

7/004.571



الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieur et de la recherche scientifique

Université de 8 Mai 1945 – Guelma -

Faculté des Mathématiques, d'Informatique et des Sciences de la Matière

Département d'Informatique



Mémoire de fin d'études de Master

Filière : Informatique

Option : Systèmes Informatiques (S.I)

Thème :

Une approche orientée données pour l'analyse
et la supervision des processus métiers

Encadré Par :

Dr. Khebizi Ali

Présenté par :

Douakha Imen

Juin 2018

Remerciement

Je tiens à exprimer ma reconnaissance à mon Directeur de mémoire Monsieur *KHEBZI* Ali. Je la remercie de m'avoir encadré, orienté, aidé et conseillé.

J'adresse mes sincères remerciements à mes professeurs intervenants et toutes les personnes qui par leurs paroles, leurs écrits, leurs conseils et leurs critiques ont guidé mes réflexions et ont accepté à me rencontrer et répondre à mes questions durant mes recherches.

Je remercie mes très chers parents, *Aziz* et *Aziza*, qui ont toujours été là pour moi, « Vous avez tout sacrifié pour vos enfants n'épargnant ni santé ni efforts. Vous m'avez donné un magnifique modèle de labeur et de persévérance. Je suis redevable d'une éducation dont je suis fier ».

Je remercie mes chers grand parents *Choussan* et *Fatma*, et tous mes oncles et tantes, mes frères *Khaled* et *Djamel*, et ma sœur *Rayane* pour leur encouragement.

Je remercie très spécialement mon oncle *Wissem* qui a toujours été là pour moi.

Je tiens à remercier *Houda*, *Sara*, *Bouchra*, *Kanane*, *Kadjer* et *Samiha* pour leur amitié, leur soutien inconditionnel et leur encouragement.

Enfin, je remercie tous mes collègues et ami(e)s que j'aime tant *Nouaim*, *Sara*, *Amira*, *Yassin*, *Soumia*, *Hamza*, *Kanane*, *Sara*, *touta*, *bichou*, *nour* ... Pour leur sincère amitié et confiance, et à qui je dois ma reconnaissance et mon attachement.

À tous ces intervenants, je présente mes remerciements, mon respect et ma gratitude.

Dédicace :

Je dédie ce mémoire à :

Mes parents :

Ma mère, qui a œuvré pour ma réussite, pour son amour, son soutien, tous les sacrifices consentis et ses précieux conseils.

Mon père, qui peut être fier et trouver ici le résultat de longues années de sacrifices et de privations pour m'aider à avancer dans la vie.

Mes frères et sœur, mes tantes et oncles qui n'ont cessé d'être pour moi des exemples de persévérance, de courage et de générosité.

Mes chers amis et collègues pour la qualité de leur travail et leur bonne humeur à toute épreuve.

Abréviations et acronymes

ACID	Atomicité Cohérence Isolation Durabilité
AFD	Automate Fini Déterministe
API	Application Programming Interface
ARIS	Academic Registration Information System
BAM	Business Activity Monitoring
BDA	Big Data Analytics
BDD	Base De Données
BP	Business Process
BPE	Business Process Engine
BPI	Business Process Intelligence
BPEL	Business Process Execution Language
BPMN	Business Process Modeling Notation
BPMS	Business Process Management System
BPR	Business Process Reengineering
CGI	Common Gateway Interface
DLL	Dynamic Link Library
DTD	Document Type Definition
ETL	Extract, Transform and Load
FSM	Finite State Machine
JDBC	Java Database Connectivity
JSON	JavaScript Object Notation
HTML	HyperText Markup Language
IDL	Interface Description Language
GUI	Graphic User Interface
MCD	Modèle Conceptuel de Données
MCT	Modèle Conceptuel de Traitement

NoSQL	Not Only SQL
OI	Office Information
OLAP	On-Line Analytical Processing
OLTP	On-Line Transaction Processing
OMG	Object Management Group
PAIS	Process Aware Information System
QoS	Quality of Service
RPC	Remote Procedure Call
SGBDR	Système de Gestion de Base de Données Relationnel
SLA	Service-Level Agreement
SOA	Service Oriented Architecture
SQL	Structured Query Language
URI	Uniform Ressource Identifier
URL	Uniform Ressource Locator
W3C	World Wide Web Consortium
WFM	Workflow Management
WfMS	Workflow Management System
XML	eXtensible Markup Language

Table des matières

Table des figures	iv
Liste des tableaux	v
1 Introduction générale	1
I Etat de l'art	3
2 Les processus métiers	4
2.1 Introduction	4
2.2 Définition des processus métiers	4
2.2.1 Types de processus métiers	5
2.3 La gestion des processus métiers	6
2.4 Les systèmes de gestion des processus métiers	6
2.5 Cycle de vie des processus métiers	6
2.5.1 Conception et analyse	7
2.5.2 Configuration	7
2.5.3 Déploiement	8
2.5.4 Évaluation	8
2.6 La modélisation des processus métiers	8
2.6.1 Les réseaux de Pétri RDP	9
2.6.2 Les automates finis déterministe A.F.D [1]	10
2.6.3 Unified Modeling Language UML	10
2.6.4 Merise : Le modèle conceptuel de traitement MCT	12
2.6.5 Business Process Model and Notation BPMN [2]	12
2.6.6 Business Process Execution Language BPEL	13
2.7 Conclusion	13
3 Les techniques d'analyse des processus métiers	14
3.1 Introduction	14
3.2 Définition et objectifs de l'analyse des processus métiers	14
3.3 Analyse au niveau modèle	15
3.3.1 Objectifs	15
3.3.2 Exemple d'analyse au niveau modèle	16
3.3.3 Les techniques d'analyse des modèles	16
3.4 Analyse au niveau données d'exécution	21
3.4.1 Objectifs	21
3.4.2 Exemple simple d'analyse au niveau données	21
3.4.3 Les techniques d'analyse des données des processus métiers	22
3.5 Conclusion	25

4	Problématique et travaux connexes	26
4.1	Introduction	26
4.2	Problématique	26
4.3	Étude des travaux de la recherche théorique	27
4.3.1	Analyse au niveau modèles	27
4.3.2	Analyse au niveau données	31
4.3.3	Analyse mixte (données/modèles)	33
4.4	Exploration des suites logicielles industrielles	36
4.4.1	IBM web sphère application server :	37
4.4.2	Suite Oracle	38
4.4.3	SAP NetWeaver BPM	38
4.4.4	Bonita BPM	39
4.5	Conclusion	40
II	Contribution	41
5	Approche d'Analyse et de Supervision des processus	42
5.1	Introduction	42
5.2	Choix du modèle des processus métiers	42
5.2.1	Spécification du modèle de processus métier	42
5.2.2	Motivations du choix de modèle	43
5.2.3	Exemple d'un processus métier illustré par les AFD	43
5.3	Concepts liés aux processus métiers	45
5.3.1	Instance d'exécution	45
5.3.2	Chemins d'exécutions	45
5.3.3	Trace et sous trace exécution	46
5.3.4	Expression régulière	49
5.3.5	Notion de sous protocole	49
5.3.6	Relation de simulation	51
5.4	Formalisation de l'approche d'analyse des données	51
5.4.1	Analyse des données au niveau modèle	52
5.4.2	Analyse des données d'exécution	55
5.5	Conclusion	65
6	Implémentation et expérimentation de l'approche	67
6.1	Introduction	67
6.2	Présentation de l'environnement de travail	67
6.2.1	Les APIs JAVA	67
6.2.2	Neo4j	67
6.2.3	GoJs	69
6.3	Architecture et fonctionnalités du système	70
6.3.1	Architecture du système	70
6.3.2	Fonctionnalités du système	71
6.3.3	Modélisation des structures de données	72
6.4	Scénario d'utilisation de Business Process Analyser	72
6.4.1	Gestionnaire des protocoles	73
6.4.2	Générateur de trace	75
6.4.3	Analyse des données	76

TABLE DES MATIÈRES

6.5 Conclusion	78
7 Conclusion générale	80
Bibliographie	81

Table des figures

2.1	Exemple simple d'un processus métier : Achat en ligne	5
2.2	Cycle de vie d'un processus métier [4]	7
2.3	Processus d'achat des produits modéliser par un RDP	9
2.4	Processus d'inscription modélisé par AFD	10
2.5	Processus d'achat des articles modélisé par un diagramme d'activités . . .	11
2.6	Processus de réservation d'hôtel modéliser par un diagramme de séquences	11
2.7	Exemple d'un processus métier modéliser par BPMN	12
3.1	Processus de remboursement des frais de déplacement modélisée en BPMN	16
3.2	processus traitement des demandes de compensation modélisé en RDP [5] .	18
3.3	Les trois principaux types de fouille de processus [5]	18
3.4	Un AFD représentant le processus d'achat d'un article	21
5.1	Processus d'inscription des étudiants en Master	44
5.2	Un exemple de sous-protocole	50
5.3	Exemple d'une relation de simulation entre deux protocoles	51
5.4	Compatibilité entre deux processus	54
6.1	Modèle de graphe de propriété étiqueté	68
6.2	Le navigateur Neo4j	69
6.3	Exemple de processus métier dessiner par GoJs	69
6.4	Architecture du système	70
6.5	Schéma de la base de données protocoles et traces	72
6.6	La page d'entrée au système	73
6.7	Formulaire de création de nouveau processus	73
6.8	Création et validation d'un nouveau processus	74
6.9	Visualisation et modification des protocoles	75
6.10	Génération des instances	75
6.11	Stockage des instances générées	76
6.12	Module d'analyse des données	77
6.13	Traitement d'un chemin complet	79

Liste des tableaux

5.1	Durée d'exécution de quelques activités des instances	47
5.2	Exemple d'instances avec leurs données descriptives	48
5.3	Un aperçu des KPI proposés	62

CHAPITRE 1

Introduction générale

Le développement d'applications informatiques qui répondent de manière efficace aux besoins variants et grandissants des utilisateurs a constitué à travers l'histoire de l'informatique un défi permanent. L'ingénierie des logiciels (software engineering) a été confrontée à plusieurs challenges pour prendre en charge de tels préoccupations. L'aboutissement du processus d'évolution historique a convergé vers une prise de conscience de l'intérêt des processus métiers et de leur positionnement au noyau du système d'informations de l'entreprise. Ainsi, il est observé durant la dernière décennie que la technologie BPM ¹ trouve beaucoup d'engouement de la part des différents acteurs de la sphère économique et elle constitue un facteur clé de réussite et de pérennité des organisations modernes.

L'émergence de la technologie BPM a favorisé le déploiement, à grande échelle, des processus métiers des entreprises. Néanmoins, le problème de l'analyse et de la supervision des processus déployés est l'une des préoccupations majeures des organisations. En effet, malgré la large adoption de cette technologie et la prolifération des environnements supportant les différentes phases du cycle de vie des processus métiers, le besoin de la prise en charge et de l'analyse des données générées par l'exécution des processus métiers se fait de plus en plus sentir. Dans cette perspective, l'exploitation des techniques d'intelligence d'affaires (business intelligence) pour l'amélioration des processus déjà déployés est l'un des soucis majeurs des fournisseurs de services. Ainsi, l'identification et la spécification des solutions aux problèmes des processus métiers déployés exigent une analyse des données stockées par les entreprises sous forme de fichiers logs d'exécution. En effet, les systèmes de gestion des processus métiers (BPMSs) enregistrent une masse importante d'informations reflétant les événements qui surgissent lors de l'exécution des processus métiers par les différentes instances ; telles que : la durée de chaque activité, les données en entrées et en sortie, les ressources déployées ainsi que les erreurs d'exécution. L'analyse et l'exploitation de ces données est porteuse de plus value pour les gestionnaires des processus et pour toute l'organisation. Par l'extraction de connaissances utiles et pertinentes ; telles que : les prédictions des exécutions, les tendances des clients, le temps moyen des procédures, . . . , les organisations peuvent améliorer la qualité de leurs processus et de leurs services fournis aux partenaires. Bien que beaucoup de travaux dans le domaine des processus métiers se sont concentrés essentiellement sur la modélisation et l'automatisation des processus, peu de travaux ont abordé les questions liées à l'analyse, à l'optimisation, au contrôle et la prédiction des processus métiers.

Ce projet de fin d'études de master visent à répondre à ces préoccupations. Nous proposons une approche qui permet d'explorer les données stockées dans les bases de données (structurées et semi-structurées) des entreprises, en vue d'extraire un ensemble d'indicateurs clés de performances ; e.g : quelles sont les sous-procédures en cours d'exécution ? Leurs statuts ? Combien de clients sont en cours d'exécuter un processus particulier ? Pour un client donné, quelles sont les activités achevées, celles qui sont en cours (commandes, factures, paiements) ? Quels sont les processus surchargés (nombre d'instances en cours, . . . etc). Pour répondre à ces exigences, nous formalisons une approche qui permet au

1. Business Process Management

tableau de bord offrant une vision globale du système de gestion des processus métiers, tout en exposant des états de synthèse extraits des données des processus métiers achevés ou en cours d'exécution. L'approche sera mise en œuvre via un prototype logiciel qui servira d'outil d'aide à la prise de décision pour le gestionnaire de protocoles.

En plus de cette introduction générale, le mémoire est structuré en deux grandes parties.

- **La première partie** est un état de l'art du domaine des processus métiers. Elle est constituée de trois chapitres. Le chapitre 2 est consacré aux processus métiers. Il aborde leur définition et leur formalisation, et les systèmes de gestion des processus métiers aussi bien que leur cycle de vie. Le chapitre 3 est dédié aux différentes techniques d'analyse des données afférentes aux processus métiers. Les différentes techniques existantes sont exposées et les méthodes et outils sont présentées. Dans le chapitre 4, une analyse des travaux connexes liés à l'analyse des données des processus métiers est exposée et différents domaines sont étudiés. A ce niveau, nous mettons en exergue la problématique abordée dans ce mémoire par la mise en évidence du déficit constaté après étude de l'état de l'art.
- **La deuxième partie** du mémoire constitue notre contribution. Elle est divisée en deux chapitres. Le chapitre 5 explicite notre approche pour l'analyse et la supervision des processus métiers. Nous y introduisons les différentes facettes de l'approche proposée, l'architecture du système ainsi que ses fonctionnalités. Le chapitre 6 est consacré à la réalisation et l'implémentation de l'approche. Nous exposons dans ce chapitre un prototype logiciel qui réalise les divers aspects introduits dans la phase de conception et nous illustrons son utilisation à travers différents scénarios.

Première partie

Etat de l'art

2.1 Introduction

les évolutions technologiques contemporaines ont complètement bouleversé les modes de fonctionnement des organisations modernes. En effet, la jonction entre les nouveaux mode de communication, induit par l'apparition d'Internet, avec les avancées spectaculaires réalisées dans le domaine des technologies de l'information (IT)¹ ont radicalement booster les organisations modernes vers de nouveaux horizons et de nouveaux degrés de performances. En se sens, les systèmes d'information modernes ont complètement dématérialisé la gestion des entreprises. Le constat actuel fait ressort le fait que les processus métiers sont au cœur des systèmes d'information modernes.

Brièvement, un processus métier exprime les activités réalisées par une organisation en collaboration avec ses différents partenaires, en vue de réaliser des objectifs de gestion. Avec l'avènement de l'Internet et l'accroissement de la connectivité, les processus métiers d'aujourd'hui sont devenus complexes et traversent les frontières de l'organisation. Par ailleurs, leur maîtrise constituent un critère de performance et de compétitivité de toute organisation qui évolue dans un contexte économique caractérisé par les fortes coalitions, les rachats et les fusions d'entreprises. Pour concrétiser cette vision stratégique basée sur les processus métiers, la discipline BPM a émergé durant la dernière décennie en tant que technologie de pointe pour la prise en charge du cycle de vie des processus métiers.

Dans ce premier chapitre, on aborde la définition des processus métiers, leur gestion, leur système de gestion aussi bien que leur cycle de vie. En fin de chapitre, leur modélisation avec des formalisme est explicitée et leur fonctionnement est illustré.

2.2 Définition des processus métiers

Un processus métier *Business process BP* est un enchaînement d'actions et d'activités exécutées en collaboration entre des acteurs dans un environnement technique et organisationnel pour fournir un service ou un produit à des clients.[6]

Chaque processus métier est déployé par une seule organisation, mais il peut interagir avec d'autres processus métiers appartenant à d'autres partenaires.

A noter que les activités d'un processus métier peuvent être des actions atomiques prises en charge par des applications, telles que les services web, comme elles peuvent être accomplies par des humains.[7]

Exemple 1 *Processus métier Achat*

La figure 1, ci-dessous illustre les différentes activités à exécuter lorsqu'un fournisseur vend un produit à un client.

Les données véhiculées par ce processus métiers portent sur plusieurs dimensions, à savoir :

1. IT : Information Technology

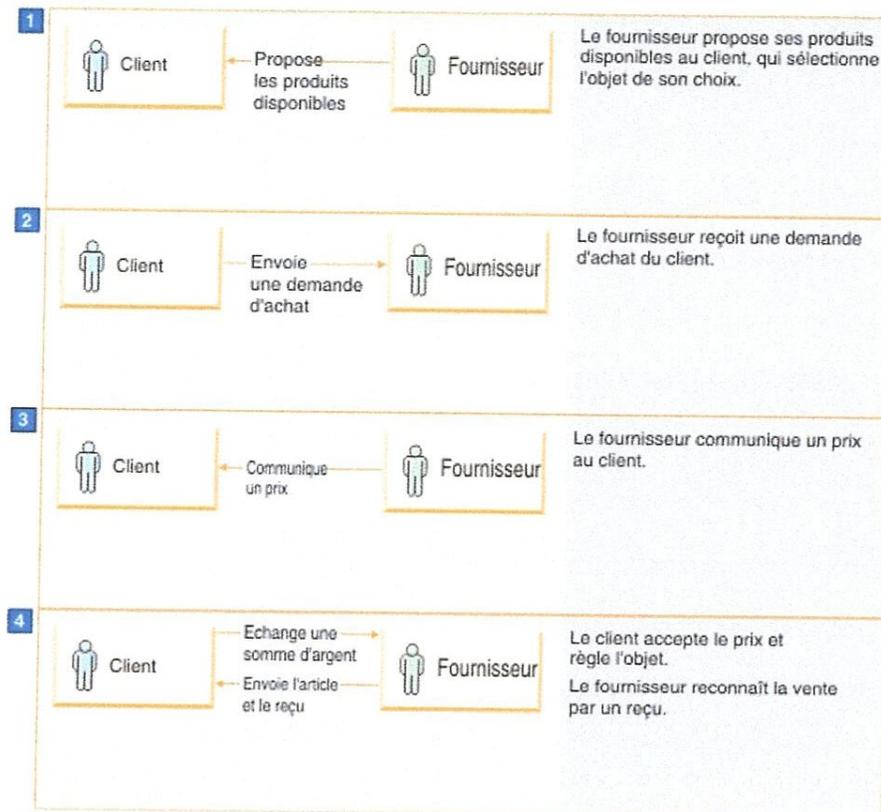


FIGURE 2.1 Exemple simple d'un processus métier : Achat en ligne

- Les acteurs : Client et Fournisseur
- Les activités : "Propose les produits disponibles" , "Envoie une demande d'achat"
- Les contraintes associées au processus métier : condition de vente, délai, ordre des activités.

Cet exemple simple (Fig.2.1) contient tous les concepts de base d'un processus métier. Dans la réalité, les processus métiers ne sont pas aussi simple que dans l'exemple précédent. En effet, afin de prendre en charge les règles de gestion, les différentes procédures et la logique métier des organisations, les processus métiers sont de plus en plus complexes.[7]

Pour mettre en exergue l'importance du concept de processus métier, nous faisons un survol des différents domaines phares qui manipulent des processus métiers de manière intensive. La sous section suivante 2.2.1 illustre des processus métiers relatifs à différents domaines d'activités.

2.2.1 Exemples de processus métiers

Les processus métier couvrent aussi bien les administrations que l'industrie, et peuvent inclure divers opérations de gestion. Nous distinguons, essentiellement :

- Fabrication : un processus d'assemblage de produits, un processus d'assurance de la qualité, un processus de maintenance corrective / préventive.
- Finance : un processus de facturation, un processus de gestion des risques, processus de gestion comptable.

- Santé : processus pour l'évaluation médicale, processus d'admission et de suivi des malades.
- Banque : processus de vérification des crédits, processus de gestion des comptes.
- Voyage : processus de réservation d'un voyage, processus de gestion des vols.
- Système de contrôle : processus de contrôle des installations, processus de gestion des urgences.
- Secteur public : demande de service gouvernemental (passeport, permis de conduire, dossier de retraite, visa ...). [8]

2.3 La gestion des processus métiers

La gestion des processus métier comprend les concepts, méthodes et techniques pour assister les utilisateurs à la conception, l'administration, la configuration, le déploiement et l'analyse des processus métier. On parle d'approche de gestion centrée processus métier ou **BPM** (Business Process Management).

Le but de l'approche BPM est la représentation explicite des processus métiers avec leurs activités et les contraintes d'exécution les concernant. Une fois les processus métiers sont définis, ils seront prêts pour le déploiement, la configuration, les analyses et les améliorations. [3]

2.4 Les systèmes de gestion des processus métiers

Traditionnellement, les processus métier sont mis en œuvre manuellement, guidés par les connaissances du personnel de l'entreprise et assistés par des règlements organisationnels et des procédures pré-établies. Néanmoins, les entreprises peuvent obtenir des avantages supplémentaires s'ils utilisent des systèmes logiciels pour coordonner les activités impliquées dans les processus métier. Ces systèmes sont appelés les systèmes de gestion des processus métier, ou Business Process Management System BPMS.

Un système de gestion des processus métier (ou BPMS simplement) est un système logiciel générique qui est piloté par des représentations explicites pour coordonner la mise en œuvre du processus métier. [3]

Aujourd'hui, les entreprises modernes exploitent les environnements BPMS pour modéliser, configurer, déployer et superviser leurs processus métiers.

A titre d'exemple l'environnement IBM web sphere ² [9] est utilisé par plusieurs compagnies et banques pour prendre en charge leur processus métier. Par exemple, le crédit lyonnais l'utilise pour gérer les comptes et les crédits de ses clients.

2.5 Cycle de vie des processus métiers

Comme le montre la figure 2.2 [3], le cycle de vie d'un processus métier est composé de phases liées entre elles. Ces phases sont organisées en une structure cyclique, montrant leurs dépendances logiques. Ces dépendances n'impliquant pas un ordonnancement temporel strict dans lequel les phases doivent être exécutées. Le cycle de vie d'un processus métier peut généralement être décomposé des étapes suivantes :

2. IBM WebSphere est un outil qui permet aux entreprises de créer et de gérer des sites Web avancés et sophistiqués à l'aide d'un ensemble d'outils IBM Java.

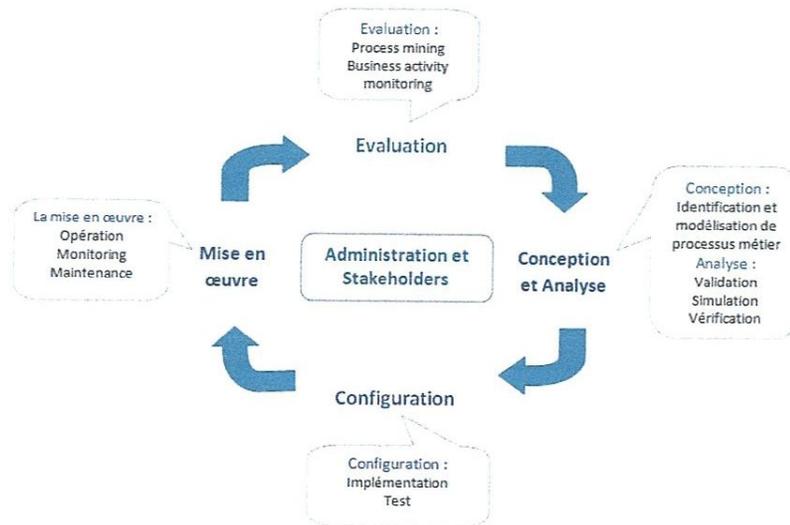


FIGURE 2.2 – Cycle de vie d'un processus métier [3]

2.5.1 Conception et analyse

Le cycle de vie de n'importe quel processus métier commence toujours par cette phase, en se basant sur des enquêtes menées sur les processus métiers et leur environnement organisationnel et technique, ces processus seront identifiés, revus, validés et représentés par des modèles.

Les techniques de modélisation des processus métier ainsi que les techniques de validation, simulation et vérification sont utilisées durant cette phase. La modélisation des processus métier est au cœur de la sous-phase technique lors de la conception du processus. Différents modèles de représentation des processus métiers sont utilisés (voir section 2.6).

Une fois que la conception d'un processus métier est terminée, on passe directement à l'étape de validation, en utilisant un atelier de travail 'Workshop', pour vérifier si toutes les instances des processus métier validées sont reflétées par le modèle.

Les techniques de la simulation peuvent être utilisées pour soutenir l'étape de validation, car certaines séquences d'exécution non désirées peuvent être simulées pour montrer des déficits dans le modèle. Il permet également aux partenaires de parcourir le processus étape par étape pour vérifier si le processus expose réellement le comportement souhaité. Au cas contraire, des opérations de redressement sont à opérer.

Ainsi, la modélisation des processus métier a un caractère évolutif dans le sens que le modèle est analysé et amélioré afin de représenter le processus métier désiré, sans aucune lacune. [3]

2.5.2 Configuration

Après avoir terminé l'étape de conception et d'analyse, le processus métier sera prêt pour être implémenté. Si on veut utiliser un système logiciel pour l'implémentation, ce dernier doit être configuré selon l'environnement organisationnel de l'entreprise et le processus métier dont il devrait contrôler le fonctionnement. Cette configuration comprend les interactions des employés avec le système ainsi que l'intégration des systèmes logiciels

existants avec le système de gestion des processus métier. [3]

2.5.3 Déploiement

La phase de mise en œuvre ou déploiement du processus englobe l'exécution réel du processus métier par des instances lancées par des clients. Chaque instance remplit les objectifs métier d'une entreprise et l'initiation d'une instance de processus suit généralement un événement déclencheur, par exemple, la réception d'une commande envoyée par un client.

Le système de gestion des processus métier contrôle activement l'exécution des instances des processus métiers définies dans le modèle, la mise en œuvre du processus doit répondre à une coordination de processus correcte, garantissant que les activités de processus sont exécutées en fonction de l'exécution des contraintes spécifiées dans le modèle de processus.

Durant la mise en œuvre du processus métier, les données d'exécution utiles sont collectées, généralement dans une certaine forme de fichiers journaux. constituent de séries ordonnées du journal d'entrées, indiquant les événements qui se sont produits au cours des processus métier.

Les journaux d'événement constituent la base de l'évaluation des processus dans la phase suivante du cycle de vie des processus métier. C'est à dire la phase de supervision et d'évaluation. [3]

2.5.4 Évaluation

La phase d'évaluation utilise les informations disponibles pour évaluer et améliorer les modèles de processus métiers et leurs implémentations, les journaux d'exécution sont analysés en utilisant un module spécifique de surveillance des activités BAM ³ et les techniques d'extraction de processus (Process mining techniques), ces techniques visent à identifier la qualité des processus métier et à l'adéquation de l'environnement d'exécution.[3]

2.6 La modélisation des processus métiers

Représenter un processus se fait par la réalisation d'un modèle qui répond à toutes ses exigences et aux besoins.

Différents modèles, dotés de niveaux d'expressivité variés, ont été proposés dans la littérature pour représenter les processus métier. Parmi ces modèles on distingue :

- Les réseaux de Pétri.
- Les automates finis déterministe A.F.D.
- Diagramme UML (diagramme d'activité, diagramme de séquence).
- Diagramme Merise (modèle conceptuel de traitement MCT).
- Business process model and notation BPMN.
- Business process execution language BPEL.

Ces modèles permettent de prendre en charge différents aspects et abstractions associés au processus métier. Dans ce qui suit, on aborde d'une manière explicite chaque modèle.

3. Business activity monitoring

2.6.1 Les réseaux de Pétri RDP

Un réseau de Pétri (ou Petri Nets) est un langage orienté graphique pour la conception, la spécification, la simulation et la vérification des systèmes. Ils sont particulièrement bien adaptés aux systèmes collaboratifs qui exigent des interactions de communication et de synchronisation.

Les réseaux historiquement ont été utilisés pour représenter les systèmes temps réels et les systèmes critiques (contrôle aérien, centrales nucléaires, ...). Vue sa richesse ce modèle est largement utilisé pour représenter les processus métiers. [10]

Définition formelle d'un RDP [11]

Un réseau de Pétri est un quadruplet (P, T, F, w) , ou :

- P est un ensemble fini de places,
- T est un ensemble fini de transitions,
- F est un ensemble fini d'arcs pondérés connectant une place a une transition et vice versa,
- w est une fonction qui associe a chaque arc un poids sous forme d'un entier strictement positif, $w : F \rightarrow \mathbb{N}^+$, les poids sont égaux a 1 par défaut.

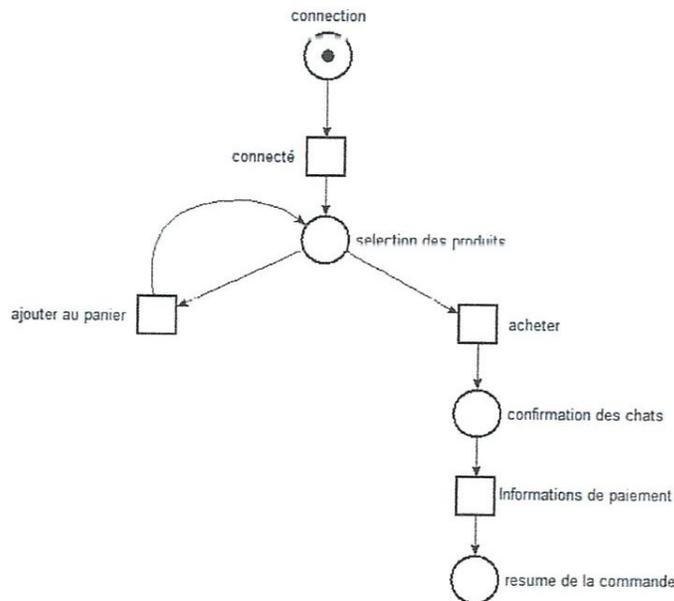


FIGURE 2.3 – Processus d'achat des produits modélisé par un RDP

2.6.2 Les automates finis déterministe A.F.D [1]

Un processus peut être considéré comme une succession de phases suite aux déclenchement des activités. De cette façon, les automates peuvent répondre parfaitement au souci de représentation des processus métiers.

Un AFD⁴ représente un processus métier, ou les états de l'automate sont les différents phases du processus métier et les transitions sont les activités.

On utilise, particulièrement les AFDs, car a un instant donné, ayant atteint un état du processus métier, l'utilisateur doit être **précis** sur les futures actions a exécuter. Par conséquent l'automate doit être déterministe.

La figure (Fig.2.6.2) ci-dessous illustre un exemple du processus métier de gestion des réinscriptions des étudiants dans une établissement universitaire.

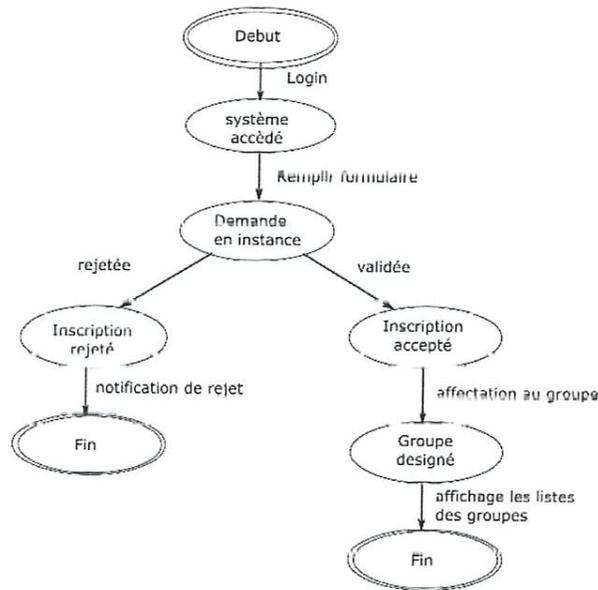


FIGURE 2.4 – Processus d’inscription modélisé par AFD [2]

2.6.3 Unified Modeling Language UML

Parmi les diagrammes offerts par la méthode UML, certain sont bien adaptés a la représentation des processus métiers et les professionnels les ont adapté en tant que technique puissante de modélisation. [19]

On distingue, notamment les deux diagrammes : le diagramme d’activités et le diagramme de séquences, qu’on va exposer dans ce qui suit.

Diagramme d’activité :

Un diagramme d’activité fournit une vue du comportement d’un système en décrivant la séquence d’actions d’un processus. Ils montrent les flux entre les actions d’une procédure. Les diagrammes d’activités peuvent, cependant, aussi montrer les flux parallèles simultanés et les flux de remplacement.

4. ou en anglais : FSM, Finite State Machine

Dans les diagrammes d'activités, on utilise des nœuds d'activité et des nœuds de contrôle pour modéliser le flux d'action et des contraintes entre les actions.[20]

Les diagrammes d'activités sont utilisés pour modéliser les processus métier, ou la partie dynamique d'un modèle. Ils peuvent être "autonomes" et représentent un processus unique, comme ils peuvent représenter plusieurs processus en interaction.[21]

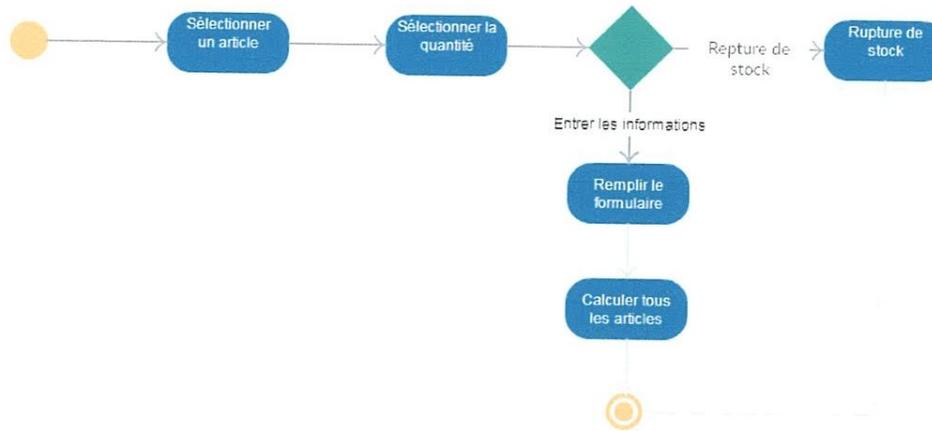


FIGURE 2.5 – Processus d'achat des articles modélisé par un diagramme d'activités

Diagramme de séquence :

Les diagrammes de séquences permettent de décrire **COMMENT** les éléments du système interagissent entre eux et avec les acteurs concernés.

Dans un contexte opérationnel un système est défini par ces objets et ces acteurs, ou :

- Les objets sont au cœur d'un système interagissant en s'échangeant des messages.
- Les acteurs interagissent avec le système au moyen d'IHM (Interfaces Homme-Machine).

Un diagramme de séquence est un diagramme d'interaction qui expose en détail la façon dont les opérations sont effectuées : quels messages sont envoyés et quand ils le sont. Les diagrammes de séquence sont organisés en fonction du temps. Les objets impliqués dans l'opération sont répertoriés de gauche.[22]

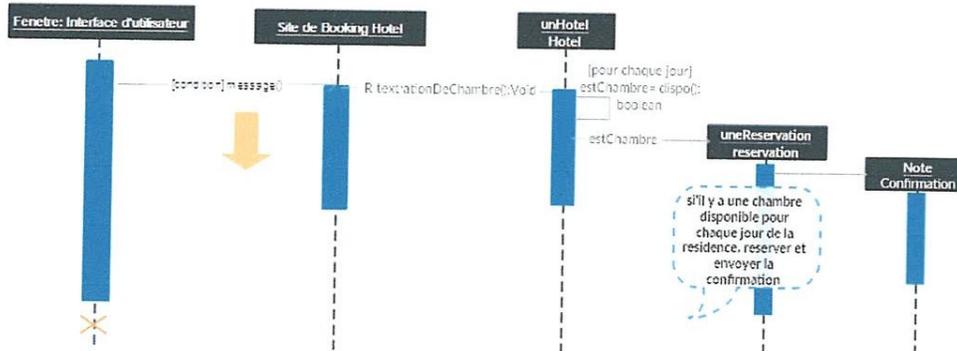


FIGURE 2.6 – Processus de réservation d'hôtel modéliser par un diagramme de séquences

2.6.4 Merise : Le modèle conceptuel de traitement MCT

Le modèle conceptuel des traitements permet de traiter la dynamique du système d'information, c'est-à-dire les opérations qui sont réalisées en fonction d'événements. Il s'agit de représenter, par un formalisme précis et en grande partie standardisé, l'ensemble des traitements que l'on doit réaliser pour répondre aux attentes du projet défini en amont de l'analyse.[23]

Ce modèle permet donc de représenter de façon schématique l'activité d'un système d'information sans faire référence à des choix organisationnels ou des moyens d'exécution, c'est-à-dire qu'il permet de définir simplement ce qui doit être fait, mais il ne dit pas quand, comment ni où.[24]

2.6.5 Business Process Model and Notation BPMN [2]

BPMN est une représentation graphique des processus métier. Il s'agit d'un ensemble d'objets graphiques et de règles définissant les connexions disponibles entre les objets.

Orienté activité et en partie inspiré d'UML. BPMN est constitué des blocs de construction de base suivants :

- Objets de flux : événements (cercles), activités (rectangles avec coins arrondis) et passerelles (losanges)
- Objets de liaison : composés principalement de flèches, elles indiquent le flux de la séquence (flèches pleines), le flux des messages (flèches pointillées) et les associations
- Voies de liaison : piscines (contour graphique) et couleurs (sous-partition de la piscine)
- Artefacts : objets de données, groupes et annotations

L'avantage de BPMN est qu'il s'agit d'un standard avec une syntaxe bien définie. En plus que la majorité des analystes d'entreprise l'utilise, ce qui facilite la collaboration. En outre, la plupart des outils de modélisation le prennent en charge, ce qui le rend beaucoup plus facile à partager et à modifier même si vous utilisez des logiciels différents.

La figure 2.7 ci-dessous montre les différents blocs de construction de BPMN.

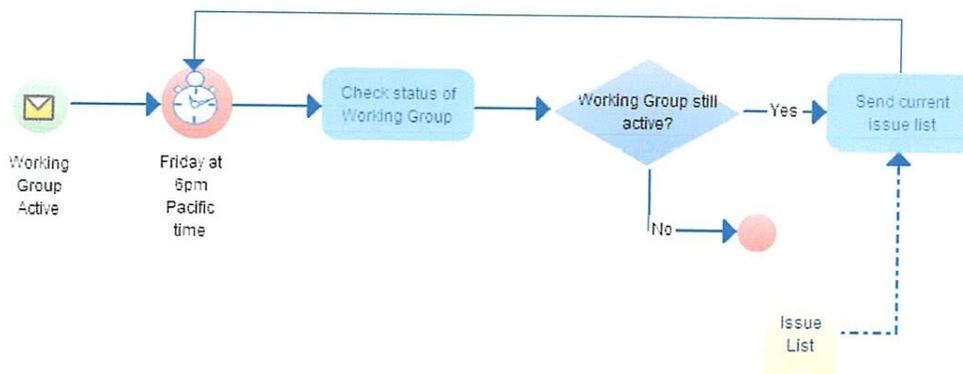


FIGURE 2.7 – Exemple d'un processus métier modéliser par BPMN

2.6.6 Business Process Execution Language BPEL

Le langage **BPEL** est un standard de l'industrie pour l'automatisation des processus métiers basés sur les services Web.

La fonctionnalité des composants logiciels peut être fournie de manière standardisée en tant que composant basé sur des services Web. Dans les architectures orientées service (SOA)⁵. Ces services sont connectés à des systèmes plus importants. Avec BPEL, les services offrant une interface de service web peuvent être combinés en un processus métier automatisé. Le processus métier automatisé est à son tour fourni en tant que service et peut donc être intégré dans d'autres processus ou services.

Contrairement à BPMN, BPEL ne définit pas une représentation graphique des processus. Les processus BPEL sont décrits en XML (Extensible Markup Language).[25]

2.7 Conclusion

Dans ce chapitre, nous avons présenté un survol du domaine des processus métier. Les concepts qui lui sont associés ont été présentés, expliqués et illustrés par des exemples. Par ailleurs, les environnements logiciels de gestion des processus métiers ont été exposés et expliqués. Nous avons terminé ce chapitre par l'exposé des différents modèles permettant la présentation des processus métiers.

Dans le prochain chapitre on va aborder les techniques d'analyse des données des processus métiers, en identifiant les diverses dimensions d'analyse et en abordant les domaines qui traitent de cette problématique.

5. Service Oriented Computing : terme suggéré par le Gartner Group dès 1996, propose de définir les échanges en terme de services.

Les techniques d'analyse des processus métiers

3.1 Introduction

La masse de données générées lors de l'exécution des processus métiers constitue un capital fondamental de toute organisation désirant assurer sa pérennité dans un environnement de plus en plus concurrentiel. Par ailleurs, la mine d'information existante est une ressource porteuse de valeur ajoutée pour toute l'entreprise. En effet, l'exploitation des données existantes, en vue d'extraire des informations utiles constitue une démarche stratégique visant à atteindre les objectifs suivants :

- améliorer les processus métiers existants, par leur enrichissement et leur évolution afin de prendre en compte les nouveaux besoins de l'environnement.
- prise en compte des contraintes liées à l'exécution des processus métiers, tels que les ressources disponibles, le temps nécessaire à l'exécution des tâches ainsi qu'à leur chronologie ;
- extraire des informations pertinentes, en vue d'assister les gestionnaires dans la prise de décision (*taux, moyenne, coût . . .*) ;
- spécification et calcul d'indicateurs de performances KPI¹ ;
- les données sont à la base de l'élaboration d'un tableau de bord pour la supervision du fonctionnement de toute l'organisation.

dans cette perspective, des techniques, des méthodes et des outils sont plus que nécessaires pour répondre à ces objectifs. Elles se situent dans un contexte globale d'analyse des processus métiers. On distingue, généralement deux niveaux d'analyse : analyse au niveau modèles et analyse au niveau des données. Nous expliquons, dans ce qui suit, la démarche d'analyse des données des processus métiers avec ses deux types d'analyse.

3.2 Définition et objectifs de l'analyse des processus métiers

L'analyse des processus métiers est une activité critique pour toute organisation. Son objectif est de permettre, d'une part l'amélioration continue des processus métiers déjà déployés par la prise en compte des facteurs qui surgissent dans son environnement, tels que les nouvelles lois et réglementations et les nouveaux besoins des utilisateurs. D'autre part elle permet d'opérer des actions de redressement et de maintenance sur les processus métiers qui manifestent des erreurs, des anomalies ou des surcharges liées à leur exploitation.

La phase d'analyse est la deuxième étape du cycle de vie des processus métiers, comme nous l'avons déjà vu dans le chapitre précédent (section 2.5). Cette phase aide à améliorer

1. Key performance indicators.

l'efficacité et l'efficience du processus, afin de s'assurer que le processus métier converge à atteindre son objectif final qui lui a été déjà spécifié dans la phase de modélisation.

L'analyse des processus métiers aide aussi à identifier et à examiner chaque partie de l'organisation, y compris les ressources exigées par le processus lui-même, les parties participantes, l'échange d'informations et autres ressources organisationnelles. Par exemple, le matériel nécessaire à l'exécution d'une tâche ou le rôle affecté à un intervenant dans la procédure. Ainsi, il apparaît que l'analyse des processus métiers est utile dans tout contexte opérationnel pour les raisons suivantes :

- Trouver les raisons derrière les retards systématiques des procédures ;
- Dégager la "bonne" et la meilleure façon de réaliser les processus ;
- Déterminer si le processus fonctionne ou non à la capacité potentielle maximale ;
- Déterminer si le processus doit être amélioré ou reconfiguré, afin de s'adapter avec les besoins de l'environnement[26].

Vue son importance, l'analyse des processus métiers a été largement étudiée par plusieurs communautés de chercheurs de différentes disciplines, particulièrement, les organisateurs, les informaticiens, les spécialistes en recherche opérationnelle ainsi que les concepteurs des systèmes d'informations. Les aspects traités portent sur plusieurs dimensions, à savoir : la validation, la simulation, la vérification et l'analyse des performances.

Dans ce projet de fin d'études, nous adoptons la perspective dynamique de la vérification et de l'analyse des performances d'une organisation en se basant sur les données véhiculées par l'exécution des processus métiers. C'est à dire, que nous effectuons notre analyse sur les modèles établis des processus métiers, aussi bien que sur les données rassemblées dans les fichiers log associées à l'exécution des processus métiers par les instances.

Ces deux dimensions sont largement discutées dans la suite de ce chapitre.

3.3 Analyse au niveau modèle

La vérification d'un modèle de processus métier consiste à analyser si le comportement de ses instances correspond ou non à un comportement attendu est spécifié préalablement.

3.3.1 Objectifs

Le comportement attendu du processus est spécifié partiellement par un modèle, ou schéma de processus métier. À rappeler qu'il existe plusieurs modèles de représentation (voir section 2.6). Cependant, le modèle de processus fournit uniquement des informations statiques et structurelles sur le processus en question. D'où apparaît l'importance de l'analyse des données générées lors de son exécution. À ce niveau, justement, la partie analyse de données peut entrer en jeu afin de remédier aux limites causées par les spécifications incluses dans les modèles.

Par exemple, dans le scénario décrit dans l'exemple 2, l'analyste peut vérifier si, dans des conditions d'exécution différentes, la somme des montants du remboursement traités en utilisant l'option "*fast track*" est inférieure à 15000 euros par trimestre de l'année. Pour effectuer cette vérification, l'analyste doit pouvoir : spécifier le comportement attendu du processus métier, ensuite fournir les entrées pour le processus, l'exécuter et suivre son comportement observé, et enfin analyser le comportement attendu afin de vérifier s'ils correspondent aux normes imposées lors de la définition du processus. La réalisation conjointe de ces tâches est appelée **la vérification** des processus métier et **l'analyse de ces performances**. [27]

3.3.2 Exemple d'analyse au niveau modèle

Pour illustrer l'importance de l'analyse des processus métiers, nous exposons ci-dessous un exemple d'analyse inspiré du domaine de remboursement des frais de voyage ;

Exemple 2 *Considérons, le processus de remboursement des frais de voyage Figure 3.2 [27]. En outre, supposons que le processus est actuellement utilisé dans un système basé sur le service, et que certains problèmes ont été identifiés par l'analyste de la société. Plus concrètement, il a remarqué qu'avec les ressources actuelles affectées à l'exploitation de ce processus, seulement 70% de toutes les demandes de remboursement sont traitées à temps. L'analyste aimerait changer le processus afin d'améliorer sa performance sans avoir besoin d'augmenter la quantité de ressources affectées au processus. De plus, il a également remarqué que le montant de nombreuses demandes de remboursement est très inférieur aux coûts opérationnels d'exécution du processus pour traiter la demande et que, dans ce cas, il peut être préférable de rembourser immédiatement l'employé sans avoir à exécuter tout le processus et engager des coûts qui ne sont pas justifiés par le montant demandé.*

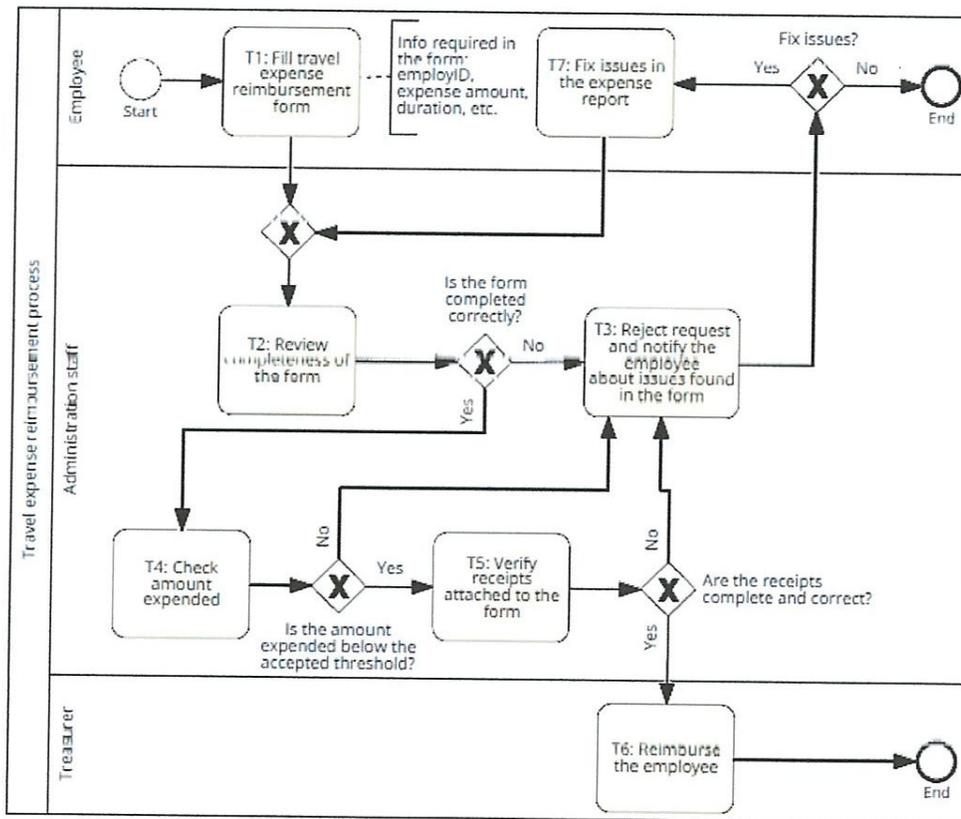


FIGURE 3.1 – Processus de remboursement des frais de déplacement modélisée en BPMN (d'après [27])

3.3.3 Les techniques d'analyse des modèles

Afin de pouvoir réaliser des analyses rentables et efficaces sur les schémas des processus métiers, plusieurs techniques ont été proposées. Elle émanent de disciplines diversifiées.

Parmi ces techniques on distingue, notamment :

- Le process mining : les techniques de fouille des processus métiers.
- Les techniques de prédiction.
- les arbre de décision concernant les processus métiers.
- Les approches de Pattern matching : recherche des motifs récurrents.

Un bref aperçu de techniques précédentes est présenté, dans ce qui suit :

Les techniques de fouille des processus métiers

Les techniques de fouille de processus² permettent d'extraire des connaissances à partir des journaux d'événements disponibles dans les systèmes d'information actuels. au fait, les informations enregistrées dans les fichiers logs expriment certains comportement liés au fonctionnement du processus métiers, et que le gestionnaire pourra récupérer en analysant les relations entre ces données.

Ces techniques fournissent de nouveaux moyens pour découvrir, surveiller et améliorer les processus dans divers domaines d'application [28]. L'intérêt croissant pour l'extraction de processus repose sur deux facteurs principaux :

- De plus en plus d'événements sont enregistrés dans les journaux logs, fournissant ainsi des informations détaillées sur l'historique des exécutions des processus métiers.
- Il est nécessaire d'améliorer et d'adapter les processus métiers fonctionnant dans des environnements en évolution rapide.

les techniques de fouille de processus fournissent une jonction importante entre les approches d'intelligence d'affaire³ et la technologie BPM, et entre les techniques de Data Mining et de Workflow, d'autre part [29]. En effet, les techniques de fouille de processus intègre la découverte de processus, la vérification de conformité (c'est à dire la surveillance des écarts en comparant le modèle et le journal), l'exploration de réseaux sociaux et organisationnels, la construction automatisée de modèles de simulation, l'extension de modèle, la prédiction de cas, et les recommandations basées sur les historiques des exécutions, . . .

Cependant, il faut d'ores et déjà signalé que la plupart des systèmes d'information stockent les données d'exécution sous une forme non structurée. Par exemple, les données d'événement sont dispersées sur de nombreuses tables ou doivent être extraites de sous-systèmes échangeant des messages. Dans de tels cas, les données d'événement existent mais certains efforts sont nécessaires pour les extraire. L'extraction de données fait partie intégrante de tout effort de fouille de processus [5].

Supposons qu'il est possible d'enregistrer séquentiellement des événements, tels que chaque événement se réfère à une activité (*c.à.d.*, *une étape bien définie dans le processus*) et est lié à une instance particulière (*c.à.d.*, *un cas d'exécution du processus*). Dans un tel contexte, il est possible d'opérer une analyse au niveau modèle, comme montré dans l'exemple suivant.

Exemple 3 *Considérons, par exemple, le traitement des demandes de compensation modélisées par la figure 3.2 ci-dessous. Les instances du processus sont des demandes individuelles et chaque instance est caractérisée par une trace d'événements qui est enregistrée dans les base de données adéquates.*

Un exemple d'une trace possible est "1. demande d'inscription, 2. examiner complètement, 3. vérifier le billet, 4. décider, 5. relancer la demande, 6. vérifier le billet, 7.

2. Process mining

3. Business intelligence

examiner complet, 8. décider, 9. payer une compensation". Ici, les noms d'activité sont utilisés pour identifier les événements. Cependant, il y a deux événements de décision qui se sont produits à des moments différents (le quatrième et le huitième événement de la trace), ont produit des résultats différents et ont pu être menés par des personnes différentes. Évidemment, il est important de distinguer ces deux décisions. Par conséquent, la plupart des journaux d'événements stockent des informations supplémentaires sur les événements. En fait, chaque fois que possible, les techniques d'extraction de processus utilisent des informations supplémentaires, telles que la ressource (humaine ou matérielle) exécutant ou initiant l'activité, et aussi l'horodatage de l'événement ou les éléments de données enregistrés avec l'événement (par exemple, la taille d'une commande).[5]

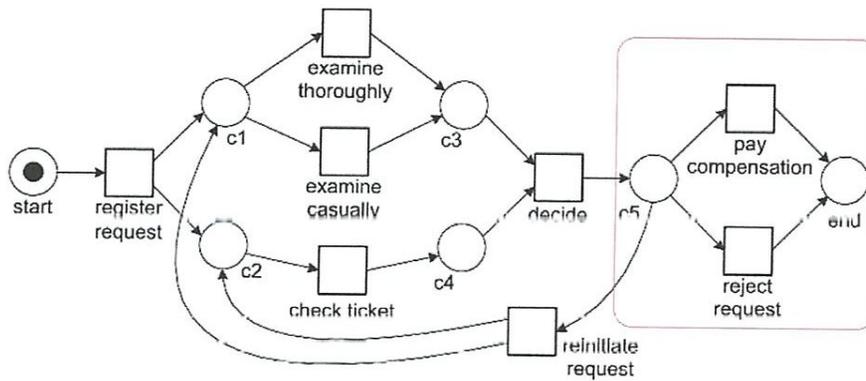


FIGURE 3.2 – processus traitement des demandes de compensation modélisé en RDP [5]

Les journaux d'événements peuvent être utilisés pour effectuer trois types d'exploration de processus [5], comme indiqué dans la figure 3.3 ci-dessous.

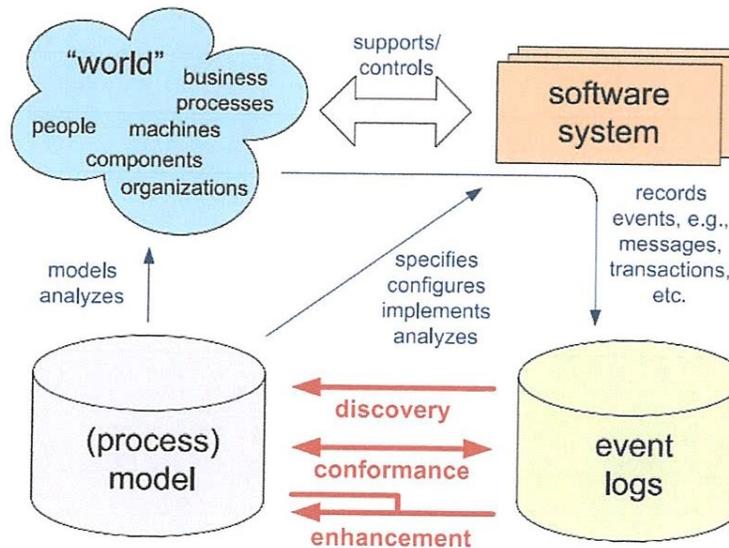


FIGURE 3.3 – Les trois principaux types de fouille de processus [5]

Le premier type de fouille de processus est la découverte. Une technique de découverte utilise un journal des événements et produit un modèle sans utiliser d'informations a priori. Un exemple est α -l'algorithme, cet algorithme prend un journal des événements et produit un réseau de Pétri expliquant le comportement enregistré dans le journal. Par exemple, l'exécution du processus représenté sur la figure 3.2, α -algorithme est capable de construire automatiquement le réseau de Pétri sans utiliser de connaissances supplémentaires. Si le journal des événements contient des informations sur les ressources, vous pouvez également découvrir des modèles liés aux ressources, par exemple, un réseau social montrant comment les gens travaillent ensemble dans une organisation.

Le deuxième type est la vérification de la conformité. Ici, un modèle de processus existant est comparé avec un journal d'événements du même processus. La vérification de la conformité peut être utilisée pour vérifier si la réalité, telle qu'elle est enregistrée dans le journal, est conforme au modèle et vice versa. Par exemple, il peut exister un modèle de processus indiquant que les commandes d'achat de plus d'un million d'euros nécessitent deux vérifications. L'analyse du journal des événements montrera si cette règle est suivie ou non en analysant le journal des événements en utilisant un modèle spécifiant ces exigences, on peut découvrir des cas potentiels de fraude. Par conséquent, la vérification de la conformité peut être utilisée pour détecter, localiser et expliquer les écarts, et pour mesurer la gravité de ces écarts.

Le troisième type est l'amélioration. L'idée est d'étendre ou d'améliorer un modèle de processus existant en utilisant des informations sur le processus réel enregistré dans un journal d'événements. Alors que le contrôle de conformité mesure l'alignement entre le modèle et la réalité, ce type vise à modifier ou à étendre le modèle a priori. Un type d'amélioration est la réparation, c'est-à-dire la modification du modèle pour mieux refléter la réalité. Par exemple, si deux activités sont modélisées séquentiellement mais en réalité peuvent se produire dans n'importe quel ordre, alors le modèle peut être corrigé pour refléter ceci. Un autre type d'amélioration est l'extension, c'est-à-dire l'ajout d'une nouvelle perspective au modèle de processus en l'inter-corrélant avec le journal. Un exemple est l'extension d'un modèle de processus avec des données de performance. Par exemple, en utilisant les horodatages dans le journal des événements du processus de "demande de compensation", on peut étendre la figure 3.2 pour montrer les goulots d'étranglement, les niveaux de service, les temps de traitement et les fréquences. De même, la figure 3.2 peut être étendue avec des informations sur les ressources, les règles de décision, les mesures de qualité, ... etc.

Les techniques de prédiction

La prédiction est de déduire des modèles de prévision et d'appliquer ces modèles aux processus en cours d'exécution, afin d'identifier la possibilité d'exceptions ou d'un comportement indésirable.

On a par exemple, "le panier de la ménagère", à partir de l'analyse de nombreuses transactions, on a pu déduire que dans 33% des cas, lorsqu'un panier contient de la soda, il contient également des chips.

Les arbres de décision des processus métiers [30]

Les arbres de décision constituent une méthode récente et efficace d'exploration de données, en vue de la prédiction d'une variable qualitative à l'aide de variables de tout

type (qualitatives et/ou quantitatives). Cette flexibilité constitue un avantage par rapport à certains outils de classification prévus pour des prédicateurs d'un seul et même type.

Il s'agit d'une méthode itérative, dite de partitionnement récursif des données. En effet, la méthode construit des classes d'individus, les plus homogènes possible, en posant une succession de questions binaires (de type oui/non) sur les attributs de chaque individu.

Contrairement à beaucoup d'outils de classification (régression logistique, SVM⁴, etc.), les arbres de décision sont extrêmement intuitifs et fournissent une représentation graphique, parlante et facile à lire, d'un protocole de classification des individus. Cette représentation graphique est sous forme d'un arbre constitué de feuilles terminales (les classes d'individus) obtenues en suivant un chemin le long des nœuds, chaque nœud correspondant à une question binaire utilisant une variable du jeu de données.

Les arbres de décision permettent donc, d'identifier très rapidement les variables les plus discriminantes d'un jeu de données, en fonction de leur présence parfois répétée le long des nœuds.

La construction de l'arbre :

- Mesure de la pureté des feuilles :

Le choix des " questions les plus discriminantes " afin de construire les nœuds de l'arbre peut se faire selon plusieurs critères : l'algorithme CART utilise l'indice de Gini, tandis que l'algorithme C4.5 utilise l'entropie probabiliste. Ces deux outils mathématiques visent à évaluer la « pureté » de chaque feuille : lorsque l'on se situe à un nœud donné de l'arbre, le but est de créer deux feuilles qui soient plus homogènes que le nœud qui les précède. Il faut donc disposer d'un moyen de mesurer cette homogénéité, ou « pureté ». Grâce à cela, à chaque nœud, le split est construit de manière à maximiser le gain d'information apporté par une question donnée sur la connaissance de la variable réponse.

- Algorithme de construction :

Les individus sont séparés en deux sous ensembles, puis chacun de ces sous-ensembles est à son tour séparé, etc. Progressivement, les individus de l'ensemble d'apprentissage sont divisés en utilisant des « questions » posées à l'aide des variables du jeu de données. Comme énoncé précédemment, ces questions sont choisies de manière à maximiser l'homogénéité des feuilles, c'est-à-dire à maximiser l'information apportée par les réponses.

On peut virtuellement continuer à construire l'arbre jusqu'à obtenir autant de feuilles que d'individus dans le jeu de données : chacune d'entre elles exprimerait alors la particularité d'un des individus

Les techniques de recherche de motifs

Nous distinguons la reconnaissance de la recherche de patterns⁵. La reconnaissance de patterns pré-suppose l'existence d'un modèle dont on va rechercher les occurrences parmi un ensemble de données. A l'inverse, la recherche de patterns (pattern mining) cherche à découvrir des motifs récurrents sans qu'aucun modèle ne soit suggéré. Cette technique est issue du domaine de la fouille de données.

Définition 1 *Un pattern (patron) est un modèle, générique qui répond à un besoin de conception spécifique, et il exprime un scénario répétitif pour un intérêt particulier.*

4. Support Vector Machines

5. Pattern matching

Dans la figure 3.4 suivante, les transitions successives $a.b^*.c$ représentent les actions : a : login, b^* : sélectionner des articles, c : confirmer l'achat.

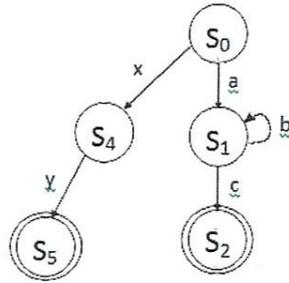


FIGURE 3.4 – Un AFD représentant le processus d'achat d'un article

Un exemple de pattern utile est de savoir : est-ce que le processus tolère la sélection de plusieurs articles ? en réponse, nous pouvons chercher l'existence du pattern (sélectionner des articles)*.

3.4 Analyse au niveau données d'exécution

Les processus métiers n'ont pas une structure de données bien définie, ils peuvent être des données structurées, semi-structurées ou bien non-structurées (type message, par exemple).

3.4.1 Objectifs

La présence de données volumineuses reflétant les différentes exécutions des processus métiers, exige une attention particulière de la part des gestionnaires des processus métiers. néanmoins, vue la diversité des structures de données pouvant stockées les informations relatives aux exécutions divers méthodes d'analyse et d'extraction de ces données en vue le jour. leur objectives visent à améliorer et atteindre les meilleurs résultats du processus métier, tout en comprenant les historiques des exécutions contenues dans les traces des instances.

3.4.2 Exemple simple d'analyse au niveau données

A titre d'exemple, dans le modèle de la figure 3.4, si on cherche à savoir : *Est-ce qu'il y a des clients qui ont annulé des achats d'articles ?* pour répondre à cette question, on doit explorer les traces d'exécution enregistrées dans les fichier logs du processus achat de l'exemple 3.4. Mais, nous avons besoins d'une spécification (technique de recherche) qui doit permettre de prendre en charge ce besoins. Dans notre exemple, il suffit de rechercher l'existence de traces contenant une séquence d'activités de la forme : $a.b^*.c.d$, avec d : annuler l'achat. En d'autres termes, pour que des annulations ont pu être effectuer, il faut que le pattern : $a.b^*.c.d$, soit vérifier, c'est à dire, il faut que la transition d existe dans les traces d'exécution. Bien évidemment, cette transition doit être conforme au modèle (elle doit exister préalablement dans le modèle).

Comme deuxième exemple, un utilisateur pourra être intéressé par connaître : quel est le nombre d'instances ayant sélectionné plus de trois articles ?

Le pattern suivant permet de répondre à ce besoin : le pattern : *a.b.b.b.b**

3.4.3 Les techniques d'analyse des données des processus métiers

L'exploration de la littérature dédiée à l'analyse des données des processus métiers fait ressortir que parmi les outils et méthodes utilisés pour analyser les données, on distingue essentiellement :

- ✕ • Processus d'analyse transactionnelle de données ou OLAP⁶ ;
- les techniques relatives aux entrepôts de données ou Data warehousing ;
- Analyse des données volumineuse ou Big data analysis.
- Intelligence d'affaire (pour business intelligence).
- Recherche de patterns au niveau données.

Dans ce qui suit, on aborde explicitement chaque technique de manière détaillée.

OLAP (On-Line Analytical Processing)

Les analyses de processus métier peuvent bénéficier du traitement analytique en ligne *OLAP* pour réduire les délais d'analyse des données de processus. Ceci est important car la grande quantité de données liées au processus générées chaque seconde doit être analysée presque en temps réel.

L'approche *OLAP* propose d'assister l'analyse et l'exploitation de données, et de fournir aux décideurs une plate-forme à partir de laquelle des informations décisionnelles sont générées [31].

Les applications *OLAP* accèdent généralement à des bases de données volumineuses à l'aide de requêtes lourdes à forte intensité. Elles englobent le support de la décision de données, en se focalisant sur l'analyse interactive des données multidimensionnelles à partir de perspectives multiples, et l'extraction de données en se concentrant sur les problèmes de complexité de calcul [31].

Il y a eu beaucoup de travaux, discutés dans un récent sondage [32] et un livre [33], traitant des méthodologies de modélisation multidimensionnelle pour les systèmes *OLAP*. Les vues conceptuelles multidimensionnelles permettent aux analystes *OLAP* de facilement comprendre et analyser les données en termes de faits et de dimensions montrant les différents points de vue d'où un sujet peut être analysé. [31] propose des éléments de données *OLAP*, tels que des partitions, des dimensions et des mesures et leur classification. Par exemple, en classifiant les mesures *OLAP* en distribution, algébrique et holistique.

Il y a beaucoup d'autres travaux [34, 35] traitant le calcul efficace des cubes de données d'*OLAP*, en incluant : des méthodes efficaces pour le calcul des cubes d'iceberg [35], les calculs des cubes d'iceberg fermés, et des propositions d'algèbre fonctionnant sur des cubes de données [34]. D'autres traitent la classification et le partitionnement de grandes bases de données, notamment : présenter une classification des requêtes *OLAP* pour décider si/comment une requête devrait être parallélisée, tirer parti des feuilles de calcul et des outils ad-hoc, et regrouper et classifier des graphes en étudiant systématiquement les méthodes d'exploitation des réseaux d'information [36]. Tous ces travaux fournissent une sorte de résumé incorporant des fonctionnalités de type *OLAP*. D'autres travaux axés sur l'exploitation minière et l'interrogation des réseaux d'information comprennent : des

6. On-Line Analytical Processing

techniques de traitement de requêtes et de matérialisation de cubes sur des réseaux informatiques [37], des cadres de synthèse pour faciliter l'analyse des données de processus modélisées comme des graphiques, et l'analyse sur un grand graphique de processus avec p-OLAP.

Data warehousing

Data warehousing ou entrepôt de données est une collection d'informations et de données dérivées de systèmes opérationnels et de sources de données externes. Un entrepôt de données est conçu pour prendre des décisions commerciales en permettant la consolidation, l'analyse et la production de rapports à différents niveaux agrégés. Les données sont saisies dans l'entrepôt à travers les processus d'extraction, de transformation et de chargement[38].

Les caractéristiques du Data Warehouse [39]

Un entrepôt de données doit avoir les quatre caractéristiques suivantes :

- **Orienté sujet** : Au coeur du Data warehouse, les données sont organisées par thème. Les données propres à un thème, les ventes par exemple, seront rapatriées des différentes bases OLTP de production et regroupées.
- **Intégré** : Les données proviennent de sources hétérogènes utilisant chacune un type de format. Elles sont intégrées avant d'être proposées à l'utilisation.
- **Non volatile** : Les données ne disparaissent pas et ne changent pas au fil des traitements, au fil du temps (Read-Only).
- **Historisé** : Les données non volatiles sont aussi horodatées. On peut ainsi visualiser l'évolution dans le temps d'une valeur donnée. Le degré de détail de l'archivage est bien entendu relatif à la nature des données. Toutes les données ne méritent pas d'être archivées.

Dans un entrepôt de données, les données provenant de nombreuses sources hétérogènes sont extraites dans une seule zone, transformées selon les besoins du système d'aide à la décision et stockées dans l'entrepôt. Par exemple, une entreprise stocke des informations relatives à ses employés, leurs salaires, produits développés, informations sur les clients, les ventes et les factures. Le directeur de l'entreprise pourrait vouloir poser une question concernant les dernières mesures de réduction des coûts; les réponses impliqueront l'analyse de toutes ces données. Il s'agit d'un service principal de l'entrepôt de données, c'est-à-dire permettant aux cadres de prendre des décisions commerciales basées sur tous ces éléments de données brutes disparates [38].

Ainsi, un entrepôt de données contribue à la prise de décision future. Comme dans l'exemple ci-dessus, un administrateur de cabinet peut interroger des données d'entrepôt pour connaître la demande du marché d'un produit particulier, des données de ventes par région géographique ou répondre à d'autres demandes. Cela donne un aperçu des étapes requises pour commercialiser plus efficacement un produit particulier. Contrairement à un magasin de données opérationnel, un entrepôt de données contient des données historiques agrégées, qui peuvent être analysées pour atteindre des décisions commerciales critiques. Malgré les coûts et les efforts associés, la plupart des grandes entreprises utilisent aujourd'hui des entrepôts de données [38].

Analyse des données volumineuses ou Big data analysis

Les entreprises modernes accumulent des quantités massives de données provenant de diverses sources. Afin de comprendre les affaires, il est nécessaire d'effectuer des analyses

sur de grandes collections hybrides de données d'exécution liées aux processus hétérogènes et partiellement non structurées. Ces données viennent de plus en plus montrer toutes les propriétés typiques des grandes données (*distribution physique large, diversité des formats, modèles de données non standard, et sémantique gérée de façon indépendante et hétérogène*), doivent être représentées sous forme de graphiques, c'est-à-dire de grands graphiques de processus[31].

La découverte et la communication de modèles significatifs des données (c'est-à-dire, les analyses) peuvent aider à comprendre les données des grandes entreprises en vue de prédire et d'améliorer les performances de l'entreprise à l'avenir. Afin de comprendre les données disponibles, nous devons les représenter, comprendre leurs relations et permettre l'analyse de ces relations du point de vue de l'exécution du processus. Pour ce faire, il est possible de représenter les données liées au processus et les relations entre elles par des graphes entité-relation. Dans ce contexte, l'analyse métier peut faciliter l'analyse des graphes de processus de manière détaillée et intelligente en décrivant les applications de l'analyse, les données et le raisonnement systématique [40]. Par conséquent, un analyste peut recueillir des informations plus complètes en utilisant des techniques telles que la modélisation, la synthèse et le filtrage[31].

Intelligence d'affaire (business process intelligence ou BPI)

L'intelligence d'affaire est un large éventail d'outils et d'applications pour la collecte, la consolidation, l'analyse et la diffusion visant à améliorer le pouvoir de décision. Les composants de Business Intelligence qui se concentrent sur la collecte et la consolidation peuvent impliquer des logiciels de gestion de données pour accéder aux variables de données, extraire, transformer et charger des outils qui améliorent également l'accès et le stockage des données dans un entrepôt de données ou un datamart. Dans les étapes d'analyse et de diffusion, chaque fois de nouveaux et différents produits sont lancés et intégrés dans le but de prendre en charge toutes les façons d'utilisation de l'information. Ces produits peuvent inclure la création de rapports, les tableaux de bord affinés contenant des indicateurs de performance personnalisés visuellement riches, des présentations utilisant des jauges, des cartes, des graphiques et d'autres éléments graphiques pour afficher plusieurs résultats ensemble. [41]

L'analyse BPI permet aux utilisateurs d'analyser les exécutions de processus terminées, à la fois d'un point de vue métier et informatique. En effet, les analystes informatiques seront intéressés par l'affichage d'informations détaillées de bas niveau, telles que le temps d'exécution moyen par nœud ou la longueur des files d'attente de ressources. Par contre, les utilisateurs métier seront plutôt intéressés par des informations de niveau supérieur, telles que le nombre d'exécutions de processus "réussies" ou les caractéristiques des processus qui n'ont pas respecté le contrat de niveau de service (SLA⁷) stipulé avec les clients. En plus de fournir un large éventail de fonctionnalités de reporting, BPI propose également plusieurs fonctionnalités pour aider les analystes à identifier les causes des comportements d'exécution de processus.[42]

Le développement d'une solution BPI présente plusieurs défis, tels que :

- Identifier l'architecture et les technologies qui peuvent fournir les fonctionnalités, et comprendre comment appliquer ou modifier ces technologies pour atteindre nos objectifs.

7. Service level agreement

- Permettre la définition de concepts et de mesures permettant une analyse qualitative des processus au niveau de l'entreprise.
- Développer des techniques pour faciliter l'utilisation de l'outil par les analystes et extraire les connaissances dont ils ont besoin, éventuellement sans écrire de code.
- Comprendre comment interagir avec le BPMS et avec les utilisateurs afin de signaler et corriger les situations critiques en temps opportun.[42]

3.5 Conclusion

Après avoir introduit les processus métiers et leur modèle de représentation dans le premier chapitre, dans ce chapitre nous nous sommes focalisés sur les techniques d'analyse des processus métiers. Dans un premier temps, nous avons introduit l'utilité et les objectifs d'une telle analyse que ce soit au niveau modèle de processus métier ou au niveau des données des traces d'exécution. Après, nous avons exposés les techniques d'analyse existantes aujourd'hui et nous avons montré que cette analyse a été portée par plusieurs domaines disparates allant du big data à l'intelligence d'affaire. D'un autre côté, les concepts instances, traces et chemins d'exécution ont été définis et expliqués par des exemples.

Problématique et travaux connexes

4.1 Introduction

Bien que durant les deux dernières décennies beaucoup de travaux de recherche de la discipline informatique ont traité, de manière plus ou moins approfondie, le problème de la modélisation, de l'optimisation et de l'analyse des processus métiers, on observe que peu de travaux se sont intéressés aux challenges exposés par l'analyse des données qui associées aux processus métiers.

En parallèle, plusieurs fournisseurs des suites BPM ont greffés à leurs produits logiciels des modules permettant la conception et l'analyse des processus métiers.

Dans ce chapitre on va explorer les travaux qui ont traité, de façon direct ou dans le cadre d'un contexte particulier, le problème de l'analyse et de la supervision de données des processus métiers. Ensuite, on exposera les apports apportés par quelques suites logicielles dans le domaine de l'analyse des processus métiers et de données d'exécution.

Mais, avant de commencer cette exploration on débute par mettre en relief le problème de recherche traité dans ce mémoire.

4.2 Problématique

Dans un environnement de gestion des processus métiers (BPM), le gestionnaire de processus métiers vise, essentiellement, à disposer d'un cadre opérationnel lui permet de :

- Disposer d'un outil qui lui facilite la prise de décision.
- La spécification d'un ensemble d'indicateurs de performances : KPI ¹, qui vont constituer des métriques d'évaluation des performances de l'entreprise et qui vont orienter sa prise de décision concernant la gestion des processus métiers. Par exemple, on peut établir un indicateur relatif à la durée de vie des instances et estimer quel sera le temps moyens des instances pour achever leur exécution. Évidemment, cette métrique, reflète le niveau d'efficacité des procédures déployées. En effet, plus le temps relatif à la durée de vie des instances est élevé, plus la lourdeur de gestion est constatée.
- Répondre à des préoccupations de type :
 1. Quels sont les processus qui présentent des goulots d'étranglement (bottlenecks)? deadlock, livelock par rapport à certaines ressources matérielles ou humaines ?
 2. Quelles sont les instances qui sont en cours d'exécution d'un processus métier précis (*achat, réservation, facturation, visa, inscription ...*) ?
 3. Quel est le statut d'une instance quelconque ? (*numéro 100, par exemple*) et qui est l'utilisateur qui a déclenché cette exécution ?

1. Key Performance Indicators

4. Quelles sont les instances qui ont exécuté une séquence d'activités particulière? (par exemple la séquence $s=a.b.c.r$ d'un processus métier P).
5. Quelles sont les instances qui n'ont pas avancé dans leur exécution depuis une certaine durée (*72 heures, 1 mois, ...*)?
6. Existe-t-il des instances d'un processus P qui ont effectué la séquence d'activités $s=a.b.c$, suivie de l'activité d? (admettant que l'activité d est une annulation)? quel est leur nombre?

La prise en charge des préoccupations précédentes, en essayant d'apporter une réponse satisfaisante aux questions soulevées par l'analyse et la supervision des données des processus métiers constituent la problématique traitée dans ce mémoire.

Les motivations de notre travail portent sur plusieurs axes, à savoir :

- améliorer la situation existante par un enrichissement des environnements BPM existants, tout en raffinant le cycle de vie des processus métiers.
- offrir un cadre général pour l'exploitation rentable des données stockées dans les bases de données associées aux processus métiers.
- doter les gestionnaires de protocole d'un tableau de bord intégrant des indicateurs de performance.
- permettre une vision globale et une gestion efficace des ressources de l'organisation (*données, ressources humaines, matérielles et financières*).
- assurer l'évolution permanente des processus métiers, en vue de leur adaptation aux besoins mouvants de l'environnement ;

Une fois notre problème bien cerné, nous abordons à présent l'analyse de quelques travaux de recherche qui ont traité ce problème selon différentes facettes.

4.3 Étude des travaux de la recherche théorique

Nous pouvons répartir les travaux de recherche analysés en trois grandes catégories. Ceux qui ont abordé le problème au niveau modèle (*ou schéma*) du processus métier, une autre classe de travaux qui traite le problème au niveau données. La troisième classe intègre simultanément les deux catégories précédentes. Dans ce qui suit, on va présenter ces trois types d'analyse d'une manière explicite.

4.3.1 Analyse au niveau modèles

La modélisation des processus métier est un concept utile dans une grande variété de situations et à de nombreuses fins. Surtout, lorsque les modèles manipulés sont incontournables pour le développement de logiciels complexes.

L'analyse des modèles de processus métiers se focalise sur la vérification de leur spécification et de leur conformité par rapport aux normes établies. L'intérêt de cette analyse est de s'assurer que les processus métiers conçus ne contiennent pas d'anomalies de conception ni d'erreurs conceptuelles induisant des contradictions avec les règles imposées par les méta-modèles utilisés pour décrire les processus métiers.

Dans cette catégorie de travaux, on trouve notamment les travaux les plus consistants suivants.

a. les techniques d'analyse de compatibilité et de remplaçabilité

Vue l'importance de la vérification de la compatibilité des processus métiers, plusieurs travaux ont abordé ce problème.

Pour gérer le cycle de vie des services web, les auteurs dans [1, 43] présentent un cadre général adéquat à la modélisation et à l'analyse de la compatibilité des protocoles de services web, mais qui restent valables pour l'analyse des protocoles des processus métiers, étant donné que tout processus métier peut être implémenté en tant que service web. D'autre par, des mécanismes de substitution des protocoles métiers sont proposés pour remplacer ceux qui ont évolué et/ou qui sont devenus incompatibles avec les nouvelles spécifications, et qui ne pourront plus supporter les mêmes conversations que celles exposées par les protocoles initiaux. Dans [44], la description des protocoles est enrichie par des contraintes temporelles et le même type d'analyse est réalisé. Les contraintes transactionnelles sont formalisées et différentes analyses de compatibilité sont étudiées dans [45]. Néanmoins, toutes les analyses précédentes sont limitées au contexte de compatibilité, et ne prennent pas en compte d'autres besoins particuliers, tels que les évolutions, la maintenance des processus métiers ou les contraintes de sécurité et celle de l'intégration dans un système global.

b. Les approches d'analyse de la migration après évolution

Les approches proposées dans le cadre de l'évolution des processus métiers ont abordé la question de l'analyse de l'impact des changements dans différentes perspectives : niveau protocole du processus métier, par rapport aux interfaces de services web et du point de vue gestion des instances en cours d'exécution.

Certaines approches existantes pour la gestion de l'**évolution dynamique** des protocoles des services web [46, 47, 48, 49, 50, 51] utilisent un modèle de protocole de processus métier qui est basé sur des automates d'états finis déterministes. En se basant sur ce modèle, les techniques proposées étudient d'une manière consistante le problème de la migration des instances lors de l'évolution des schémas des protocoles. Un objectif principal poursuivi par ces approches réside dans l'identification des instances qui ne sont pas affectées par les changements. En conséquence, ces instances peuvent migrer sans difficultés vers le nouveau protocole, et le processus de migration reste transparent pour les clients du service.

Dans [47, 48], une analyse basée sur le modèle de protocole précédent est conduite. Elle prend la forme d'une hiérarchie de classes de remplaçabilité pour gérer la migration des instances du protocole évolué. La classe la plus contraignante ; *remplaçabilité complète* implique une forte relation entre un protocole \mathcal{P} et sa nouvelle version \mathcal{P}' , ce qui nécessite que l'ensemble des chemins d'exécution de \mathcal{P} soit inclus dans l'ensemble des chemins d'exécution de \mathcal{P}' . Par conséquent, dans ce cas de *remplaçabilité complète*, toutes les instances de \mathcal{P} sont, à la fois, compatibles dans le *passé* et compatibles dans le *futur* avec \mathcal{P}' . Donc, elles peuvent être migrées en toute sécurité vers \mathcal{P}' . Dans le cas où la *compatibilité complète* n'est pas satisfaite entre \mathcal{P} et \mathcal{P}' , d'autres classes de remplaçabilité moins contraignantes peuvent être utilisées.

Malgré les contributions majeurs apportées par les approches existantes pour la gestion de l'impact du changement des processus métiers, elles se basent toutes sur des stratégies de migration prédéfinies. La conséquence immédiate est que les gestionnaires de protocoles ne peuvent pas définir leur propres stratégies de migration qui répondront à des besoins

spécifiques de migration, d'où le besoin incessant d'une analyse plus concrète sur les données des protocoles des processus métiers.

c. Les méthodologies générales d'analyse des processus métiers

Les modèles de processus métier sont non seulement utiles pour produire des logiciels qui correspondent aux exigences de l'entreprise, mais ils peuvent également servir de partie utile de la gestion des connaissances, de l'analyse des coûts, de l'analyse des processus métier, etc. Pour cela, les auteurs dans [44] présentent le principe de deux méthodologies de modélisation des processus métiers :

- **Select Perspective :**

Select Perspective est une approche éprouvée et pragmatique pour fournir des solutions logicielles performantes dans une grande variété de contextes. L'avantage de la solution logicielle est mesuré par deux indicateurs :

- Si elle offre des services utiles (*services dont quelqu'un dans l'organisation a besoin*),
- Si elle fournit ces services par rapport à un niveau requis.

Cette méthode peut être mise en œuvre dans le cadre des processus métiers. dans ce cas, l'instance de modélisation des processus métier utilisée par les produits Select Perspective est un modèle de domaine d'une réalité commerciale et d'exigences métier qui peut, si elle est créée correctement, garantir l'atteinte du niveau requis pour les deux indicateurs précédents.

- **ARIS :**

L'architecture ARIS constitue le cadre de développement et d'optimisation des systèmes d'information intégrés ainsi qu'une description de leur mise en œuvre. Le concept ARIS sert comme un modèle pour la création, l'analyse et l'évaluation des chaînes de processus de gestion d'entreprise.

La conception d'ARIS est basée sur un concept d'intégration dérivé d'une analyse holistique des processus métiers. Lors de la modélisation complète des processus métier, un modèle très complexe peut en résulter. Pour réduire une telle complexité, l'architecture d'ARIS comprend différentes vues, dont les relations mutuelles sont strictement définies. Une seconde approche de réduction de la complexité est l'analyse de différents niveaux descriptifs. Cela garantit une description cohérente des problèmes liés à la gestion d'entreprise jusqu'à leur mise en œuvre technique.

Dans la dernière partie de l'article [44], les auteurs dressent une comparaison entre les deux méthodologies et arrivent aux conclusions suivantes.

- La méthode Select perspective est orientée objet alors qu'ARIS est une méthodologie structurée.
- La méthode Perspective est beaucoup plus spécialisé q'ARIS. ARIS se considère non seulement comme une méthodologie mais aussi comme un cadre conceptuel global pour la création de méthodologies et de méthodes de modélisation/gestion de processus métier.
- La méthode Perspective est une méthodologie agile strictement orientée vers le développement des SI, alors que ARIS est un cadre général couvrant tous les domaines d'analyse métier possibles.
- L'intention d'ARIS est la modélisation, l'association, la mesure et l'optimisation de chaque chose/personne/entité/processus/activité pertinente.

[45] dans cette lignée de travaux, une méthodologie d'analyse des processus métier est décrite. Elle a été développée au département des technologies de l'information de

l'Université d'économie de Prague. Les raisons du développement d'une telle méthodologie proviennent de plusieurs sources :

- Le besoin d'identifier la substance réelle des activités à soutenir par le système d'information.
- Dans le domaine de la ré-ingénierie des processus métier (**BPR**), il existe également un besoin réel de moyens (*techniques, outils et méthodes*) pour identifier la substance des processus à (re)concevoir.
- Les produits informatiques supportant la gestion des workflows deviennent de plus en plus intéressants et utiles. Comme conséquence de ce fait, la nécessité de la théorie qui répond à la question fondamentale : " Quelle est l'origine du flux de travail et où les règles substantielles pour cela doivent être recherchées ? " est de plus en plus réelle. une approche exacte de l'analyse conceptuelle du workflow est toujours manquante dans ce domaine.

Dans cet article une présomption de base pour suivre trois principes généraux (mutuellement dépendants) de la méthodologie est élaborée :

- Principe de modélisation.
- Principe des différentes architectures de processus.
- Principe d'abstraction.

Finalement, l'approche proposée explique le processus d'analyse des processus métiers et ses différentes phases.

d. Modélisation déclarative des processus métiers

La documentation sur les processus métier a proposé une multitude d'approches ou de paradigmes de modélisation des processus métier. Mais, le paradigme le plus dominant est celui de l'orientation déclaratif ou paradigme déclaratif.

Les auteurs dans [54], se concentrent sur l'approche déclarative, dans le but de fournir une analyse de la modélisation déclarative des processus en soulignant les principes généraux et leur impact, et en discutant des approches déclaratives modernes de modélisation des processus métiers. Les approches de modélisation des processus déclaratifs se concentrent sur ce qui devrait être fait pour atteindre les objectifs de l'entreprise, sans prescrire comment un état final devrait être atteint. Par conséquent, ces modèles spécifient un ensemble de contraintes, de règles métier, de conditions d'événements ou d'autres expressions (*logiques*) qui définissent les propriétés et les dépendances entre les activités d'un processus métier.

Dans [54], les auteurs ont présenté un aperçu des différentes approches de la modélisation de processus déclaratifs, allant de la complémentation de modèles impératifs à la logique déclarative jusqu'à atteindre des approches de modélisation déclaratives de processus, à part entière. Ces approches déclaratives diffèrent les unes des autres dans les préoccupations commerciales qui peuvent être spécifiées, l'espace d'état requis, les types de contraintes disponibles et les paradigmes de représentation et de raisonnement utilisés. Cependant, leurs objectifs, la flexibilité et les types d'applications restent les mêmes.

Le premier ensemble de principes que ils ont indiqué comment la modélisation des processus déclaratifs gère les préoccupations des entreprises. Le paradigme opte pour la prise en compte explicite des préoccupations des entreprises et pour la prise en compte de la modalité, ce qui se traduit par une flexibilité, une traçabilité et une adaptabilité accrues.

Le deuxième ensemble décrit l'absence de flux de contrôle pré-calculés et son effet positif sur la réduction du biais d'hypothèse. Les modèles sous-spécifiés qui en résultent

offrent une flexibilité supplémentaire.

Troisièmement, l'utilisation de la granularité au niveau de l'activité et le concept de cycle de vie de l'activité, les techniques de modélisation de processus déclaratives sont capables de produire des contraintes de transition plus spécifiques qui sont instantanément transférées à tous les processus contenant cette activité.

Quatrièmement, les approches de modélisation des processus déclaratifs mettent l'accent sur les agents, considèrent le travail de coordination comme des tâches régulières, évitent de modéliser la logique de communication et embrassent la perspective de la troisième personne. Enfin, les modèles de processus déclaratifs pourraient avoir tendance à échanger une certaine compréhensibilité pour une plus grande expressivité.

4.3.2 Analyse au niveau données

Contrairement à l'analyse précédente qui se concentre sur les contraintes imposées sur la spécification du processus métier (*c'est à dire son modèle*), les travaux d'analyse des processus métiers par rapport au niveau données s'intéresse uniquement aux données générées lors de l'exécution des processus métiers. Ces données sont souvent stockées dans fichiers logs ou dans des bases de données adéquates.

Plusieurs domaines ont traité l'analyse des données des processus métiers. Dans ce qui suit, une étude des domaines les plus significatifs est exposée.

a. L'entrepôt de données des processus métiers

Améliorer les procédures de travail est vital à toutes les entreprises. L'amélioration des processus métiers nécessite une analyse réelle des données collectées, comme première étape de base.

Dans ce contexte les auteurs dans [55] présentent les défis de l'analyse de processus qui sont les suivants :

- la conception de solutions ad-hoc pour chaque processus métier exécuté dans l'entreprise est trop coûteuse. Par conséquent, des approches génériques doivent être recherchées ;
- le niveau d'abstraction auquel les processus doivent être analysés est beaucoup plus élevé par rapport à l'information disponible dans l'environnement d'exécution du processus ;
- le besoin croissant de co-développement de l'analyse des processus et de la solution d'automatisation des processus ainsi que l'ampleur du problème rendent difficile la gestion des changements fréquents dans les sources des données de processus.

Pour relever ces défis, les auteurs ont proposé un modèle conceptuel pour l'entrepôt de données de processus. Plus précisément, l'article apporte les contributions suivantes :

- Analyser et classer les exigences d'analyse pour l'entrepôt de données de processus.
- Fournir un modèle d'entrepôt configurable qui peut répondre aux besoins des rapports complexes pour n'importe quel processus, en tenant également compte des contraintes de performance. Le modèle aborde les problèmes récurrents des clés tels que le compromis entre le besoin de modéliser l'hétérogénéité (*chaque processus est différent*) et celui de définir une représentation uniforme pour tous les processus (*pour soutenir la réutilisabilité et l'analyse croisée des processus*).
- Montrer comment extraire des données de bas niveau sur les processus exécutés vers des vues de plus haut niveau du même processus, adaptées à la création de

rapports. L'approche est basée sur la définition de processus abstraits et ensuite la cartographie de la progression du processus vers des événements se produisant dans les systèmes sources.

- Décrire comment extraire, transformer et charger des données de processus, et en particulier comment maintenir de façon semi-automatique les procédures ETL² à la suite de changements dans les applications source.
- Montrez comment la solution peut être rapidement prototypée en utilisant un environnement d'émulation pour obtenir rapidement les réactions des utilisateurs.

b. Approches basée sur Big Data (données volumineuses)

Les processus métier sont omniprésents et interviennent dans plusieurs secteurs : marketing, santé, gestion financière et bien sûr métier, et génèrent une quantité importante de données dites big data. Ces dernières années, la gestion des modèles et des données des processus métiers est très difficile. D'une part, les processus métiers doivent être puissants en termes de modélisation. D'une autre part, le support d'analyse de Big Data permet de trouver des connaissances appropriées pour adopter des modèles de processus métier.

La contribution de [56] est de proposer une framework pour faciliter l'amélioration des processus métiers basée sur l'analyse de Big Data. Cette framework décrit la surveillance des processus métiers depuis la phase de modélisation, le déploiement jusqu'à l'analyse des données pertinentes au moyen d'outils d'analyse des données volumineuses.

La première section de l'article est dédiée au BPM actuel et explique le cycle de vie des processus métiers. Après l'introduction des concepts du big data, les auteurs exposent le concept de 3V (**volume, variété et vitesse**) présenté par Laney en 2001, et expliquent le cycle de vie des Big Data. Ensuite, ils définissent l'analyse de Big Data (Big Data Analytics BDA), puis ils présentent les principales techniques adoptées pour analyser le Big Data. Les auteurs mettent l'accent sur l'analyse de Big Data et son domaine d'activité applicable, et montrent la relation entre l'analyse des Big Data et l'analyse des processus métiers. Ils ont montré aussi l'insuffisance des systèmes de gestion de bases de données relationnelles (SGBDR), et favorisent les bases de données NoSQL³, comme XLM, et citent ses points forts et ses avantages. En effet, les auteurs stipulent que les systèmes relationnels accordent un peu d'attention aux données semi-structurées et non structurées. Ils se concentrent et prennent en charge uniquement les données structurées. De plus, les SGBDR évoluent avec du matériel coûteux, ce qui ne permet pas de gérer le volume croissant de données.

L'architecture du framework proposé est expliquée en détails. Pour la conception de leur travail, ils ont choisi d'adopter MongoDB pour stocker les données.

c. Les techniques On-line Analytical Processing (OLAP)

Après la phase de collecte des données, différents types d'analyses s'imposent. Analyser les données des systèmes transactionnels consiste à les collecter dans un entrepôt de données (*à l'aide d'outils d'extraction, de transformation et de chargement ETL*), puis à utiliser un outil adéquat pour découper les données selon différentes dimensions. On-line Analytical Processing (OLAP) fait référence à l'activité générale d'interrogation et de présentation de données textuelles et numériques à partir d'entrepôts de données et/ou de data marts à des fins d'analyse.

2. ETL : Extract, Transform and Load

3. Not Only SQL

Le terme OLAP a suivi le développement du concept de base de données standard Online Transactional Processing OLTP. OLTP fait référence à l'activité générale de mise à jour, d'interrogation et de présentation des données textuelles et numériques des bases de données à des fins opérationnelles. En d'autres termes, OLTP englobe toutes les transactions quotidiennes effectuées sur les systèmes de base de données opérationnels.

Les trois fonctionnalités OLAP de base utilisées régulièrement par les analystes sont communément appelées :

- Slice et Dice
- Pivot (Rotation)
- Drill Down et Drill Up

Les outils OLAP sont désormais une partie essentielle du processus de prise de décision pour chaque organisation qui collecte de grandes quantités de données. Plus les données accumulées dans ses opérations sont importantes, plus les capacités OLAP essentielles deviennent importantes pour une organisation. L'attrait universel de OLAP réside dans la simplicité de la structure et de la conceptualisation, et par conséquent, dans sa simplicité d'utilisation.

A titre d'illustration de l'importance des travaux ayant abordé l'analyse des données des processus métiers au niveau données, dans [57], les auteurs donnent un aperçu des techniques OLAP et expliquent comment elles sont utilisées pour l'aide à la décision.

Dans ce travail, les auteurs présentent les fonctions spécifiques d'un système OLAP et ces plates formes. Ils spécifient aussi, la connexion entre les systèmes OLAP et les référentiels de données analytiques. Ensuite, un aperçu des fonctionnalités communes à tous les outils OLAP est présenté.

Dans cette section on a montré comment les techniques OLAP sont liées à des données référentiels analytiques, tels que les entrepôts de données et les data marts, et qu'elles offrent à l'utilisateur un moyen efficace d'accéder aux données des processus métiers pour faciliter l'analyse de données et permettre une prise de décision de gestion efficace.

A partir de cette étude on peut conclure que, naturellement, l'approche OLAP a ses limites. Une analyse plus approfondie des données peut être réalisée avec des méthodes plus complexes, telles que les procédures statistiques ou l'exploration de données. Cependant, dans la plupart des scénarios liés aux activités, ces méthodes complexes ne doivent être appliquées qu'après l'analyse basée sur OLAP. Ainsi, Une analyse plus approfondie révèle que le modèle dimensionnel est essentiel pour OLAP. Si les données sous-jacentes n'étaient pas organisées de manière dimensionnelle, avec une table de faits au centre connectée à un certain nombre de tables de dimensions, les trois opérations OLAP de base ne pouvaient pas être exécutées efficacement. De ce qui précède, une d'autres techniques d'analyse doivent entrer en force pour prendre en charge les besoins des gestionnaires de processus métiers.

4.3.3 Analyse mixte (données/modèles)

Pour comprendre le comportement des systèmes d'information aussi bien que les processus qui les supportent, deux grandes orientations ont vue le jour : les techniques d'adaptation et la technologie BPM.

a. Approches d'adaptation des processus métiers

Dans ce type travaux, l'accent est mis sur le calcul et la prise en charge des différences entre les protocoles et/ou les interfaces résultant de la mise en œuvre des changements.

- Dans [58], l'auteur propose un ensemble de contributions permettant une vérification de la cohérence et de la conformité des modèles de processus métier après chaque changement, mais aussi d'établir une évaluation *a priori* de l'impact structurel et qualitatif des modifications. L'analyse de l'impact est basée sur la relation générique dite de dépendance. En effet, une telle relation permet de dire : *si une entité B dépend d'une autre entité A alors le changement de A aura un impact sur B*. Dans ce travail l'analyse de l'impact est abordée au niveau schéma du processus métier aussi bien que sur les données exprimées par les effets sur les instances qui sont en cours d'exécution.

Les auteurs dans [59], posent le problème de l'amélioration des processus métiers des grandes et moyennes entreprises. Ils suggèrent que l'identification des besoins métiers et la spécification des solutions aux problèmes de l'entreprise passe inévitablement par l'analyse des données générées et stockées dans les systèmes d'information afin de faciliter la prise des décisions. Pour cela, ils présentent différents aspects de l'analyse des données et des modèles des processus métiers. En ce sens, ils spécifient plusieurs dimensions garantissant l'adaptation des processus métiers aux changements de lois et réglementations. Les analyses opérées portent sur plusieurs dimensions : le modèle du processus, les données manipulées, les flux d'activités réalisées ainsi que les états atteints par les instances actives.

b. État des lieux de la technologie BPM actuelle

Au cours de la dernière décennie, la gestion des processus métier (BPM) est devenue une discipline mature, avec un ensemble bien établi de principes, méthodes et outils combinant les technologies de l'information, les sciences de gestion et l'ingénierie industrielle.

Malgré l'attention accordée au BPM dans les universités et l'industrie, il existe un écart considérable entre : les technologies et approches BPM les plus modernes et l'utilisation réelle par les praticiens du BPM et leurs besoins. Par exemple, beaucoup d'organisations utilisent les systèmes BPM pour exécuter automatiquement leurs processus opérationnels. Dans de nombreux cas, les processus sont codés en dur dans les applications (*par exemple, les systèmes ERP tels que SAP ou les systèmes développés en interne*). Pour cela, les auteurs dans [59], réfléchissent sur l'état actuel du BPM et sur ce qui pourrait être fait pour combler le fossé entre la recherche BPM et l'utilisation pratique des technologies BPM. Ils arrivent à la conclusion que la recherche BPM, il y a souvent eu une focalisation excessive sur des artefacts spécifiques (*tels que les modèles de processus*). Cependant, de meilleurs modèles ne produisent pas automatiquement de meilleurs processus. Par conséquent, la recherche devrait être mieux alignée sur l'objectif initial de BPM d'améliorer les processus métiers, plutôt que d'améliorer les modèles de processus.

D'un autre point de vue, les indicateurs clés de performance (KPI) relatifs aux performances communes des processus tels que le temps, la qualité, les coûts et la conformité sont souvent mentionnés dans la recherche sur l'amélioration des processus, mais les résultats de recherche et la technologie BPM associée contribuent souvent à améliorer les KPI. En même temps, beaucoup de bonnes idées ne sont pas adoptées : elles ne sont pas implémentées dans les systèmes d'information que les gens utilisent réellement. De plus, la résistance organisationnelle peut constituer un obstacle majeur à la réussite des initiatives de BPM.

Le constat général est que la tendance actuelle s'oriente vers l'automatisation des processus. Cependant, la majorité des approches de recherche BPM existantes semblent toujours reposer sur les hypothèses utilisées par les systèmes de gestion du flux de travail

WFM⁴ et les systèmes d'information de bureau OI⁵. La gestion des processus s'est révélée beaucoup plus "épineuse" que ne le pensaient les pionniers du domaine.

Plusieurs questions demeurent posées : comment définir un meilleur processus ? Les auteurs dans [51] ont tenté de répondre à cette question par déduire que dans la littérature BPM et WFM. Se concentrer sur les modèles de processus, et ils croient que de meilleurs modèles mèneront à de meilleurs processus.

Ce type de question s'impose pour plusieurs raisons :

1. Les modèles de processus utilisés pour l'analyse des performances peuvent ne pas ressembler à la réalité. Ils s'appuient sur les informations de ceux qui participent au processus, à travers des ateliers ou des entretiens. Ils peuvent décrire une situation idéalisée ou particulière et ne sont donc souvent pas utiles pour fournir les informations nécessaires.
2. Ces modèles conceptuels sont rarement utilisés pour guider la mise en œuvre d'une solution d'automatisation de processus.

Malgré tout le travail sur la flexibilité, les systèmes BPM sont toujours perçus comme étant trop restrictifs, mais très coûteux. Par conséquent, ils se disputent qu'une focalisation sur l'automatisation n'aidera pas à combler le vide. Les modèles de processus ne sont utiles que s'ils aident réellement à améliorer les processus. Pour cela, ils préconisent de se concentrer sur le processus plutôt que sur son modèle. Cela ne signifie pas que les modèles de processus doivent être abandonnés, mais plutôt qu'ils doivent être créés dans un but précis.

Un meilleur processus est donc celui qui contribue le mieux à atteindre les objectifs stratégiques d'une organisation. Et pour mesurer les améliorations, les auteurs ont proposé d'utiliser indicateurs clés de performance (KPI) qui sont des quantités qui peuvent être déterminées sans ambiguïté pour un processus métier donné, en supposant que les données permettant de calculer ces mesures de performance sont disponibles.

Après avoir identifié les KPI, la question que les auteurs dans [51] ont posé après est la suivante :

"comment améliorer le processus en termes de KPI?", c'est-à-dire, comment améliorer les KPIs du processus pour que ceux-ci atteignent les objectifs envisagés.

Deux directions de recherche possibles sont discutées. La première, est réalisé par Six Sigma [52], elle consiste à l'utilisation des outils d'analyse statistiques aux données organisationnelles, pour mesurer et réduire le degré de variabilité des processus métiers. Par exemple, en termes d'erreurs ou défauts dans les processus métiers, et de contrôler le fonctionnement des processus dans les objectifs souhaités. Cependant, Six Sigma se concentre sur l'amélioration des processus métier en quantifiant statistiquement les changements de performance des processus, les données utilisées pour telles analyses sont généralement collectées manuellement. Cela rend l'utilisation de telles techniques, lorsqu'elle est effectuée correctement, très coûteuse et prend beaucoup de temps. En outre, Six Sigma regarde rarement à l'intérieur des processus de bout en bout. L'accent est mis sur une étape spécifique du processus ou sur des mesures globales. Ce problème peut être évité en utilisant des techniques qui extraient automatiquement des connaissances de processus à partir de données d'événements consignées par des systèmes d'informations communs.

La deuxième direction, est le domaine de fouille de processus métier [4, 53, 54, 55]. Cette orientation vise l'exploitation des données existantes des exécution en vue de remon-

4. Workflow Management

5. Office Information

ter d'un niveau d'abstraction et aboutir à des modèles de processus ou modèle comportementaux. Il existe toute une gamme de méthodes et d'outils pour exploiter ces données pour découvrir automatiquement un modèle de processus, vérifier sa conformité aux modèles de référence existants ou pour déterminer les causes d'évictions ou de variantes de processus. L'avantage de s'appuyer sur des données enregistrées par rapport aux données collectées manuellement est que toute idée extraite de ces données est basée sur des preuves, plutôt que sur la confiance humaine. De plus, les artefacts extraits par extraction de processus, peuvent être améliorés avec des informations de performance de processus (en direct). Cela permet aux organisations d'examiner les processus de bout en bout. Cependant, un large éventail de techniques ont été développées dans ce domaine, la communauté de la recherche a principalement consacré son attention à la qualité des artefacts produits plutôt que d'améliorer les processus réels pour lesquels ces journaux sont disponibles.

La solution proposée dans [56], est de combler le fossé actuel entre l'extraction de processus et Six Sigma. Par exemple, les techniques d'extraction de processus pourraient être utilisées pour extraire des mesures de performance de processus détaillées et précises, et les techniques Six Sigma pourraient être utilisées pour identifier les causes de variabilité et évaluer l'impact de différents changements de processus sur les indicateurs clés de performance KPI du processus.

L'article [57] traite une autre technique pour obtenir de meilleurs processus. Celle là, consiste à appliquer les techniques de la recherche opérationnelle au domaine des processus métiers en utilisant diverses techniques mathématiques, telles que la simulation, la théorie des files d'attente, l'optimisation et les statistiques. De nombreux problèmes d'amélioration des processus peuvent en fait être attribués à des problèmes typiques étudiés par la recherche opérationnelle, car il existe généralement un certain nombre de contraintes et d'options qui compliquent la recherche de solutions optimales. D'une certaine manière, l'objectif est d'optimiser un processus en fonction d'indicateurs clés de performance donnés (généralement le temps et l'utilisation des ressources).

La valeur de lier la recherche opérationnelle et BPM a été réalisé par [58], qui a préconisé l'utilisation des files d'attente pour évaluer les conditions dans lesquelles les changements de processus dans le contexte des initiatives susceptibles de ré-engineering de processus BPR approprié. Plus récemment, des techniques de recherche opérationnelle ont été appliquées pour résoudre les problèmes de contention des ressources dans les processus métier [59, 60]) ou pour identifier une allocation optimale des ressources humaines aux activités de traitement [61]). Cependant, à part ces quelques exceptions, les techniques de recherche opérationnelle n'ont pas encore été systématiquement appliquées pour résoudre les problèmes d'amélioration des processus.

4.4 Exploration des suites logicielles industrielles

Dans cette section on va faire un survol du domaine industriel afin de voir ce qui se passe du côté des firmes de production des logiciels et quels sont les possibilités offertes par les outils commerciaux dans le cadre de l'analyse des données des processus métiers.

4.4.1 IBM web sphère application server :

IBM WebSphere est un kit de produits logiciels qui aide les entreprises à créer des applications ou à intégrer des applications, des bases de données et des systèmes intégrés.

Il fournit toute une gamme d'environnements d'exécution Java EE7 disponibles sur site ou sur tout cloud public, privé ou hybride. Utile dans plusieurs contextes, Web Sphère s'adapte à tous les besoins de l'entreprise.

Le logiciel IBM webSphere⁶ pour les environnements SOA⁷ permet l'exécution des processus métiers dynamiques et inter-connectés, tout en offrant des infrastructures applicatives à hautes performances. Ces dernières sont adaptées à toutes les situations rencontrées par les entreprises [62].

Les différentes versions existantes, telles que : webSphere Business Compass, Business Modeler Advanced ou Rational Application Developer sont des outils web collaboratifs offrant des interfaces ergonomiques pour la conception et le développement des processus métiers. Dans ces environnements intégrés, les utilisateurs peuvent afficher et commenter les brouillons des processus publiés, et peuvent collaborer avec différents participants pour définir de bonnes pratiques en matière de modèles. Reposant sur l'édition webSphere Business Modeler Basic, la version Business Modeler Advanced permet aux professionnels de générer la modélisation, la simulation et l'analyse des processus métiers pour mieux comprendre, documenter et déployer leurs activités. webSphere Business Modeler Advanced aide à accélérer le développement des processus en vue d'une adaptation aux besoins métier évolutifs actuels [62].

D'une manière générale, les outils présentés par IBM aident les développeurs à concevoir, développer, assembler, tester et déployer des services web. Ils intègrent des outils de test et d'analyse, qui permettent d'identifier et de corriger les problèmes d'évolution. Cependant, des insuffisances sont constatées pour la gestion de la migration des instances actives. En effet, cette dernière est laissée à la charge de l'utilisateur qui est contraint de prendre des décisions, au cas par cas, pour trancher sur la poursuite des activités quand une nouvelle version du processus métier est déployée.

Caractéristiques et avantages de Web Sphère Application Server :

- Optimiser l'infrastructure des applications afin de réduire les coûts grâce à des fonctionnalités hybrides suffisamment souples pour permettre un déploiement et une gestion des applications sur tout cloud et service d'annuaire.
- Connecter les applications Java existantes au cloud et dégager de nouvelles sources de valeur avec la gestion du cycle de vie des API⁸ et des services cloud comme IBM Watson (US), IBM Cloudant et IBM dashDB.
- Créer et déployer rapidement des applications cloud natives et web, ainsi que des micro services grâce à un environnement d'exécution de production léger et composable comportant une console d'administration unique pour les applications et les API Java et Node.js.
- Être opérationnel en quelques minutes dans un environnement cloud hébergé grâce à IBM WebSphere Application Server on Cloud.

6. <https://www-01.ibm.com/software/websphere/subscriptionandsupport/business-process-manager/versions/>

7. Service Oriented Architecture

8. Application Programming Interface

4.4.2 Suite Oracle

Oracle Business Process Analysis Suite (Oracle BPA) [63] est un outil de modélisation et d'analyse destiné aux analystes métier. Oracle BPA Suite étend la valeur métier des processus BPEL [18] et favorise la collaboration entre l'analyse métier et l'informatique. Il permet aux propriétaires de processus, aux analystes métier qualifiés et aux architectes d'effectuer la modélisation et l'analyse de processus, la simulation et la publication de modèles de processus. La suite Oracle BPA prend également en charge l'exécution et la surveillance de ces modèles de processus pour les développeurs informatiques avec BPEL et Oracle BAM.

Plus précisément, Oracle BPA Suite contient les composants suivants :

- Architecte fournit une modélisation de processus à l'aide de BPEL, basée sur une architecture orientée services robuste, pour l'élaboration de processus métier.
- Éditeur fournit une présentation basée sur les rôles des modèles de processus à un portail Web.
- Le référentiel stocke et fournit un accès simultané aux modèles de processus métiers et aux artefacts associés.
- Les développeurs informatiques peuvent utiliser Oracle JDeveloper pour utiliser ces modèles afin de créer des applications composites SOA.
- Le simulateur simule les modèles de processus en fonction d'un ensemble d'événements discrets pour effectuer une analyse «what if». Cette simulation permet aux analystes métier de détecter les goulets d'étranglement, de déterminer le temps de traitement des processus, de comparer les alternatives de processus et de quantifier l'impact des modifications de l'organisation.
- Oracle SOA Extensions fait passer les processus métier des concepts à l'exécution en partageant les plans de processus avec Oracle SOA Suite.

Oracle BPA permet la collaboration entre les utilisateurs professionnels et informatiques en diffusant des informations. Les modèles d'entreprise favorisent le développement rapide de BPEL en propageant les changements métier en temps réel vers l'informatique. Les modifications peuvent également s'appliquer au travail en cours sans perte de données.

4.4.3 SAP NetWeaver BPM

SAP NetWeaver BPM [64] est une plate-forme d'automatisation de processus, il prend en charge la modélisation des processus métier basés sur BPMN [2], et il fournit également un environnement d'exécution pour exécuter ces processus (Process Server).

"Process Composer" est l'environnement de conception pour la modélisation des processus métiers et intègre de nombreux autres outils SAP pilotés par les modèles dans "SAP NetWeaver Developer Studio", tels que :

- Règles compositeur pour décrire les règles métier (y compris les tables de décision)
- Service compositeur pour les services composés construits sur d'autres services
- "Composite Application Framework" pour les services d'application et les objets métier
- Outils d'interface utilisateur tels que Web Dynpro Java / ABAP, Visual Composer et outils SAPUI5
- Outils ESR pour connecter et importer des services à partir du référentiel de services d'entreprise
- Outils Eclipse standard comme par exemple les outils JEE

La modélisation avec "SAP Process Composer" :

Un modèle de processus est une abstraction d'un processus et décrit tous les aspects du processus, qui sont :

- **Activités** : étapes qui sont exécutées dans le processus
- **Rôles** : utilisateurs ou systèmes exécutant les activités
- **Artefacts** : objets, tels que des documents commerciaux.

4.4.4 Bonita BPM

Bonita BPM [65] a été créé en France en 2001. Actuellement, c'est l'un des BPM le plus utilisé au monde. La société a plus de 1000 clients comme Orange, Xerox, Accenture, La Redoute ou Cisco, répartis dans plusieurs pays.

BonitaSoft propose 4 éditions différentes de son logiciel dont une open source qui s'appelle Community et 3 payantes qui sont Teamwork, Efficiency et Performance.

Bonita BPM est un outil qui permet de créer, exécuter, surveiller et améliorer des applications métier qui connectent des interfaces utilisateur personnalisées à des processus de back-office fiables connectés aux données et aux autres systèmes d'information métier pour créer des expériences utilisateur engageantes et satisfaisantes.

Le point fort de cet outils est l'architecture découplée, qui offre une séparation claire entre la logique métier, les données et les interfaces utilisateur. Le concept spécifique appliqué est un contrat qui agit comme une interface. Cela apporte plus de flexibilité et d'agilité : l'équipe front-end et l'équipe de workflow peuvent collaborer et développer en parallèle, et peuvent tous les deux faire des changements lorsque l'application est en production. Une fois que le concepteur d'interface utilisateur et les développeurs de processus sont d'accord sur une définition de contrat, ils peuvent chacun travailler indépendamment.

Bonita BPM interfaces

Pour utiliser Bonita [66] nous avons besoin du Studio Eclipse pour la partie modélisation des processus, du Portail web pour le test des processus ainsi que du serveur intégré pour générer les processus dans le Portail web depuis le Studio Eclipse.

Ci-dessous, une explication brève de chaque interface :

- **Bonita BPM Studio :**

Le Studio Eclipse est divisé en 5 zones visibles sur la figure ci-dessous :

- La coolbar
- La zone du diagramme
- Les éléments BPM
- La vue arborescente / synthèse
- La partie de configuration des diagrammes

Le Studio Eclipse de Bonita BPM est aux couleurs du logo de BonitaSoft. On y trouve différents liens d'accès rapide notamment pour démarrer la création d'un diagramme, ouvrir ou importer un projet existant, des liens vers des vidéos tutoriels pour les débutants, de la documentation, etc

- **Analyse du Portail web :**

Le Portail web comporte deux vues distinctes : le test des processus et le portail administratif. Il s'ouvre lors du lancement du projet et on peut directement le tester.

- **Le serveur :**

L'utilisateur n'a pas d'interaction avec le serveur, il fonctionne en arrière-plan sans qu'il n'y ait besoin d'avoir des connaissances en serveur. Il se lance à l'ouverture de Bonita BPM et s'arrête à la fermeture du logiciel, ce qui en facilite l'utilisation. Il n'y a pas besoin de le configurer ni de le démarrer ou l'arrêter.

Fonctionnalités de Bonita BPM

Bonita Business Process Management permet de :

- **Modéliser** les processus métiers graphiquement à l'aide de la norme BPMN [2].
- **Gérer** des données complexes facilement avec la gestion des données métier et bénéficiez d'un chargement plus rapide, un affichage plus facile des formulaires web et une persistance sans interruption.
- **Construire** des interfaces utilisateurs attrayantes en utilisant un concepteur d'interface utilisateur graphique de pointe, basé sur une technologie Web.
- **Effectuer** des tâches et administrez des processus en utilisant le portail web Bonita BPM.
- **Créer** des interfaces utilisateurs personnalisées si le portail standard ne convient pas l'utilisateur.

4.5 Conclusion

Dans ce chapitre nous avons abordé une étude approfondie des travaux connexes ayant traité la question de l'analyse des données des processus métiers.

Nous avons commencé le chapitre par mettre en relief la problématique de recherche abordé dans ce mémoire et nous avons explicité les motivations qui ont conduit à ce travail. Après, nous avons exposé divers travaux de la recherche académique qui sont en relation avec notre sujet. Ce type de travaux a été classé en trois catégories suivant la portée de l'analyse : modèle, données ou mixte. Ensuite, on a essayé d'explorer le domaine des suites logicielles existantes afin de déceler les propriétés et les fonctionnalités des systèmes existants.

En conclusion, nous pouvons affirmer que malgré la richesse et la diversité des approches, techniques et outils d'analyse des données des processus métiers certaines insuffisances sont encore résiduelles. Elles concernent, essentiellement, l'absence d'approches qui visent à extraire des éléments structurels (*chemins, sous-protocoles, état, ...*) de la spécification des processus métiers (modèles). D'autre part, un manque flagrant en outils qui exploitent les données semi-structurées en vue de filtrer certaines données représentant un intérêt de gestion particulier. La dernière observation concerne les KPI. En effet, malgré la batterie des différents indicateurs proposés, ceux-ci restent généralistes et ne tiennent pas compte des réalités particulières des données des processus métiers.

Pour surmonter les insuffisances précédentes constatées lors de la phase analyse des travaux connexes, dans la suite du mémoire nous présentons notre contribution.

Deuxième partie
Contribution

Approche d'Analyse et de Supervision des processus

5.1 Introduction

Dans les environnements de gestion des processus métiers, ces derniers sont déployés et invoqués par des milliers, voir des millions d'utilisateurs simultanément. Chaque invocation d'un processus métier correspond à une instance d'exécution et génère des données afférentes aux activités réalisées, aux ressources utilisées, aux utilisateurs impliqués ainsi qu'au temps relatif à la réalisation des activités. Par ailleurs, les outils BPMs existants offrent des mécanismes pour le stockage des données générées. On parle de données persistantes des processus métiers.

Dans ce chapitre, on va proposer une approche permettant l'analyse et la supervision des données relatives aux processus métiers. Nous commençons par la présentation des outils théoriques, des concepts et notions supportant notre approche, puis nous détaillerons le cadre formel qui soutient l'approche proposée. Nous terminons le chapitre par l'exposé de l'architecture et les fonctionnalités du système que nous avons conçu.

5.2 Choix du modèle des processus métiers

Dans cette section, nous présentons d'abord la spécification formelle du modèle de processus métiers utilisé dans notre approche, puis nous motivons notre choix pour ce type de modèle et enfin, nous l'illustrons avec un exemple réel.

5.2.1 Spécification du modèle de processus métier

Comme nous avons déjà vu dans le premier chapitre, il existe plusieurs modèles de représentation, dotés de niveau d'expressivité variés pour représenter les processus métiers. En effet, chaque modèle prend en charge différents type de contraintes et de propriétés, telles que les contraintes d'ordre (*chronologie des activités*), les contraintes temporelles (*délais d'exécution des activités*) ou encore les contraintes transactionnelles liées aux effets de compensation décrivant l'impact des activités sur le monde du client ou celui du fournisseur [37].

Pour notre travail, nous avons choisi les automates d'états finis déterministes (AFD) pour la modélisation et la représentation des processus métiers. Selon cette spécification, un processus métier est défini comme suit.

Définition formelle d'un AFD

Selon [67], un processus métier est défini par le quintuplet $A = (Q, M, R, q_0, F)$, où :

- Q est un ensemble fini d'états ;

- $q_0 \in Q$ est l'état initial de l'automate ;
- $F \subset Q$ est l'ensemble des états finaux (ou terminaux) de l'automate ;
- M est un ensemble fini de symboles (l'alphabet) ;
- R est une fonction de $Q * M$ dans Q , appelée fonction de transition de l'automate. Cette fonction est basée sur les triplets de la forme : (q_1, m, q_2) permettant de transiter d'un état q_1 vers un état q_2 , suite à l'invocation de l'activité m .

Selon cette formalisation, un processus métier est représenté par un automate d'état finis déterministe ou l'ensemble des états expriment les phases par lesquelles passe le processus métier durant l'accomplissement de la logique métier alors que les transitions sont les activités (*opérations ou messages*) à accomplir pour passer d'un état à un autre.

5.2.2 Motivations du choix de modèle

Bien que différents modèles sont proposés pour la représentation des processus métiers, les automates d'états finis déterministes sont considérés comme l'un des modèles les plus utilisés. Les avantages offerts par de tels formalismes sont les suivants :

- Sont des outils simples et populaires ;
- Les AFDs sont des outils formels et dotés d'une assise théorique solide ;
- Permettent la vérification et la validation des modèles ;
- Ils sont largement utilisés dans les systèmes dynamiques et réactifs aussi bien que les systèmes de contrôle.

D'autre part, bien que nous formalisons les processus métiers en tant que AFD, les principes et les techniques proposés restent valables pour d'autres formalismes, tels que les réseaux de Petri, les diagrammes UML, les diagrammes BPMN et les diagrammes EPC. En effet, divers techniques et outils permettent de transformer d'autres modèles en des automates. A titre d'exemple, les techniques exposées dans [72] transforme les réseaux de pétri en AFD. Les techniques [73] transforment les diagrammes UML vers des AFDs, alors que dans [74, 75] les diagrammes BPMN et BPEL sont convertis en AFDs.

5.2.3 Exemple d'un processus métier illustré par les AFD

La figure 5.1 ci-dessous, illustre un exemple simple de processus métier qui peut être exploiter pour la gestion des inscriptions des étudiants en Master. Ce processus est déployé par l'administration de n'importe quel département universitaire, dans le but de traiter les demandes d'inscription à une formation de Master. Dans le modèle schématisé dans la figure ci-dessous, les noms des états (*e.g : Demande déposée, Attestations délivrées, ...*), n'ont pas de signification pour le protocole. Ils expriment simplement des étapes de son déroulement. Par contre, les noms des transitions (*e.g : Envoie documents, Évaluation demande, ...*) représentent les activités réelles invoqués, par l'étudiant demandeur (*client*) ou l'administration (*fournisseur*) et permettent au protocole de passer d'une étape à une autre, lors de sa progression. Comme il est montré dans l'exemple, nous avons utilisé des abréviations pour les noms des activités, pour faciliter et alléger leurs manipulations (*e.g : Envoie documents est abrégée en ED, et Soumettre Recours est abrégée en SR, ...*).

Dans la réalité, ce processus sera invoqué par plusieurs utilisateurs en même temps, ce qui engendre la génération d'un nombre élevé d'instances du même processus qui sont en cours d'exécution simultanément. Dans les applications grand public, telles que les applications de e-gouvernement (*passes-ports, permis de conduire, ...*), les bibliothèques électroniques, e-learning, commerce-électroniques, le nombre des instances peut atteindre

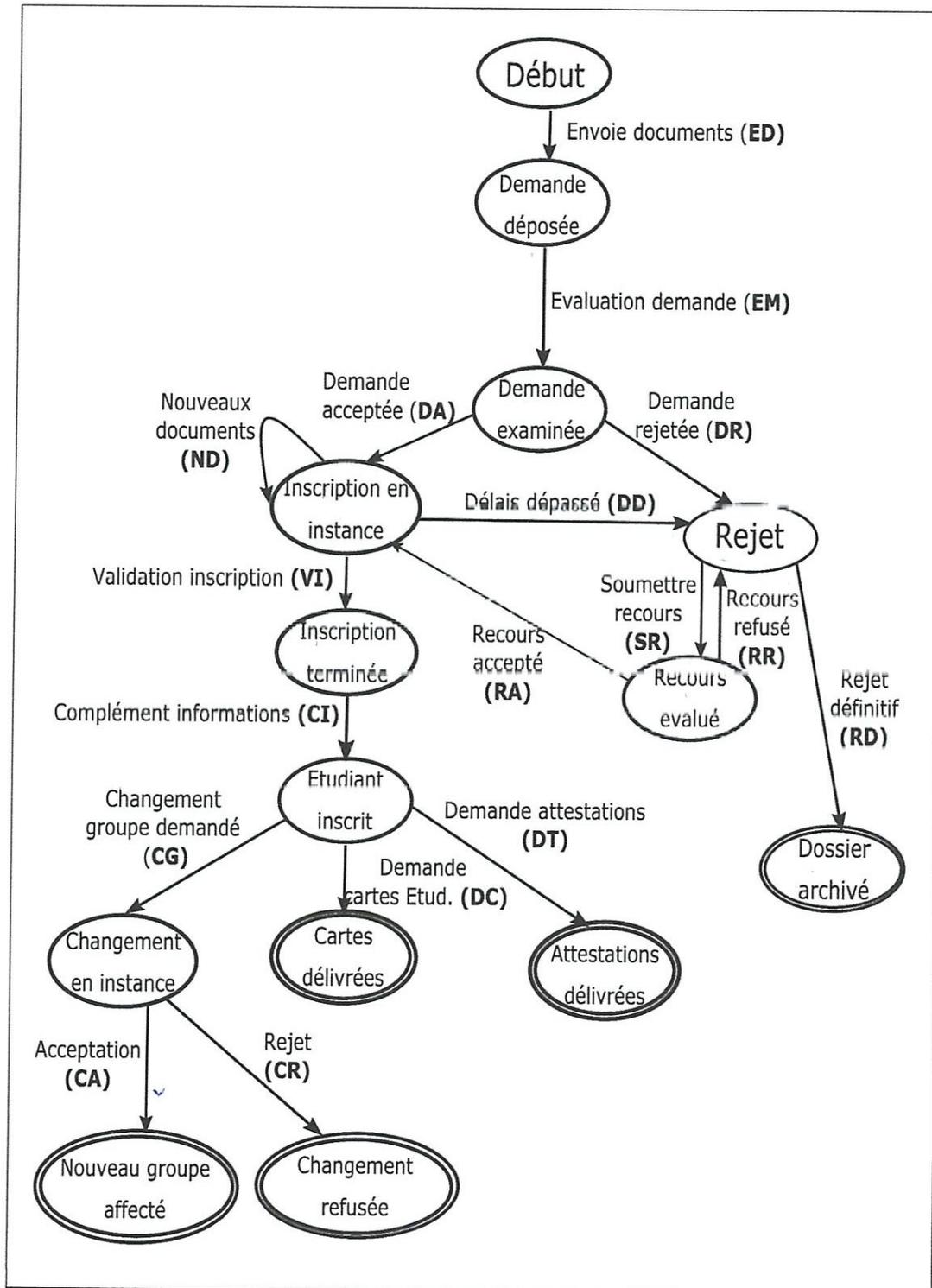


FIGURE 5.1 – Processus d'inscription des étudiants en Master

des millions, où chacune a atteint un état spécifique, suite à l'exécution d'un chemin particulier. Par ailleurs, d'autres instances peuvent avoir terminé leur interactions et se trouve, à cet effet, en situation complète. Les données générées par l'exécutions des instances d'un processus métier sont stockées dans des bases de données adéquates.

Dans ce qui suit, nous exposons l'ensemble des notions et concepts théoriques utiles à la formalisation de notre approche.

5.3 Concepts liés aux processus métiers

Nous introduisons, ci-après les concepts associés à la manipulation des protocoles métiers modélisés sous formes d'AFDs.

5.3.1 Instance d'exécution

Définition 2 Une instance d'exécution I d'un processus métier correspond à une invocation de ce même processus par un utilisateur spécifique. Ce dernier peut être un être humain, une machine ou un autre processus métier.[3, 72]

Comme le processus métier peut être invoqué par un nombre important de clients, à un instant donné t , nous pouvons avoir plusieurs instances dont chacune ayant atteint un niveau d'exécution (état) spécifique. Nous illustrons, ci-après un ensemble d'instances d'exécution relatives au processus métier de la figure 5 1.

Exemple 4 Le processus métier inscription peut être invoqué par les instances suivantes :

- I_1 : exécutée par l'agent Imen a 10h00 le 23-02-2018, et qui a atteint l'état "Inscription terminée".
- I_2 : exécutée par l'agent Mohamed a 17h30 le 23-02-2018, et qui a atteint l'état "Recours évalué".
- I_3 : exécutée par l'agent numéro 1234 a 20h00 le 05-03-2018, et terminée son exécution, elle est a l'état "Attestations délivrées".
- I_4 : exécutée par l'agent numéro 1000 a 14h10 le 05-03-2018, terminée son exécution, elle est a l'état "Dossier archivé".

Les deux instances I_1 et I_2 sont encore actives, par contre I_3 et I_4 sont achevées ou complètes.

5.3.2 Chemins d'exécutions

Étant donné que pour une instance I ayant atteint un état spécifique Q_k , elle doit exécuter la séquence d'activités adéquate, menant de Q_0 vers Q_k . Or, cet état peut être joint par divers chemins possibles offerts par la spécification des processus métiers. Donc, il est impératif d'exprimer d'une manière formelle, cette notion de chemin emprunté ou chemin d'exécution emprunté par une instance.

Définition 3 Un chemin d'exécution c d'un processus métiers \mathcal{P} exprime une alternance entre les états parcourus et les activités exécutées par l'instance I , lors de son interaction avec le processus métier.[72]

Exemple 5 Les chemins d'exécution des instances de l'exemple 4 précédant sont les suivants.

- $c_1 = \text{Début.ED.Demande déposé.EM.Demande examinée.DA.Inscription en instance}$
- $c_2 = \text{Début.ED.Demande déposé.EM.Demande examinée.DR.Rejet.ER.Recours évalué}$
- $c_3 = \text{Début.ED.Demande déposé.EM.Demande examinée.DA.Inscription en instance.VI.Inscription terminée.CI.Étudiant inscrit.DT.Attestations délivrées}$
- $c_4 = \text{Début.ED.Demande déposé.EM.Demande examinée.DR.Rejet.SR.Rejet.RD.Dossier archivé}$

Un chemin est dit **complet** s'il correspond à une instance terminée (*ayant atteint un état final*), le cas contraire, le chemin est dit **incomplet**.

Exemple 6 Prenant par exemple les instances I_1, I_2, I_3, I_4 de l'exemple 4 précédent, les instances I_3 et I_4 sont des instances terminées, donc leurs chemins d'exécution sont complets. Par contre les instances I_1 et I_2 sont encore en cours d'exécution ou instances actives.

5.3.3 Trace et sous trace exécution

Définition 4 Une trace d'exécution $\delta(I)$ d'une instance I correspond à l'historique ou séquence d'activités exécutées par I lors de son interaction avec le processus métier, du début de l'invocation de ce processus métier jusqu'à son état actuel[77].

Exemple 7 Traces des instances

Pour bien illustrer la notion de trace, on prend au hasard quelques traces d'instances, en tenant en compte uniquement les noms des activités. Ces instances sont relatives au processus métier de la figure 5.1.

- $\delta(I_1) : ED.EM.DA^1$
- $\delta(I_2) : ED.EM.DR.RD$
- $\delta(I_3) : ED.EM.DA.VI.CI.DT$
- $\delta(I_4) : ED.EM.DR.SR$
- $\delta(I_5) : ED.EM.DR.SR.RA$
- $\delta(I_6) : ED.EM.DA.ND.ND.VI$
- $\delta(I_7) : ED.EM.DR.SR.RR.SR.RR.RD$
- $\delta(I_8) : ED.EM.DA.ND.DD$
- $\delta(I_9) : ED.EM$

D'autre part, une trace intègre, en plus des données relatives aux activités, d'autre informations relatives aux propriétés d'exécution. On spécifie ces propriétés dans ce qui suit :

- InstID : est l'identificateur de l'instance. C'est un numéro unique attribué par le système.
- User : L'utilisateur ayant invoqué l'instance ;
- Time : Le temps de démarrage de l'instance.
- Time-Stamp : Le temps écoulé depuis le lancement de l'instance jusqu'au moment actuel.
- Sit : désigne la situation de l'instance (*e.g : en cours, bloquée ou achevée*) ;

1. Le symbole "." est utilisé pour exprimer la succession d'activités.

- InstDATA : expriment les données manipulées par l'instance (informations spécifiques à l'occurrence ; i.g : nom, prénom, prix, ...);
- Ressource : Les ressources exigées par l'instance, (*e.g* : matériel exigé, ...);
- Coût : Représente les charges totales induites par l'invocation des différentes activités d'une trace. A noter qu'à chaque activité est associé un montant exprimant les couts des ressources engagées (*heure main d'œuvre, heure machine, frais de transaction,...*).
- Séquence d'activités : représente la séquence d'activités exécutées par l'instance.

Une propriété importante des activités associées aux traces d'exécution est leur durée. En effet, on associe lors de la description des processus métiers une valeur seuil pour l'achèvement de cette activité. Dépasser ce délai l'instance sera bloquée ou annuler. Néanmoins, dans la réalité effective chaque activité prendra un laps de temps qui diffère d'une instance à l'autre. Ainsi, il est impératif de donner les valeurs réelles des temps d'exécution relatif à chaque activité d'une instance donnée. Cette propriété sera utilisée pour calculer certains métriques de gestion (retard, délais, ...).

Le tableau 5.1 montre les durées d'exécution des activités de processus pour trois instances différentes.

Activité → Instance ↓	ED	EM	DA	DR	RD	VI	CI	DT
I_1	5 s	7 s	10 s	-	-	-	-	-
I_2	3 s	4 s	-	12 s	2 s	-	-	-
I_3	10 s	6 s	25 s	-	-	3 s	7 s	15 s

TABLE 5.1 – Durée d'exécution de quelques activités des instances

Une sous-trace $\beta' = st(l,i,\delta)$ de δ est une séquence d'activités de δ qui commence par δ_i , et dont la taille est l . Par exemple, $\beta' = st(3,1,\beta) = ED.EM.DA$.

L'enrichissement de la base des traces de l'exemple 7 par la prise en compte des données supplémentaires est exposé dans le tableau suivant :

Trace	InstID	User	Time	T-εstamp	Sit	InstDATA	Ressource	coût (um^2)
$\delta(I_1)$	100	Imen	14/05/2018 22 :54	15 s	E	Douakha Imen imiid16@gmail.com	Imprimante Internet	5 um
$\delta(I_2)$	110	Rayane	14/05/2018 00 :15	3 s	E	Chaabna Rayane rayan.ch@gmail.com	Scanner	10 um
$\delta(I_3)$	153	Khaled	14/05/2018 14 :15	5 s	A	Chaabna Khaled Khaled97@gmail.com	Imprimante Opérateur	3 um
$\delta(I_4)$	288	Aziza	14/05/2018 10 :20	5 s	B	Chaabna Aziza azza.belg@gmail.com	Opérateur Internet	7 um
$\delta(I_5)$	28	Ali	24/05/2018 03 :20	9 s	A	Khebizi Ali ali.kh@gmail.com	Imprimante Internet	9 um
$\delta(I_6)$	88	Houda	24/05/2018 19 :35	4 s	B	Lachaal Houda Houda.lach@gmail.com	Scanner	17 um
$\delta(I_7)$	115	Hanane	04/08/2018 20 :25	6 s	E	Zitouni Hanane lanane.ztn@gmail.com	Internet	6 um
$\delta(I_8)$	214	Bouchra	17/04/2018 22 :05	2 s	E	Zeghdoudi Bouchra bouchra.zegh@gmail.com	Opérateur Scanner	12 um
$\delta(I_9)$	90	Sara	01/09/2018 23 :10	10 s	A	Boumlit Sara sara.b92@gmail.com	Imprimante	18 um

TABLE 5.2 – Exemple d'instances avec leurs données descriptives

2 um : Unité monétaire.

5.3.4 Expression régulière

Les expressions régulières sont largement utilisées dans la discipline informatique et particulièrement lors de la manipulation des automates. Une expression régulière *Reg* est un moyen formel permettant de définir et de manipuler des ensembles de caractères [78].

Ces ensembles peuvent contenir :

- des suites de caractères définies explicitement,
- des répétitions de suites de caractères,
- des choix entre plusieurs suites de caractères.

Dans notre travail, nous exploitons cette notion pour spécifier la séquence d'activités exécutée lors de l'invocation des processus métiers. Plus formellement, étant donnée un **AFD** représentant un processus métier quelconque, une expression régulière est donnée par les éléments suivants :

- a_i sont les activités appartenant à M ; i.e ; $a_i \in M$
- "." est l'opérateur de concaténation ;
- "*" est l'opérateur star-Kleene [79] ;
- "(" et ")" pour l'application agrégé d'un opérateur.

Avec cette formalisation, une expression régulière peut exprimée la séquence d'activités, i.e ; la trace d'une instance

Exemple 8 *A titre d'illustration , on peut exprimer plusieurs expressions régulières de l'exemple de la figure 5.1.*

$Reg_1 : ED.EM.DR.SR$

$Reg_2 : VI.(EM)^{*}.(CG)^{*}$

$Reg_2 : VI.(CI.CG)^{*}$

D'autre part, le langage reconnu (ou accepté) par un automate (**AFD**) est l'ensemble des mots qui correspondent à un calcul de l'automate partant d'un état initial et s'arrêtant dans un état final.

5.3.5 Notion de sous protocole

Afin de permettre l'analyse et la supervision des processus métiers, nous exposons ci-dessous la notion de sous protocole qui est très utile pour nos développements futurs.

Spécification formelle d'un sous protocole

Soit $P = (Q, q_0, F, M, R)$ un protocole de service, et soit $q \in Q$ un état de P . le sous-protocole $C_{SP}(P, q)$ de P est le protocole : $C_{SP}(P, q) = (Q'_s, q'_{0s}, F'_s, M'_s, R'_s)$, obtenu à partir de P comme suit : [77]

- q est l'état initial de $C_{SP}(P, q)$.
- $Q'_s \subseteq Q$ contient l'ensemble des états de Q accessibles a partir de q en utilisant la relation R de P .
- $F'_s = Q'_s \cap F$.
- $R'_s = (q, q', m) \in R$ s.t $q, q' \subseteq Q'_s$.
- $M'_s \subset M$ est constitué des messages M qui apparaissent dans les transitions de R'_s .

Conformément à cette formalisation, le sous-protocole $C_{SP}(P, q)$ permet de capturer toutes les exécutions possibles de P à partir de l'état q . En fait, il spécifie tous les chemins d'exécution futurs de P qui commencent par l'état q et qui se terminent à l'un des états finaux de P .

Exemple d'un sous-protocole

Exemple 9 A titre d'illustration, prenant l'exemple de la figure 5.1 et dont nous voulons extraire un sous-protocole relative à la prise en charge des demandes de changement de groupe et des demandes des divers documents par les étudiants. Cette procédure décrit les actions incluses dans la logique métier du protocole principal. Le sous protocole associé est illustré ci-dessous :

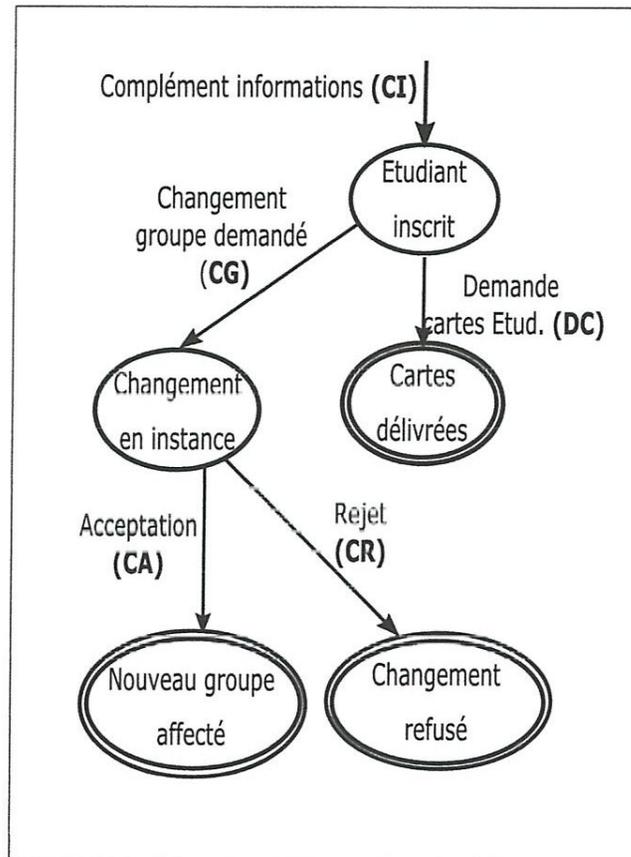


FIGURE 5.2 – Un exemple de sous-protocole

5.3.6 Relation de simulation

Soient $P = (Q, q_0, F, M, R)$ et $P' = (Q', q'_0, F', M', R')$ deux protocoles de service. Un état $q_1 \in Q$ est simulé par un état $q'_1 \in Q'$, noté $q_1 \ll q'_1$, si et seulement si les deux conditions suivantes sont vérifiées : [3]

- $\forall a \in M$ et $\forall q_2 \in Q$ avec : $(q_1, q_2, a) \in R$, il existe une transition $(q'_1, q'_2, a) \in R'$, telle que : $q_2 \ll q'_2$,
- Si $q_1 \in F$, alors $q'_1 \in F'$, On dit que P est simulée par P' , et on note $P \ll P'$, si : $q_0 \ll q'_0$.

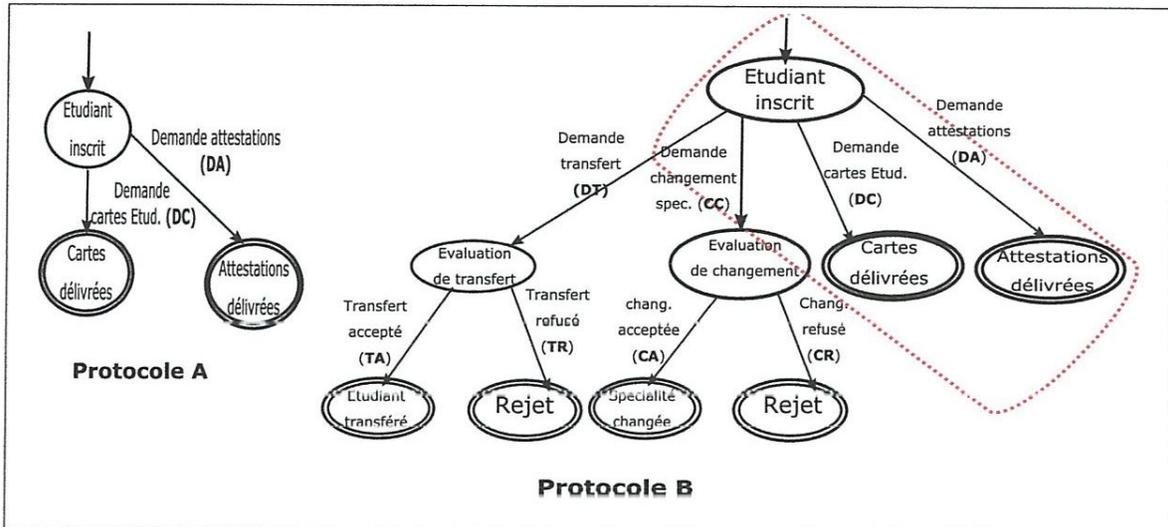


FIGURE 5.3 – Exemple d’une relation de simulation entre deux protocoles

Exemple 10 Dans la figure 5.3 ci-dessus, le **Protocole A** réalise l’opération de délivrance des documents "Cartes et attestations des étudiants". Ce protocole est opérationnel au niveau de l’administration du département. D’une autre côté le **Protocole B** assure plusieurs taches, tels que les traitements des transferts, changement de spécialité, en plus la délivrance des documents aux étudiants. ce dernier protocole B est utilisé au niveau du rectorat de l’université. Le **Protocole B** peut simuler le **Protocole A**, c’est à dire, qu’il peut produire les mêmes exécutions (e.g : quand le protocole A exécute la transition DA le protocole B peut l’exécuter. Aussi, si le Protocole A exécute la transition DC le Protocole B peut l’exécuter). Par contre, le **Protocole A** ne simule pas le **Protocole B**.

5.4 Formalisation de l’approche d’analyse des données

Les objectifs visés par ce projet de fin d’études consistent à proposer une méthode pour l’analyse et la supervision des données des processus métiers.

Dans cette perspective, nous percevons les analyses à réaliser selon deux dimensions **complémentaires**. La première se focalise sur les schémas des protocoles des processus métiers, et la seconde se concentre sur les données d’exécution. La suite de ce chapitre expose ces deux dimensions de manière détaillée.

5.4.1 Analyse des données au niveau modèle

L'analyse des données à ce niveau traite les structures exposées par la description des schémas des processus métiers ou les modèles des processus. On s'intéressera, particulièrement à l'extraction de certains motifs d'intérêt, à la recherche de certains chemins et aussi à la vérification de l'existence de certaines sous-procédures. Ces trois éléments expriment des besoins de gestion spécifiques.

a. Recherche de motifs d'intérêt

Dans le cadre de la gestion du cycle de vie des processus métiers, un gestionnaire de protocole pourra être intéressé par explorer le schéma de protocole à la recherche de certaines situations décrivant des règles métiers ou contraintes particulières. C'est le scénario, par exemple, lorsqu'il veut vérifier *Est-ce que le processus métier permet de faire un sélection répétitive d'articles ?*, ou encore, *Est-ce qu'on peut annuler l'achat après paiement de la marchandise ?*.

Pour prendre en charge de telles préoccupation, nous proposons l'utilisation du concept de **motif** ou **pattern de conception**; concept très répandu dans le domaine de génie logiciel.

Définition d'un motif de recherche

Définition 5 *Un patron de conception ou motif², est un modèle abstrait permettant de représenter des situations génériques, pouvant se manifester de manière récurrente lors de l'exécution des processus métiers [80, 81].*

Conformément à cette définition, un pattern n'est autre qu'un sous-chemin d'un chemin d'exécution complet.

Utilité de la notion de motif de recherche Souvent, le gestionnaire de protocole désire chercher et extraire des comportements spécifiques relatifs à une règle de gestion particulière. Généralement, ces règles concernent : (i) La succession d'activités : *quelle (s) est (sont) la ou les activité(s) qui succède (ent) l'activité X ?*
(ii) L'antériorité : *Une activité X est précédée par quelle (s) activité (s) ?*
(iii) L'itération : *Le processus métier tolère t-il l'exécution répétitive d'une activité quelconque ?*

Exemple d'utilisation des patterns A partir de la fig.5.1, les trois patterns suivants expriment différents besoins de gestion.

- La succession : *Est-ce que le protocole donne à l'étudiant le droit de faire un recours après que son dossier a été refusé ?* Le pattern qui présente cette préoccupation est formalisé par : *DR.SR*;
- L'antériorité : *Pourquoi "l'inscription est en instance" ?* Plusieurs chemins mènent à l'état "inscription en instance", et les patterns correspondant sont : *..ER.RA* ou *..EM.DA* ou *..DA.ND*;
- L'itération : Acceptation de l'inscription sous réserve
Tolérer les dossiers incomplets, mais à conditions de déposer de nouveaux documents complémentaires. Le pattern correspondant à cette situation est : *DA.ND**.

2. En anglais : Pattern

b. Traitement des chemins complets

Au lieu de traiter un simple pattern, tel que exposé précédemment, le gestionnaire de protocoles peut s'intéresser à toute une procédure de gestion, dès le début jusqu'à la fin du processus. Donc, il peut aborder tout un chemin complet, conduisant à un état final de l'automate. Plus formellement, un chemin $c = S_0.a_0.S_1.a_1\dots S_n.a_n.s_{n+1}$, est complet s'il vérifie les deux conditions : (i) : s_0 est un état initial, i.e; $s_0 \in Q_0$ et (ii) : s_{n+1} est un état final, i.e; $s_{n+1} \in F$. Par exemple, dans le protocole de la fig.5.1, le chemin : *Début.ED.Demande déposée.EM.Demande examinée.DR.Rejet.RD.Dossier archivé*, est un chemin complet, qui commence par l'état initial **Début** $\in Q_0$ suivi par une suite alternée de transitions et activités, et qui se termine à l'état final **Dossier archivé** $\in F$.

Utilité de la notion de chemin complet La notion de chemin complet est très utile lors de la supervision des processus métiers. En effet, elle est incontournable pour les raisons suivantes :

- Besoins de substitution : Lorsque l'on désire remplacer une méthode de travail par une autre plus élaborée ou pour des raisons d'amélioration de la logique métier ;
- Pour des raisons d'évolution : Les changements qui surgissent dans l'environnement de l'organisation imposent des modifications de certaines procédures de travail. Par exemples, les changements dans les lois et les réglementations ;
- Maintenance des processus métiers : le processus de supervision exige une opération de contrôle continue sur les instances d'exécution. En cas de constatation d'anomalies ou d'incohérences, le gestionnaire de protocoles intervient pour opérer des actions correctives ;
- Pour des raisons de compatibilité : L'interaction de plusieurs processus métiers dans un cadre de coopération ou de composition de processus [82] impose la mise en conformité des procédures de travail des intervenants.

Exemple d'utilisation En se basant sur l'exemple de la figure 5.1, on peut exprimer les besoins suivants :

- Si le gestionnaire veut substituer le chemin complet :
Début.ED.Demande déposée.EM.Demande examinée.DA.Inscription en instance. VI.Inscription terminée, CI.Étudiant inscrit.DC, carte délivrées,
à fin de simplifier et alléger l'exploitation du processus métier, il peut par exemple, supprimer la transition **Validation inscription VI** et l'activité **Inscription terminée**, tout en reliant directement les activités **Inscription en instance** et **Étudiant inscrit** par la transition **Complément informations CI** ;
- La maintenance et l'évolution : On prend le chemin complet suivant :
Début.ED.Demande déposée.EM.Demande examinée.DR.Rejet.SR.Recours évalué.RR. Rejet.RD.Dossier archivé,
Ce chemin contient une boucle . . . *Rejet.ER.Recours évalué.RR.Rejet . . .*, si le gestionnaire ne pose pas des opérations de contrôle sur cette boucle, le client "**Étudiant**" peut soumettre des recours d'une manière itérative et sans fin. Pour éviter la surcharge du protocole, le gestionnaire peut limiter le nombre de recours par exemple à trois (03) au maximum.
- La compatibilité : Prenant l'exemple d'un étudiant qui veut déposer le dossier de bourse.
La figure 5.4, montre qu'il est impératif d'être inscrit, pour déposer le dossier de

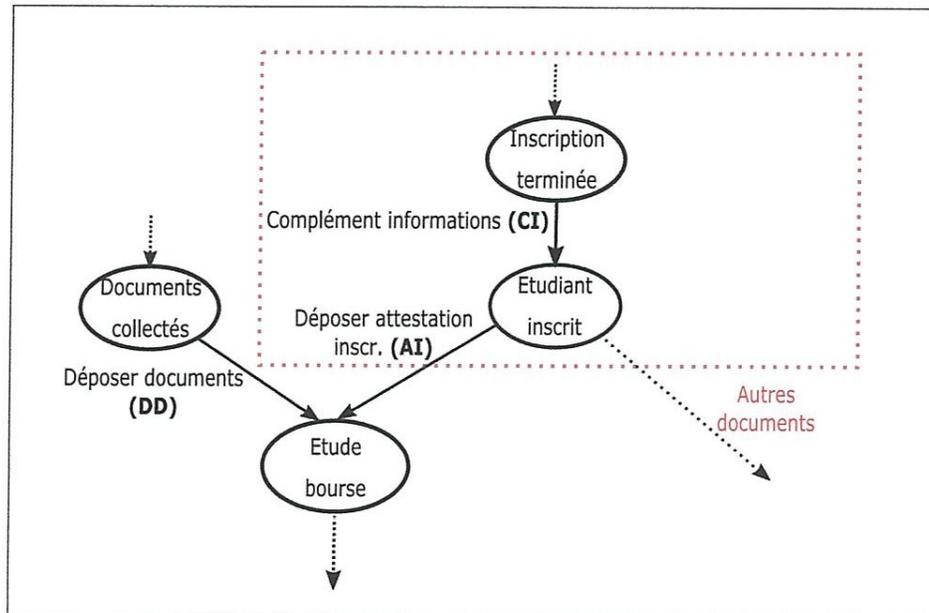


FIGURE 5.4 – Compatibilité entre deux processus

bourse. Le processus de bourse doit être compatible avec le processus d'inscription.

c. Exploitation des sous-protocoles

Dans la notion de sous-protocole, telle que donnée dans la définition 5.3.5 permet de prendre en charge plusieurs contraintes de gestion, en offrant différentes possibilités d'exécution à partir d'un état particulier s_q . Cela, sous entend que le comportement offert à partir de l'état s_q donne plusieurs possibilités d'exécution futures. En ce sens, différents chemins d'exécution sont données à l'utilisateur, à partir de l'état s_q jusqu'à l'un des états finaux de l'automate.

En se référant au sous-protocole de la figure 5.2, les chemins d'exécution possible à partir de l'état sont :

- *Etudiant inscrit.DC.Cartes délivrées ;*
- *Etudiant inscrit.CG.Changement en instance.CA.Nouveau groupe affecté ;*
- *Etudiant inscrit.CG.Changement en instance.CR.Changement refusée.*

Utilité de la notion de sous-protocole

- Intégration de sous-protocoles SP_1 d'un protocole P_1 dans un deuxième protocole P_2 .
- Possibilité de partage de sous-protocoles dans le cadre d'une composition des processus métiers.
- Substitution d'un sous-protocole par un autre qui fait la même tâche en cas de défaillance ou panne.
- évolution des processus métiers par un autre qui offre, au moins, les mêmes fonctionnalités.
- Sélection de sous protocoles : C'est le cas lorsqu'on cherche des protocoles qui satisfont un ensemble de critères de gestion, alors que nous sommes intéressés uniquement par les fonctionnalités offertes par un sous protocole spécifique. Dans

cette situation, il est impératif de se focaliser sur le sous protocole désiré et d'évaluer les critères de qualité du processus métiers afin de sélectionner le meilleur. Par critère de qualité de service (QoS), on sous entend un ensemble de caractéristiques décrivant des propriétés afférentes aux processus métiers, telles que le temps d'exécution, la disponibilité du processus, les performances, la sécurité, . . .

Exemples d'utilisation des sous-protocoles Prenant la fig.5.3, et quelques exemples réels, pour illustrer l'utilité de la notion des sous-protocoles.

- La substitution : Dans l'exemple de la fig.5.3, si dans le **protocole B** le processus qui est responsable à délivrer les documents est tombé en panne, le **protocole A** peut le remplacer.
- Plusieurs raisons, comme les modifications dans les lois et réglementations peuvent induire des changements dans la spécifications des processus métiers. Dans ce cas une nouvelle version du processus métier qui intègre ces changements (*Ajout d'activités, suppression, déplacement, modification,...*) doit être déployée. Souvent une partie seulement du processus métier est affectée par ces changements. De ce fait, la notion de région de changement est exprimée par le concept de sous-protocole. Dans ce contexte, le sous protocole peut servir d'outil très efficace pour la prise en compte des changements et de leurs impacts.
- A titre d'exemple, lors de la recherche d'un processus qui permet d'assurer la réservation d'un voyage qui intègre la réservation du vol, de l'hôtel et la location d'un taxi et enfin la réservation d'un théâtre, on s'intéresse au sous protocole de réservation d'un hôtel. Ce choix est justifié par le critère de sécurité. Dans ce contexte, après réponse à la requête de recherche, plusieurs processus métiers vérifient les critères de recherches introduits. Néanmoins, la sélection va porter sur celle qui satisfait notre critère de sélection. En ce sens, la notion de sous-protocole s'avère très utile. En d'autres termes, on va se baser uniquement sur le sous-protocole relatif à la réservation d'hôtel pour choisir le meilleur protocole à invoquer.

5.4.2 Analyse des données d'exécution

Contrairement aux analyses précédentes qui se basent sur les structures ou schémas des processus métiers, dans cette section on aborde une autre perspective d'analyse articulée autour des données réelles d'exécution des processus métiers et exprimées par les informations contenues dans les traces des instances d'exécution.

La première analyse est purement exploratrice, car elle offre différentes vues aux utilisateurs, cependant la deuxième est basée sur des métriques quantitatives associées aux données d'exécution.

a. Exploitation des données des instances

Dans cette analyse, on propose un ensemble de critères de filtrages des données des instances actives, afin de permettre au gestionnaire de protocoles d'avoir une vision claire, d'estimer, d'évaluer et par conséquent d'orienter sa stratégie de gestion.

A cette fin, nous nous procédons en deux étapes.

1. Nous commençons par identifier un ensemble de critères qui ne sont autre que certains attributs des traces d'exécution et qui vont servir de paramètres utiles à

la sélection des données désirées.

2. Dans un deuxième temps, un modèle général de filtrage, indépendant des structures de données et qui prend en compte les critères précédents est formalisé.

Spécification des critères de filtrage En se basant sur la formalisation des traces d'exécution introduite dans la section 5.3.3, on peut extraire des données répondant à certains critères spécifiques.

En se référant à la définition 5.3.3, parmi les huit propriétés décrivant les traces, nous avons retenu six critères pertinents à la recherche des données des instances d'exécution et qui vont servir d'élément de sélection lors de la formulation des requêtes. Ces critères sont : *User*, *Time*, *Time-stamp*, *Sit*, *Ressource* et *coût*. A noter que ces attributs sont définis dans un domaine de valeurs \mathcal{D} . Par exemple l'attribut *Sit* est défini dans l'ensemble E, A, B ³

Soit δ_i une trace qui a un ensemble d'attributs $A_{i1}, A_{i2} \dots, A_{in}$, sa spécification formelle est la suivante :

$$\delta_i(A_{i1}, A_{i2} \dots A_{in})$$

Soient " v_1, v_2, \dots, v_n " des valeurs associées aux attributs " $A_{i1}, A_{i2} \dots, A_{in}$ ". A présent, nous considérons Σ l'ensemble des traces, tel que :

$$\Sigma = \{\delta_1, \delta_2, \dots, \delta_n\}$$

Cet ensemble de traces est le point de départ de notre approche d'exploration des données. Il va servir de base pour la formulation des requêtes d'interrogation des données.

Une requête d'interrogation des données sur l'ensemble des traces Σ est formalisée comme suit :

$$R_{\Sigma} = \sigma_{(A_{1 \square v_1}) \wedge (A_{2 \square v_2}) \dots \wedge (A_n \square v_n)} \quad (5.1)$$

A noter que ce modèle est inspiré de celui utilisé dans le domaine des bases de données relationnelles. Néanmoins, la différence réside dans le fait que les données persistantes peuvent être stockées dans des base de données non relationnelles, c'est à dire des bases de données orientées graphe. Dans ce cas le filtrage à opérer, tel que exprimé dans l'équation (5.1) reste valable et sera efficient indépendamment du type de base de données utilisé. Ainsi, nous pouvons extraire des données en explorant des base de données orientées graphe.

Dans ce qui suit, nous illustrons l'utilisation du modèle proposé sur une base de donnée orientée graphe.

Les bases de données orientées graphe Les graphes sont la forme la plus générique de stocker des données de manière visuelle dans le monde des structures de données. Les graphes stockent les données sous la forme de nœuds (blocs de données) où un nœud pointe vers un autre. Nous pouvons atteindre n'importe quel bloc de données d'un autre.[83]

3. E : en cours, A : Achevée et B : Bloquée

directement à ce besoin. Les graphes sont naturellement additifs, ce qui signifie que nous pouvons ajouter de nouveaux types de relations, de nouveaux nœuds et de nouveaux sous-graphes à une structure existante sans perturber les requêtes existantes et les fonctionnalités des applications. Ces choses ont généralement des implications positives sur la productivité des développeurs et le risque du projet. En raison de la flexibilité du modèle graphe, nous n'avons pas à modéliser à l'avance notre domaine de façon exhaustive, une pratique qui est tout sauf téméraire face à l'évolution des besoins de l'entreprise. La nature additive des graphiques signifie également que nous avons tendance à effectuer moins de migrations, réduisant ainsi les frais généraux de maintenance et des risques.[83]

- **Agilité :**

Nous voulons être en mesure d'évoluer notre modèle de données en phase avec le reste de notre application, en utilisant une technologie alignée avec les pratiques de livraison de logiciels incrémentales et itératives d'aujourd'hui. Les bases de données orientées graphes modernes nous permettent d'effectuer un développement sans friction et une maintenance des systèmes gracieuse. En particulier, la nature sans schéma du modèle de données graphe, couplée à la nature testable de l'API de la base de données orientée graphe et du langage de requête, nous permet de faire évoluer une application de manière contrôlée.[83]

Exploitation des données des bases orientées graphes avec Cypher Pour exploiter les bases de données graphe, le langage déclaratif **Cypher** servira d'outil clé d'exploration. Dans ces bases de données, **Cypher** joue le même rôle que SQL dans les bases de données relationnelles. En effet, il permet d'interroger en vue d'extraire des données vérifiant certains critères.

Fonctionnement de Cypher Cypher est un langage déclaratif permet d'enquêter et de mettre à jour un graphe. Inspiré du SQL, on y retrouve beaucoup de concepts familiers, comme les clauses : WHERE, ORDER BY, SKIP, LIMIT, ...

Cypher définit un ensemble de fonctions. Parmi les plus courantes on trouve MATCH et WHERE. Ces fonctions sont un peu différentes des mêmes fonctions en SQL. MATCH est utilisé pour décrire le modèle de recherche. Il est principalement basé sur les relations. WHERE est utilisé pour l'ajout de contraintes aux modèles [86].

Cypher définit aussi des fonctions d'écriture, de mise à jour et de suppression des données. CREATE et DELETE sont utilisées pour créer et supprimer les nœuds et les relations. SET et REMOVE sont utilisées pour définir les valeurs des propriétés et affecter aux labels des nœuds. [86]

Exemple 11 *Création de protocole E-Learning*

```
CREATE (ELearning :Protocole {nom : 'E-Learning', desc : 'Processus métier pour un site de e-learning'})
```

```
CREATE (e1 :Etat {nom : 'Début', type : 'initial'})
```

```
CREATE (e2 :Etat {nom : 'Connecté', type : 'simple'})
```

```
CREATE (e3 :Etat {nom : 'SupportChoisi', type : 'simple'})
```

```
CREATE (e4 :Etat {nom : 'NiveauChoisi', type : 'simple'})
```

```
CREATE (e5 :Etat {nom : 'Évalué', type : 'simple'})
```

```
CREATE (e6 :Etat {nom : 'ApprentissageTerminé', type : 'simple'})
```

```
CREATE (e7 :Etat {nom : 'Succes', type : 'final'})
```

```

CREATE (e8 :Etat {nom : 'Echec', type : 'final'})
CREATE (ELearning)-[:CONTIENT]->(e1)
CREATE (ELearning)-[:CONTIENT]->(e2)
CREATE (ELearning)-[:CONTIENT]->(e3)
CREATE (ELearning)-[:CONTIENT]->(e4)
CREATE (ELearning)-[:CONTIENT]->(e5)
CREATE (ELearning)-[:CONTIENT]->(e6)
CREATE (ELearning)-[:CONTIENT]->(e7)
CREATE (ELearning)-[:CONTIENT]->(e8)
CREATE (e1)-[:TRANSITION {nom : 'Loggin'}]->(e2)
CREATE (e2)-[:TRANSITION {nom : 'Choisir Support', abr : 'CS', duree :9 }]->(e3)
CREATE (e3)-[:TRANSITION {nom : 'Niveau et Tuteur', abr : 'NT', duree :15 }]->(e4)
CREATE (e4)-[:TRANSITION {nom : 'Apprentissage', abr : 'AP', duree :25 }]->(e5)
CREATE (e5)-[:TRANSITION {nom : 'Evaluation', abr : 'EL', duree :20 }]->(e6)
CREATE (e6)-[:TRANSITION {nom : 'Note Valide', abr : 'NV', duree :6 }]->(e7)
CREATE (e6)-[:TRANSITION {nom : 'Note Invalide', abr : 'NI', duree :6 }]->(e8)

```

Les bases de données orientées graphe permettent aussi, d'extraire les chemins d'exécution d'un processus spécifique d'une manière simple et rapide, en utilisant des fonctions spécialisées pour extraire les paths, contrairement aux bases de données relationnelles traditionnelles, qui nécessitent un grand nombre de 'jointure' entre les tables, et cela cause une perte de temps, surtout dans le cas des Big data.

La requête Cypher suivante permet d'extraire les chemins d'exécution complets de processus "E-learning".

```

MATCH p = (n :Etat type : 'initial')-[:rs :TRANSITION*]->(m :Etat type : 'final')
WHERE ( :Protocole nom : "E-Learning") -[:CONTIENT]->(n)
RETURN extract(n IN rs | n.nom)

```

La réponse :

```

["Loggin", "ChoisirSupport", "choisirNiveauEtTuteur", "Apprentissage", "Evaluation", "NoteValide"]
["Loggin", "ChoisirSupport", "choisirNiveauEtTuteur", "Apprentissage", "Evaluation", "NoteInvalide"]

```

Exemple 12 Interrogation des données des bases orientées graphes

A titre d'illustration, nous exposons dans ce qui suit quelques exemples d'exploitation des données de la base à partir des données de la table 5.2.

- R_1 : Quelles sont les instances qui ont démarré avant le 23/05/2018 ?

-Requête Cypher correspondante :

```

MATCH (i :Instance)
WHERE i.date='2018-05-23'
RETURN i

```

-La réponse :

```

{"InstID" :100, "User" : "Imen", "Date" : "2018-05-14", "Time" : "22 :54", "TStamp" : "15",
"Sit" : "E", "cout" : 5}
{"InstID" :110, "User" : "Rayane", "Date" : "2018-05-14", "Time" : "00 :15", "TStamp" : "3",
"Sit" : "E", "cout" : 10}

```

```

    {"InstID":153, "User": "Khaled", "Date": "2018-05-14", "Time": "14:15", "TStamp": "6",
    "Sit": "A", "cout": 3}
    {"InstID":288, "User": "Aziza", "Date": "2018-05-14", "Time": "10:20", "TStamp": "7",
    "Sit": "B", "cout": 7}
    {"InstID":214, "User": "Bouchra", "Date": "2018-05-17", "Time": "22:05", "TS-
    tamp": "2", "Sit": "E", "cout": 12}

```

- R_2 : Quelles sont toutes les instances lancées par l'utilisateur "Imen" ?

-Requête Cypher correspondante :

```

MATCH (u :User { prenom : 'Imen' }),(i :Instance)
WHERE id(u)=i.idUser
RETURN i

```

-La réponse :

```

{"InstID":100, "User": "Imen", "Date": "2018-05-14", "Time": "22:54", "TStamp": "15",
"Sit": "E", "cout": 5}

```

- R_3 : Quelles sont les instances qui exigent l'utilisation de l'imprimante ?

-Requête Cypher correspondante :

```

MATCH p=(i :Instance)-[r :ESTCOMPOSE]->(a :Activite)
where a.ressource='imprimante'
return i

```

-La réponse :

```

{"InstID":100, "User": "Imen", "Date": "2018-05-14", "Time": "22:54", "TStamp": "15",
"Sit": "E", "cout": 5}
{"InstID":153, "User": "Khaled", "Date": "2018-05-14", "Time": "14:15", "TStamp": "6",
"Sit": "A", "cout": 3}
{"InstID":28, "User": "Ali", "Date": "2018-05-24", "Time": "03:20", "TStamp": "9",
"Sit": "A", "cout": 9}
{"InstID":90, "User": "Sara", "Date": "2018-09-01", "Time": "23:10", "TStamp": "10",
"Sit": "A", "cout": 18}

```

- R_4 : Le nombre des traces qui sont en cours d'exécution.

-Requête Cypher correspondante :

```

MATCH (i :Instance { sit : 'E' })
RETURN count(i)

```

-La réponse : 4.

- R_5 : Quelle est l'instances qui a une durée minimale ?

-Requête Cypher correspondante :

```

MATCH (i :Instance)-[r :ESTCOMPOSE]->(a :Activite)
RETURN i, sum(a.duree)
ORDER BY sum(a.duree)
LIMIT 1

```

-La réponse :

```

{"InstID":214, "User": "Bouchra", "Date": "2018-05-17", "Time": "22:05", "TS-
tamp": "2", "Sit": "E", "cout": 12}

```

b. Identification et spécification des KPI

Au delà de l'extraction des données brutes, dans cette section nous visons à identifier et à formaliser un ensemble d'indicateurs utiles pour des raisons de performances et d'efficacité de la gestion de toute l'entreprise. On parle d'indicateurs clés de performance ou **KPI**⁷.

Les KPI sont des métriques qui expriment des mesures quantitatives ou qualitatives relatives aux données d'exécution des processus métiers.

Définition Les indicateurs de performance clés KPI définissent un ensemble de valeurs sur lesquelles se base l'organisation pour évaluer son efficacité. Chaque indicateur de performance clé est une valeur mesurable qui montre l'efficacité avec laquelle une entreprise atteint ses objectifs de gestion clés.

Les organisations utilisent des KPI à plusieurs niveaux pour évaluer leur succès dans la réalisation des objectifs escomptés. Les KPI de haut niveau peuvent se concentrer sur la performance globale de l'entreprise, tandis que les KPI de bas niveau peuvent se concentrer sur des processus opérationnels dans des départements, tels que les ventes, le marketing ou fidélisation des clients. Dans notre approche, on aborde uniquement les indicateurs de bas niveau à cause de leur importance et de leur relation directe avec les données opérationnelles générées par les exécutions des processus métiers. D'autre part, on signale que les KPI de haut niveau relève de la dimension stratégique de l'entreprise et sortent de notre contexte d'études. Bien que les indicateurs de haut niveau sont souvent élaborés à partir de ceux de bas niveau.

Du fait que les indicateurs exploitent les données déjà enregistrées, alors il ne peuvent mesurer que ce qui est déjà passé dans l'historique, de sorte que le seul type de mesure soit descriptif et constitue une donnée réelle. Tout KPI qui tente de mesurer quelque chose dans un futur état comme prédictif, diagnostique ou perspectif n'est plus un "**indicateur**" mais un plutôt "**un pronostic**" (éventuellement basée sur un KPI).

Catégories des KPI Il existe deux catégories de mesure pour les KPI

Les faits quantitatifs Ce sont des valeurs objectives qui ne sont soumises à aucune "distorsion des sentiments personnels, des préjugés ou des interprétations présentées avec une valeur spécifique. Ces indicateurs sont objectifs, dans le sens où ils sont numériques et mesurés par rapport à une norme établie.

Valeurs qualitatives Basées sur ou influencées par des appréciations, goûts ou opinions personnels et présentées comme toute valeur numérique ou textuelle qui représente une interprétation de ces éléments.

Dans notre analyse on aborde seulement les KPI de la catégorie quantitative, du fait qu'ils véhiculent des valeurs numériques quantifiables associées aux données d'exécution des processus métiers.

Objectifs des KPI Les KPI permettent de répondre à des besoins de performance au niveau de l'organisation. Ils visent les objectifs suivant :

- Amélioration le rendement de l'entreprise ;

7. Key Performance Indicator

- Réduction des coûts ;
- Améliorer le cycle de vie des processus métiers ;
- Augmentation de la satisfaction de la clientèle ;
- Offrir un outil de gestion et d'aide à la prise de décision.

En d'autres termes, l'utilisation des KPI permettra de réduire les frais généraux, de dépasser les erreurs potentielles, d'éliminer des retards d'exécution ainsi que l'optimisation des coûts.

Dans ce qui suit, nous proposons un ensemble d'indicateurs pertinents dans le cadre de l'exploitation des données d'exécution des processus métiers. Pour chaque indicateur identifié, nous expliquons son intérêt et sa sémantique, nous donnons sa formalisation et nous l'illustrons par un exemple.

Symbole	Nom du KPI
ρ	Pourcentage des instances achevées
λ	Temps moyens d'exécution du processus
α	Temps moyen pour terminer une activité
β	Temps moyen nécessaire à la terminaison d'une instance
γ	Taux de retard des instances achevées

TABLE 5.3 – Un aperçu des KPI proposés

Nous détaillons par la suite chacun des KPI proposés.

1. Pourcentage des instances achevées : ρ

Sémantique du KPI Exprime le taux des instances ayant terminé leur exécution par rapport au nombre total d'instances du processus métier. Il sert à estimer l'efficacité globale du système. On dira qu'un processus métier est **fluide** si son taux d'instances achevées est élevé. Par contre un processus métier, dont le taux ρ est faible ($\leq 10\%$) exprime un problème dans l'exécution du processus, ce qui justifie le fait que les instances démarrent leur exécution mais n'arrivent pas à atteindre des états finaux, et restent en état d'attente de ressources. Cela peut être justifié par des raisons d'inter-blocage ou de lourdeur de la procédure, ce qui engendre un abandon excessif par les utilisateurs.

Syntaxe du KPI

$$\rho = \frac{\Sigma InstA}{\Sigma Instances} \times 100 \quad (5.2)$$

Où :

$$\Sigma Instances = InstA + InstE + InstB$$

Et :

$InstA$: Le nombre des instances achevées ;

$InstE$: Le nombre des instances en cours ;

$InstB$: Le nombre des instances bloquées.

Exemple 13 En utilisant le KPI " ρ " on peut calculer le taux des instances achevées à partir de la table des traces 5.2.

On a :

$$InstA = 3, \bar{InstE} = 4, \bar{InstB} = 2$$

$$\rho = \frac{3}{9} \times 100 = 33,33\%$$

2. Temps moyens d'exécution du processus : λ

Sémantique du KPI Le temps moyen d'exécution du processus est la somme des durées de ses instances exécutées (Achevées, bloquées et en cours) divisé par le nombre total d'instances. Où, la durée d'exécution d'une instance se mesure par le cumul des durées des activités déjà exécutées par cette instance.

Ce KPI permet de déduire si le processus fonction correctement ou si des goulots d'étranglement sont observés. Généralement, lorsque cet indicateur est faible explique une bonne maîtrise et une bonne optimisation des processus métiers de l'organisation. D'un autre point de vue, lorsque que le temps moyen d'exécution est élevé, cela réfère qu'il y a un inter-blocage dans le processus.

Syntaxe du KPI

$$\lambda = \frac{\Sigma DInst}{\Sigma Instances} \quad (5.3)$$

Où :

$DInst$: est la durée des instances et qui vaut la somme des durées des activités de cette instance. Autrement, $DInst = \Sigma activits$ contenues dans cette instance.

Exemple 14 *Considérant l'exemple de la table des traces 5.2 contenant les données d'exécution d'un processus métier. On peut alors calculer son indicateur λ comme suit :*

$$DInst = 60s$$

$$\lambda = \frac{60s}{9} \approx 6,66s$$

3. Temps moyen pour terminer une activité : α

Sémantique du KPI Cette mesure sert à estimer la durée moyenne nécessaire pour qu'une activité termine son exécution. Sa formule est donnée par le rapport entre la somme des durées de toutes les occurrences de l'activité lors de l'exécution de processus et le nombre total de ces occurrences.

Syntaxe du KPI

$$\alpha = \frac{\Sigma D_a}{Occur_a} \quad (5.4)$$

Où :

D_a : La durée de l'activité ;
 $Occur_a$: Le nombre des occurrences de l'activité.

Exemple 15 *A titre d'illustration, on calcule le temps moyen de l'activité "ED" de la table 5.1, en supposant que cette activité a été exécutée seulement par les trois traces "I₁, I₂ et I₃".*

On a :

$$D_{ED_{I_1}} = 5s, D_{ED_{I_2}} = 3s, D_{ED_{I_3}} = 10s.$$

Alors :

$$\alpha = \frac{\sum D_{ED}}{3} = \frac{5s + 3s + 10s}{3} = \frac{18}{3} = 6s.$$

4. Temps moyen nécessaire à la terminaison d'une trace : β

Sémantique du KPI Ce paramètre permet de mesurer le temps moyen nécessaire à l'achèvement des instances. Il se base sur le cumul des temps des activités incluses dans les traces des instances. Il reflète des goulots d'étranglements existants dans les nœuds d'un processus métiers, soit à cause du manque de ressources utiles à l'avancement d'un état vers autre, soit à des situations d'inter-blocage.

Syntaxe du KPI

$$\beta = \sum_{i=1}^n (I_{\alpha_i}) \quad (5.5)$$

Où :

I_{α_i} : La durée de l'activité i de l'instance I .

Exemple 16 *A titre d'exemple, la trace de l'instance $\delta(I_1)$ du tableau 5.1, est le cumul des durées d'exécution des activités ED, EM et DA.*

Alors :

$$\beta = \sum_{i=1}^3 (I_{\alpha_i}) = I_{ED} + I_{EM} + I_{DA} + 7s + 5s + 10s = 22s$$

5. Taux de retard des instances achevées : γ

Sémantique du KPI Le taux de retard des instances achevées est le rapport entre la durée prévue et la durée réelle pour terminer son exécution. C'est à dire l'instance nécessite X heures, alors que sur 100 instances, 40% terminent leurs exécution à X+2 heures.

Ce KPI permet de détecter les erreurs et de prévoir s'il y a des blocages lors de l'exécution. Par exemple, après une analyse des fichiers log, on trouve que les 40% des instances utilisent la même ressource, ce qui conduit à faire un retard.

Syntaxe du KPI

$$\gamma = \frac{\sum_{i=1}^n T_{I_i}}{\sum I_i} \quad (5.6)$$

Où :

$$T_{I_i} = \frac{\sum_{j=1}^m I_{p_{\alpha_j}}}{\sum_{j=1}^m I_{r_{\alpha_j}}} * 100$$

Et :

$I_{p_{ai}}$: La durée prévue de l'activité i de l'instance I .
 $I_{r_{ai}}$: La durée réelle de l'activité i de l'instance I .

Exemple 17 On prend par exemple les 3 trace $\delta(I_1)$, $\delta(I_2)$ et $\delta(I_3)$ du tableau 5.1. Ce tableau contient les durées réelles d'exécution de chaque activité par les instances I_1, I_2 et I_3
 Et supposons que la durée prévue d'exécution des activités est :

$ED = 3s$, $EM = 4s$, $DA = 8s$, $DR = 9s$, $RD = 2s$, $VI = 3s$, $CI = 5s$ et $DT = 10s$.

Alors :

La durée réelle de terminaison des instances :

$$I_1 : \sum_{i=1}^3 I_{r_{ai}} = 5 + 7 + 10 = 22s.$$

$$I_2 : \sum_{i=1}^4 I_{r_{ai}} = 3 + 4 + 12 + 2 = 21s.$$

$$I_3 : \sum_{i=1}^6 I_{r_{ai}} = 10 + 6 + 25 + 3 + 7 + 15 = 66s.$$

La durée prévue de terminaison des instances :

$$I_1 : \sum_{i=1}^3 I_{p_{ai}} = 3 + 4 + 8 = 15s.$$

$$I_2 : \sum_{i=1}^4 I_{p_{ai}} = 3 + 4 + 9 + 2 = 18s.$$

$$I_3 : \sum_{i=1}^6 I_{p_{ai}} = 3 + 4 + 8 + 3 + 5 + 10 = 33s.$$

Le taux de retard de chaque instance :

$$T_{I_1} = \frac{15}{22} * 100 = 68\%$$

$$T_{I_2} = \frac{18}{21} * 100 = 86\%$$

$$T_{I_3} = \frac{33}{66} * 100 = 50\%$$

$$\Rightarrow \gamma = \frac{68+86+50}{3} = 68\%$$

5.5 Conclusion

Dans ce chapitre, nous avons exposé notre approche d'analyse et de supervision des données des processus métiers. Nous avons commencé par présenter les modèles formels sous-jacents notre approche, tout en exposant les concepts et notions utiles à nos développements.

Une fois ces ingrédients bien maîtrisés, notre approche d'analyse porte sur trois dimensions complémentaires, à savoir :

- Identification de trois types de structures à rechercher dans les définitions des processus métiers : les patterns, les chemins et les sous-protocoles et illustration de leur intérêt.
- Spécification d'une technique pour l'interrogation des données des processus métiers. Elle est générique dans le sens où elle est indépendante du type de modèle de données manipulé (structurée, semi-structurée,...)
- Identification et formalisation d'un ensemble d'indicateurs clé de performances, utile à la supervision des processus métiers. chaque indicateur a été formalisé, explicité et son usage à été montré avec un exemple.

Une fois ces éléments bien cernés, nous avons abordé l'architecture et le fonctionnement du système à implémenter. Dans ce sens, les différents composants ont été détaillés et leur interactions ont été établies. Par ailleurs, les structures des bases de données autour desquelles s'articule le système ont été conçues.

Dans le prochain chapitre, on présente la réalisation d'un prototype logiciel qui implémente l'approche proposée.

Implémentation et expérimentation de l'approche

6.1 Introduction

Après avoir terminé l'étape de formalisation de notre approche, on aborde dans ce chapitre la phase d'implémentation qui constitue le dernier volet de ce rapport et qui a pour objectifs de mettre en œuvre notre approche. Pour ce faire, on va commencer tout d'abord par préciser l'environnement et les outils de développement du système. Ensuite, on va présenter l'architecture générale de notre approche. Et en fin les interfaces et les fonctionnalités du notre prototype réalisé.

6.2 Présentation de l'environnement de travail

Durant l'implémentation de notre approche on a utilisé Neo4j comme un système de gestion de base de données graphe, qui utilise Cypher comme un langage de requête et de parcours de graphe. Nous avons choisi Java comme langage de programmation sous l'environnement de développement Eclipse. D'autres bibliothèques et logiciels aussi, tels que : Bootstrap, Fontawsome et GoJs. Nous exposons par la suite les outils les plus importants pour la réalisation de notre application.

6.2.1 Les APIs JAVA

Java [88] est un langage de programmation et une plate-forme informatique qui ont été créés par Sun Microsystems, est un langage orienté objet très connu et populaire. Nous avons réalisé notre projet avec Java 8, et à l'aide d'autres bibliothèques Java, comme : Gson [89] utilisée pour convertir des objets Java en leur représentation JSON. Il peut également être utilisé pour convertir une chaîne JSON en un objet Java équivalent, SparkJava [90] est un framework d'application web libre et open-source et Neo4j JDBC Driver [91] qui est constitué d'un ensemble d'interfaces et de classes qui permettent l'accès, à partir de programmes Java, à des données d'une base de données Neo4j.

6.2.2 Neo4j

Neo4j est une base de données de graphes native NoSQL open-source codé par Java et Scala, il fournit un back-end transactionnel ACID¹-compatible pour les applications. Il a été développé à partir des années 2003, et il est disponible publiquement depuis 2007. Ceci en fait l'une des premières bases de données orientées graphes, mais aussi l'une des plus évoluées et robustes. Il est utilisé aujourd'hui par des milliers d'entreprises

1. Atomicité, Cohérence, Isolation et Durabilité, ces composants garantissent qu'une transaction de base de données est terminée en temps opportun.

et d'organisations dans presque toutes les industries, y compris les services financiers, le gouvernement, l'énergie, la technologie, la vente au détail et la fabrication.[92] Neo4j utilise un langage de requête graphe déclaratif, simple et efficace : **Cypher** vu au chapitre précédent paragraphe 5.4.2.

Des connecteurs clients sont disponibles pour différents langages : Java, Python, .NET, ... [93]. Nous utilisons l'API correspond au Java, pour connecter Java avec Neo4j [94].

Les éléments bases de Neo4j

La figure ci-dessous fig.6.1 illustre les éléments de base de la base de données Neo4j.

-**Les nœuds** sont les principaux éléments de données, connectés entre eux via des relations qui peuvent être une ou plusieurs, et même récursives. Elles peuvent avoir une ou plusieurs propriétés (c'est-à-dire, des attributs stockés en tant que paires clé / valeur). Elles ont une ou plusieurs étiquettes qui décrivent leur rôle dans le graphique.

-**Les relations** connectent deux nœuds. Elles peuvent avoir une ou plusieurs propriétés. Et elles sont directionnelles.[95]



FIGURE 6.1 – Modèle de graphe de propriété étiqueté

Le navigateur Neo4j

Le navigateur Neo4j est un client web accessible. Il est parfait pour exécuter des requêtes graphiques ad-hoc, avec juste assez de possibilités pour prototyper une application basée sur Neo4j [96]. La figure 6.2 illustre un navigateur Neo4j.

- Le champs numéro "1" est l'espace de la saisie des requêtes Cypher ;
- Le champs numéro "2" est l'espace d'affichage du résultat de la requête qu'il peut être afficher sous forme de graphe, de tableau, de texte ou de un code.

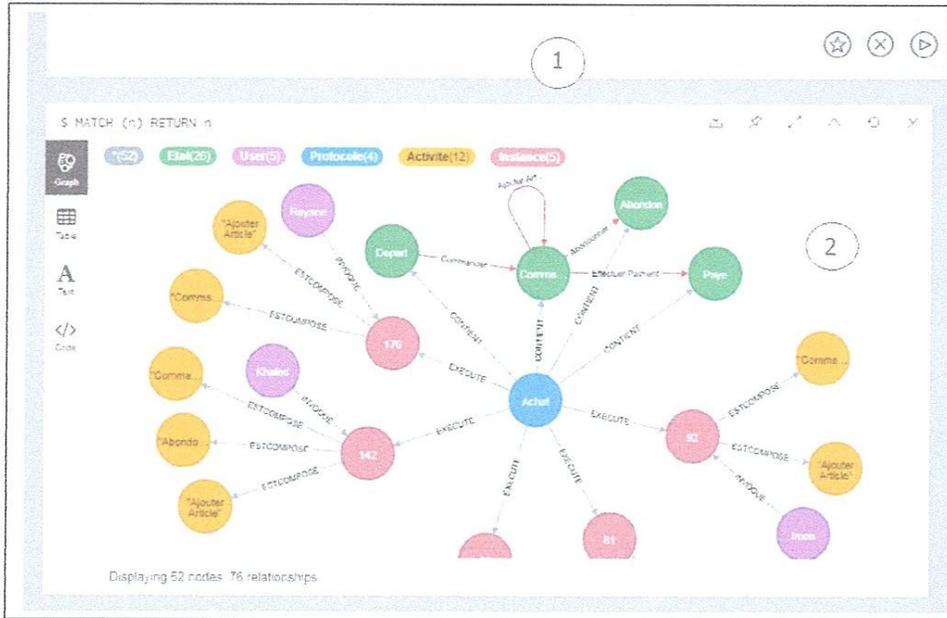


FIGURE 6.2 – Le navigateur Neo4j

6.2.3 GoJs

GoJS est une bibliothèque JavaScript riche en fonctionnalités permettant d'implémenter des diagrammes interactifs personnalisés et des visualisations complexes sur les navigateurs Web et les plates-formes modernes. GoJS facilite la construction de diagrammes JavaScript de nœuds complexes, de liens et de groupes avec des modèles et des mises en page personnalisables.

GoJS offre de nombreuses fonctionnalités avancées pour l'interactivité utilisateur telles que glisser-déposer, copier-coller, édition de texte sur place, info-bulles, menus contextuels, mises en page automatiques, modèles, liaison de données et modèles, gestion transactionnelle des états et des palettes, palettes, des vues d'ensemble, des gestionnaires d'événements, des commandes et un système d'outils extensible pour les opérations personnalisées. [92]

La figure 6.3 illustre un processus métier "Achat" dessiné par GoJs.

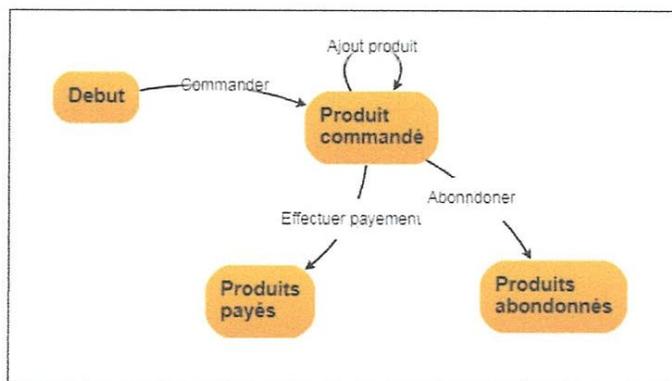


FIGURE 6.3 – Exemple de processus métier dessiné par GoJs

6.3 Architecture et fonctionnalités du système

L'intégration des différentes contributions proposées dans le cadre de notre approche à conduit à l'élaboration d'un système complet pour l'analyse et la supervision des données des processus métiers. Nous décrivons dans cette section l'architecture du système et ses fonctionnalités.

6.3.1 Architecture du système

Nous proposons une organisation modulaire des modules d'analyse et de supervision proposés. Ces composants s'articulent autour d'une base de données stockant les informations des processus métiers. Bien évidemment, pour réaliser les objectifs attendus, les différents éléments sont en interaction, d'une part entre eux, et d'autre part avec la bases de données.

La fig.6.4, illustre l'architecture du système ainsi que les interactions entre les différents composants.

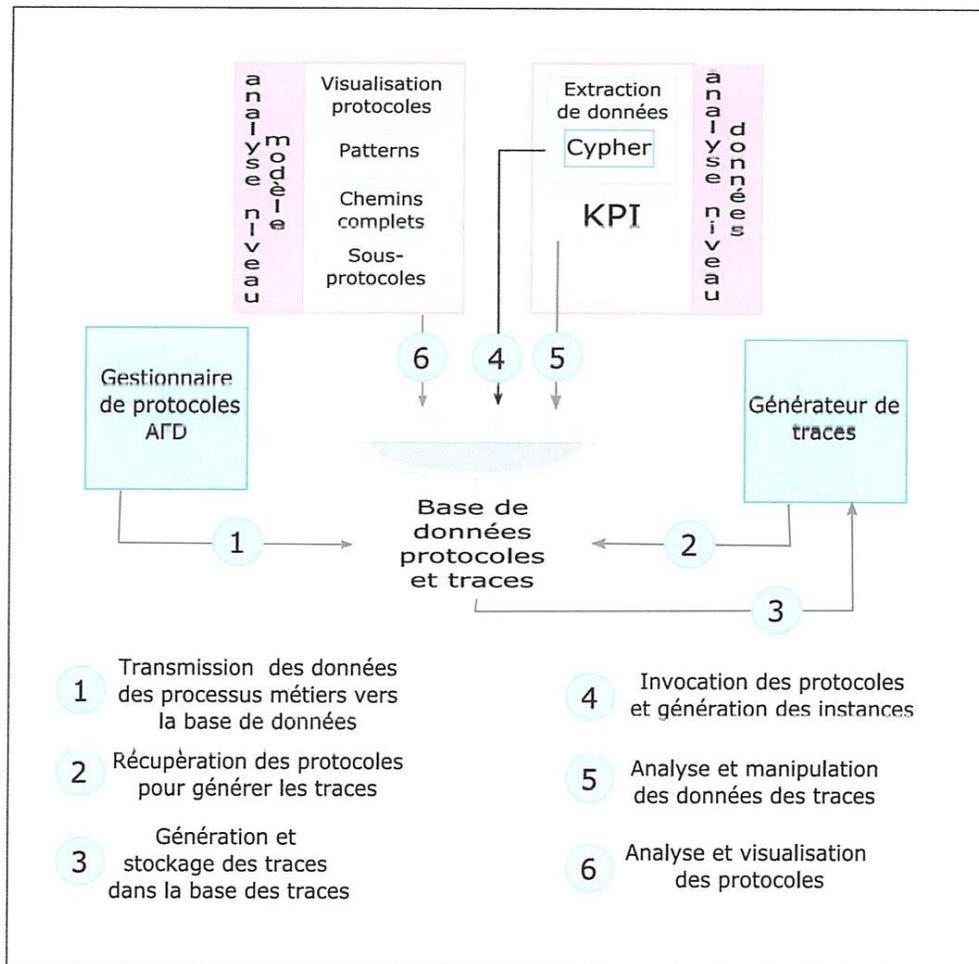


FIGURE 6.4 – Architecture du système

6.3.2 Fonctionnalités du système

Le système est composé de deux modules responsables a générer les protocoles et les traces, deux modules d'analyse sur les deux niveaux : modèle et données, et finalement une base de données stocke les protocoles et les traces générés par les générateurs. Les cinq modules sont liés entre eux par des fonctions numérotées de 1 à 6.

Dans ce qui suit, nous allons expliquer les fonctionnalités de chaque module.

Base de données protocoles et traces

La base de données utilisée est une base orientée graphe comme nous avons déjà dit. Comme les protocoles sont sous forme de graphes (c'est-à dire les données sont de type graphe), alors la meilleur façon pour les stocker et les manipuler sont les bases de données orientées graphes, et c'est le critère principal pour notre choix. La spécification des protocoles est sous forme de requête Cypher, décrivant les états par des nœuds et les transitions entre les états par des relations, ce nœuds et relations contient toutes les informations nécessaires du protocole. A cause de sa simplicité et rapidité, la base sert aussi à stocker et sauvegarder toutes les traces, activités, . . . et leurs données générées lors d'exécution des protocoles.

Gestionnaire de protocole AFD

Ce module est le point de départ du système d'analyse de données. Il donne la possibilité à l'utilisateur de créer son propre processus métier d'une manière graphique et interactive. A cet effet, l'utilisateur doit d'abord, entrer le nom et la description de processus, comme montrer dans la figure 6.7.

Ensuite il modélise son processus avec un outils. Une fois, achevée, la structure décrite sera par la suite passé à l'étape de validation, qui a pour but de détecter les erreurs (processus qui n'a pas un état initial ou qui contient un état isolé). Il affiche les erreurs s'il les trouve, sinon il passe à l'étape de sauvegarde et stock le processus dans la base de données. La figure 6.8, illustre les étapes expliquées précédemment.

Ce module permet aussi de visualiser graphiquement les protocoles pour faciliter la manipulation.

Générateur de traces

Le générateur de traces est un outil qui donne à l'utilisateur d'application le droit de choisir n'importe quel processus stocké dans la base et générer le nombre de traces qu'il veut. Les traces seront générées aléatoirement, stockées dans la base de données et affichées sur l'écran.

Analyse niveau modèle

Dans notre travail, nous allons analyser les protocoles du système par l'utilisation des techniques d'analyse basées sur les patterns, les chemins complets et les sous-protocoles, tels que exposés précédemment. Ces techniques seront exploitées dans la phase d'implémentation, afin d'extraire des éléments descriptifs des processus métiers.

Analyse niveau données

L'analyse au niveau données sert à extraire les informations souhaitées à partir des traces stockées dans les fichiers logs (stockés dans la base de données). En plus de la visualisation et de filtrage des données. Nous avons choisi la technique des KPIs ou indicateur clés de performance. A l'aide de ces métriques, le gestionnaire peut savoir l'état de son système. Par exemple, "Le nombre des utilisateurs qui ont commandé des articles, mais n'ont pas payé?", si le nombre des utilisateurs est élevé, il va déduire qu'il y a une panne quelque part, et la réparer par la suite.

6.3.3 Modélisation des structures de données

Nous nous basons sur le modèle Nœud/Relation pour modéliser les données manipulées par les processus métiers. Ce modèle de graphe de propriétés étiqueté va sembler familier, si vous avez déjà travaillé avec un diagramme de relation d'entité.

Les nœuds sont les entités du graphe. Ils peuvent contenir n'importe quel nombre d'attributs (paire clé-valeur) appelés propriétés. Les nœuds peuvent être étiquetés avec des étiquettes représentant leurs différents rôles dans notre domaine. En plus de contextualiser les propriétés de nœud et de relation, les étiquettes peuvent également servir à attacher des informations d'index ou de contrainte de méta-données à certains nœuds.[87]

Les relations fournissent des connexions orientées, nommées et pertinentes sur le plan sémantique entre deux entités de nœud (par exemple, le protocole CONTIENT un état). Une relation a toujours une direction, un type, un nœud de début et un nœud final. Comme les nœuds, les relations peuvent également avoir des propriétés.[87]

La modélisation de la bases de données protocoles et traces est décrite dans ce qui suit.

Schéma de la base de données

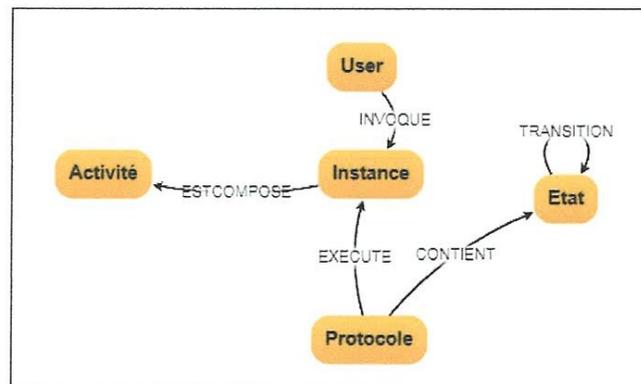


FIGURE 6.5 – Schéma de la base de données protocoles et traces

6.4 Scénario d'utilisation de Business Process Analyser

Dans ce qui suit, nous présentons un scénario d'utilisation réel de notre système. Les différentes étapes sont détaillées ci-dessous :

6.4.1 Gestionnaire des protocoles

Dans cette section l'utilisateur peut visualiser un protocole existant, ou aussi créer un nouveau.

L'entrée à notre système est montrée dans la figure 6.6. L'utilisateur au début il peut choisir de créer un nouveau protocole ou bien de visualiser un protocole déjà existant.

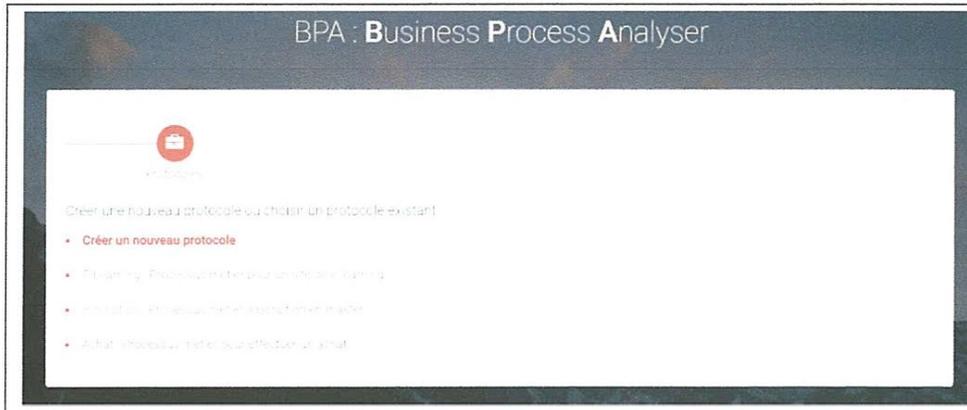


FIGURE 6.6 – La page d'entrée au système

Les deux fonctionnalités de gestionnaire de protocole sont bien expliquées dans ce qui suit.

La création et la validation d'un nouveau processus métier

La première étape : L'utilisateur peut créer un nouveau protocole en cliquant sur "Créer nouveau protocole" comme montrer dans la figure 6.6. La fenêtre illustrer dans la figure 6.7 est l'étape qui permet de nommer et donner la description au protocole.

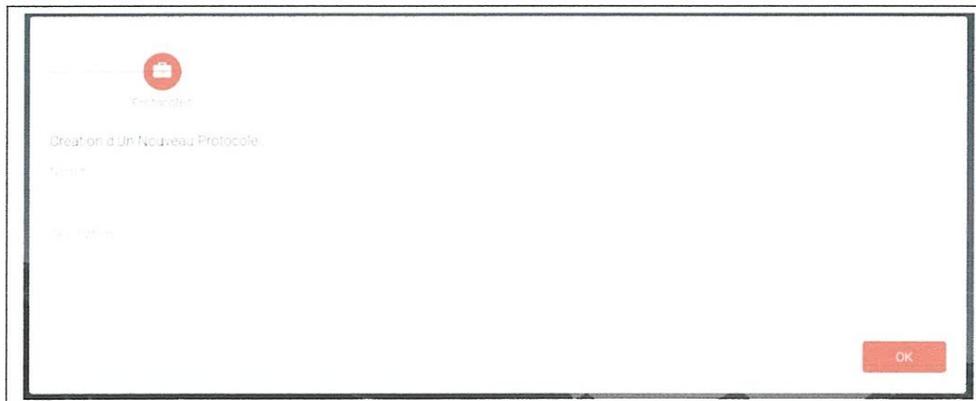


FIGURE 6.7 – Formulaire de création de nouveau processus

La deuxième étape est la création du protocole graphiquement. Il peut aussi spécifier toutes les information liées à chaque transition (le nom, la durée prévue et les ressources exigées). Le scénario est bien détaillé dans la figure 6.8 ci-dessous.

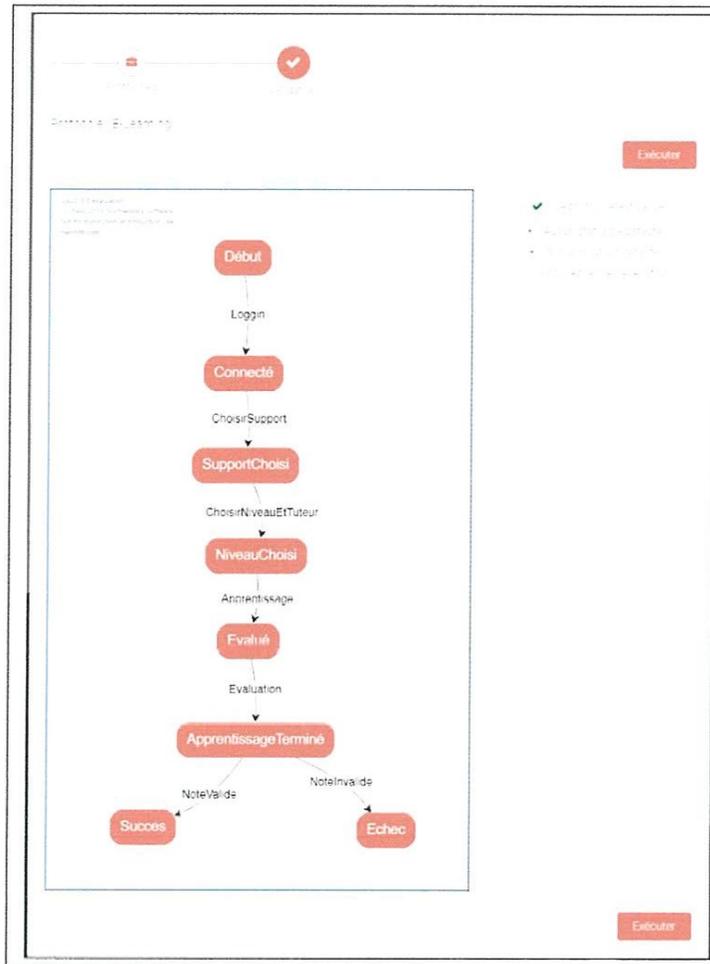


FIGURE 6.8 – Création et validation d'un nouveau processus

Visualisation des protocoles

Le système permet de visualiser un protocole, en choisissant son nom dans la panneau fig.6.6. Cette action va exposer le protocole graphiquement. Il donne aussi, le droit pour modifier puis sauvegarder le protocole comme illustrer dans la figure 6.9 suivante.

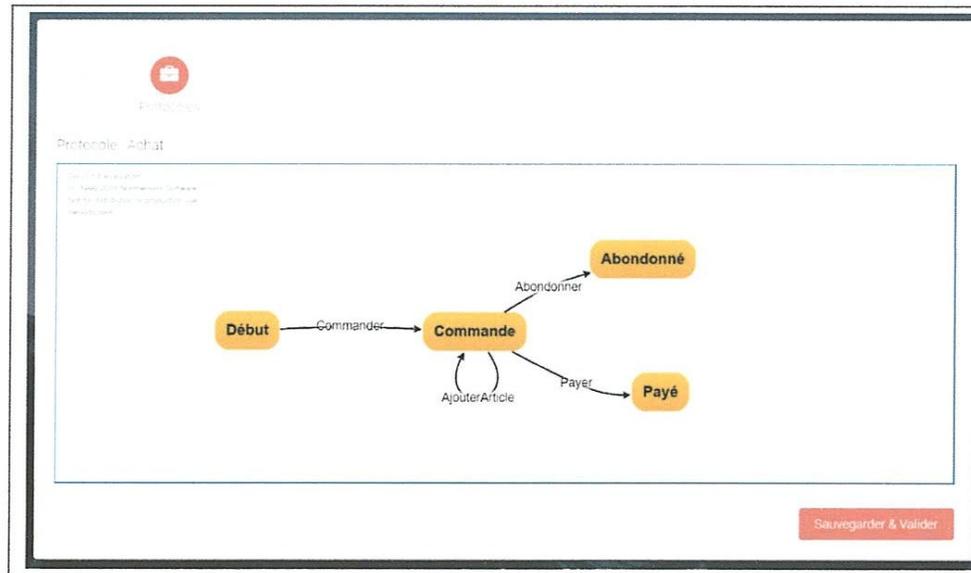


FIGURE 6.9 – Visualisation et modification des protocoles

6.4.2 Générateur de trace

Le protocole a été soit choisi au début parmi les protocoles existant, soit il est un nouveau protocole crée. L'utilisateur ici, entre le nombre des instance qu'il veut dans le champs de saisi, puis il lance la génération. La figure 6.10 montre les étapes de génération.

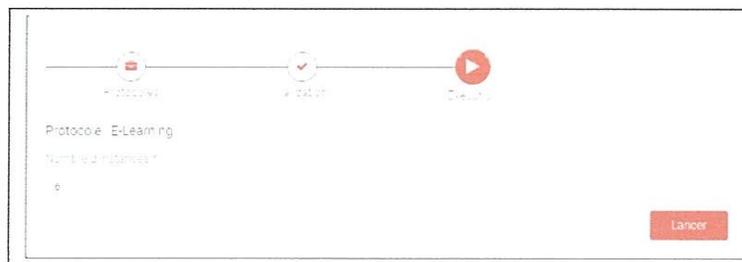


FIGURE 6.10 – Génération des instances

Les instances générées seront stockées dans la base de données. La figure 6.11 montre la base de données après la génération des instances.

Les nœuds en rose sont les instances. La génération des instances génère aussi toutes ces relations avec les autres nœuds.

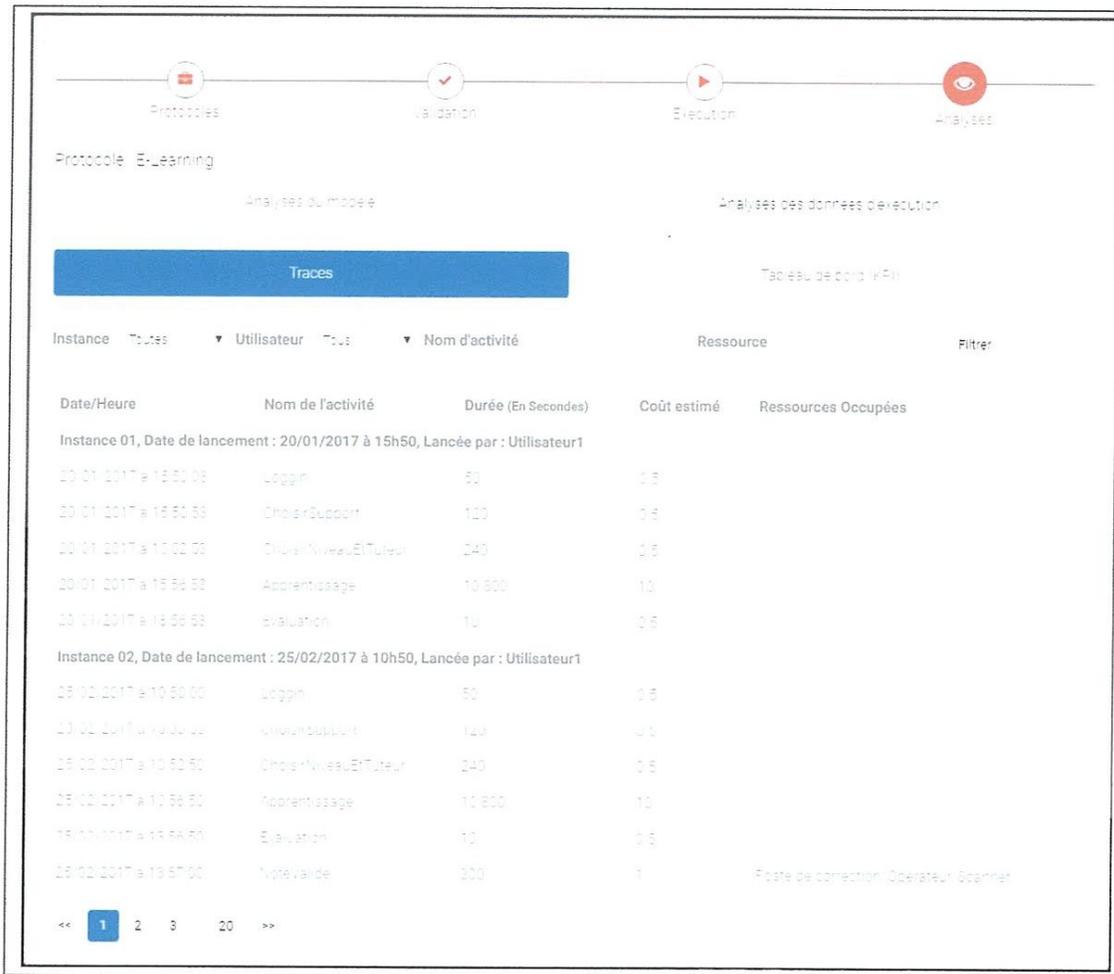


FIGURE 6.12 – Module d'analyse des données

Analyses niveau modèle

Cette partie est divisées en trois : rechercher des motifs d'intérêt, traitement des chemins complets et exploitation des sous-protocoles.

La figure 6.13 illustre l'extraction des chemins complets à partir du protocole. Le chemin est surligné en bleu. La liste des chemins complets de processus est à droite.

Ce module d'analyse permet aussi de rechercher aux patterns d'antériorité, succession et d'itération de n'importe quel état de protocole.

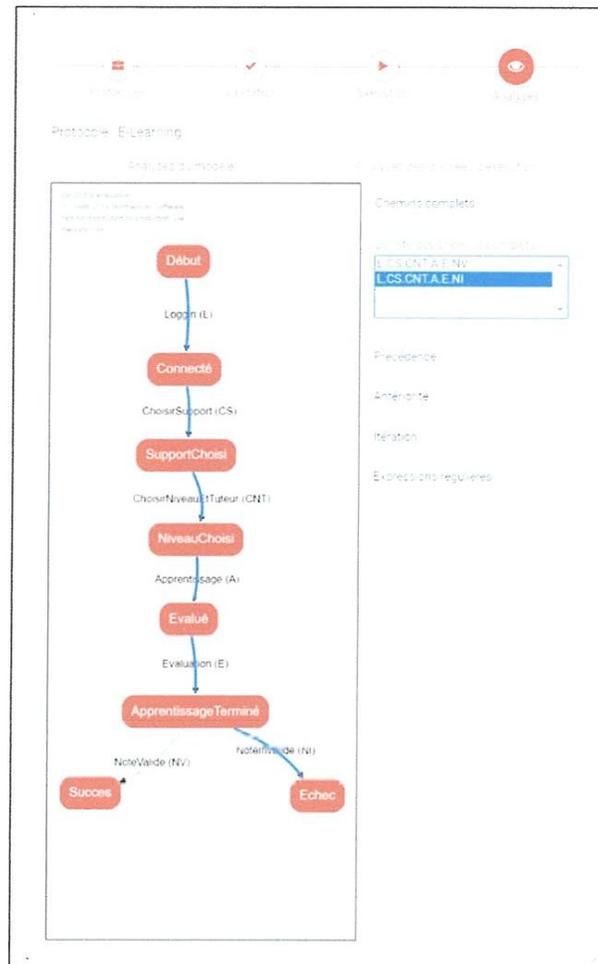


FIGURE 6.13 Traitement d'un chemin complet

6.5 Conclusion

Dans ce dernier chapitre de notre projet, nous avons implémenté notre prototype qui répond aux objectifs fixés au début. Notre prototype BPAnalyser prend en compte tous les aspects conceptuels et formels proposés dans le cadre de notre approche (modélisation du processus métier, génération et stockage des données dans la base de données graphe et l'interrogation et la mise à jour des données par les opérateurs et les primitives proposées). Malgré que les principales fonctionnalités sont réalisées et comme notre application est en première version, il nous reste quelques choses à faire, afin qu'elle soit optimale.

Conclusion générale

Les processus métiers sont une technologie émergente qui prend de plus en plus de l'ampleur et de l'intérêt au niveau des organisations. Dans ce projet de fin d'études de master, nous avons abordés le problème de l'analyse et de la supervision des données des processus métiers.

Après la phase d'analyse l'état de l'art et de maîtrise du domaine des processus métiers, nous avons fait ressortir le déficit flagrant en ce qui concerne le manque d'approches et d'outils adéquats pour l'exploitation efficace et rentable de fichiers logs enregistrant toutes les transactions associées aux instances exécutant les processus métiers.

Dans un deuxième temps, nous avons proposé une approche pour l'extraction des données des processus métiers. A ce niveau, nous avons commencé par exposer les outils formels utiles à la modélisation de notre approche, puis nous avons exposer une technique pour la supervision et l'analyse des données des processus métiers. Notre contribution porte sur plusieurs dimensions, à savoir :

- formalisation d'une technique d'analyse des données au niveau schéma des processus métiers. Elle permet d'extraire des structures d'intérêt.
- proposition d'une approche pour la visualisation des données des processus métiers, indépendamment de la nature de la base de données (*structurée vs semi-structurée*).
- identification et spécification d'un ensemble d'indicateurs clés de performance intégré dans un tableau de bord pour la supervision des données des processus métiers.
- Élaboration de la structure du système à implémenter, tout en spécifiant la modélisation des bases de données manipulées.
- Implémentation de l'approche proposée dans un environnement de programmation.

Bien que le prototype logiciel réalisé reste primitif et exige des améliorations, nous pouvons affirmer que les premiers résultats d'expérimentation sont très satisfaisants. En effet, lors de la phase d'expérimentation, nous avons trouvé des difficultés pour acquérir une base de données réelles contenant des fichiers logs réels. Ce manque nous a conduit à opérer par simulation des traces d'exécution générées par le système que nous avons réalisé.

Au termes de ce projet, nous pouvons affirmer, sans risque de se tromper, que les acquis capitalisés sont nombreux et portent sur plusieurs dimensions. Les plus importantes sont les suivantes :

- Découverte d'un nouveau domaine de recherche qui est celui des processus métiers. Il constitue la nouvelle orientation des systèmes d'information modernes.
- Maîtrise de la méthodologie de recherche et sa mise en œuvre dans un projet réel.
- Approfondissement des connaissances dans le domaine de la programmation.

Enfin, nous espérons que ce travail trouvera une continuité de la part des promotions futures en vue de l'améliorer et l'enrichir.

Bibliographie

- [1] Boualem Benatallah, Fabio Casati, and Farouk Toumani. Representing, analysing and managing web service protocols. *Data & Knowledge Engineering*, 58(3) :327–357, 2006.
- [2] creatly blog : Business process modeling techniques with examples. <https://creately.com/blog/diagrams/business-process-modeling-techniques/#bpmn>.
- [3] Mathias Weske. Business process management foundations. In Business Process Management. Springer, 2012.
- [4] Wil M. P. van der Aalst. *Process Mining : Discovery, Conformance and Enhancement of Business Processes*. Springer Publishing Company, Incorporated, 1st edition, 2011.
- [5] Equipe Conseil Softeam Supervisée par Philippe Desfray. Le guide pratique des processus métiers. *Softeam*, 2008.
- [6] Appian : Business process definition. <https://www.appian.com/bpm/definition-of-a-business-process>.
- [7] Ibm knowledge center, processus métier. https://www.ibm.com/support/knowledgecenter/fr/SSNJFY_8.0.1/com.ibm.wbpm.wid.bpel.doc/topics/cunder.html.
- [8] Pnmsoft : Business process. <http://www.pnmsoft.com/resources/bpm-tutorial/business-process/>. Accessed : 2016.
- [9] Ibm : Websphere developer domain, platform overview. <https://www.ibm.com/developerworks/websphere/products/platformoverview.html>.
- [10] Creatly blog : Business process modeling techniques with examples. <https://creately.com/blog/diagrams/business-process-modeling-techniques/#petri>. Accessed : 16 August 2017.
- [11] Tadao Murata. Petri nets : Properties, analysis and applications. *Proceedings of the IEEE*, 77(4) :541–580, 1989.
- [12] Bruno Bouzy. Unified modeling language. 2014.
- [13] Ibm knowledge centre : diagramme d’activité. https://www.ibm.com/support/knowledgecenter/fr/SS8PJ7_9.1.2/com.ibm.xtools.modeler.doc/topics/cactd.html.
- [14] Modeliosoft : outil uml diagramme d’activité. <https://www.modeliosoft.com/fr/ressources/exemples-de-diagrammes/outil-uml-diagrammes-activite.html>.
- [15] Définition des diagrammes de séquence uml 1.5. <https://www.modeliosoft.com/fr/ressources/exemples-de-diagrammes/outil-uml-diagrammes-activite.html>.
- [16] Guide pratique modèle conceptuel des traitements modèle organisationnel des traitements. <http://gigatechnologie.com/wp-content/uploads/2014/01/MCT-MOT.pdf>.

- [17] Modele conceptuel des traitements. <http://web.maths.unsw.edu.au/~lafaye/CCM/merise/mct.htm>.
- [18] Springer gabler : Business process execution language (bpel). <https://wirtschaftslexikon.gabler.de/definition/business-process-execution-language-bpel-52690/version-275808>.
- [19] Business process analysis : Definition, techniques, and examples. <https://tallyfy.com/business-process-analysis/>.
- [20] Jorge Saldivar, Carla Vairetti, Carlos Rodríguez, Florian Daniel, Fabio Casati, and Rosa Alarcón. Analysis and improvement of business process models using spreadsheets. *Information Systems*, 57 :1–19, 2016.
- [21] Julia Rudnitckaia. Process mining. data science in action. *University of Technology, Faculty of Information Technology*, pages 1–11.
- [22] W. v. Aalst. Using process mining to bridge the gap between bi and bpm. *Computer*, 44 :77–80, 12 2011.
- [23] Frédéric Santos. Arbres de décision. 2015.
- [24] Boualem Benatallah, Sherif Sakr, Daniela Grigori, Hamid Reza Motahari-Nezhad, Moshe Chai Barukh, Ahmed Gater, Seung Hwan Ryu, et al. *Process analytics : concepts and techniques for querying and analyzing process data*. Springer, 2016.
- [25] Oscar Romero and Alberto Abelló. A survey of multidimensional modeling methodologies. *International Journal of Data Warehousing and Mining*, 5(2) :1, 2009.
- [26] Erik Thomsen. *OLAP solutions : building multidimensional information systems*. John Wiley & Sons, 2002.
- [27] Elke Rundensteiner, Volker Markl, Ioana Manolescu, Sihem Amer-Yahia, Felix Naumann, and Ismail Ari, editors. *EDBT '12 : Proceedings of the 15th International Conference on Extending Database Technology*, New York, NY, USA, 2012. ACM.
- [28] Jiawei Han, Jian Pei, Guozhu Dong, and Ke Wang. Efficient computation of iceberg cubes with complex measures. In *ACM SIGMOD Record*, volume 30, pages 1–12. ACM, 2001.
- [29] Jiawei Han, Yizhou Sun, Xifeng Yan, and S Yu Philip. Mining knowledge from data : An information network analysis approach. In *Data Engineering (ICDE), 2012 IEEE 28th International Conference on*, pages 1214–1217. IEEE, 2012.
- [30] Qiang Qu, Feida Zhu, Xifeng Yan, Jiawei Han, S Yu Philip, and Hongyan Li. Efficient topological olap on information networks. In *International Conference on Database Systems for Advanced Applications*, pages 389–403. Springer, 2011.
- [31] Techopedia : Data warehouse (dw). <https://www.techopedia.com/definition/1184/data-warehouse-dw>.
- [32] Piloter.org : Data warehouse, entrepôt de données. <https://www.piloter.org/business-intelligence/datawarehouse.htm>.
- [33] *Helix : Online enterprise data analytics*, 01 2011.
@bookbpi, title=Business Intelligence : Data Mining and Optimization for Decision Making, author=Vercellis, C., isbn=9781119965473, url=https ://books.google.dz/books?id=Yl_yAn2bhZ0C, year=2011, publisher=Wiley

- [34] Fabio Casati, Umesh Dayal, Mehmet Sayal, and Ming-Chien Shan. Business process intelligence. *Software Technology Laboratory, HP Laboratories, Palo Alto, California*, 04 2002.
- [35] Boualem Benatallah, Fabio Casati, and Farouk Toumani. Web service conversation modeling : A cornerstone for e-business automation. *IEEE Internet Computing*, 8(1) :46–54, 2004.
- [36] Julien Ponge, Boualem Benatallah, Fabio Casati, and Farouk Toumani. Fine-grained compatibility and replaceability analysis of timed web service protocols. In *ER*, pages 599–614, 2007.
- [37] Ali Khebbizi. External behavior modeling enrichment of web services by transactional constraints. In *Proceedings of the PhD Symposium at the 6th International Conference on Service Oriented Computing (ICSOC 2008), Sydney, Australia, 1 December 2008*, 2008.
- [38] Nannette Liske, Niels Lohmann, Christian Stahl, and Karsten Wolf. Another approach to service instance migration. In *ICSOC-ServiceWave '09*, pages 607–621, Berlin, Heidelberg, 2009. Springer-Verlag.
- [39] Seung Hwan Ryu, Fabio Casati, Halvard Skogsrud, Boualem Benatallah, and Régis Saint-Paul. Supporting the dynamic evolution of web service protocols in service-oriented architectures. *ACM Trans. Web*, 2(2) :1–16, May 2008.
- [40] Seung Hwan Ryu, Régis Saint-Paul, Boualem Benatallah, and Fabio Casati. A framework for managing the evolution of business protocols in web services. In *Proceedings of the fourth Asia-Pacific conference on Conceptual modelling - Volume 67, APCCM '07*, pages 49–59, Ballarat, Australia, 2007. Australian Computer Society, Inc.
- [41] Mike P. PAPAZOUGLOU. The challenges of service evolution. In *Proceedings of the 20th international conference on Advanced Information Systems Engineering, CAiSE '08*, pages 1–15, Berlin, Heidelberg, 2008. Springer-Verlag.
- [42] Ahmed Azough, Emmanuel Coquery, and Mohand-Said Hacid. Supporting web service protocol changes by propagation. In *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology - Volume 01, WI-IAT '09*, pages 438–441, Washington, DC, USA, 2009. IEEE Computer Society.
- [43] Yi Wang and Ying Wang. A survey of change management in service-based environments. *SOCA*, 7(4) :259–273.
- [44] Tomáš Růpl. Business process modelling—methods and methodologies. *Systémová integrace*, 12(3) :27–42, 2005.
- [45] Vaclav Repa. Methodology for business processes analysis. In *Evolution and Challenges in System Development*, pages 147–157. Springer, 1999.
- [46] Stijn Goedertier, Jan Vanthienen, and Filip Caron. Declarative business process modelling : principles and modelling languages. *Enterprise Information Systems*, 9(2) :161–185, 2015.
- [47] Fabio Casati, Malu Castellanos, Umeshwar Dayal, and Norman Salazar. A generic solution for warehousing business process data. In *Proceedings of the 33rd international conference on Very large data bases*, pages 1128–1137. VLDB Endowment, 2007.

- [48] Asma Hassani and Sonia Ayachi Gahnouchi. A framework for business process data management based on big data approach. *Procedia Computer Science*, 121 :740–747, 2017.
- [49] Nenad Jukic, Boris Jukic, and Mary Malliaris. Online analytical processing (olap) for decision support. In *Handbook on Decision Support Systems 1*, pages 259–276. Springer, 2008.
- [50] Oussama Mohammed Kherbouche. *Contribution à la gestion de l'évolution des processus métiers*. PhD thesis, Université du Littoral Côte d'Opale, -France-, 2013.
- [51] Wil M. P. Aalst, Marcello La Rosa, and Flávia Santoro. Business process management - don't forget to improve the process! 58 :6, 10 2015.
- [52] T. Pyzdek. *The Six Sigma Handbook, Revised and Expanded*. Mcgraw-hill, 2003.
- [53] W.M.P. van der Aalst, A.J.M.M. Weijter, and L. Maruster. Workflow mining : Discovering process models from event logs. *IEEE Transactions on KDE.*, 2003.
- [54] W. M. P. van der Aalst, B. F. van Dongen, J. Herbst, L. Maruster, G. Schimm, and A Weijters. Workflow mining : A survey. *D.K.E.*, 47(2), 2003.
- [55] Ana Karla A. de Medeiros, W.M.P. van der Aalst, and A. J. M. M. Weijters. Workflow mining : Current status and future directions. In *CoopIS/DOA/ODBASE, L.N.C.S., vol. 2888*, pages 389–406, 2003.
- [56] Sherif Sakr Daniela Grigori Hamid Reza Motahari-Nezhad Moshe Chai Barukh Ahmed Gater Seung Hwan Ryu Seyed-Mehdi-Reza Beheshti, Boualem Benatallah. *Process Analytics, Concepts and Techniques for Querying and Analyzing Process Data*, chapter Business Process Data Analysis, pages 107–134. Springer Publishing Company, Incorporated, 1st edition, 2016.
- [57] J.J. Moder and S.E. Elmaghraby. *Handbook of Operations Research : Foundations and Fundamentals*. Chapman & Hall, 2000.
- [58] John A Buzacott. Commonalities in reengineered business processes models and issues. *Management Science*, 42(5) :768–782, 1996.
- [59] Avishai Mandelbaum and Sergey Zeltyn. Data-stories about (im) patient customers in tele-queues. *Queueing Systems*, 75(2-4) :115–146, 2013.
- [60] Arik Senderovich, Matthias Weidlich, Avigdor Gal, and Avishai Mandelbaum. Queue mining—predicting delays in service processes. In *International Conference on Advanced Information Systems Engineering*, pages 42–57. Springer, 2014.
- [61] Raffaele Conforti, Massimiliano de Leoni, Marcello La Rosa, Wil MP van der Aalst, and Arthur HM ter Hofstede. A recommendation system for predicting risks across multiple business process instances. *Decision Support Systems*, 69 :1–19, 2015.
- [62] Ibm software. Technical report, www-01.ibm.com/software.
- [63] Oracle business process management suite. <http://www.oracle.com/us/technologies/bpm/suite/overview/index.html>.
- [64] Business process management. https://help.sap.com/saphelp_nw73ehp1/helpdata/en/10/e808e319284dc8b0eac42d1d95735a/frameset.htm.
- [65] Bonitasoft. <https://www.bonitasoft.com/>.
- [66] Bonita bpm portal interface overview. <https://documentation.bonitasoft.com/6.x-7.2/bonita-bpm-portal-interface-overview-0>.

- [67] Chapitre 2 : Automates finis déterministes (afd). <http://ehess.modelisationsavoirs.fr/marc/ens/langages/2006/AutChapitre2.pdf>.
- [68] Franck Cassez and Olivier-H Roux. Structural translation from time petri nets to timed automata. *Electronic Notes in Theoretical Computer Science*, 128(6) :145–160, 2005.
- [69] Xi Wang, Huaikou Miao, and Liang Guo. Towards automatic transformation from uml model to fsm model for web applications. *Journal of Software Engineering and Applications*, 1(01) :68, 2008.
- [70] Chun Ouyang, Marlon Dumas, Arthur HM Ter Hofstede, and Wil Mp Van Der Aalst. Pattern-based translation of bpmn process models to bpm web services. *International Journal of Web Services Research*, 5(1) :42, 2008.
- [71] Andreas Wombacher, Peter Fankhauser, and Erich Neuhold. Transforming bpm into annotated deterministic finite state automata for service discovery. In *Web Services, 2004. Proceedings. IEEE International Conference on*, pages 316–323. IEEE, 2004.
- [72] Ali Khebizi, Hassina Seridi-Bouchelaghem, Bouallem Benatallah, and Farouk Toumani. A declarative language to support dynamic evolution of web service business protocols. *Serv. Oriented Comput. Appl.*, 11(2) :163–181, June 2017.
- [73] Chapitre 1 : Expressions régulières et automates. <http://ehess.modelisationsavoirs.fr/marc/ens/langages/2006/AutChapitre1.pdf>.
- [74] John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages, and Computation (3rd Edition)*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2006.
- [75] Ramy Ragab Hassen. *Automatic composition of protocol-based Web services*. PhD thesis, Université Blaise Pascal-Clermont-Ferrand II, 2009.
- [76] John Vlissides, Richard Helm, Ralph Johnson, and Erich Gamma. Design patterns : Elements of reusable object-oriented software. *Reading : Addison-Wesley*, 49(120) :11, 1995.
- [77] Yehia Elshater, Patrick Martin, and Ehab Hassanein. Using design patterns to improve web service performance. In *Services Computing (SCC), 2015 IEEE International Conference on*, pages 746–749. IEEE, 2015.
- [78] Daniela Berardi, Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Massimo Mecella. Automatic composition of e-services that export their behavior. In *ICSOC*, pages 43–58, Dec. 2003.
- [79] GauravVaswani ShetaliPatil and Anuradha Bhatia. Graph databases-an overview. *1Student, ME Computers, Terna College of Engg, Navi Mumbai*, 2 :657–660.
- [80] Ian Robinson, Jim Webber, and Emil Eifrem. *Graph Databases : New Opportunities for Connected Data*. O’Reilly Media, Inc., 2nd edition, 2015.
- [81] Mingxi Wu. What are the major advantages of using a graph database? <https://dzone.com/articles/what-are-the-pros-and-cons-of-using-a-graph-databa>, 2017.
- [82] Cypgher(language. [https://fr.wikipedia.org/wiki/Cypher_\(langage\)](https://fr.wikipedia.org/wiki/Cypher_(langage))), 2018.
- [83] Java. https://www.java.com/fr/download/faq/whatis_java.xml.
- [84] Gson user guide. <https://github.com/google/gson/blob/master/UserGuide.md#TOC-Overview>.

- [85] Sparkjava : Quick start. <http://sparkjava.com/>.
- [86] Neo4j : The all-new, officially supported neo4j-jdbc driver 3.0. <https://neo4j.com/blog/official-neo4j-jdbc-driver-3-0/>.
- [87] What is neo4j. https://neo4j.com/developer/graph-database/#_what_is_neo4j.
- [88] Neo4j : Une autre façon de manipuler les données. <https://www.d-booker.fr/content/68-introduction-bases-de-donnees-neo4j>.
- [89] Using neo4j from java. <https://neo4j.com/developer/java/>.
- [90] Neo4j,. <https://neo4j.com/product/>.
- [91] Neo4j browser user interface guide. https://neo4j.com/developer/guide-neo4j-browser/#_installing_and_starting_neo4j_browser.
- [92] Gojs : Interactive javascript diagrams in html. <https://gojs.net/latest/index.html>.
- [93] What is a graph database? <https://neo4j.com/developer/graph-database/>.