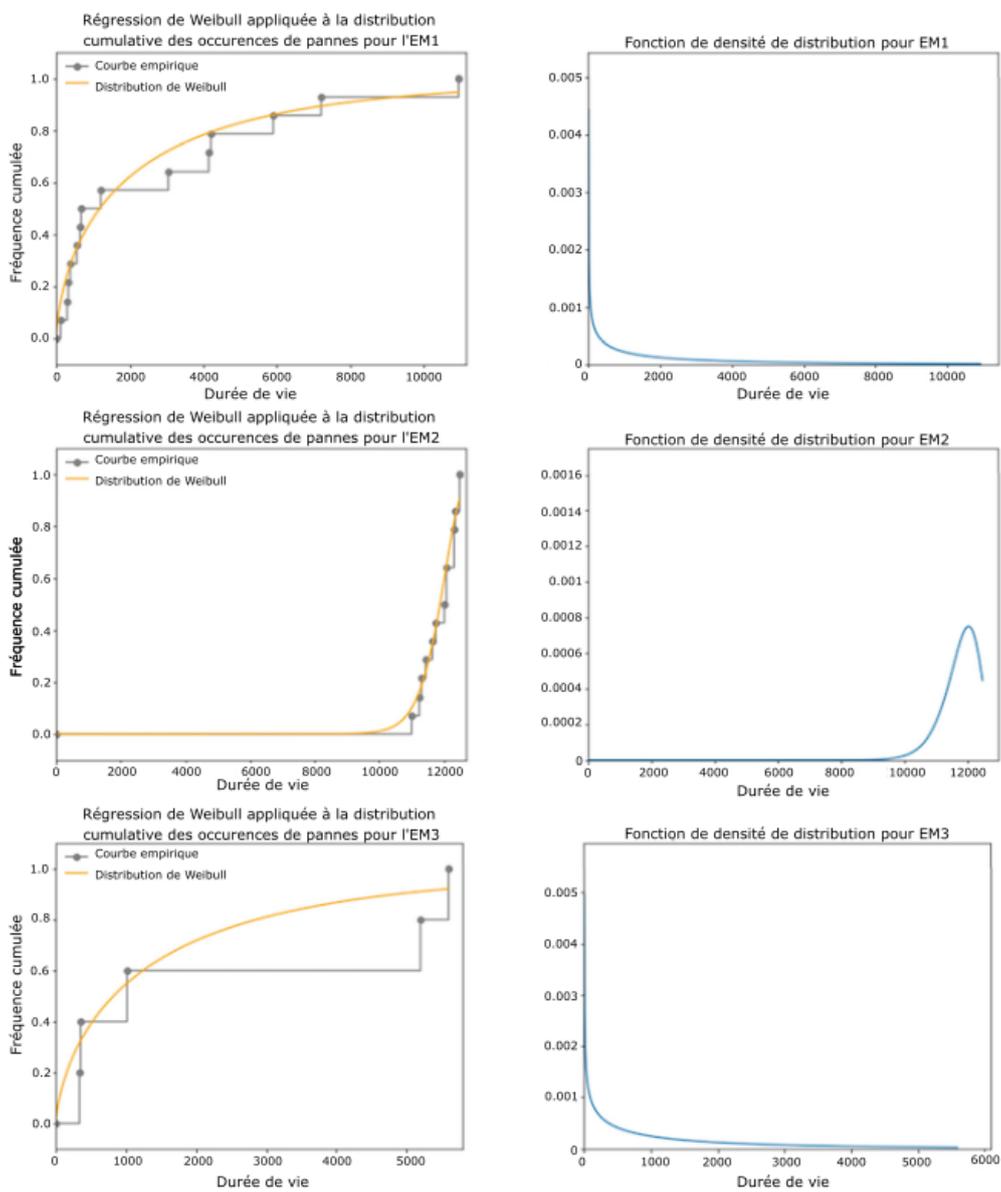
où  $p$  désigne le nombre de paramètres ( $p = 2$  for une distribution de Weibull simple et  $p = 4$  pour une distribution double),  $n$  désigne la taille de l'échantillon et  $L$  est la valeur maximale de la fonction de vraisemblance. Le modèle conservé est celui possédant la valeur la plus faible pour ce critère.

Enfin, un test de Kolmogorov-Smirnov est réalisé pour vérifier si le modèle obtenu correspond bien à une distribution de Weibull (NWABI et al. 2014). Dans l'éventualité où le test serait négatif, il faudrait alors modéliser les courbes empiriques par une autre distribution.

La Figure 11.5 présente quelques exemples illustrant le résultat des modélisations par lois de Weibull avec des phases de rodage, de fin de vie et de production variées par éléments maintenables et modèles de pièces. Ce choix de modélisation permet de tenir compte de la position de la pièce dans son profil de dégradation.

Il est également possible de réaliser des distributions ne tenant compte que du modèle de la pièce. Bien que cette approche soit moins précise, elle apporte des informations suffisantes d'un point de vue logistique pour l'approvisionnement des pièces et la planification de la tâche de maintenance. On peut, par ailleurs, remarquer que cette modélisation peut attribuer la même probabilité de défaillance à deux pièces à condition qu'elles soient du même modèle et qu'elles aient été placées en même temps sur l'appareil. Elles ont alors la même durée de vie et suivent le même modèle de dégradation ce qui leur assure des probabilités de défaillance identiques. Tenir compte de la position sur l'appareil permet d'avoir deux courbes différentes pour ces pièces minimisant le risque d'avoir une égalité de leurs probabilités.

Ce protocole expérimental permet d'identifier un élément maintenable, c'est-à-dire un modèle de pièce à une position précise, ou un modèle de pièce défaillant par l'exploitation des connaissances techniques capitalisées et structurées dans une base graphe. Cette identification s'appuie sur des règles probabilistes basées sur des distributions de Weibull. La prochaine étape du cas d'étude consiste à évaluer la performance de ce protocole grâce aux modes de pannes provenant du troisième jeu de données. .



**Figure 11.5** – Exemples de d'approximation des courbes cumulatives de défaillances empiriques par les lois de Weibull par élément maintenable et par modèle de pièce

# Chapitre 12

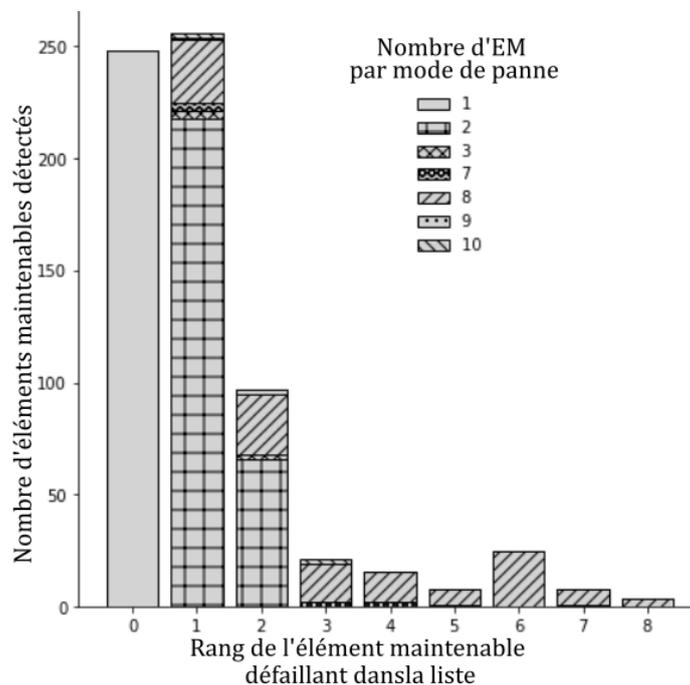
## Analyse des résultats

Afin d'évaluer la performance du graphe avec les modèles précédemment présentés, le troisième jeu de données est utilisé. Il regroupe un ensemble de 683 modes de pannes qui ont été collectés sur un an pour 4 compagnies. Pour chacun de ces modes de panne, plusieurs éléments maintenables peuvent être responsables. L'objectif du cas d'étude est de générer une liste des éléments maintenables potentiellement responsables pour chaque mode de panne ordonnée par probabilités de défaillance décroissantes. La performance du diagnostic dépend de la place de l'élément réellement défectueux dans chaque liste.

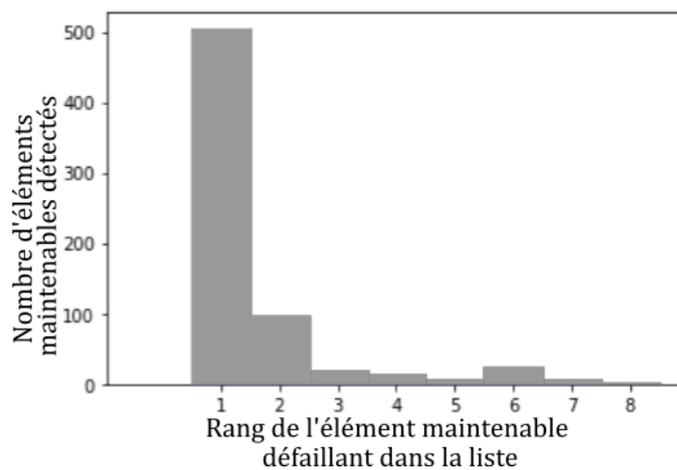
Deux approches peuvent être utilisées pour cette analyse. La première consiste à modéliser le risque de défaillance d'un élément maintenable en tenant compte du modèle de pièce, tandis que la seconde ne considère que le modèle de la pièce sans tenir compte de l'influence de sa position sur l'avion pour l'estimation de sa dégradation. La première approche présente l'avantage d'identifier précisément l'équipement défaillant. Cependant, plus le nombre de positions est élevé, plus le risque d'erreur est grand. En revanche, la deuxième est suffisante pour connaître le modèle d'équipement à prévoir, d'un point de vue logistique, et anticiper la maintenance.

La Figure 12.1 présente les résultats de l'analyse des places des éléments défectueux dans les liste avec la première approche. Le rang "0" correspond aux cas pour lesquels le mode de panne n'est lié qu'à une seule pièce. Dans le jeu de données disponible, cette situation représente 36% des cas. Le rang "1" désigne les situations dans lesquelles l'élément ayant la plus grande probabilité de défaillance a bien été déposé pour panne et représente 59% des cas en excluant le rang "0". Le rang "2" signifie que l'élément déposé avait la deuxième plus grande probabilité de défaillance et ainsi de suite. Pour chaque rang, la proportion de modes de pannes par nombre d'éléments maintenables accusés est indiquée par un motif. On remarque notamment que parmi les résultats du rang "1", 85% des listes ne contenaient que deux éléments, 11% en contenaient 8 et les 4% restants se répartissent entre 3, 7, 9 et 10 possibilités.

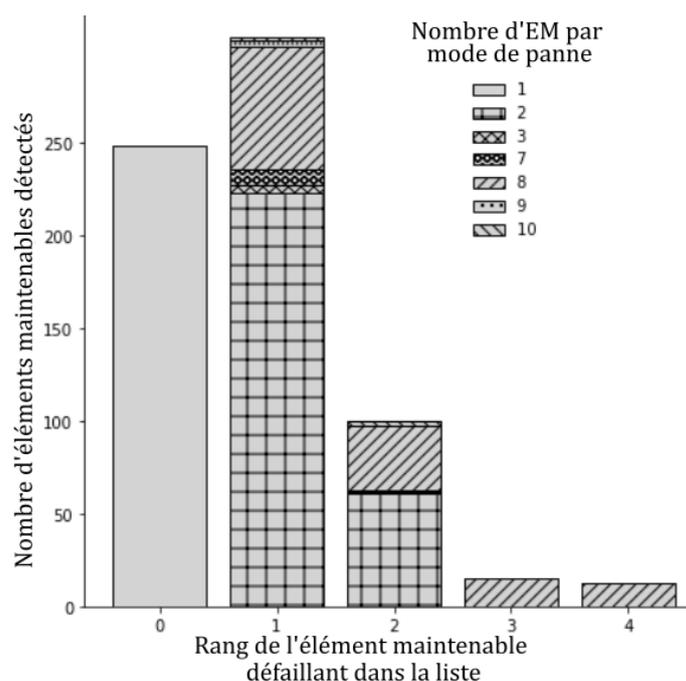
Afin d'évaluer la proportion d'identifications correctes, les rangs "0" et "1" sont agrégés dans la figure 12.2. Avec cette approche, l'élément ayant été déposé pour panne est correctement identifié dans 74% des cas. Si on regarde les 2 premières solutions, cette proportion monte jusqu'à 88%.



**Figure 12.1** – Résultats de l'identification d'un élément maintenable défaillant selon le modèle de la pièce



**Figure 12.2** – Résultats de l'identification d'un élément maintenable défaillant selon le modèle de la pièce sans distinction des listes de taille 1



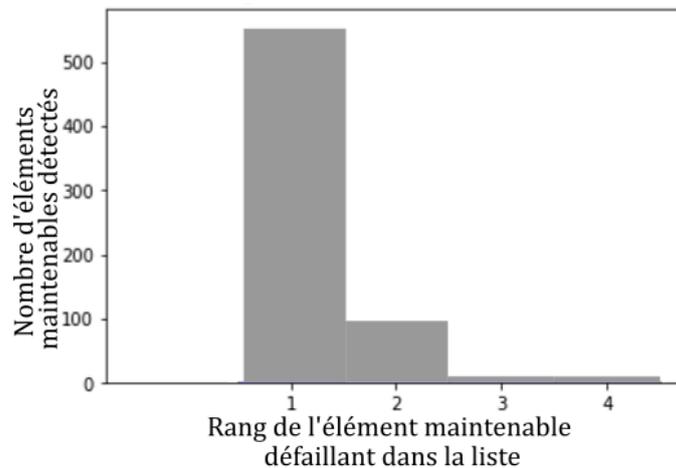
**Figure 12.3** – Résultat de l'identification d'un modèle de pièce défaillant

La deuxième approche a pour but d'identifier uniquement le modèle de pièce lié à la défaillance. Bien que l'identification soit moins précise que dans la première approche, elle est suffisante d'un point de vue logistique pour l'approvisionnement des pièces, en particulier celles qui ne sont pas réparables et/ou impliquent une immobilisation de l'appareil, en prévision de la réalisation de la tâche de maintenance.

Le même processus que pour la première approche est utilisé pour évaluer l'identification du modèle de pièce défaillant. Les histogrammes obtenus sont présentés en figures 12.3 et 12.4. Le rang "0" ne dépendant pas du calcul des probabilités, il n'est pas affecté par ce changement d'approche et représente toujours 36% des défaillances. En revanche, le modèle défaillant est correctement identifié dans le rang "1" dans 71% des cas en excluant le rang "0". Si on regarde les 2 premiers résultats de la liste, cette proportion monte à 94%. La figure 12.4 indique que cette approche identifie correctement le modèle de la pièce défaillante dans 81% des cas si l'on inclut le rang "0".

En conclusion, l'approche PHM proposée répond à des enjeux actuels de la maintenance aéronautique civile comme la prise en compte de la complexité des systèmes, les exigences de fiabilité et de réduction des coûts. Le cas d'étude présenté applique l'approche par raisonnement proposée au moyen d'un SE comprenant une base de faits sous format graphe et des règles probabilistes basées sur des distributions de Weibull à un sous-système d'aéronef. Ces règles visent à identifier un élément maintenable défaillant à partir de modes de panne identifiés.

L'application de cette approche à un échantillon de 683 modes de pannes collecté sur un an indique que l'exploitation du graphe structurant les données techniques de l'appareil permet d'identifier la pièce défaillante dans 36% des cas. L'ajout d'une règle probabiliste basée sur les distributions de Weibull permet d'améliorer cette



**Figure 12.4** – Résultats de l'identification d'un modèle de la pièce défaillant sans distinction des listes de taille 1

performance en identifiant correctement la pièce défaillante dans 74% des cas et monte jusqu'à 81% si l'on cherche uniquement le modèle de la pièce. Ces résultats peuvent par ailleurs être améliorés par le fait que certains modes de panne peuvent entraîner la défaillance de plusieurs pièces. Cela signifie que certaines listes possèdent des éléments défaillants dans les rangs 1 et 2.

Bien que les résultats présentés soient prometteurs, le but de ce cas d'étude est d'illustrer l'implémentation d'une règle dans un cas d'étude réel. Ici, la règle choisie est une règle d'identification d'élément défaillant basée sur un modèle probabiliste relativement simple : la distribution de Weibull. Cependant, l'approche proposée permet d'intégrer tous types de modèles : probabilistes, physiques ou d'apprentissage entraînés. L'utilisation de modèles plus complexes dans le SE pourrait ainsi fournir de meilleurs résultats. Une étude comparative pourrait être réalisée en ce sens.

# Chapitre 13

## Conclusion et perspectives

### 13.1 Bilan des travaux

La maintenance est aujourd’hui confrontée à des enjeux économiques majeurs dont :

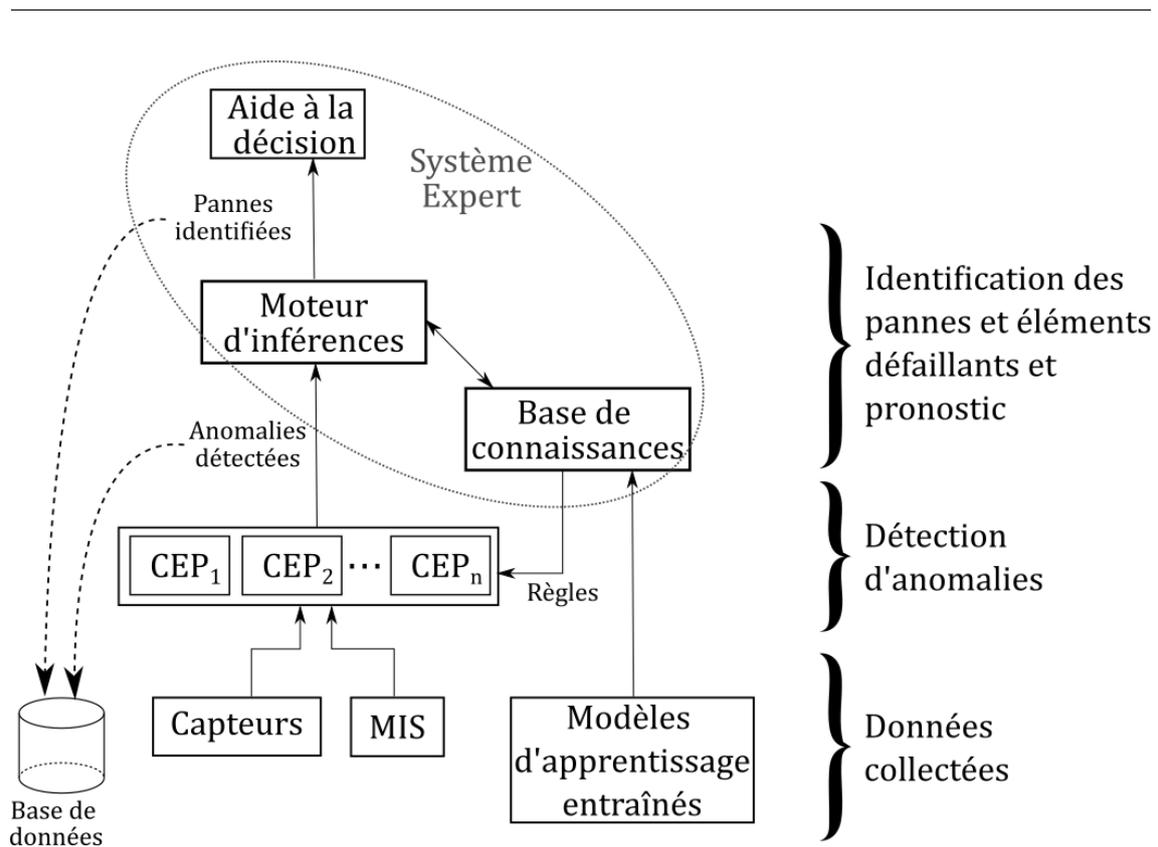
- une forte complexité des systèmes qui rend difficile l’identification des pannes,
- des exigences de fiabilité élevées, en particulier sur des systèmes critiques,
- des objectifs de réductions des coûts.

Face à ces enjeux, la collecte massive de données, rendue possible par le développement des systèmes de collecte, représente une opportunité de mutation des stratégies de maintenance correctives et systématiques vers des approches prévisionnelles comme le PHM.

Dans ce manuscrit nous avons défini une approche PHM pour des systèmes répondant à quatre caractéristiques qui seront considérées comme des hypothèses :

- l’automatisation du processus de maintenance est rendue nécessaire par les complexités fonctionnelles et techniques des systèmes considérés,
- ils sont équipés de systèmes de collecte de données renvoyant les informations nécessaires à l’évaluation de son état de santé. Du fait de la complexité du système, ces données sont caractéristiques d’un contexte “big data”,
- l’application de modèles d’apprentissage n’est pas toujours possible du fait d’un faible volume d’occurrences de pannes. Ces données relèvent d’un problème de type “small data”,
- une connaissance experte capitalisée et structurée est disponible au travers d’une documentation technique, de logs et feedback. Cette documentation permet d’exploiter des informations sur la composition et la fiabilité du système surveillé.

Afin de développer une stratégie de maintenance prévisionnelle adaptée à ces systèmes, une approche par raisonnement est proposée dans ce manuscrit. La figure 13.1 présente le schéma de l’architecture technique proposée.



**Figure 13.1** – Présentation de l'architecture technique

Afin de détecter les anomalies dans les données d'entrée tout en répondant aux problématiques de volume, de vélocité et de variété des données, une architecture dirigée par les événements est implémentée au moyen d'un ensemble de CEP organisés en parallèle. Cette détection se base sur l'application de règles de surveillance métier provenant d'une expertise humaine ou de modèles d'apprentissages entraînés. Ces règles sont stockées dans la base de connaissances d'un SE.

Ce SE est également chargé d'analyser les anomalies détectées pour identifier les pannes affectant ou sur le point d'affecter le système surveillé. Cette identification permet ensuite de déterminer la ou les pièces impactées par la panne.

La première contribution de ce manuscrit consiste à définir les composants du SE de manière à répondre aux exigences et contraintes liées à la maintenance prévisionnelle.

Le SE est constitué d'une base de connaissances adaptée à la maintenance conditionnelle et répondant à des exigences de :

- scalabilité pour convenir à des systèmes de complexité variables,
- d'adaptabilité des concepts manipulés pour correspondre à n'importe quel type de système,
- d'efficacité dans l'exploitation des données.

L'exigence d'adaptabilité est adressée au moyen d'un méta-modèle générique indépendant du système considéré. Pour gérer la scalabilité et l'efficacité, nous proposons

d'implémenter la base de connaissances sous forme de réseau d'informations hétérogènes. Cette proposition s'accompagne de la définition d'un processus d'application des règles adapté dans le moteur d'inférences.

La seconde contribution repose sur le paramétrage des CEP par les règles de surveillances de la base de connaissances. La stratégie adoptée est d'employer une approche d'architecture dirigée par les modèles. Deux transformations de modèles successives permettent de traduire les règles métiers dans un format compréhensible et exploitable par un CEP, ici l'EPL.

Pour illustrer et tester l'implémentation de la première contribution, un cas d'étude sur un sous-système d'aéronef a été réalisé. Celui-ci consiste à instancier la base de connaissances du SE avec des données de documentation technique et à identifier un composant défaillant à partir de modes de pannes identifiés. Cette identification est réalisée en s'appuyant sur des règles probabilistes établies à partir de distributions de Weibull.

Le résultat du cas d'étude montre que l'approche proposée permet d'identifier précisément l'élément défaillant dans 74% des cas et ce résultat monte à 81% si l'on cherche uniquement le modèle de la pièce défaillante.

## 13.2 Limites et perspectives

Les principales limites des travaux proposés ont trait au périmètre d'étude concerné. En effet, une première contrainte pour la mise en place de l'approche proposée réside dans la présence d'un système de collecte de données sur l'appareil suffisamment complet et pertinent pour permettre l'évaluation de son état de santé en temps réel. La qualité des données de surveillance est un pré-requis impactant fortement la qualité des analyses qui en découlent. De plus, un historique de ces données est nécessaire à la conception de modèles statistiques ou d'apprentissage.

Une seconde limite repose sur la disponibilité de ressources documentaires exploitables. Ces ressources sont indispensables à l'implémentation de la base de connaissances et, par extension, à la mise en place d'une approche par raisonnement. Ces documentations techniques doivent fournir toutes les informations nécessaires sur la composition du système et les pannes et symptômes pouvant l'affecter pour fournir un socle solide au processus déductif.

Une troisième limite concerne le type de composants traités. L'approche présentée dans ce manuscrit se concentre en effet sur des composants physiques réparables et/ou remplaçables. En revanche la maintenance de composants comme les logiciels répond à d'autres pratiques et logiques, tels le débogage ou la réinitialisation, qui ne sont pas abordées dans le périmètre considérés.

Le périmètre des travaux présentés dans ce manuscrit ouvre également la voie à de nombreuses perspectives. Ces perspectives peuvent se classer en deux catégories :

- les perspectives techniques liées au développement de l'architecture,
- les perspectives fonctionnelles relatives à l'impact métier de l'approche proposée.

---

Une première perspective technique l'intégration et l'évaluation de nouveaux modèles pour la détection des anomalies, l'identification des pannes et éléments défaillants et le pronostic des RUL. En particulier, la mise en place de modèles d'apprentissage entraînés pour le pronostic est un des enjeux principaux de l'approche proposée. Les travaux basés sur les résultats du simulateur de turboréacteur C-MAPSS, collectés et mis en ligne par la NASA, sont des sources d'inspiration intéressantes dans ce cadre (HSU et al. 2018 ; J. WANG et al. 2018). Le développement et l'intégration de modèles d'apprentissages basés sur ce type de données constitue ainsi une première perspective technique.

Une seconde perspective technique consisterait à élargir le périmètre expérimental au processus complet en intégrant la détection d'anomalies avec les CEP, l'identification des pannes et le pronostic. Un tel cas d'étude permettrait d'éprouver l'approche dans son intégralité, de bout en bout, sur un système réel. Cette expérience pourrait également servir de base au calcul des probabilités à chaque étape, puis de la probabilité cumulée à l'issue du processus. Cette probabilité cumulée pourrait alors contribuer à évaluer l'état de santé du système. KUMAR et al. (2009) propose pour cela d'employer des approches par logique floue.

Une troisième perspective technique peut être identifiée au niveau de l'adaptabilité de l'approche proposée. En effet, le cas d'étude proposé illustre son application pour un sous-système aéronautique. Il serait également intéressant d'éprouver la généralité de l'approche à des systèmes variés dans des secteurs différents comme le ferroviaire ou pour la maintenance de robots ou machines industrielles. Cette perspective technique permettrait ainsi d'améliorer la maturité de l'approche.

D'un point de vue fonctionnel, les travaux présentés dans ce manuscrit ouvrent la porte à des perspectives liées à l'impact des analyses produites sur la réalisation de la maintenance. Ce champ d'application dépasse le cadre de la maintenance prévisionnelle et s'inscrit dans le périmètre de la maintenance prescriptive. D'après MARQUES et al. (2019), la maintenance prescriptive a pour but d'étendre le périmètre de la maintenance prévisionnelle en fournissant des recommandations dynamiques en temps réel sur les tâches à effectuer. Au delà de tenir compte des prévisions basées sur l'analyse des tendances dans les données, elle considère également les spécificités de la stratégie et du processus de maintenance de l'exploitant pour générer des recommandations adaptées.

Les impacts de la maintenance prévisionnelle peuvent s'étendre aux trois niveaux de décisions : stratégique, tactique et opérationnels.

La capacité d'évaluation en temps réel de l'état de santé d'un système, offerte par la maintenance prévisionnelle, permet en effet d'envisager de nouvelles possibilités de gestion de la disponibilité des appareils. Ces stratégies doivent répondre aux questions de regroupement des tâches de maintenance et de réduction du nombre et/ou des durées d'immobilisation. Ces critères doivent être définis de manière à satisfaire les objectifs de coûts de maintenance et de fiabilité requis pour le système considéré.

Au niveau tactique, la maintenance prescriptive se traduit par une planification plus dynamique qu'avec une maintenance systématique du fait de la prise en compte de la

dégradation du système en temps réel. Cette planification a pour but d'optimiser la durée de vie des équipements tout en rationalisant la programmation des interventions et en tenant compte du risque de défaillance. Cette planification s'appuie ainsi directement sur les choix stratégiques précédemment évoqués.

Enfin, la maintenance prescriptive a pour rôle d'accompagner le décideur à un niveau opérationnel, notamment sur les questions de planification et de logistique. En effet, le diagnostic et le pronostic en temps réel amènent à faire des choix sur la base des informations générées. Parmi les questions à adresser dans le cadre d'une maintenance prescriptive, on peut citer :

- Pour la planification :
  - Faut-il prévoir une opération de maintenance ?
  - Laquelle ? Quelle est la procédure à réaliser ?
  - Quelles sont les personnes, compétences et qualifications requises pour la réaliser ?
  - Quelles sont les pièces et outils nécessaires ? Sont-ils disponibles, où et quand ?
  - Quand peut-on programmer l'intervention ? Quelle est la meilleure option ?
- Pour la logistique :
  - Peut-on réparer la pièce ?
  - Faut-il commander des pièces ? si oui, dans quels délais ?

Toutes ces questions relèvent de la proposition de scénarios complets pour l'utilisateur selon son mode de fonctionnement.

Pour élaborer de tels scénarios, il est nécessaire de s'appuyer sur les orientations stratégiques et tactiques préalablement définies. La prise en compte de ces problématiques est un enjeu déterminant dans la valorisation des informations générées dans l'approche présentées dans cette thèse. Y répondre constitue ainsi une perspective fonctionnelle majeure.



# Table des figures

1.1	Présentation de l'architecture fonctionnelle . . . . .	3
2.1	Classification des typologies de maintenance . . . . .	8
2.2	Classification et évolution historique de la maintenance . . . . .	11
2.3	Les 7 étapes du PHM (JOUIN et al. 2013) . . . . .	12
2.4	Défauts et techniques de prétraitement des données (FAMILI et al. 1997)	17
2.5	Principe du mécanisme de Publication/Souscription (MARÉCHAUX 2006) . . . . .	19
2.6	Exemple de réseau bayésien (BEN-GAL 2008) . . . . .	22
2.7	Illustration de détection de pattern par apprentissage non supervisé extrait de (HANACHI et al. 2018) . . . . .	25
2.8	Exemple de détection d'anomalie par les modèles (HANACHI et al. 2018)	26
2.9	Courbes de dégradation de santé d'un bien (SIKORSKA et al. 2011) . .	29
2.10	Répartition des méthodologies de pronostic dans la littérature (LEI et al. 2018) . . . . .	30
2.11	Illustration de pronostic par extrapolation (SIKORSKA et al. 2011) . .	31
2.12	Exemple de pronostic par approche statistique (HENG et al. 2009) . .	31
2.13	Comparaison des performances de 11 algorithmes d'apprentissage sur 4 jeux de données issus du simulateur C-MAPPS (ZHANG et al. 2017)	33
2.14	Procédure de pronostic par fusion de modèles physiques et orienté données d'après (LIAO et al. 2014) . . . . .	34
3.1	Rappel de l'architecture fonctionnelle de la solution proposée . . . . .	39
3.2	Positionnement de la partie II dans l'architecture fonctionnelle . . . . .	42
4.1	Relations entre donnée, information et connaissance (WATSON 2008)	45

---

4.2	Éléments constitutifs d'un système expert . . . . .	47
5.1	Exemple de graphe conceptuel (FU et al. 2017) . . . . .	50
5.2	Exemple de HIN au travers d'un réseau bibliographique (BASCANS 2017) . . . . .	51
5.3	Illustration de la notion de modèle (BÉZIVIN et al. 2004) . . . . .	56
5.4	Relation entre système, modèle et méta-modèle (BÉZIVIN et al. 2004)	56
5.5	Ontologie OntoProg (NUNEZ et al. 2017) . . . . .	58
5.6	Ontologie de documentation technique (KOUKIAS et al. 2015) . . . . .	59
5.7	Ontologie de diagnostic SemDia (MEHDI et al. 2017) . . . . .	60
5.8	Ontologie de surveillance et de maintenance de processus industriel (ELHDAD et al. 2013) . . . . .	61
5.9	Ontologie de diagnostic de pannes (F. XU et al. 2018) . . . . .	62
5.10	Structure de l'ontologie MFPO (CAO et al. 2019) . . . . .	63
5.11	Méta-modèle de maintenance conditionnelle (GUILLÉN et al. 2016) . .	64
5.12	Modèle de structure de la base de connaissances . . . . .	67
5.13	Exemple de base de connaissances . . . . .	69
6.1	Moteur d'inférences : processus d'application des règles . . . . .	72
6.2	Illustration du processus d'application des règles . . . . .	73
6.3	Positionnement de la partie III dans l'architecture fonctionnelle . . .	76
7.1	Principe de fonctionnement d'une EDA basée sur des CEP (BARTHE 2013; FÜLÖP et al. 2012) . . . . .	78
8.1	Schéma de principe de la vision MDA (BENABEN 2012) . . . . .	82
8.2	Principes de transformation de modèles (TRUPTIL et al. 2010) . . . . .	83
8.3	Transformation de modèles de la base de connaissances vers le langage EPL . . . . .	84
8.4	Méta-modèle des règles génériques (SARAZIN et al. 2020b) . . . . .	85
8.5	Méta-modèle EPL (BOUBETA-PUIG et al. 2014) . . . . .	88
9.1	Illustration de la détection d'une suppression dans la pièce . . . . .	93
9.2	Modèles CIM des règles de détection . . . . .	93
9.3	PIM de la première règle . . . . .	95

---

9.4	Modèles PIM décrivant la seconde règle . . . . .	96
9.5	Règle EPL pour le premier exemple . . . . .	97
9.6	Modèle de règle EPL pour la 2ème règle . . . . .	98
10.1	Évolution du trafic aéronautique payant international et intérieur de 2010 à 2019 (ICAO 2019b) . . . . .	104
11.1	Illustration du graphe obtenu par l'implémentation de la base de faits d'un sous-système d'avion . . . . .	109
11.2	Illustration d'une branche de la base de faits d'un point de hiérarchique	109
11.3	Processus d'identification d'un élément maintenable défaillant à partir d'un mode de panne . . . . .	109
11.4	Illustration des trois phases d'une courbe en baignoire . . . . .	111
11.5	Exemples de d'approximation des courbes cumulatives de défaillances empiriques par les lois de Weibull par élément maintenable et par modèle de pièce . . . . .	114
12.1	Résultats de l'identification d'un élément maintenable défaillant selon le modèle de la pièce . . . . .	116
12.2	Résultats de l'identification d'un élément maintenable défaillant selon le modèle de la pièce sans distinction des listes de taille 1 . . . . .	116
12.3	Résultat de l'identification d'un modèle de pièce défaillant . . . . .	117
12.4	Résultats de l'identification d'un modèle de la pièce défaillant sans distinction des listes de taille 1 . . . . .	118
13.1	Présentation de l'architecture technique . . . . .	120



# Liste des tableaux

2.1	Principales caractéristiques de l'EDA . . . . .	20
2.2	Répartition des publications par domaine . . . . .	27
2.3	Nombre de publications par application des systèmes experts . . . . .	28
5.1	Tableau comparatif des solutions de graphes de connaissances . . . . .	52
5.2	Tableau récapitulatif des notations de la base de connaissances . . . . .	54
8.1	Règles de mapping du modèle CIM vers le PIM . . . . .	87
8.2	Règles de mapping du PIM vers le PSM . . . . .	90
9.1	Application des règles de transformation du CIM au PIM pour la première règle . . . . .	94
9.2	Application des règles de mapping du CIM au PIM pour la seconde règle (model a) . . . . .	95
9.3	Application des règles de mapping du CIM au PIM pour la seconde règle (model b) . . . . .	95
9.4	Application des règles de mapping du PIM au PSM pour la première règle . . . . .	97
9.5	Application des règles de mapping du PIM au PSM pour la 2ème règle (modèle a) . . . . .	97
9.6	Application des règles de mapping du PIM au PSM pour la 2ème règle (modèle b) . . . . .	98



# Acronymes

**IMT** Institut Mines-Télécom : depuis 2017, intégration dans un seul institut des écoles de télécom et de la plupart des écoles des mines, sous la tutelle du ministère de l'Économie et des Finances. .... 1



# Bibliographie

- (Aalst 2014) Wil M. P. van der Aalst. “Data Scientist: The Engineer of the Future.” en. In: *Enterprise Interoperability VI*. Ed. by Kai Mertins, Frédérick Bénaben, Raúl Poler, and Jean-Paul Bourrières. Proceedings of the I-ESA Conferences. Springer International Publishing, 2014, pp. 13–26 (cit. on p. 14).
- (Académie française 2021) Académie française. *Dictionnaire de l’Académie française*. fr (cit. on p. 7).
- (Activus 2020) Activus. *Activus Software*. fr-fr (cit. on p. 108).
- (Aditiya et al. 2017) Nur Ashar Aditiya, Muhammad Rizky Dharmawan, Zaqiatud Darojah, and D Raden Sanggar. “Fault diagnosis system of rotating machines using Hidden Markov Model (HMM).” In: *2017 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC)*. IEEE. 2017, pp. 177–181 (cit. on p. 24).
- (AFNOR 2018) AFNOR. *NF EN 13306 - Maintenance — Terminologie de la maintenance*. Jan. 2018 (cit. on pp. 7, 8).
- (Agrawal et al. 2000) Rakesh Agrawal, Roberto Bayardo, and Ramakrishnan Srikant. “Athena: Mining-based interactive management of text databases.” In: *International Conference on Extending Database Technology*. Springer, 2000, pp. 365–379 (cit. on p. 49).
- (Angelova et al. 2012) Ralitsa Angelova, Gjergji Kasneci, and Gerhard Weikum. “Graffiti: graph-based classification in heterogeneous networks.” In: *World Wide Web 15.2 (2012)*, pp. 139–170 (cit. on p. 51).
- (Badkook 2016) Basem Badkook. “[Study to Determine the Aircraft on Ground \(AOG\) Cost of the Boeing 777 Fleet at X Airline](#).” en. In: *American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)* 25.1 (Sept. 2016), pp. 51–71 (cit. on pp. 9, 103).
- (Barthe 2013) Anne-Marie Barthe. “Prise en charge de l’agilité de workflows collaboratifs par une approche dirigée par les événements.” 2013EMAC0003. PhD thesis. 2013 (cit. on pp. 20, 78, 79).
- (Bascans 2017) Jérémy Bascans. “Modèles de mémoires d’entreprise avec intégration automatique d’informations.” 2017TOU30144. PhD thesis. 2017 (cit. on pp. 49–52).
- (Belaunde et al. 2003) Mariano Belaunde, Cory Casanave, Desmond DSouza, Keith Duddy, William El Kaim, Alan Kennedy, William Frank, David Frankel, Randall Hauch, and Stan Hendryx. *MDA Guide Version 1.0. 1*. 2003 (cit. on pp. 55, 82).

- 
- (Beleulmi et al. 2014) Salima Beleulmi, Ahmed Bellaouar, and Mohammed Benidir. “Reliability Modeling of Transport Systems: Influence of the Running-in Period of Life.” In: *Physics Procedia* 55 (2014), pp. 75–82 (cit. on pp. 111, 112).
- (Bellman 1978) Richard Bellman. *An introduction to artificial intelligence: Can computers think?* Thomson Course Technology, 1978 (cit. on p. 21).
- (Ben-Gal 2008) Irad Ben-Gal. “[Bayesian Networks](#).” en. In: *Encyclopedia of Statistics in Quality and Reliability*. American Cancer Society, 2008 (cit. on p. 22).
- (Benaben 2012) Frederick Benaben. “Conception de Système d’Information de Médiation pour la prise en charge de l’Interopérabilité dans les Collaborations d’Organisations.” Habilitation à diriger des recherches. Institut National Polytechnique de Toulouse, Oct. 2012 (cit. on p. 82).
- (J. Bezivin et al. 2001) J. Bezivin and O. Gerbe. “Towards a precise definition of the OMG/MDA framework.” In: *Proceedings 16th Annual International Conference on Automated Software Engineering (ASE 2001)*. Nov. 2001, pp. 273–280 (cit. on p. 55).
- (Jean Bezivin et al. 2006) Jean Bezivin, Fabian Büttner, Martin Gogolla, Frederic Jouault, Ivan Kurtev, and Arne Lindow. “Model Transformations? Transformation Models!” en. In: *Model Driven Engineering Languages and Systems*. Ed. by Oscar Nierstrasz, Jon Whittle, David Harel, and Gianna Reggio. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2006, pp. 440–453 (cit. on p. 82).
- (Bézivin et al. 2004) Jean Bézivin and Jean-Pierre Briot. “Sur les principes de base de l’ingénierie des modèles.” In: *Obj. Logiciel Base données Réseaux* 10.4 (2004), pp. 145–157 (cit. on pp. 55, 56).
- (Blanchard et al. 1995) Benjamin S. Blanchard, Dinesh C. Verma, and Elmer L. Peterson. *Maintainability : a key to effective serviceability and maintenance management*. English. New York : John Wiley & Sons, Inc, 1995 (cit. on p. 11).
- (Bloch et al. 1983) H. P. Bloch and F. K. Geitner. *Practical machinery management for process plants. Volume 2: Machinery failure analysis and troubleshooting*. English. Tech. rep. Gulf Publishing Co., Houston, TX, Jan. 1983 (cit. on pp. 20, 29).
- (Borst 1999) Willem Nico Borst. “Construction of engineering ontologies for knowledge sharing and reuse.” In: (1999) (cit. on p. 54).
- (Boubeta-Puig et al. 2014) Juan Boubeta-Puig, Guadalupe Ortiz, and Inmaculada Medina-Bulo. “A model-driven approach for facilitating user-friendly design of complex event patterns.” In: *Expert Systems with Applications* 41.2 (2014), pp. 445–456 (cit. on pp. 81, 83, 88, 89).
- (Campos 2009) Jaime Campos. “[Development in the application of ICT in condition monitoring and maintenance](#).” In: *Computers in Industry* 60.1 (Jan. 2009), pp. 1–20 (cit. on p. 14).
- (Cao et al. 2019) Qiushi Cao, Ahmed Samet, Cecilia Zanni-Merk, François de Bertrand de Beuvron, and Christoph Reich. “An ontology-based approach for failure classification in predictive maintenance using fuzzy C-means and SWRL rules.” In: *Procedia Computer Science* 159 (2019), pp. 630–639 (cit. on p. 63).
- (Carrino et al. 2016) Francesco Carrino, Elena Mugellini, Omar Abou Khaled, Nabil Ouerhani, and Juergen Ehrensberger. “[iNUIT: Internet of Things for Urban Innovation](#).” en. In: *Future Internet* 8.2 (June 2016), p. 18 (cit. on p. 15).

- (Q. Chen et al. 2014) Q. Chen, S. Abdelwahed, and A. Erradi. “[A Model-Based Validated Autonomic Approach to Self-Protect Computing Systems](#).” In: *IEEE Internet of Things Journal* 1.5 (Oct. 2014), pp. 446–460 (cit. on p. 15).
- (X. Chen et al. 2020) Xiaojun Chen, Shengbin Jia, and Yang Xiang. “A review: Knowledge reasoning over knowledge graph.” In: *Expert Systems with Applications* 141 (2020), p. 112948 (cit. on p. 49).
- (Chevallet 1992) Jean-Pierre Chevallet. “Un modèle logique de recherche d’informations appliqué au formalisme des graphes conceptuels: le prototype ELEN et son expérimentation sur un corpus de composants logiciels.” PhD thesis. Grenoble 1, 1992 (cit. on p. 50).
- (Corset et al. 2003) Franck Corset, Gilles Celeux, and André Lannoy. “Introduction du Retour d’Expérience dans les Réseaux bayésiens.” In: *Qualita*. Nancy, France, 2003 (cit. on p. 50).
- (Couallier V. 2016) & Bayle F. Couallier V. Denis L. “MTBF computations for aeronautical maintained repairable systems with dormant failure mode of control device.” In: *20e Congrès de maîtrise des risques et de sûreté de fonctionnement Communication 4E /2* (2016), pp. 1–8 (cit. on p. 112).
- (J. F. Davis et al. 1999) J. F. Davis, M. J. Piovoso, K. A. Hoo, and B. R. Bakshi. “[Process Data Analysis and Interpretation](#).” In: *Advances in Chemical Engineering*. Ed. by James Wei. Vol. 25. Academic Press, Jan. 1999, pp. 1–103 (cit. on p. 16).
- (R. Davis et al. 1993) Randall Davis, Howard Shrobe, and Peter Szolovits. “What is a knowledge representation?” In: *AI magazine* 14.1 (1993), pp. 17–17 (cit. on pp. 21, 51).
- (Ebeling 1997) CE Ebeling. “Reliability and Maintainability Engineering, The McRaw-Hill Companies.” In: *INC, USA* (1997) (cit. on p. 111).
- (Elhdad et al. 2013) Rabee Elhdad, Naveen Chilamkurti, and Torab Torabi. “[An ontology-based framework for process monitoring and maintenance in petroleum plant](#).” In: *Journal of Loss Prevention in the Process Industries* 26.1 (2013), pp. 104–116 (cit. on pp. 60, 61).
- (Elsayed et al. 1979) E. A. Elsayed and Balbir S. Dhillon. “[Repairable systems with one standby unit](#).” In: *Microelectronics Reliability* 19.3 (Jan. 1979), pp. 243–245 (cit. on p. 8).
- (Engel et al. 2000) S. J. Engel, B. J. Gilmartin, K. Bongort, and A. Hess. “Prognostics, the real issues involved with predicting life remaining.” In: *2000 IEEE Aerospace Conference. Proceedings (Cat. No.00TH8484)*. Vol. 6. 2000, 457–469 vol.6 (cit. on p. 29).
- (Etzion et al. 2011) Opher Etzion and Peter Niblett. *Event processing in action*. Manning, 2011 (cit. on p. 79).
- (Famili et al. 1997) A. Famili, Wei-Min Shen, Richard Weber, and Evangelos Simoudis. “[Data Preprocessing and Intelligent Data Analysis](#).” en. In: *Intelligent Data Analysis* 1.1 (Jan. 1997), pp. 3–23 (cit. on pp. 16–18).
- (Feigenbaum et al. 1981) Edward A Feigenbaum, Avron Barr, and Paul R Cohen. “The handbook of artificial intelligence.” In: (1981) (cit. on p. 21).
- (Ferri et al. 2013) Felipe Augusto Sviaghin Ferri, Leonardo Ramos Rodrigues, João Paulo Pordeus Gomes, Ivo Paixão de Medeiros, Roberto Kawakami Harrop Galvão, and Cairo Lúcio Nascimento. “Combining PHM information and system archi-

- 
- ecture to support aircraft maintenance planning.” In: *2013 IEEE International Systems Conference (SysCon)*. IEEE. 2013, pp. 60–65 (cit. on p. 35).
- (Fertier 2018) Audrey Fertier. “Interprétation automatique de données hétérogènes pour la modélisation de situations collaboratives: Application à la gestion de crise.” PhD thesis. Nov. 2018 (cit. on p. 15).
- (Fiege et al. 2003) Ludger Fiege, Gero Mühl, and Felix C Gärtner. “Modular event-based systems.” In: *The Knowledge Engineering Review* 17.4 (2003), pp. 55–85 (cit. on p. 20).
- (Finch et al. 1986) Byron J. Finch and James P. Gilbert. “[Developing maintenance craft labor efficiency through an integrated planning and control system: A prescriptive model.](#)” In: *Journal of Operations Management*. Special Combined Issue 6.3 (May 1986), pp. 449–459 (cit. on p. 8).
- (Fu et al. 2017) Zhangjie Fu, Fengxiao Huang, Kui Ren, Jian Weng, and Cong Wang. “Privacy-preserving smart semantic search based on conceptual graphs over encrypted outsourced data.” In: *IEEE Transactions on Information Forensics and Security* 12.8 (2017), pp. 1874–1884 (cit. on p. 50).
- (Fülöp et al. 2012) Lajos Jenő Fülöp, Árpád Beszédes, Gabriella Tóth, Hunor Demeter, László Vidács, and Lóránt Farkas. “Predictive complex event processing: a conceptual framework for combining complex event processing and predictive analytics.” In: *Proceedings of the Fifth Balkan Conference in Informatics*. 2012, pp. 26–31 (cit. on p. 78).
- (Gedik et al. 2008) Bugra Gedik, Henrique Andrade, Kun-Lung Wu, Philip S Yu, and Myungcheol Doo. “SPADE: The System S declarative stream processing engine.” In: *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. 2008, pp. 1123–1134 (cit. on p. 79).
- (Gómez-Pérez et al. 2006) Asunción Gómez-Pérez, Mariano Fernández-López, and Oscar Corcho. *Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*. Springer Science & Business Media, 2006 (cit. on p. 55).
- (T. R. Gruber 1993) Thomas R Gruber. “A translation approach to portable ontology specifications.” In: *Knowledge acquisition* 5.2 (1993), pp. 199–220 (cit. on p. 54).
- (T. Gruber 1993) Tom Gruber. *What is an Ontology*. 1993 (cit. on p. 55).
- (Guille et al. 2013) Adrien Guille, Hakim Hacid, Cecile Favre, and Djamel A. Zighed. “[Information Diffusion in Online Social Networks: A Survey.](#)” In: *SIGMOD Rec.* 42.2 (July 2013), pp. 17–28 (cit. on p. 14).
- (Guillén et al. 2016) Antonio J. Guillén, Adolfo Crespo, Juan Fco. Gómez, and Maria Dolores Sanz. “[A framework for effective management of condition based maintenance programs in the context of industrial development of E-Maintenance strategies.](#)” In: *Computers in Industry* 82 (Oct. 2016), pp. 170–185 (cit. on pp. 64, 66, 68, 91).
- (Haddad et al. 2012) Gilbert Haddad, Peter A Sandborn, and Michael G Pecht. “An options approach for decision support of systems with prognostic capabilities.” In: *IEEE Transactions on reliability* 61.4 (2012), pp. 872–883 (cit. on pp. 34, 35).
- (Hanachi et al. 2018) Houman Hanachi, Christopher Mechefske, Jie Liu, Avisekh Banerjee, and Ying Chen. “Performance-based gas turbine health monitoring,

- diagnostics, and prognostics: A survey.” In: *IEEE Transactions on Reliability* 67.3 (2018). Publisher: IEEE, pp. 1340–1363 (cit. on pp. 11, 24–26).
- (Heng et al. 2009) Aiwina Heng, Andy CC Tan, Joseph Mathew, Neil Montgomery, Dragan Banjevic, and Andrew KS Jardine. “Intelligent condition-based prediction of machinery reliability.” In: *Mechanical Systems and Signal Processing* 23.5 (2009), pp. 1600–1614 (cit. on p. 31).
- (Hsu et al. 2018) Che-Sheng Hsu and Jehn-Ruey Jiang. “Remaining useful life estimation using long short-term memory deep learning.” In: *2018 IEEE International Conference on Applied System Invention (ICASI)*. IEEE. 2018, pp. 58–61 (cit. on p. 122).
- (Hu et al. 2018) L. Hu and Q. Ni. “IoT-Driven Automated Object Detection Algorithm for Urban Surveillance Systems in Smart Cities.” In: *IEEE Internet of Things Journal* 5.2 (Apr. 2018), pp. 747–754 (cit. on p. 15).
- (Hurvich et al. 1989) Clifford M Hurvich and Chih-Ling Tsai. “Regression and time series model selection in small samples.” In: *Biometrika* 76.2 (1989), pp. 297–307 (cit. on p. 113).
- (ICAO 2019a) ICAO. *Le monde du transport aérien en 2019*. 2019 (cit. on p. 103).
- (ICAO 2019b) ICAO. *Présentation des résultats statistiques du transport aérien en 2019*. 2019 (cit. on pp. 103, 104).
- (Isermann 1984) Rolf Isermann. “Process fault detection based on modeling and estimation methods—A survey.” In: *Automatica* 20.4 (July 1984), pp. 387–404 (cit. on p. 26).
- (ISO 2012) ISO. *ISO 13379-1:2012 Condition monitoring and diagnostics of machines — Data interpretation and diagnostics techniques — Part 1: General guidelines*. May 2012 (cit. on p. 23).
- (ISO 2016) ISO. *NF ISO 13381-1 - Surveillance et diagnostic des machines — Pronostic — Partie 1 : Lignes directrices générales*. Feb. 2016 (cit. on p. 29).
- (J. Seltman 2018) H J. Seltman. *Experimental design and analysis*. July 2018 (cit. on p. 18).
- (Jackson 1998) Peter Jackson. *Introduction to Expert Systems*. 3rd. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1998 (cit. on pp. 26, 46).
- (Jardine et al. 2006) Andrew K. S. Jardine, Daming Lin, and Dragan Banjevic. “A review on machinery diagnostics and prognostics implementing condition-based maintenance.” In: *Mechanical Systems and Signal Processing* 20.7 (Oct. 2006), pp. 1483–1510 (cit. on pp. 14, 25, 26).
- (Josuttis 2007) Nicolai M Josuttis. *SOA in practice: the art of distributed system design*. " O'Reilly Media, Inc.", 2007 (cit. on p. 19).
- (Jouin et al. 2013) Marine Jouin, Rafael Gouriveau, Daniel Hissel, Marie-Cécile Péra, and Noureddine Zerhouni. “Prognostics and Health Management of PEMFC – State of the art and remaining challenges.” In: *International Journal of Hydrogen Energy* 38.35 (Nov. 2013), pp. 15307–15317 (cit. on p. 12).
- (Kalogirou 2003) Soteris A. Kalogirou. “Artificial intelligence for the modeling and control of combustion processes: a review.” In: *Progress in Energy and Combustion Science* 29.6 (Jan. 2003), pp. 515–566 (cit. on pp. 46, 47).
- (Kelly 1989) Anthony Kelly. “Maintenance and its management.” In: Conference Communication, 1989 (cit. on p. 8).

- 
- (Kitchin et al. 2015) Rob Kitchin and Tracey P Lauriault. “Small data in the era of big data.” In: *GeoJournal* 80.4 (2015), pp. 463–475 (cit. on pp. 2, 38, 104).
- (Koukias et al. 2015) Andreas Koukias, Dražen Nadoveza, and Dimitris Kiritsis. “An ontology-based approach for modelling technical documentation towards ensuring asset optimisation.” In: *International Journal of Product Lifecycle Management* 8.1 (2015), pp. 24–45 (cit. on pp. 57, 59).
- (Kumar et al. 2009) Edwin Vijay Kumar, SK Chaturvedi, and AW Deshpandé. “Maintenance of industrial equipment: degree of certainty with fuzzy modelling using predictive maintenance.” In: *International Journal of Quality & Reliability Management* (2009) (cit. on p. 122).
- (Lai 2014) Chin-Diew Lai. “Generalized weibull distributions.” In: *Generalized Weibull Distributions*. Springer, 2014, pp. 107–111 (cit. on p. 112).
- (Laylavi et al. 2017) Farhad Laylavi, Abbas Rajabifard, and Mohsen Kalantari. “Event relatedness assessment of Twitter messages for emergency response.” In: *Information Processing & Management* 53.1 (Jan. 2017), pp. 266–280 (cit. on p. 15).
- (Lee et al. 2014) Jay Lee, Fangji Wu, Wenyu Zhao, Masoud Ghaffari, Linxia Liao, and David Siegel. “Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications.” In: *Mechanical Systems and Signal Processing* 42.1 (Jan. 2014), pp. 314–334 (cit. on pp. 10, 12, 24).
- (Lehner et al. 2000) Franz Lehner and Ronald K Maier. “How can organizational memory theories contribute to organizational memory systems?” In: *Information Systems Frontiers* 2.3 (2000), pp. 277–298 (cit. on p. 46).
- (Lei et al. 2018) Yaguo Lei, Naipeng Li, Liang Guo, Ningbo Li, Tao Yan, and Jing Lin. “Machinery health prognostics: A systematic review from data acquisition to RUL prediction.” In: *Mechanical systems and signal processing* 104 (2018), pp. 799–834 (cit. on pp. 10, 29, 30).
- (Levine et al. 1990) Pierre Levine and Jean-Charles Pomerol. *Systèmes interactifs d’aide à la décision et systèmes experts*. fr. Hermès, 1990 (cit. on p. 46).
- (Liao et al. 2014) Linxia Liao and Felix Köttig. “Review of hybrid prognostics approaches for remaining useful life prediction of engineered systems, and an application to battery life prediction.” In: *IEEE Transactions on Reliability* 63.1 (2014), pp. 191–207 (cit. on pp. 33, 34).
- (Liebowitz 1995) Jay Liebowitz. “Expert systems: A short introduction.” In: *Engineering Fracture Mechanics* 50.5-6 (1995). Publisher: Elsevier, pp. 601–607 (cit. on pp. 46, 47).
- (Long et al. 2008) Hao Long and Xinmin Wang. “Aircraft fuel system diagnostic fault detection through expert system.” In: *2008 7th World Congress on Intelligent Control and Automation*. IEEE. 2008, pp. 7104–7107 (cit. on p. 27).
- (Macmann et al. 2016) Owen B Macmann, Timothy M Seitz, Alireza R Behbahani, and Kelly Cohen. “Performing diagnostics & prognostics on simulated engine failures using neural networks.” In: *52nd AIAA/SAE/ASEE Joint Propulsion Conference*. 2016, p. 4807 (cit. on p. 24).
- (Maréchaux 2006) Jean-Louis Maréchaux. “Combining service-oriented architecture and event-driven architecture using an enterprise service bus.” In: *IBM developer works* 12691275 (2006) (cit. on p. 19).

- (Marques et al. 2019) Henrique Marques and Alessandro Giacotto. “Prescriptive Maintenance: Building Alternative Plans for Smart Operations.” In: *FT2019. Proceedings of the 10th Aerospace Technology Congress, October 8-9, 2019, Stockholm, Sweden*. 162. Linköping University Electronic Press. 2019, pp. 231–236 (cit. on p. 122).
- (Mazac 2015) Sébastien Mazac. “Approche décentralisée de l’apprentissage constructiviste et modélisation multi-agent du problème d’amorçage de l’apprentissage sensorimoteur en environnement continu : application à l’intelligence ambiante.” fr. PhD thesis. Université Claude Bernard - Lyon I, Oct. 2015 (cit. on p. 23).
- (Mazhar Rathore et al. 2017) M. Mazhar Rathore, Awais Ahmad, Anand Paul, Won-Hwa Hong, and HyunCheol Seo. “Advanced computing model for geosocial media using big data analytics.” en. In: *Multimedia Tools and Applications* 76.23 (Dec. 2017), pp. 24767–24787 (cit. on p. 15).
- (Mehdi et al. 2017) Gulnar Mehdi, Evgeny Kharlamov, Ognjen Savković, Guohui Xiao, E Güzel Kalaycı, Sebastian Brandt, Ian Horrocks, Mikhail Roshchin, and Thomas Runkler. “Semantic rule-based equipment diagnostics.” In: *International Semantic Web Conference*. Springer. 2017, pp. 314–333 (cit. on pp. 58, 60).
- (Michelson 2006) Brenda M Michelson. “Event-driven architecture overview.” In: *Patricia Seybold Group 2.12* (2006), pp. 10–1571 (cit. on pp. 19, 20).
- (Michie et al. 1982) Donald Michie et al. *Introductory readings in expert systems*. Vol. 1. CRC Press, 1982 (cit. on p. 46).
- (MOF 2002) OMG MOF. *OMG Meta Object Facility (MOF) Specification v1. 4*. 2002. URL: <https://www.omg.org/cgi-bin/apps/doc> (cit. on p. 55).
- (MOF 2016) OMG MOF. *OMG Meta Object Facility (MOF) Specification v2.5.1*. Oct. 2016. URL: <https://www.omg.org/spec/MOF> (cit. on p. 55).
- (Musaev et al. 2015) A. Musaev, D. Wang, and C. Pu. “LITMUS: A Multi-Service Composition System for Landslide Detection.” In: *IEEE Transactions on Services Computing* 8.5 (Sept. 2015), pp. 715–726 (cit. on p. 15).
- (Nunez et al. 2017) David Lira Nunez and Milton Borsato. “An ontology-based model for prognostics and health management of machines.” In: *Journal of Industrial Information Integration* 6 (2017), pp. 33–46 (cit. on pp. 57, 58).
- (Nwobi et al. 2014) Felix Noyanim Nwobi and Chukwudi Anderson Ugomma. “A comparison of methods for the estimation of Weibull distribution parameters.” In: *Metodoloski zvezki* 11.1 (2014). Publisher: Anuska Ferligoj, p. 65 (cit. on p. 113).
- (Pecht et al. 2010) Michael Pecht and Rubyca Jaai. “A prognostics and health management roadmap for information and electronics-rich systems.” In: *Microelectronics Reliability* 50.3 (Mar. 2010), pp. 317–323 (cit. on p. 25).
- (Peele et al. 1984) TT Peele and RL Chapman. “Determining maintenance manpower requirements.” In: *Plant Engineering* 38.30 (1984). Publisher: CAHNERS-DENVER PUBLISHING CO 8773 S RIDGELINE BLVD, HIGHLANDS RANCH, CO . . . , pp. 62–65 (cit. on p. 8).
- (Perera et al. 2014) C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos. “Context Aware Computing for The Internet of Things: A Survey.” In: *IEEE Communications Surveys Tutorials* 16.1 (2014), pp. 414–454 (cit. on p. 14).

- 
- (Pham et al. 1988) DT Pham and PTN Pham. “Expert systems in mechanical and manufacturing engineering.” In: *The International Journal of Advanced Manufacturing Technology* 3.3 (1988), pp. 3–21 (cit. on pp. 46, 47).
- (Preston et al. 1972) K Preston and JR Carvalko. “On determining optimum simple Golay marking transformations for binary image processing.” In: *IEEE Transactions on Computers* 100.12 (1972), pp. 1430–1433 (cit. on p. 24).
- (Rich et al. 1991) E Rich and K Knight. “Introduction to Artificial Networks.” In: *Mac Graw-Hill Publications, New York* (1991) (cit. on p. 21).
- (Russell et al. 1995) Stuart Russell, Peter Norvig, and A Intelligence. “A modern approach.” In: *Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs* 25.27 (1995), pp. 79–80 (cit. on p. 21).
- (Sandborn 2005) Peter Sandborn. “A decision support model for determining the applicability of prognostic health management (PHM) approaches to electronic systems.” In: *Annual Reliability and Maintainability Symposium, 2005. Proceedings.* IEEE. 2005, pp. 422–427 (cit. on pp. 34, 35).
- (Sarazin et al. 2021) Alexandre Sarazin, Jérémy Bascans, Jean-Baptiste Sciau, Jiefu Song, Bruno Supiot, Aurélie Montarnal, Xavier Lorca, and Sébastien Truptil. “Expert system dedicated to condition-based maintenance based on a knowledge graph approach: Application to an aeronautic system.” In: *Expert Systems with Applications* (2021), p. 115767 (cit. on pp. 3, 39, 73).
- (Sarazin et al. 2020a) Alexandre Sarazin, Sebastien Truptil, Aurélie Montarnal, Jérémy Bascans, and Xavier Lorca. “Model Transformation from CBM to EPL Rules to Detect Failure Symptoms.” In: *International Conference on Model-Driven Engineering and Software Development.* Springer. 2020, pp. 200–224 (cit. on pp. 3, 39, 75).
- (Sarazin et al. 2020b) Alexandre Sarazin, Sébastien Truptil, Aurelie Montarnal, Jacques Lamothe, Julien Commanay, and Laurent Sagaspe. “Towards Model Transformation from a CBM Model to CEP Rules to Support Predictive Maintenance.” In: *MODELSWARS 2020-The 8th International Conference on Model-Driven Engineering and Software Development.* Vol. 1. SciTePress. 2020, pp. 205–215 (cit. on pp. 75, 85).
- (Schalkoff 1990) Robert J Schalkoff. *Artificial intelligence: an engineering approach.* McGraw-Hill New York, 1990 (cit. on p. 21).
- (Sheut et al. 1994) C. Sheut and L. J. Krajewski. “A decision model for corrective maintenance management.” In: *International Journal of Production Research* 32.6 (June 1994), pp. 1365–1382 (cit. on p. 8).
- (Sicular 2013) Svetlana Sicular. *Gartner’s Big Data Definition Consists of Three Parts, Not to Be Confused with Three “V”s.* Feb. 2013 (cit. on pp. 14, 77).
- (Siegel 2014) J Siegel. *MDA Guide, revision 2.0.* 2014 (cit. on p. 81).
- (Sikorska et al. 2011) JZ Sikorska, Melinda Hodkiewicz, and Lin Ma. “Prognostic modelling options for remaining useful life estimation by industry.” In: *Mechanical systems and signal processing* 25.5 (2011), pp. 1803–1836 (cit. on pp. 29–31, 33).
- (Simon 2013) Phil Simon. *Too big to ignore: the business case for big data.* Vol. 72. John Wiley & Sons, 2013 (cit. on p. 23).

- (Simpson 1978) Vincent P. Simpson. “[Optimum Solution Structure for a Repairable Inventory Problem](#).” In: *Operations Research* 26.2 (Apr. 1978), pp. 270–281 (cit. on p. 8).
- (Sowa 1983) J.F. Sowa. *Conceptual structures: Information processing in mind and machine*. English. United States: Addison-Wesley Pub., Reading, MA, Jan. 1983 (cit. on p. 50).
- (Studer et al. 1998) Rudi Studer, V Richard Benjamins, and Dieter Fensel. “Knowledge engineering: Principles and methods.” In: *Data & knowledge engineering* 25.1-2 (1998), pp. 161–197 (cit. on p. 54).
- (B. Sun et al. 2010) Bo Sun, Shengkui Zeng, Rui Kang, and Michael Pecht. “Benefits analysis of prognostics in systems.” In: *2010 Prognostics and System Health Management Conference*. IEEE. 2010, pp. 1–8 (cit. on p. 24).
- (Y. Sun et al. 2012) Yizhou Sun and Jiawei Han. “Mining heterogeneous information networks: principles and methodologies.” In: *Synthesis Lectures on Data Mining and Knowledge Discovery* 3.2 (2012). Publisher: Morgan & Claypool Publishers, pp. 1–159 (cit. on pp. 51, 52).
- (Taylor et al. 2009) Hugh Taylor, Angela Yochem, Les Phillips, and Frank Martinez. *Event-driven architecture: how SOA enables the real-time enterprise*. Pearson Education, 2009 (cit. on p. 19).
- (Truptil 2011) Sébastien Truptil. “Etude de l’approche de l’interopérabilité par médiation dans le cadre d’une dynamique de collaboration appliquée à la gestion de crise.” 2011INPT0012. PhD thesis. 2011 (cit. on p. 55).
- (Truptil et al. 2010) Sébastien Truptil, Frédérick Bénaben, Nicolas Salatge, Chihab Hanachi, Vincent Chapurlat, Jean-Paul Pignon, and Hervé Pingaud. “Mediation Information System Engineering for Interoperability Support in Crisis Management.” en. In: *Enterprise Interoperability IV*. Ed. by Keith Popplewell, Jenny Harding, Raul Poler, and Ricardo Chalmeta. Springer London, 2010, pp. 187–197 (cit. on p. 83).
- (Tsang 1995) Albert HC Tsang. “Condition-based maintenance: tools and decision making.” In: *Journal of quality in maintenance engineering* (1995). Publisher: MCB UP Ltd (cit. on p. 9).
- (Tu et al. 2007) Fang Tu, Sudipto Ghoshal, Jianhui Luo, Gautam Biswas, Sankaran Mahadevan, Link Jaw, and Kelly Navarra. “PHM integration with maintenance and inventory management systems.” In: *2007 IEEE Aerospace Conference*. IEEE. 2007, pp. 1–12 (cit. on pp. 34, 35).
- (Tukey 1977) John W. Tukey. *Exploratory Data Analysis*. Anglais. 1st ed. Reading, Mass: Pearson, Jan. 1977 (cit. on p. 18).
- (Vianu 1997) Victor Vianu. “Rule-based languages.” In: *Annals of Mathematics and Artificial Intelligence* 19.1 (1997), pp. 215–259 (cit. on p. 79).
- (Vichare et al. 2006) N. M. Vichare and M. G. Pecht. “[Prognostics and health management of electronics](#).” In: *IEEE Transactions on Components and Packaging Technologies* 29.1 (Mar. 2006), pp. 222–229 (cit. on p. 11).
- (Wagner 2017) William P Wagner. “Trends in expert system development: A longitudinal content analysis of over thirty years of expert system case studies.” In: *Expert systems with applications* 76 (2017), pp. 85–96 (cit. on p. 27).

- 
- (J. Wang et al. 2018) Jiujian Wang, Guilin Wen, Shaopu Yang, and Yongqiang Liu. “Remaining useful life estimation in prognostics using deep bidirectional lstm neural network.” In: *2018 Prognostics and system health management conference (PHM-Chongqing)*. IEEE. 2018, pp. 1037–1042 (cit. on p. 122).
- (S. Wang et al. 2017) Senzhang Wang, Xiaoming Zhang, Jianping Cao, Lifang He, Leon Stenneth, Philip S. Yu, Zhoujun Li, and Zhiqiu Huang. “Computing Urban Traffic Congestions by Incorporating Sparse GPS Probe Data and Social Media Data.” In: *ACM Transactions on Information Systems (TOIS)* 35.4 (Aug. 2017), p. 40 (cit. on p. 15).
- (Z. Wang et al. 2012) Z. Wang, J. L. Zarader, and S. Argentieri. “A novel aircraft engine fault diagnostic and prognostic system based on SVM.” In: *2012 IEEE International Conference on Condition Monitoring and Diagnosis*. Sept. 2012, pp. 723–728 (cit. on p. 24).
- (Watson 2008) Richard T Watson. *Data management, databases and organizations*. John Wiley & Sons, 2008 (cit. on p. 45).
- (Winston 1992) P Winston. *Artificial Intelligence, chapter 25*. 1992 (cit. on p. 21).
- (Xiaoxue et al. 2019) Liu Xiaoxue, Bai Xuesong, Wang Longhe, Ren Bingyuan, Lu Shuhan, and Li Lin. “Review and Trend Analysis of Knowledge Graphs for Crop Pest and Diseases.” In: *IEEE Access* 7 (2019). Conference Name: IEEE Access, pp. 62251–62264 (cit. on pp. 27, 48, 49).
- (Xie et al. 1996) Ming Xie and Chin Diew Lai. “Reliability analysis using an additive Weibull model with bathtub-shaped failure rate function.” In: *Reliability Engineering & System Safety* 52.1 (1996), pp. 87–93 (cit. on p. 112).
- (F. Xu et al. 2018) Feixiang Xu, Xinhui Liu, Wei Chen, Chen Zhou, and Bingwei Cao. “Ontology-based method for fault diagnosis of loaders.” In: *Sensors* 18.3 (2018), p. 729 (cit. on pp. 61, 62).
- (Z. Xu et al. 2017) Z. Xu, N. Y. Yen, H. Zhang, X. Wei, Z. Lv, K. R. Choo, L. Mei, and X. Luo. “Social Sensors Based Online Attention Computing of Public Safety Events.” In: *IEEE Transactions on Emerging Topics in Computing* 5.3 (July 2017), pp. 403–411 (cit. on p. 15).
- (Yan et al. 2018) Hao Yan, Yongxiang Xu, Feiyang Cai, He Zhang, Weiduo Zhao, and Chris Gerada. “PWM-VSI fault diagnosis for a PMSM drive based on the fuzzy logic approach.” In: *IEEE Transactions on Power Electronics* 34.1 (2018), pp. 759–768 (cit. on p. 24).
- (Yang et al. 2020) Zebin Yang, Aijun Zhang, and Agus Sudjianto. “Enhancing explainability of neural networks through architecture constraints.” In: *IEEE Transactions on Neural Networks and Learning Systems* (2020) (cit. on p. 48).
- (Žarković et al. 2017) Mileta Žarković and Zlatan Stojković. “Analysis of artificial intelligence expert systems for power transformer condition monitoring and diagnostics.” In: *Electric Power Systems Research* 149 (2017), pp. 125–136 (cit. on p. 27).
- (Zhang et al. 2017) C. Zhang, P. Lim, A. K. Qin, and K. C. Tan. “Multiobjective Deep Belief Networks Ensemble for Remaining Useful Life Estimation in Prognostics.” In: *IEEE Transactions on Neural Networks and Learning Systems* 28.10 (Oct. 2017), pp. 2306–2318 (cit. on pp. 32, 33).

(Zhang L. 2008) Tang L Zhang L. Xie M. &. “On Weighted Least Squares Estimation for the Parameters of Weibull Distribution.” In: *Recent Advances in Reliability and Quality in Design*. Springer, 2008, pp. 57–84 (cit. on p. 112).



# Table des matières

<i>Remerciements</i> .....	iii
<i>Sommaire</i> .....	v
<hr/>	
<b>1 Introduction générale</b> .....	1
<b>Partie I État de l’art et positionnement</b> .....	5
<b>2 État de l’art</b> .....	7
2.1 La notion de maintenance .....	7
2.1.1 La maintenance corrective .....	8
2.1.2 La maintenance préventive .....	9
2.2 Le pronostic et la gestion de la santé - PHM .....	11
2.2.1 Présentation du PHM .....	11
2.2.2 Surveillance .....	14
2.2.3 Détection d’anomalies et identification des pannes .....	20
2.2.4 Pronostic .....	28
2.2.5 Aide à la décision .....	34
<b>3 Verrous scientifiques et positionnement</b> .....	37
3.1 Identification des verrous scientifiques .....	37
3.2 Présentation de l’approche proposée .....	38
<b>Partie II Présentation de la solution de diagnostic</b> .....	41
<b>4 Principe d’un système expert</b> .....	45
<b>5 Base de connaissances</b> .....	49
5.1 Stockage des données .....	49
5.2 Structure des connaissances .....	54
5.2.1 Ontologie OntoProg .....	57
5.2.2 Ontologie de documentation technique .....	57
5.2.3 Ontologie de diagnostic SemDia .....	58
5.2.4 Ontologie de surveillance et de maintenance de processus industriel .....	60

5.2.5	Ontologie de diagnostic de pannes	61
5.2.6	Ontologie de prévision de pannes industrielles	63
5.2.7	Méta-modèle de maintenance conditionnelle	64
5.2.8	Méta-modèle retenu pour le SE	66
<b>6</b>	<b>Moteur d'inférences</b>	<b>71</b>
<hr/>		
<b>Partie III</b>	<b>Transformation de modèle</b>	<b>75</b>
<b>7</b>	<b>Architecture dirigée par les évènements</b>	<b>77</b>
<b>8</b>	<b>Architecture dirigée par les modèles</b>	<b>81</b>
8.1	Principe	81
8.2	Transformation des règles de surveillance vers un modèle de règles génériques	84
8.2.1	Métamodèle des règles génériques	84
8.2.2	Transformation du CIM vers le PIM	86
8.3	Transformation des règles génériques en EPL	88
8.3.1	Présentation du méta-modèle EPL	88
8.3.2	Transformation du PIM vers le PSM	89
<b>9</b>	<b>Illustrations</b>	<b>91</b>
9.1	Description de système	91
9.2	Présentation des CIM	92
9.3	Présentation des PIM	94
9.4	Présentation des PSM	96
<hr/>		
<b>Partie IV</b>	<b>Cas d'étude</b>	<b>101</b>
<b>10</b>	<b>Les enjeux de la maintenance dans le domaine aéronautique</b>	<b>103</b>
<b>11</b>	<b>Protocole expérimental</b>	<b>107</b>
11.1	Présentation des données	107
11.2	Identification d'un composant défaillant	108
11.3	Le modèle statistique : la loi de Weibull	110
<b>12</b>	<b>Analyse des résultats</b>	<b>115</b>
<hr/>		
<b>13</b>	<b>Conclusion et perspectives</b>	<b>119</b>
13.1	Bilan des travaux	119
13.2	Limites et perspectives	121
<hr/>		
	<i>Table des figures</i>	125
	<i>Liste des tableaux</i>	129
	<i>Acronymes</i>	131

*Bibliographie* . . . . . 133  
*Table des matières* . . . . . 145



## Résumé

### Mise en oeuvre d'une approche d'estimation de besoin de maintenance par système à base de connaissances

Le domaine de la maintenance fait aujourd'hui face à de nouveaux enjeux majeurs dus à l'émergence du numérique et à la multiplicité des données. De plus, la complexité croissante des systèmes maintenus engendre une augmentation des risques et de la diversité des pannes pouvant les affecter. Cette problématique survient par ailleurs dans un contexte d'accroissement des exigences relatives à la fiabilité et à la rentabilité des systèmes dans le but de réduire les impacts humains et économiques de leurs défaillances.

Dans ce contexte, la capacité nouvelle de gestion des flux de données modifie les fondements de la maintenance à travers l'émergence de nouvelles approches telles que le Prognostics and Health Management (PHM). Les travaux menés dans le cadre de cette thèse ont pour objet la mise en oeuvre d'une approche de maintenance prévisionnelle de type PHM permettant de réaliser la détec-

tion d'anomalies, le diagnostic et le pronostic de la durée de vie d'un système complexe dont l'observation et l'analyse sont soumises à des contraintes de volume, vitesse et variété des données manipulées.

Dans l'étude menée, la gestion de ces contraintes est assurée par la mise en place d'une architecture dirigée par les événements permettant de détecter les anomalies sur la base des données provenant du système surveillé. Cette architecture repose sur un système à base de règles chargé de l'analyse des anomalies pour fournir un diagnostic des pannes et un pronostic de la durée de vie restante. Cette approche a pour but de valoriser les connaissances expertes capitalisées tout en permettant l'intégration de modèles d'apprentissage artificiels entraînés.

Les résultats de cette étude ont été appliqués dans un cas d'étude portant sur l'identification d'équipements défaillants sur un sous-système d'aéronef.

MOTS-CLÉS : PHM, Modélisation, Système expert, EDA

---

## Abstract

### Implementation of a knowledge-based approach to estimate maintenance needs

The maintenance domain is currently facing challenges related to the advents of the digital landscape and data proliferation. Furthermore, the maintained systems' surging complexity entails increasing risks and failure diversity. This problematic comes about in a context of intensified requirements regarding safety and profitability aiming at reducing the failures' human and economic impacts.

In this context, the new data streams processing possibilities bring about fundamental changes to maintenance through the emergence of approaches like Prognostics and Health Management (PHM). The works presented in this thesis propose an implementation of a PHM-like predictive maintenance approach designed to perform anomaly detection, diagnostics and remaining useful life

(RUL) prognostics. This study is performed on complex systems for which data monitoring and processing come with high volume, velocity and variety constraints.

In this study, these constraints are dealt with through the implementation of an event driven architecture to detect anomalies among monitored data. It is supported by a rule-based system designed to analyze the detected anomalies and provide diagnostics and RUL prognostics. This approach aims at taking advantage of capitalized knowledge while managing the integration of trained machine learning models.

This approach was applied in a case study in the aeronautical field to identify failed equipment on aircrafts' subsystems.

KEYWORDS: PHM, Modeling, Expert System, EDA