



Integrating a business process analysis and optimization step using BPMN model in a general process design and development approach : application to a paperless mail flow process

Ahmad Shraideh

► To cite this version:

Ahmad Shraideh. Integrating a business process analysis and optimization step using BPMN model in a general process design and development approach : application to a paperless mail flow process. Other. Ecole Centrale de Lille, 2009. English. NNT : 2009ECLI0021 . tel-00579520

HAL Id: tel-00579520

<https://theses.hal.science/tel-00579520>

Submitted on 24 Mar 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N° d'ordre :

1	1	1
---	---	---

ÉCOLE CENTRALE DE LILLE

THÈSE

présentée en vue d'obtenir le grade de

DOCTEUR

en

Spécialité : Automatique et Informatique Industrielle

par

Ahmad SHRAIDEH

Doctorat délivré par l'École Centrale de Lille

Analyse et optimisation d'un processus à partir d'un modèle BPMN dans une démarche globale de conception et de développement d'un processus métier:

Application à la dématérialisation de flux courrier du projet GOCD (PICOM)

Soutenue le 8 décembre 2009 devant le jury d'examen :

Président	Jean-Claude GENTINA	Professeur à l'École Centrale de Lille
Rapporteur	Isabel DEMONGODIN	Professeur à l'Université Paul Cézanne à Marseille
Rapporteur	Jean-Claude BOCQUET	Professeur à l'École Centrale de Paris
Examineur	Jean-Claude HENNET	Professeur, Directeur de Recherche au CNRS
Examineur	Michel BIGAND	Maître de conférences HdR à l'ERGI, Ecole Centrale de Lille
Examineur	François PÉRÈS	Maître de conférences à l'École Nationale d'Ingénieurs de TARBES
Directeur de thèse	Pascal YIM	Professeur à l'École Centrale de Lille
Co-directeur de thèse	Hervé CAMUS	Maître de Conférences à l'École Centrale de Lille
Invité	Olivier BERUT	Directeur Développement à Okaidi

Thèse préparée au sein des Laboratoires LAGIS

École Doctorale SPI 072

À mes parents

À mes frères et sœurs

Au peuple Palestinien

Remerciements

En premier lieu, je tiens à exprimer ma profonde gratitude aux directeurs de cette thèse le Professeur Pascal YIM et à Hervé CAMUS, Maître de Conférences. Je veux les remercier pour leur encadrement plein d'enthousiasme et de rigueur et pour la confiance dont ils ont fait preuve à mon égard. De plus, je tiens à leur manifester ma sincère reconnaissance pour leurs grandes qualités humaines qui ont fait que cette thèse se passe dans la bonne humeur.

Ces remerciements ne seraient pas complets si je n'y associais pas toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce travail, en particulier, mes collègues, tout le personnel du LAGIS et de l'École Centrale de Lille pour leur bonne humeur et leur disponibilité.

Contents

List of Figures	9
List of Tables	10
Résumé	12
Introduction	21
1 From business process modeling to workflow analysis and optimization	28
1.1 Introduction	28
1.2 Problematic: mail flow optimization COFIDIS	29
1.3 Workflow reengineering	32
1.3.1 Proposed standards to be used in workflow reengineering process	33
1.3.1.1 BPMN	34
1.3.1.2 WS-BPEL	42
1.3.2 BPMN to BPEL	46
1.3.2.1 Event handler based transformation	48
1.3.2.2 Pattern based transformation	49
1.3.2.3 BPMN2BPEL tool	54

1.3.3	BPMN limits	56
1.4	Proposed framework for workflow reengineering	57
1.4.1	Framework for workflow reengineering	57
1.4.2	Petri nets	60
1.4.2.1	Colored Petri nets	64
1.4.2.2	Timed Petri nets	64
1.5	Modeling and operational analysis for the current workflow at COFIDIS	65
1.6	Workflow analysis	70
1.7	Workflow optimization	75
1.8	Conclusion	85
2	Business process optimization and help in decision taking	87
2.1	Introduction	87
2.2	Related Problems	88
2.2.1	Bin packing problem BPP	88
2.2.2	Generalized assignment problem GAP	90
2.2.3	The suitability of BPP and GAP to represent COFIDIS contracts assignment problem	92
2.3	The Generalized assignment problem with identified first-used bins (GAPIFB)	94
2.4	GAPIFB and decision making tool	97
2.4.1	Short-term approach	99
2.4.2	Mid-term approach	105
2.4.3	Simulation and test results	109
2.5	Conclusion	113
3	Conclusion and perspectives	116

List of Figures

1	Process lifecycle	22
1.1	Workflow reengineering process	33
1.2	Workflow reengineering process with proposed standard and technologies	35
1.3	Events Objects	38
1.4	Activities Objects	39
1.5	Gate Ways Objects	39
1.6	Connecting Objects	40
1.7	Swimlanes(Pool and Lanes)	41
1.8	Artifacts in BPMN	41
1.9	Example of BPEL Process	44
1.10	Mapping well-structured componants into BPEl	51
1.11	A complete example in using the FOLD function	52
1.12	Quasi Structured componants transformation	53
1.13	Example of acyclic componant neither safe nor sound	54
1.14	Model transformation in order to use BPMN2BPEL tool as proposed by Pau et al.	55
1.15	Proposed framework for workflow reengineering	59
1.16	An example of Petri nets.	61
1.17	Workflow Net	63

1.18	Timed Petri nets	65
1.19	Current mails workflow within COFIDIS	66
1.20	New proposed mails workflow	68
1.21	The mapping of task, events, and gateways to Petri-net as proposed by Dijkman	71
1.22	Funding project process and the corresponding Petri nets model	73
1.23	Modified Petri nets model to represent continuous funding project process	73
1.24	Petri nets model for the proposed workflow. a) Handling a contract by a collaborator. b) Client response.	74
1.25	Behavioral and structural analysis	75
1.26	A part of the colored Petri nets that simulate contracts work- flow at time stamp 80.	78
1.27	Petri nets representing two enterprises operators and their cor- responding processing time for two different contract type(RED, BLUE).	82
2.1	New workflow with decision making support	98
2.2	Interactive decision-making process	100
2.3	Postponed contracts flow over D days	106

List of Tables

1.1	Competence matrix example	30
1.2	Daily tasks table example for two services	31
1.3	Results extracted from simulating contracts assignment process using colored Petri nets.	78
1.4	Result from two different contracts assignments for the same contracts sample. Different contracts assignments needed different number of work time units.	79
2.1	Simulation results of the short-term approach for the first stage	110
2.2	First stage results for the short-term approach and the mid-term approach	112
2.3	Average profit of using mid-term approach over 10 random samples	112

Résumé

Cette thèse a été réalisée dans le cadre du projet " Gestion et Optimisation de la Chaîne Documentaire " (GOCD), projet labellisé par le Pôle de compétitivité des Industries du Commerce (PICOM). Le projet a pour but de concevoir et de développer un nouveau workflow et un outil d'aide à la décision. Ce système s'inscrit dans la démarche de dématérialisation de flux courriers dans l'entreprise COFIDIS. Nous nous intéressons ici à la gestion des demandes de crédit envoyées par les clients sous la forme de contrats. Ceux-ci dans la nouvelle organisation devront être, dans un premier temps, scannés, puis identifiés, triés et envoyés au département chargé d'étudier la recevabilité de cette demande. Cette étude est réalisée par des collaborateurs dont le degré de compétences influe sur le temps de traitement de chaque dossier. Après examen, le dossier peut être soit refusé, soit directement accepté et dans ce cas il est archivé. Enfin si celui-ci est jugé incomplet par le collaborateur, une information est transmise au client concerné (via mail, téléphone, courrier, ...) pour une demande de complément d'information (pièce d'identité, relevé bancaire, justificatif de domicile, ...). Le dossier est alors mis en attente jusqu'à réception des documents manquants. L'outil de décision sera en partie utilisé par les chefs de service en charge de répartir les dossiers à leurs équipes de collaborateurs. Actuellement la répartition des dossiers se fait selon une heuristique (outil DISPATCH) limitée à une vision

court terme (quotidien) du flux courrier.

Une première analyse et retour d'existant du processus actuel ont abouti à certains dysfonctionnements : difficulté pour rattacher des documents complémentaires envoyés par un client à sa demande initiale de crédit, éventuelles pertes d'information notamment des dossiers incomplets mis en attente par les collaborateurs, mauvaise répartition des dossiers suivants les équipes ; appel non justifié d'intérimaires pour venir épauler les équipes permanentes en charge des études de demande de crédit. Cette analyse a été prise en compte dans la conception du nouveau processus métier de gestion des flux dématérialisés courrier.

Un premier modèle a été réalisé en utilisant BPMN (Business Process Management Notation). Cette notation est particulièrement adaptée pour les processus d'entreprise avec différentes phases de traitement et validation. Le choix de BPMN s'inscrit dans le cadre d'un framework complet de conception - développement de workflow composé de BPMN et de BPEL (Business Process Execution Language). Il propose l'utilisation de BPEL pour implémenter et exécuter l'application finale de workflow en tant que services web. Les motivations derrière le choix de BPMN et BPEL dans ce cadre de travail sont soulignées. Les approches pour générer du code BPEL à partir d'un model BPMN sont présentées.

Cependant BPMN est un langage orienté description car il doit être facilement compréhensible par toutes les personnes du projet depuis les utilisateurs jusqu'aux responsables en passant par les analystes métier et les développeurs. La flexibilité offerte par BPMN peut conduire à des propriétés indésirables du processus tels que blocage et inaccessibilité. Il est donc difficile de vérifier de bonnes propriétés du modèle. De plus, BPMN a été conçu pour fournir des modèles Orientés Processus. Les données ou les ressources

y sont donc peu représentées. Peu de formalisation sur l'aspect comportemental des flux a été faite. En conséquence, l'analyse de performance sur un modèle BPMN est quasi inexistante et d'autant plus sur l'aspect optimisation de flux.

Afin de surmonter ces problèmes, nous proposons d'insérer dans le framework deux nouvelles phases. Ces deux phases sont appliquées à partir de l'obtention du modèle BPMN. La première est une phase de vérification et de validation et la deuxième une phase d'optimisation. Ces deux phases sont réalisées en transformant le modèle BPMN vers un langage formel. Notre choix s'est porté sur les Réseaux de Petri.

Il existe dans la littérature des travaux qui proposent la transformation par étapes du modèle BPMN en un modèle Réseau de Petri. Nous présentons ces approches et nous les adaptons au processus qui nous intéresse ici. Mais pour prendre en compte l'aspect discret du problème (gestion des dossiers clients) ainsi que les contraintes ressources spécifiques (matrice de compétence des collaborateurs, c'est-à-dire le temps de traitement des dossiers dépendant du niveau de compétence du collaborateur et du type de dossiers et disponibilité des ressources), nous avons apporté ces informations supplémentaires par rapport au modèle initial BPMN, qui n'est pas capable de gérer ces données et ces contraintes de ressources. Nous avons d'abord utilisé les Réseaux de Petri colorés pour représenter facilement et lisiblement la diversité des chemins liés aux nombreux types de dossiers à étudier.

Pour pouvoir vérifier les propriétés structurelles et comportementales du processus métier, nous avons utilisé les Réseaux de Petri ordinaires en passant par un dépliage de celui-ci pour vérifier qu'il n'y aura pas de présence de blocage pour telle ou telle couleur par exemple. Les RdP ordinaires sont utilisés pour valider et vérifier les propriétés telles que la vivacité (deadlock

free, liveness) et la finitude (boundedness). Cette recherche de propriétés donne de précieuses informations sur le processus ainsi conçu. Les problèmes que l'on peut identifier à cette étape ne sont pas forcément visibles avec le modèle BPMN. Ils sont dus en général à des erreurs humaines de conception ou à une mauvaise interprétation des spécifications de BPMN. Cette phase de vérification est donc indispensable pour valider dans un deuxième temps le modèle du processus après retour et correction si nécessaire sur la conception du processus et donc sur le modèle BPMN.

Une fois le modèle validé sur les aspects structurels et comportementaux, il s'agit dans une deuxième phase d'étudier les performances du workflow afin de l'optimiser. Notre choix dans ce travail a été d'utiliser dans un premier temps les réseaux de Petri colorés pour simuler les performances à l'aide de l'outil CPN Tool. Ce qui nous a permis de constater la présence d'un goulet d'étranglement au niveau du traitement d'acceptation des demandes de crédit. C'est donc à ce niveau qu'une analyse de performance sera faite pour optimiser la répartition des dossiers entre les collaborateurs avec appel si nécessaire à des intérimaires pour supporter l'excédent de charge de travail.

Pour cela, nous sommes repartis sur le modèle déplié du Réseau de Petri. Il permet en effet une analyse fine des performances du workflow en évaluant précisément les charges de travail de chaque ressource (ici les collaborateurs). Le modèle Réseau de Petri nous a permis à l'aide du modèle mathématique sous-jacent de formuler mathématiquement le problème de répartition des charges de travail suivant les collaborateurs et intérimaires. La résolution de ce problème va permettre d'offrir aux décideurs, c'est-à-dire les chefs de services, des indicateurs quant au nombre minimum d'intérimaires à engager au jour le jour pour absorber la charge de travail quotidien, de choisir entre différents critères d'optimisation (équilibre de charge, minimisation du

max des charges de travail, ...) pour la meilleure politique de répartition des dossiers. De plus à partir de l'historique des arrivées de courrier et en estimant l'arrivée de courriers sur une période donnée (une semaine, voire un mois), il va être possible d'ajuster et de limiter le surcoût financier engendré par les appels ponctuels et quotidiens de ces intérimaires en lissant la surcharge occasionnée par une campagne marketing sur une semaine par exemple en prenant en compte la disponibilité de leurs collaborateurs.

A partir de l'analyse de performance du modèle Réseau de Petri, nous avons défini une nouvelle variante du problème d'affectation (bin packing problem) et proposé une résolution à intégrer dans le processus d'aide à la décision. Le problème ainsi obtenu a été appelé " Generalized Assignment Problem with Identified First-used Bins " (GAPIFB).

Ce problème se situe entre deux problèmes connus en Recherche Opérationnelle : le Bin Packing Problem (BPP) et le Generalized Assignment Problem (GAP). Nous montrons qu'aucun de ces deux problèmes, ni leurs variantes, ne correspond à notre problème. Celui-ci est un problème d'affectation ayant une application concrète et généralisable en entreprise. Nous présentons la formulation mathématique de ce type de problème comme un problème linéaire en entiers (PLE). Pour résoudre ce problème, nous avons choisi la résolution par contraintes et utilisé le solveur CPLEX développé par ILOG. Ce n'est pas bien entendu la seule méthode de résolution. Pour l'ordre de grandeur du problème qui nous intéresse, c'est-à-dire quelques milliers de dossiers à traiter par jour et une centaine de collaborateurs et intérimaires confondus, le temps de résolution exacte du problème avec CPLEX est de l'ordre de la milliseconde. Ce qui est largement suffisant pour les contraintes de calcul demandé ici, c'est-à-dire calcul prévisionnel hors ligne pour nourrir les indicateurs de l'outil d'aide à la décision.

Nous proposons d'intégrer cette résolution du problème d'affectation dans un outil d'aide à la décision. Cet outil aide les décideurs à prendre leur décision sur un horizon à court-terme (décisions quotidiennes) et moyen terme (décisions sur plusieurs jours).

A court terme, l'approche permet d'affecter les contrats de façon optimal et en temps quasi réel. Pour arriver à résoudre ce problème, nous l'avons décomposé en deux phases distinctes. La première consiste à évaluer la charge de travail engendré par l'arrivée de courrier (GAPIFB problem) : soit le système est en sous-charge c'est-à-dire que l'équipe des collaborateurs est capable de traiter l'ensemble de la charge soit en surcharge - incapacité de traiter le jour-même tout le flux courrier. Dans ce cas, nous calculons le nombre minimum d'intérimaires nécessaires pour absorber la charge de travail dans la journée. A partir de cette donnée et du choix effectué par le chef de service, un deuxième calcul (GAP problem) d'optimisation est lancé afin d'évaluer la meilleure répartition selon l'objectif demandé par le décideur. C'est ce qui permet de pouvoir rapidement réagir en cas de perturbation du flux normal (le nombre de contrats incomplets non conforme à la prévision qui a été faite le matin même par exemple). Par un contrôle continu des paramètres prévisionnels de charge et de temps de traitement pour les différents opérateurs, ces variations peuvent être rapidement détectées et on peut anticiper sur les dérives occasionnées en recalculant la réaffectation des dossiers à traiter pour la journée.

Sur un horizon à moyen terme, cet outil permet aux décideurs de ne pas systématiquement faire appel le jour même à des intérimaires si on voit qu'il est possible de lisser la charge sur les jours suivants. Il faut faire un compromis entre les surcouts engendrés par l'appel d'intérimaire et celui engendré par le retard pris dans l'étude de demande de crédit.

Cette approche a été appliquée sur des échantillons de données réelles venant de Cofidis. D'autres échantillons de même ordre de grandeur ont été également générés automatiquement afin d'avoir des résultats significatifs en termes de temps de résolution. Même si dans certains cas, la preuve de l'optimalité de la solution peut largement dépasser l'ordre de la milliseconde, il reste qu'une solution reste toujours très rapidement trouvée. Une heuristique simple consisterait à borner le temps de résolution et de ne conserver que la dernière meilleure solution trouvée. Dans tous les cas, l'approche que nous proposons reste toujours meilleure par rapport à celle actuellement utilisée pour " dispatcher " les dossiers entre collaborateurs qui n'est basée que sur des règles heuristiques très (voire trop simples). Sur le court terme, notre approche permet d'évaluer la charge engendrée par l'arrivée de courriers et d'économiser ainsi l'appel d'un nombre non nécessaires d'intérimaires. L'approche à moyen terme que nous avons proposée est déjà en soit une innovation puisque l'optimisation de la répartition sur plusieurs jours n'avait pas encore été envisagée. Nous montrons que nous pouvons par cette approche économiser jusqu'à 25% sur les surcouts salariaux. Le seul problème réside dans l'estimation correcte des taux de dossiers incomplets ainsi que des quantités de courrier à arriver dans les jours futurs. Nous pensons que cela ne posera pas trop de problèmes même s'il ne faut pas négliger cette étape de prédiction. En effet Cofidis conserve un historique complet de ces informations et a une solide expérience sur les retombées de différentes campagnes marketing saisonnières (type le Tour de France cycliste) sur le nombre de dossiers reçus.

En termes de perspectives, plusieurs pistes sont à envisager. Tout d'abord une formalisation et une généralisation de la méthode de transformation du modèle BPMN en Réseaux de Petri pourra être étudié. Concernant les ap-

proches de résolution du GAFPIB problem, on peut améliorer l'approche liée à la programmation sous contrainte en apportant de nouvelles contraintes qui permettront de couper plus rapidement des branches entières de l'arbre de recherche qui n'apporteront pas de solution optimale. Dans un deuxième temps, comme ce n'est pas l'approche optimale d'autres approches de résolution pourront être étudiés pour des problèmes se rapprochant permettant la mise en place d'un algorithme spécifique de résolution de ce type de problème en un temps minimal.

Introduction

Information technology (IT) has become one of the key successes of any modern enterprise and organization. It touches all business aspects and overlaps with external process, especially with the new outsourcing tendency in enterprises world. Thus the need for more complex and flexible systems to manage and analyze business processes has become more important than any time before, to save cost and improve services and business process quality.

In business process reengineering process, many notions appear and use the different process aspects such as Business Process Management (BPM), Workflow, Business Activity Monitoring (BAM) and Decision-making Support systems (DMSS). In the next few paragraphs we try to present and define briefly each of these notions.

Business Process Management (BPM) is defined as "the art of modeling, managing and optimizing business processes in order to increase business performance". This means managing the entire business process lifecycle which includes analyzing, executing, monitoring, and ensuring business process integrity and optimality. This can be realized through employing recent technologies and standards in BPM domain. Figure 1 demonstrates a business process lifecycle. A business process is composed of a sequence of activities and each activity is composed of roles and actions. These roles and actions are performed by one or more actors to achieve business objectives

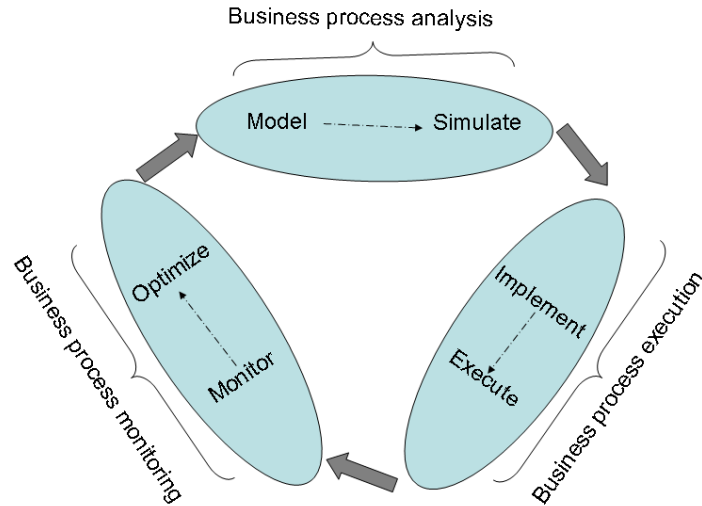


Figure 1: Process lifecycle

and to create additional value for both enterprises and customers. An actor can be one person, a working group, or even a computer program or external enterprises[Osyc 85]. Sometimes confusion between Business Process Modeling and Business Process Management can occur since they have the same acronym (BPM). Business Process Modeling is the act of representing a process using text, diagram and notations. These processes can be the current processes ("as is") or the wished improved process ("to be"). Business Process Modeling is only one phase of the different phases within Business process management lifecycle.

Another important emerging technology for business process is known as Workflow. Workflow is the automation of business process through multiple participants by passing task from one participant to another. It permits to know which action or actions will be performed, by which actor or actors,

and in which order.

According to Workflow Management Coalition¹ (WfMC), workflow is defined as: "The automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules."

Any product developed to support workflow is called workflow management system (WfMS). WfMC has defined WfMS as: "A system that defines, creates, and manages the execution of workflows through the use of software, running on one or more workflow engines, which is able to interpret the process definition, interact with other workflow participants and, where required, invoke the use of IT tools and applications."

So what are the differences between BPM and workflow? In fact, BPM and workflow can be used to define, test and use business process. Still, some people distinguish between the two notions in that BPM is used more for business process that is done entirely by machine and in fast manner, whereas workflow focus on operation or process that need human intervention and decision and that take more time to be executed. In general, workflow systems deal with operations that are often distributed over a large numbers of actors. Thus, both BPM and workflow systems task are overlapped and complementary.

In order to seize BPM and workflow advantages, business process progress must be managed and controlled in real time. This allows decision makers to detect early any unexpected events or dysfunctions that threaten or disturb the normal business process flow. And anticipate them by taking the correct

¹The Workflow Management Coalition (WfMC) is a global organization of adopters, developers, consultants, analysts, as well as university and research groups engaged in workflow and BPM. <http://www.wfmc.org/>

decision in the right time.

In the past, companies depend on managers' experiences to detect and to prevision any improbable problems in the system. Nowadays, companies become large and larger day after day, and business process by consequence is more and more complicated. Thus, this task became nearly impossible, and managers need more help. Business Activity Monitoring (BAM) and Decision-making Support systems (DMSS) come to answer and to fill out this need. Business Activity Monitoring (BAM) is a software product that in general aimed to provide real-time access to critical business performance indicators in order to improve business operations speediness and effectiveness". It provides managers with the necessary indicators and relevant information to give him clearer vision on the system. Some of these indicators and information can be invisible, and require more work to be induced.

BAM can provide real time indicators to alert supervisors in real time by using simple business rules. It can also provide indicators that are obtained from gathering and analysis data for hours, days or even weeks. This gives mid-term vision and help in near future planning. Finally these indicators can be obtained from complex and deep analysis of company information system by using sophisticated business intelligent techniques. These can be very helpful to senior managers to help them taking decision in strategic level for the enterprise.

The project GOCD

This thesis is realized as apart of the French competitive cluster "Industrie du commerce". The French competitive cluster is a French governmental effort that aims to bring competition key factors in one industrial policy

to encourage and support projects that are initiated by economic or academic players. It aims to face current market challenges and competitively especially the new trade globalization and the arrival of a knowledge-based economy, where innovation and research are its primary drivers. The French competitive cluster associates each of companies, research centers and educational institutions under a common strategy to work in partnership and to advance collaboration between them.

Within this context, the project *GOCD* was initialized. *GOCD* is the French acronym for "Gestion et Optimisation de la Chaîne Documentaire" which stands for Management and optimization of document life cycle. The industrial partners in this projects includes both of *COFIDIS* and *ALFEA*. *COFIDIS* is a French company specialized in consumer credit business. *ALFEA* is an information system consulting company that specialized in Enterprise Content Management (ECM) and Decision Support System (DSS). The academic partner in this project is our university "L'Ecole Centrale de Lille" represented by *LAGIS* laboratory "Laboratoire d'Automatique, Génie Informatique et Signal".

Project objectives

The main objective for the project *GOCD* is to install a new paperless workflow system and decision making tool to replace the current paper based system. The new workflow system aims to manage and optimize received mails at *COFIDIS* in all mails lifecycle, starting by their arrival and ending by archiving them. In the other hand, the decision making tool will interest in optimizing mails assignment process by providing decision makers with the suitable Key Performance Indicator. At the same time, mails traceabil-

ity is kept over the different handling phases and monitoring mails handling progress at any time.

In this thesis, we propose and study a new framework to model, simulate, optimize and evaluate the performance of workflow. The proposed framework allows i) more flexibility for workflow reengineering process ii) deep and thin analysis for different business process which allows the identification of business process weakness and deficiencies. The proposed framework uses the most modern standards and technologies. It covers all development phases, starting from modeling phase, passing through process analyzing and optimizing phase and ending by the finale implementation phase. Within the optimization phase a new assignment problem has been emerged. This problem is formulated as an integer linear programming problem, and solved using exact method.

Thesis organization

This thesis is organized as the following. In the next chapter, we start by presenting in details current workflow problem at COFIDIS. This is followed by presenting a normal workflow reengineering process. A brief introduction of standards and technologies used in workflow reengineering is presented and the motivation behind choosing these standards and technologies. Challenges that hinder these choices are demonstrated. To overcome these challenges, new framework for workflow reengineering is proposed. And the reengineering process for COFIDIS workflow is taken as an applied example.

In chapter two, we demonstrate the new assignment problem that emerged due to the application of the optimizing phase of the new framework. The new problem is discussed in detailed with other assignment problems exist

in literature. Similarities and differences between these problems are underlined. We show that none of the problems exist already in literature corresponds completely to the new emerged problem. The new problem is called "Generalized Assignment Problem with Identified First-use Bins (GAPIFB)" and it discussed in details. The new GAPIFB is defined and its mathematical model is presented. An improvement of the GAPIFB problem is discussed then to cover long period of time. We show how to employ the new GAPIFB in generating useful KPIs and dashboards for decision maker. And we terminate this chapter by tests and simulation results. Last chapter resumes our conclusions and perspective of this thesis.

Chapter 1

From business process modeling to workflow analysis and optimization

1.1 Introduction

As we have said in the introduction, this thesis is a part of the GOCD project. It aims to install a new paperless workflow system and decision making tool for the French credit company COFIDIS. In the next section, we detail current mail flow within the enterprises COFIDIS. This will be followed by presenting a normal workflow reengineering process. A brief introduction of most recent standards and technologies used in this domain is presented followed by the motivation behind each choice. Challenges that hinder these choices are demonstrated, and a framework to overcome these problems is proposed. At each step of the proposed framework, we present in parallel its application on COFIDIS problem.

1.2 Problematic: mail flow optimization COFIDIS

Workflow is a key factor for the success of modern and developed enterprises. It permits the ease and the fast transformation of data and information between different internal and external enterprises actors, especially with the new globalized world and enterprises fusion tendency. However designing and implementing an efficient and optimized workflow is not trivial task and needs a lot of effort and analysis. In this work, we propose a new framework to model, optimize and implement company's workflow applied to a real case study of mails flow at COFIDIS.

Every day, the French credit company *COFIDIS* receives from the post office thousands of mails containing contracts and credit demands of different types. For facility, we will use the term contracts for both contracts and credit demands. Contracts types and the quantity of each type change from day to day. They can only be known in the morning of each day. This variation in quantities and types is due to marketing policies followed by COFIDIS. Each received contract should be handled by only one company collaborator. When contract treatment is started, it must be finished at the same day. Each collaborator has its own bin. This bin is filled in the morning with the daily loads of different contracts that must be handled throughout the day.

Each collaborator has different skills and experiences with respect to different contracts types. As a result, the needed time to handle a contract of one type changes from one collaborator to other. The expected time of a collaborator to handle contract of certain type is defined by a two dimensional competence matrix, where collaborators and contracts types represent matrix dimensions. This matrix is built according to the knowledge of company managers regarding their collaborators skills through time. Collaborators daily load may change from one day to another due to human resources con-

siderations. Thus, daily contracts load can not be estimated until contracts are assigned to collaborators. Each contract type has a different importance and profit to company activity. Handling contracts at their arrival day is always preferable.

Currently, company activities are distributed in many services and each service consists of many sectors. Each sector composed of collaborators and their supervisor.

Sectors supervisors are responsible to define the daily tasks table for their collaborators. This includes collaborator daily load, maximum/minimum load for each contract type and the number of allowed overtime hours for each collaborator by day. All this information is expressed in unit of time and they are used later to distribute contracts to collaborators in the contract assignment process.

An examples of competence matrix and tasks table can be seen in table 1.1 and table 1.2.

Collaborator ID	Type 1	Type 2	...	Type j
Col 1	2	1	...	4
Col 2	4	2	...	5
...
...
Col n	1	3	...	3

Table 1.1: Competence matrix example

Currently, contracts are distributed to company collaborators depending on the previous described tasks table and according to common rules i.e. "assign current contract to the first collaborators that is not overloaded, in the condition that his global load and his maximum load for the type of

Services	Sector	Col	Load	Type	Min	Max	Overtime
Service A	Sec1	Op1	70	T1	10	20	10
				T2	5	20	
				T3	15	20	
	Sec1	Oo2	65	T1	15	30	15
				T2	5	10	
	Sec2	Oo3	85	T1	0	20	15
				T2	10	30	
Service B	Sec1	Op4	60	T1	5	20	10
				T2	10	20	
				T3	15	20	
	Sec2	Oo5	70	T1	10	30	15
				T2	0	20	
	Sec3	Oo6	65	T1	10	20	10
				T2	20	30	

Table 1.2: Daily tasks table example for two services

the current contract are not violated". This distribution is not optimal, but hoped to be approximated to the optimal one. If collaborators capacity in some sector is overloaded, sector responsible should take the right decision to reduce company expenses. The decision can be either to allow collaborators to take their overtime hours, to postpone some contracts to the next days or to calls temporary workers to handle the overloaded contracts.

The major concern to any decision maker is to reduce company's expenses. To realize this in our problem, manager must know if company's current resources are capable to handle the totality of daily received contracts or not? And if not, what will be more expensive to hire temporary workers or

to pay the overtime hours for company collaborators? And to answer the last question, managers must be able to estimate precisely the exact number of temporary works that will be needed to handle the overloaded contracts.

Decision maker may desire to achieve other operational and marketing objectives, even if this will lead to non optimal distribution. One objective can be giving high priority to certain types of contracts that can not be delayed or to some contracts types that are considered more profitable to the company than the other types. Another objective could be to handle important contracts uniquely by company collaborators and not by temporary workers, since company collaborators have best experience and skills than temporary workers. Thus, guaranty service quality for these type of contracts. Load balancing between company collaborators can be significance objective from the social vision of point. Maximizing profitability rate by collaborator can be an interesting objective from economic vision.

1.3 Workflow reengineering

Workflow systems are considered as one of the key successes for modern and developed enterprises, especially with modern enterprise that took the choice of outsourcing or the choice of diffusion with other enterprises. This requires a flexible and a robust reengineering process that is capable to modify or even to completely replace installed workflow system with a new one without affecting enterprise activities.

In general, this process starts by modeling current workflow system. This gives a complete vision on the different steps that the workflow traverses from the beginning to the end of a process, which permits a deep analysis from operational point of view. This analysis permits to identify and underline

system dysfunctions and drawbacks. Knowing system deficiencies help system analyst and designers to conceive and design the new workflow system. The new model is then used as the base for developing and implementing the workflow system. Figure (1.1) shows the different steps in a classical workflow reengineering process.

In the next section, we present and discuss the most recent standards and technologies that can be used for modeling and implanting workflow systems.

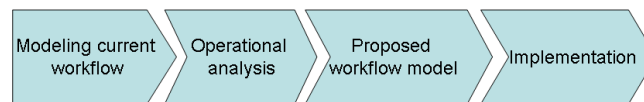


Figure 1.1: Workflow reengineering process

1.3.1 Proposed standards to be used in workflow reengineering process

First and third steps in the workflow reengineering process are modeling both the existed workflow system and the expected final workflow system. The choice of the modeling notation will have a direct impact on the workflow reengineering process. A complete and understandable notation will facilitate system analyst's work. For this reason, we propose to use Object Management Group (OMG)¹ adopted modeling notation for workflow and business process modeling the *BPMN*. In the next subsection, we present the origin and the motivations for choosing *BPMN* as a modeling notation.

¹OMG: is an international, open membership, not-for-profit computer industry consortium. OMG Task Forces develop enterprise integration standards for a wide range of technologies, and an even wider range of industries. <http://www.omg.org/>

For the implementation step, it is crucial to choose fixable technology that enables systems interoperability and integrity, especially with the new globalization world and the enormous development in networks and internet. Service Oriented Architecture (SOA) and web services appear as the new tendency to answer these questions. In SOA, business application is exposed as web services that propose one or more functionalities for each service. Still, creating and exposing services is not an easy task. We need to know how these services are organized and the dependencies between them. In addition, business processes are always the subject of changes as more and more enterprises take the choice of mergers and acquisitions. For the implementation of web services, the *Web Services Business Process Execution Language (WS-BPEL, known also as BPEL)* looks as the best answer as we will see later in this chapter. In the next subsections, we briefly introduce both BPMN and BPEL and the motivation behind each choice for both modeling and implementation business process and workflow. Thus, the graph in Figure (1.1) becomes the one in Figure (1.2) after adding the proposed modeling standard and technologies.

1.3.1.1 BPMN

BPMN is an acronym for Business Process Modeling Notation. It is defined as "a graphical notation that depicts the steps in a business process". This graphical notation is used to draw business processes in a workflow. It was developed by Business Process Management Initiative (BPMI) which has merged with Object Management Group since 2005. The main objective of BPMN was to have understandable notations for both business users (managers and employers) and system analyst and developers, in order to remove any confusion and facilitate ideas exchange between them. BPMN richness

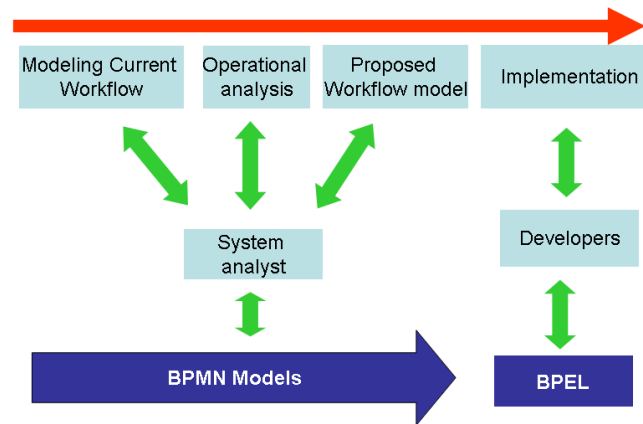


Figure 1.2: Workflow reengineering process with proposed standard and technologies

and completeness regarding all business process patterns is one of its most advantages over other modeling notation. Another important characteristic of BPMN is its ability to generate BPEL executable code from BPMN diagram. In the BPMN specification[OMG 06], they gives an ample on the transformation of BPMN diagram to BPEL process. However, this transformation is not completed and limited, which led to many other works in this domain as we will see later in this chapter.

The BPMI had combined the best ideas exist in other standards such as UML Activity Diagram, UML EDOC Business Processes, IDEF, ebXML BPSS, Activity-ecision Flow (ADF) Diagram, RosettaNet, LOVeM, and Event-Process Chains (EPCs) to create BPMN[Whit 04a].

BPMN is designed to describe and conceive only business process. Thus it does not cover other organizations model types such as enterprise organizational structures or data models.

Motivation

As we said before, BPMN is designed to deliver an understandable notation for all business users. The use of an understandable notation will reduce and remove any confusion between different system users in all levels. This includes business managers, system analysts, IT developers and simple employers. For this purpose, BPMN uses rich library of symbols that cover all business process details and can treat all workflow pattern defined by the Workflow Patterns initiative² [Whit 04b].

BPMN offers also the ability to have different end-to-end models. It can be used to model private business processes (internal workflow), abstract process (process sequence which informs external user in how to contact and communicate with the process), and finally interactions between two or more business entities (collaboration process with our partners).

Another important motivation for choosing BPMN as a modeling notation is that, it has already been adopted by more than 40 modeling software producer, and many other producers are planning to incorporate BPMN in they future products. Such enterprises that have already adopted BOMN are IBM, ILOG, IDS-Scheer, SAP, Intalio. As a result, more and more analysts and system architects have become familiar with BPMN and adopt it in theirs business process modeling. Another important motivation to choose BPMN for business process modeling is the possibility of generating automatic executable BPEL code from a BPMN diagram. Although the proposed transformation in BPMN specification is not complete, current researches are

²The Workflow Patterns initiative is a joint effort of Eindhoven University of Technology and Queensland University of Technology which started in 1999. The aim of this initiative is to provide a conceptual basis for process technology. <http://www.workflowpatterns.com/>

promising. We believe it is only a question of time before obtaining a complete code generated from a BPMN diagram.

But why to chose BPMN instead of UML activity diagram to model business process, knowing that UML is the most famous modeling standard for software development.

The answer is, while UML is mostly addressed to help in software reengineering and development, BPMN is designed to be used in business process management and to be understandable for both managers and developers. In other words, UML is used to model applications in an object orient approach and BPMN is used to model business process in a process-oriented approach. The two notations are not in competition, but they are two different views for the system.

BPMN Element

BPMN is designed to ease and simplify business process representation for both business managers and IT specialists. As a result, OMG has chosen in its specification[OMG 06] a set of distinguishable graphical elements that are famous and well known to most business process modelers. For example, business activity is represented by using rounded-corner rectangle, whereas decision is represented using diamonds shape. The different elements are organized in four categories:

1. Flow Objects
2. Connecting Objects
3. Swimlanes
4. Artifacts

In the next sections, we present each category in more details.

Flow Objects

In this category we find three core elements of BPDs, which are:

1. Events

An Event is represented by a circle. It can be of three types, Start, Intermediate or End event. Each event represents something that happens during business process. They can affect the flow of the process and usually have a cause or a result. See figure 1.3.

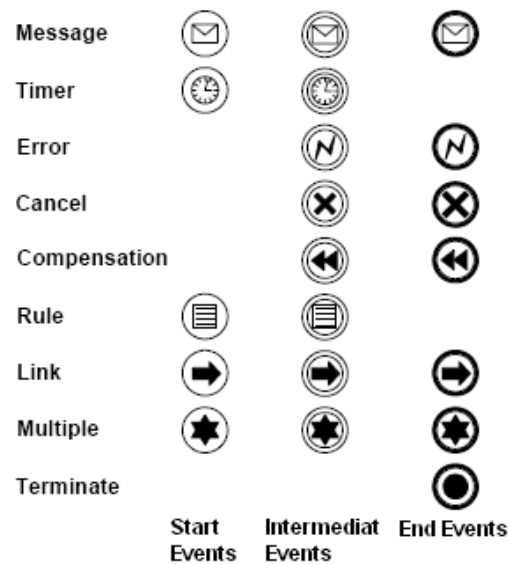


Figure 1.3: Events Objects

2. Activity

An Activity is a work or task within business process. It is represented by a rounded-corner rectangle and can be atomic or nonatomic. The different types of Activities are: Task, Sub-Process, loop task and sub-process loop task. See figure 1.4.

3. Gateways

A Gateway is represented as a diamond shape and is used to control

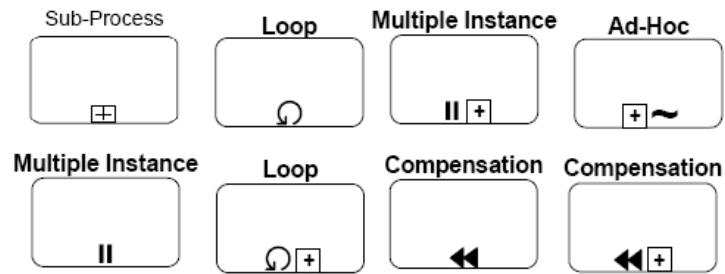


Figure 1.4: Activities Objects

the divergence and convergence of business process sequence flow, such as, the forking, merging, and joining of paths. See figure 1.5.

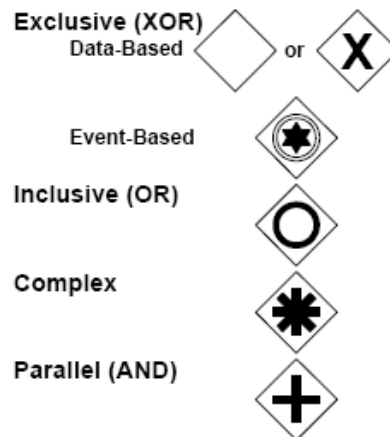


Figure 1.5: Gate Ways Objects

Connecting Objects

The connecting objects are used to connect Flow Objects within a diagram. There exist three types of Connecting Objects.

- Flow: represented by solid line and arrowhead to shows in which order the activities will be performed. To indicate a default choice of decision

a diagonal slash is used on the ongoing line.

- Message Flow: A Message Flow is represented with a dashed line and an open arrowhead. It shows what the messages flow between two process participants.
- Association: An Association is represented with a dotted line and a line arrowhead. It is used to associate an Artifact, data or text to a Flow Object.

See figure 1.6.

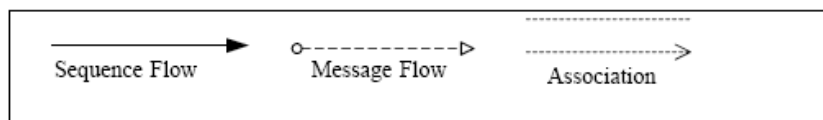


Figure 1.6: Connecting Objects

Swimlanes

Swimlanes are used to organize and group BPMN elements into separate visual categories to show different functionality or responsibilities. BPMN supports swimlanes with two main constructs. See Figure 1.7.

The two types of BPD swimlane objects are:

- Pool: A Pool represents a Participant in a Process; it is usually used in the context of B2B situations.
- Lane: Lane is a sub-partition within a Pool. Lanes are used to organize and categorize activities

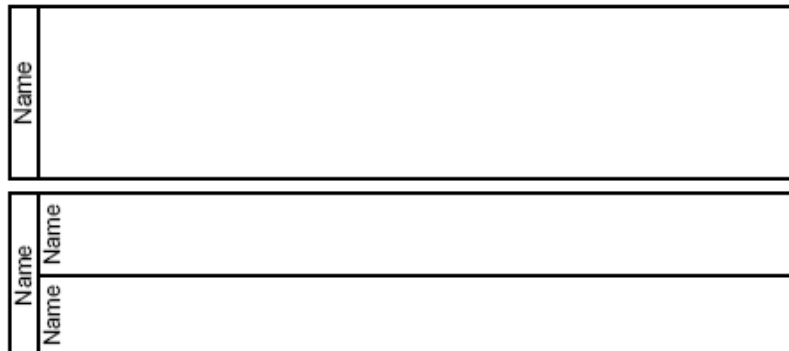


Figure 1.7: Swimlanes(Pool and Lanes)

Artifacts

Artifacts are used to add more information to BPMN diagram in order to clear the ambiguousness in the diagram. See figure 1.8.

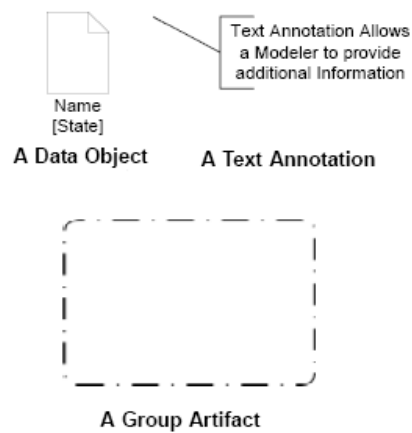


Figure 1.8: Artifacts in BPMN

- Data Object: used to explain which data is required in the diagram.
- Group: is used to group different activities without affect the flow in the diagram.

- Annotation: is used to give more information and commentaries about the diagram.

For the complete list, readers are referred to [OMG 06].

1.3.1.2 WS-BPEL

Service oriented architecture (SOA) and web services appear as the best answer to implement and realize company's application. In SOA, business processes are exposed as web services that can be integrated and used by different application and different users. This concept is similar to the concept already used by object brokers DCOM and CORBA. The Organization for the Advancement of Structured Information Standards (OASIS)³, has adapted the *Web Services Business Process Execution Language (WS-BPEL, known also as BPEL)* as standards for web services implementation. According OASIS, WS-BPEL is defined as "a language enabling users to describe business process activities as Web services and define how they can be connected to accomplish specific tasks[Arki 05]". WS-BPEL is an XML-based language enable task-sharing across multiple organizations using a combination of Web services. It uses the Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL), and the Universal Description, Discovery, and Integration (UDDI).

BPEL comes from a combination of two early workflow languages, XLANG language designed by Microsoft and Web Services Flow Language (WSFL) designed by IBM. The later based on the concept of directed graphs, whereas the former based on a block-structured language. In 2003, BPEL was submitted to OASIS for standardization, and this gave WS-BPEL more acceptances

³OASIS: is a not-for-profit consortium that drives the development, convergence and adoption of open standards for the global information society. <http://www.oasis-open.org>

in industrial world.

Many platforms support the execution of BPEL processes. Some of them provide graphical editing tools for defining BPEL processes. However, these tools reflect in general the syntax and the requirements of BPEL process. From the Commercial engines that support BPEL we mention, BPEL Process Manager "Oracle", WebSphere and BPWS4J "IBM", and Biztalk "Microsoft". Other open source engines include ActiveBPEL, PXE, Twister and BEXEE.

The advantages of WS-BPEL come from its ability to be used between or within different enterprises applications, where each application can be exposed as web services with its own functionalities. Since BPEL is designed to cope with the Service Oriented Architecture (SOA), this leads to standardize enterprises applications and increase their interpretability in efficient and easy manner. The increasing use of web services technology will in parallel increase the importance of WS-BPEL.

The BPEL process appears for the external world as one Web service. It uses several interfaces with a set of port types to communicate with the external services, and provides operations like any other Web service.

It can be either synchronous or asynchronous process. An asynchronous process is used for long time operations whereas synchronous process is used for operations that return a result in a relatively short time. When a synchronous process is used, it blocks its client until the process is finished and returns the result to the client. An asynchronous process does not block the client; instead it returns the results by a call-back.

BPEL follows the orchestration paradigm in implanting business process where a central process takes control over all involved Web services and coordinates operations execution on them. Involved web services have no

idea about how they are organized or what are the dependencies between them. This information is known only by process coordinator. See figure 1.9

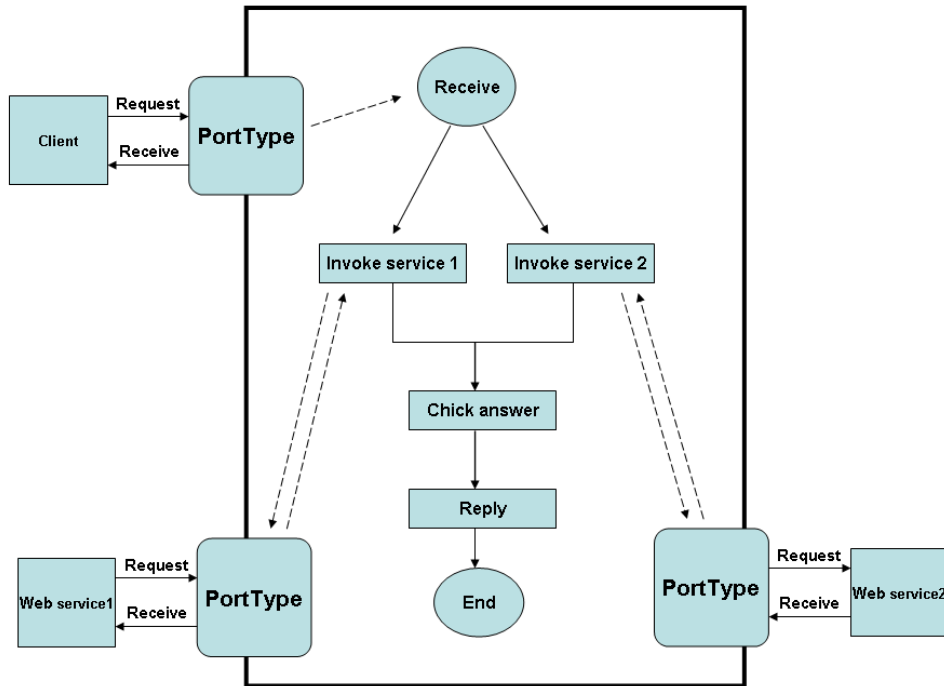


Figure 1.9: Example of BPEL Process

Since XLANG was in the foundation of BPEL, some common components between BPEL and XLANG can be remarked. Such component includes conditional expressions, structured loops, variables and handlers. These permit to built structural process.

A BPEL process consists of activities which can be either primitive (basic) activities or structured activities. Primitive activities are basic constructs and are used for common tasks. The primitive activities can be combined using structured activities to represent business process.

In the following we show some primitive activities and their functionalities:

- `<invoke>` invokes other Web services,
- `<receive>` waits client message to invoke the business process,
- `<reply>` generating a response for synchronous operations,
- `<assign>` manipulates data variables,
- `<wait>` used to wait some time.

Some structured activities are listed below:

- Sequence (`<sequence>`), which allows us to define a set of activities that will be invoked in an ordered sequence
- `<sequence>` allows activities to be invoked in sequential ordered,
- `<flow>` allows parallel invocation of a set of activities,
- `<switch>` for implementing branches,
- `<pick>` to select one of a number of alternative paths(conditions)

In order to communication with the external services, a BPEL process declares partner links. These partner links interact with external web services either by invoking operations on other web services, or by receiving invocations from clients. Each BPEL process has at least one client partner link, because it must have a client that invokes the BPEL process.

BPEL document validation

As we have said before, BPEL is an XML based language. And a valid BPEL document must correspond to the BPEL specification published by OASIS. This specification determines the structure, the tags, the attributes of a valid document and additional constraints written in a natural language. However, several individual attempts were done to defined a proper BPEL metamodel from its specification[Bare 06, Bord 04]. These attempts were neither complete nor constrained as the authors admit themselves. They represent only individual efforts to interpret BPEL metamodel from its specification. Others attempts were done within bigger software projects such as eclipse BPMN modeler. Yet none of these metamodel has been reviewed or standardized by OASIS consortium, and by consequence, they are not reliable and can not be considered as conformable to OASIS recommendations and specifications.

In [Akeh 04], the author uses the Unified Modelling Language (UML) and the Object Constraint Language (OCL) to provide a model for XML based BPEL languages. Using this model, the author shows how OCL can be used to precise the used natural language in OASIS specification. This work is used then to create a validation tool for any BPEL document. This tool reduces development time by ensuring the conformability of BPEL document in automatic manner.

1.3.2 BPMN to BPEL

One important motivation to use BPMN in business process modeling is the possibility to generate executable BPEL code from the BPMN diagram. This is a crucial step in the end-to-end development process for process-oriented systems. The mapping of BPMN to BPEL code is a challenge process,

since BPMN and BPEL represent two fundamentally different classes of languages (BPMN is graphical oriented languages and BPEL is block-structured language)[Reck 06, Wohe 05]. The reclamation of generating BPEL code from BPMN diagram in BPMN specification is not really accurate. The specification supposes that the designer will follow certain restrictions and rules, which do not reflect business process reality. For example, in the specification they suppose that every loop must have one entry point and one exit point. Another example, they suppose also that each AND-split corresponds to AND-join[Whit 04a, Whit 05]. Add to that, some control-flow patterns which are allowed in BPMN (such multi-merge, arbitrary cycle and spawning instances) can't be mapped to a single BPEL process. Some tools in the market propose an automatic transformation between BPMN and BPEL. However these tools are either incomplete (need human intervention) or impose certain restriction on BPMN diagram to facilities the transformation process. The use of unstructured cycles is a good example for these restrictions.

A transformation method is considered good, if the generated code is understandable for human and there is no need for a major developer's interventions. Having understandable code is important issue for developers who will refine and modify this code later. Modifying and developing code is indispensable since business process is in continuous change and progress.

Existing approaches to map BPMN to BPEL code are proposed by business process management group in Queensland University⁴ with the collaboration of Eindhoven University⁵ in Netherlands. These approaches can be are classified into three categories (event handler based transformation, pattern

⁴<http://www.bpm.fit.qut.edu.au>

⁵<http://w3.tue.nl/en/>

based transformation and control link based transformation). Each time a new approach is proposed, it aims to improve the readability and the completeness of others approaches.

1.3.2.1 Event handler based transformation

The first proposed transformation approach depends on BPEL event handler construct. This approach is applicable only to a core subset of BPMN and in the condition that the diagram doesn't contain live-lock. This approach is continued to be used in every later works, as there is always some parts of BPM diagram that can not be translated with the new approaches [Ouya 07, Ouya, Ouya 06a, Ouya 06b].

The importance of BPEL event handler comes from the possibility to have multiple simultaneously active instances in a single process instances. Each event handle in BPEL process is associated with a scope and it is enabled when the associated scope is under execution. When an associated event is triggered for some reason the body of its handler is executed. This approach is consists of three steps

1. Finding the precondition sets for all activities
2. Translate the precondition set into event condition rules (ECRs)
3. Translate ECRs into BPEL code

A precondition set is a combination of events and conditions to be held in order to execute certain activity in a process. For every activity several precondition set and several ways to execute it can be found. For more details, readers are refereed to [Ouya 06a].

1.3.2.2 Pattern based transformation

Well-structured pattern based translation

Even the event handler based technique works well in generating BPEL code from BPMN diagram, still the generated code is unreadable and very complicated for developers who will work on it later. In [Ouya 06b, Ouya 06c], they proposed a new approach based on exploiting BPEL structural nature to obtain more readable code. The idea is to discover structured components (patterns) that can be mapped directly into BPEL construct without any modification, and to use the ECR to map the rest of the diagram. For this end, the diagram is divided into well-structured component and non-structured component according to precise rules and definitions. In this transformation, we consider only a well-formed core of Business process diagram (BPD). A BPD is a well-formed core if its elements satisfy the following conditions:

- Start events have an indegree of zero and an outdegree of one,
- End events have an outdegree of zero and an indegree of one,
- Task and intermediate events have an indegree of one and an outdegree of one,
- Fork or decision gateways have an indegree of one and an outdegree of more than one,
- Join or merge gateways have an outdegree of one and an indegree of more than one,
- Event-based XOR decision gateways must be followed by intermediate events or receive tasks,

- There is a default flow among all the outgoing flows of a gateway,
- Every object is on a path from a start event to an end event.

From the well-formed diagram, seven well structured components can be found. These components can then be mapped into the corresponding BPEL construct. The example in figure 1.10 demonstrates this mapping which is presented in [Ouya 06b].

In this approach, they propose also the use of *FOLD* function which replaces every well-formed component by a task object. This task object can be then used to perform iterative reduction of a componentized BPD until no component is left. Finally, ECR method is used to transfer the rest of BPD into BPEL code.

Figure 1.11, demonstrates the use of *FOLD* proposed in [Ouya 06b].

Quasi-structured pattern transformation

A new attempt to have more readable code is demonstrated in [Ouya 07]. In this paper, we do not only try to detect the perfect structured components, but also we search quasi structured component that can be redefined and modified in order to be transformed by the previous mentioned technique. The modified components should not affect the original process semantic. As an example, we can modify the incompatible part of the BPMN diagram by split a gateway into two gateways. In this paper, three types of quasi-structured components are refined (FLOW, SWITCH and PICK component). See figure 1.12.

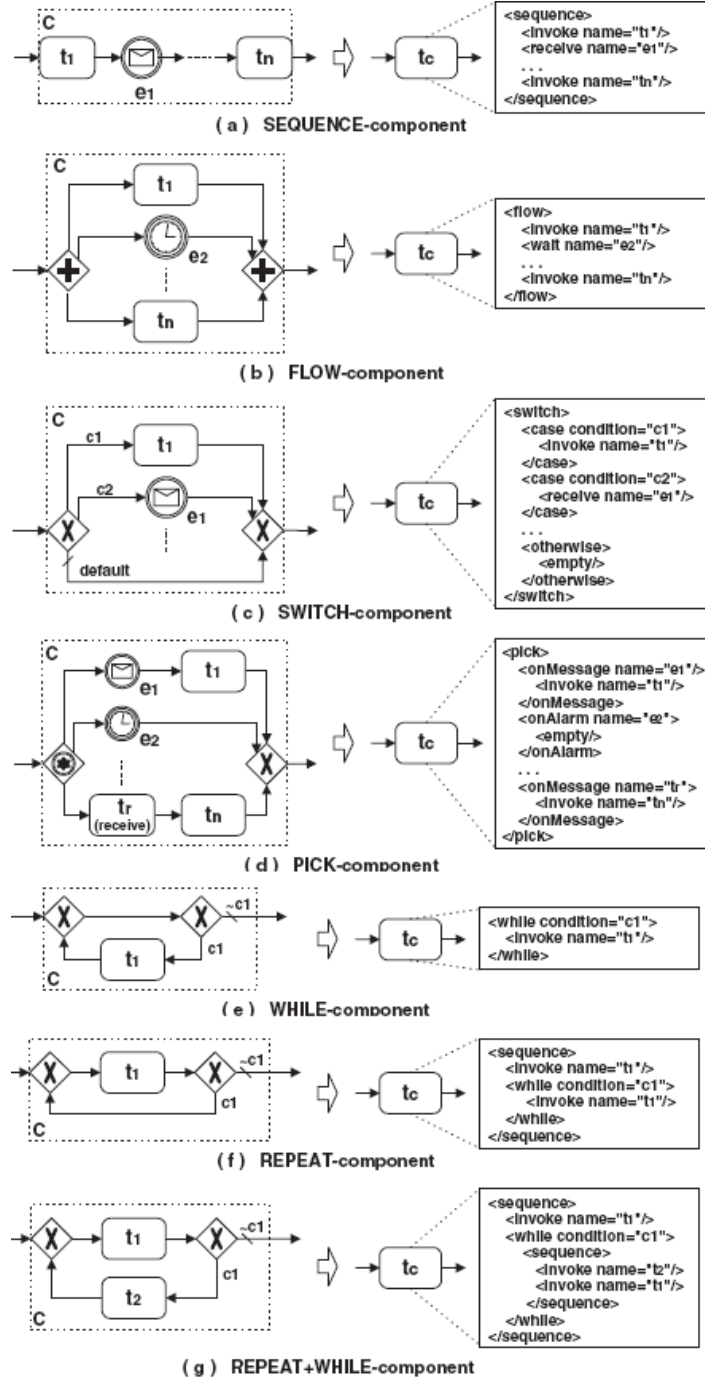


Figure 1.10: Mapping well-structured components into BPEL

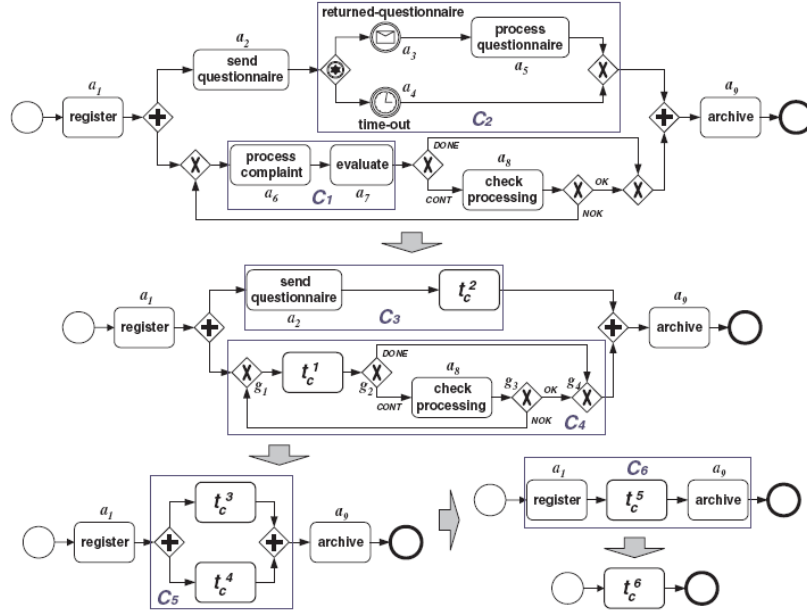


Figure 1.11: A complete example in using the FOLD function

Control link based transformation

Although the fact of using well-structured and quasi- structured components has increased code readability, it doesn't solve the problem of acyclic BPMN diagram. So, another BPEL construct was searched to solve this problem and the use of the non-structured BPEL construct the *control links* is proposed. When a control link exists between activity A and activity B , it indicates that activity B cannot start before activity A , unless activity A has been completed or skipped. And to execute activity B , its associated join condition must be evaluated to true. The join condition is represented by tokens carried to activity B . The token is true if its propagated node had been executed, if not (the node is skipped) the token is false. The advantage of this construct lays in its ability to define directed graphs. It is important before mapping a graph to ensure its soundness (no deadlock) and safeness (no mul-

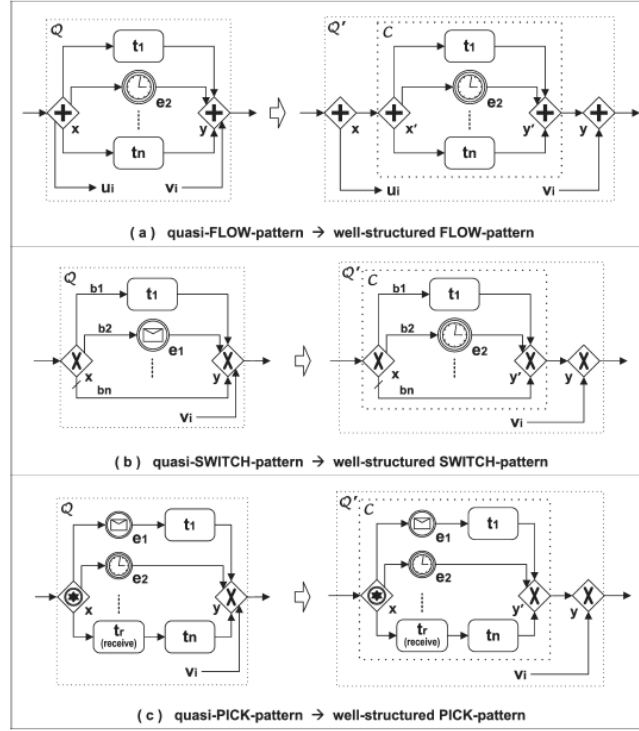


Figure 1.12: Quasi Structured components transformation

tuple instances of the same activity are executed concurrently). Figure 1.13 demonstrates how could acyclic component be neither safe nor sound when the condition at G2 is false.

Since BPMN has no formal semantic, Petri-net is used in [Ouya, Dijk 07] to define a formal semantic for a sub set of BPMN and to verify its soundness and safeness. Component that can be transferred using the control link construct is called (Synchronizing process component). The synchronizing process component is transformed into BPEL code by translating the control-flow relation between all its tasks and event objects into a set of control links.

For a complete example and details on using control link construct in mapping acyclic BPMN, we refer reader to [Ouya, Ouya 07].

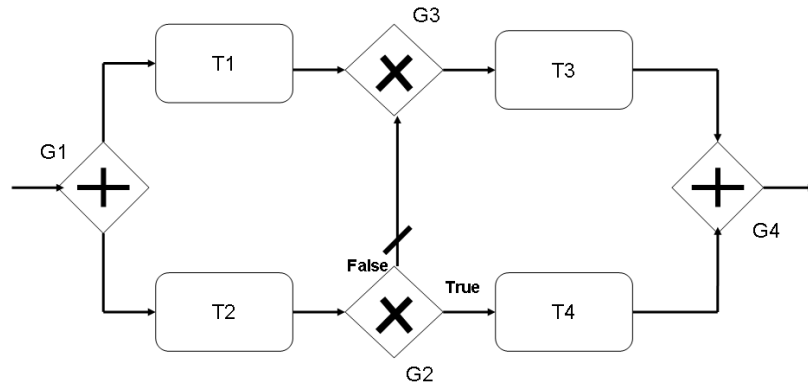


Figure 1.13: Example of acyclic component neither safe nor sound

1.3.2.3 BPMN2BPEL tool

In the previous section, we presented the different approaches used to transform BPMN models into BPEL process. These approaches based mainly on recognizing BPMN well-structured patterns and transform them into the suitable BPEL constructs. These approaches are grouped and implemented in one open source tool called BPMN2BPEL⁶.

This tool takes BPMN model as an input and it outputs a BPEL process. The input file must conform to a particular XML format representing the BPMN meta-model created by BPM group at Queensland University and inspired from BPMN specification. BPMN2BPEL tool integration to any others modeling tools is not easy since BPMN2BPEL uses a particular XML format to represent BPMN and there is no exchange format in BPMN specification. As a result, each BPMN modeling tool defines and uses its

⁶<http://www.bpm.fit.qut.edu.au/projects/babel/tools/>

own BPMN meta-model and exchange format and requires an additional step before using BPMN2BPEL tool in their products. This step consists of transforming BPMN model obtained from other modeling tool to the corresponding BPMN2BPEL XML input format. Pau et al. present in [Gine 07] a model to model transformation to bridge this gap. This approach is currently applied to eclipse Service-oriented architecture Tools Platform (STP)⁷. It uses BPMN STP modeler to create BPMN models and uses ATLAS Transformation Language⁸ (ATL) to transform these models into the corresponding BPMN2BPEL model. Figure 1.14 depicts the needed step before using BPMN2BPEL tool.

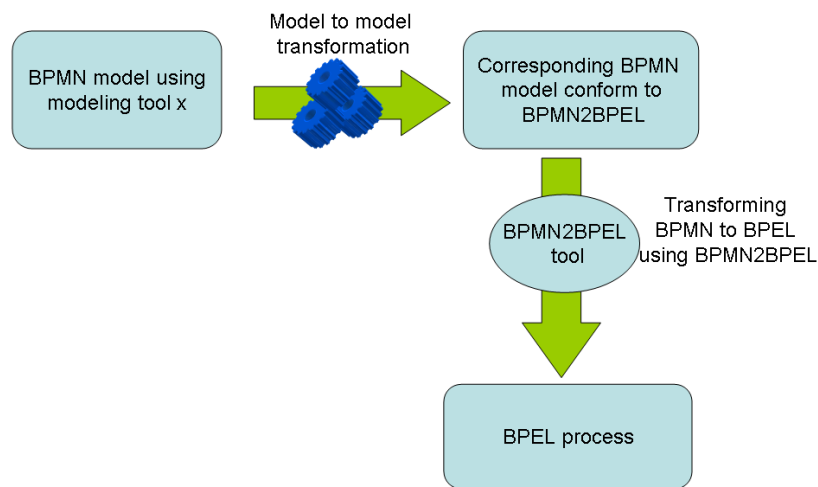


Figure 1.14: Model transformation in order to use BPMN2BPEL tool as proposed by Pau et al.

⁷ <http://www.eclipse.org/stp/>

⁸ A model transformation language and toolkit developed by the ATLAS Group (INRIA and LINA) <http://www.eclipse.org/m2m/atl/>

1.3.3 BPMN limits

In spite of BPMN advantages, BPMN suffers from serious problems regarding its semantic. The last BPMN specification (V1.2 January 2009), does not contain a formal semantic definition for BPMN [Ouya]. This specification is written in verbal way which allows different interpretations of the same pattern. Thus, the flexibility offered by BPMN can also lead to undesirable properties for business process such as deadlocks and livelock. Since one of the important objects of using BPMN is to generate executed BPEL code, any problem in BPMN model will be directly reflected in the obtained code and in the implemented process. More, since BPMN is a process oriented notation, little attention was dedicated to represent data or resources availability [Aals 06]. As a result, one can not realize performance analysis or evaluation for the target system. Verifying that the new workflow system works correctly is a good thing, still we need to ensure that it works efficiently. Recently, OMG has issued request for proposals for version 2.0 of BPMN. The new BPMN 2.0 must contain a dedicated meta-model for the BPMN, a graphical notation to represent different business processes, an interchange format to exchange business process models without losing semantic integrity and an extension to model a choreography process. The debate about BPMN metamodel is not terminated. And the final decision regarding BPMN metamodel that will be taken by OMG is not known. In the next sections we propose an approach to avoid these problems.

1.4 Proposed framework for workflow reengineering

1.4.1 Framework for workflow reengineering

As we have discussed in section 1.3.1.1, BPMN has been adopted by the OMG as the standard to model and represent business process workflow. This makes it the natural modeling choice for any business analysts to model and represent business process.

However, BPMN suffers from serious problems due to the lack of formal semantics. Within the last BPMN specification (V1.2 January 2009), OMG does not define a dedicated meta-model for BPMN. This specification is written in verbal way which allows different interpretations of the same pattern. The flexibility offered by BPMN can also lead to undesirable properties for business process such as deadlocks and unreachability. Since one of the important objects of using BPMN is to generate executed BPEL code, any problem in the model will be directly reflected in the obtained code and in the implemented process. Unfortunately there is no way to verify BPMN structural properties.

Recently, OMG has issued request for proposals for version 2.0 of BPMN. The new BPMN 2.0 must contain a dedicated meta-model for the BPMN, a graphical notation to represent different business processes, an interchange format to exchange business process models without losing semantic integrity and an extension to model a choreography process. BPMN acronym in the new BPMN 2.0 will stand for Business Process Metamodel and Notation and no more to Business Process Modeling Notation.

The debate about BPMN metamodel is not terminated. And the final decision regarding BPMN metamodel that will be taken by OMG is not

known. Will OMG keep BPDM as BPMN metamodel or they will adopt a new dedicated metamodel for BPMN. To answer this question we must wait OMG BPMN 2.0 specification.

Another issue in BPMN diagram is that there is no way to perform any performance evaluation and optimization to test and compare the proposed workflow model presented by BPMN with the old one. Verifying that the new workflow system works correctly is a good thing, still we need to ensure that it works efficiently. Thus, we need more analysis to identify workflow shortcoming and bottlenecks. As a consequence, system designers and analyst can solve and optimize these problems with the best manner.

To answer these issues, we propose at first to use BPMN to model the *as-is* workflow. This will allow a deep understanding of the current workflow and permits more precise operational analysis to design the wished *to-be* workflow. The new workflow is then passes through an additional phase of verification, validation and optimization, as we have proposed in [Shra 09], before being adopted to generate BPEL code. This phase can be realized by transforming BPMN model to any of modeling languages with strong formal semantics. Our choice in this work was to use Petri net as the target formal language. However, other important works in this domain exist. One of them depend on the transformation of BPMN into the Calculus of Orchestration of Web Services (COWS) [Pran 08]. The transformation of BPMN to a set of Communicating Sequential Processes (CSP) process and events [Wong 08] is another attempt in the domain. This method is suitable when BPMN model size is small, since the size of CSP model produced from a small BPMN model can be considerable. In [Puhl 06], static analysis is proposed after transforming a subset of BPMN to π -calculus. However this method does not cover error handling that can be represented in BPMN and this approach

is restricted only to a very small size of BPMN models.

The choice of using Petri nets is justified by that i) Petri nets are particularly suitable to model systems behavior in terms of flow. ii) Since BPMN is flow-oriented, Petri nets seems to be the natural candidate for formally defining BPMN models semantics. iii) Petri nets has taken a lot of attention and has been studied from a theoretical point of view for several decades. And there exist many tools that able to perform automated analysis on it.

In the next subsection, we present briefly the Petri nets. Readers interested in more details about Petri nets are refereed to[Mura 89, Jens 03]. Readers familial with Petri nets can skip this subsection.

After introducing Petri nets in our framework, the final framework phases to install a new workflow system become as demonstrated in Figure 1.15. In the next subsection, a brief introduction to Petri nets is presented

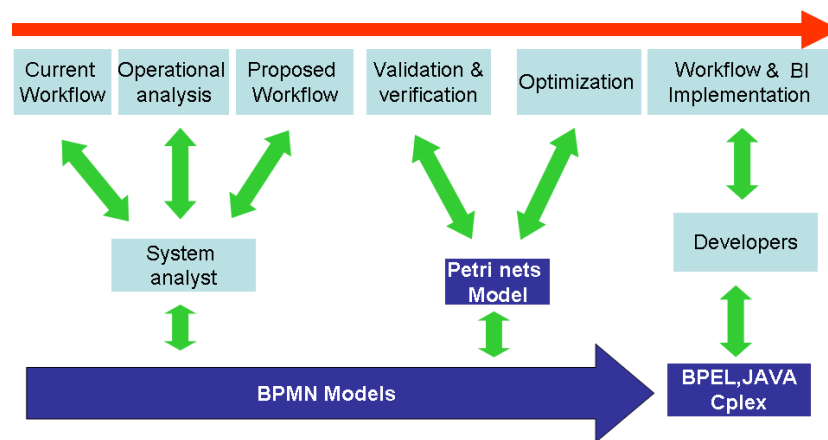


Figure 1.15: Proposed framework for workflow reengineering

1.4.2 Petri nets

Petri nets took its name from the early work of Carl Adam Petri[Petr 62]. Since that, this work was extended and studied extensively. For more detailed bibliography, readers are referred to[Mura 89]. A classical Petri net is a directed bipartite with two types of nodes called place and transition. Places are connected to transitions by directed arcs. Places are represented by circles whereas transitions are represented by rectangles.

Petri net places can contain zero or any positive number of *t tokens* at any time. Tokens are represented by black dots in the places. The state of a Petri net; also known as marking M , is the distribution of tokens over Petri net places.

A Petri net state changes when at least one transition fires. A transition may fire if it is enabled. A transition is enabled iff each input place p of t contains at least one token. When a transition t fires, it consumes one token from each input place p of t and produces one token in each output place.

The example in Figure (1.16) represents a Petri net before and after transition firing. The Figure (1.16).a, shows the system before transition t_1 is fired. The marking of the Petri net before firing is: $3p_1+1p_2+1p_3$. Figure (1.16).b, represents the same Petri net but after t_1 has fired. So, the new marking for this net becomes $2p_1+0p_2+2p_3$.

Formally, Petri net is 4-tuple $N = (P, T, Pre, Post)$, where

- P : is a finite set of places ($|P| = n$),
- T : is a finite set of transition ($|T| = m, P \cap T = \phi, P \cup T \neq \phi$),
- $Pre(Post)$: is the pre-(post-) incidence function representing the input (output) arcs, $Pre : P \times T \rightarrow IN = \{0, 1, 2, \dots\} (Post : P \times T \rightarrow IN)$.

For a given Petri net $N(P, T, Pre, Post)$ and marking M_1 :

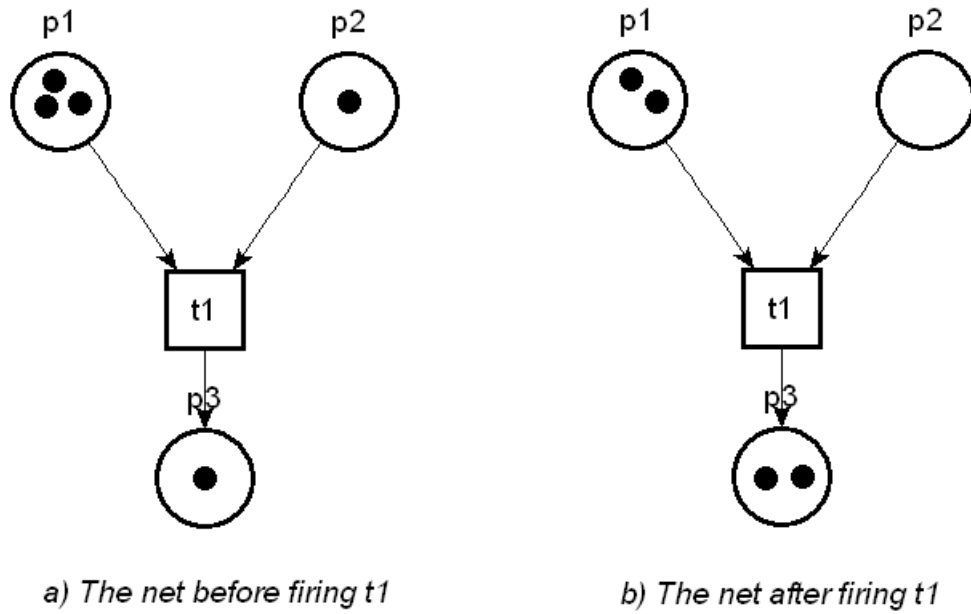


Figure 1.16: An example of Petri nets.

- $M_1 \xrightarrow{t} M_2$: transition t is enabled in state M_1 and firing t in M_1 results in state M_2
- $M_1 \xrightarrow{\sigma} M_n$: the firing sequence $\sigma = t_1 t_2 t_3 \dots t_{n-1}$ leads from state M_1 to state M_n , i.e., $M_1 \xrightarrow{t_1} M_2 \xrightarrow{t_2} \dots \xrightarrow{t_{n-1}} M_n$

A state M_n is called reachable from state M_1 (represented by $M_1 \xrightarrow{*} M_n$) iff there is a firing sequence $\sigma = t_1 t_2 t_3 \dots t_{n-1}$ such that $M_1 \xrightarrow{\sigma} M_n$.

Ordinary nets are Petri nets whose *pre* and *post* incidence functions take values in $\{0, 1\}$. The incidence function of a given arc in non-ordinary nets is called *weight* or *multiplicity*.

A place p is called an input place of a transition t iff there exists a directed arc from p to t . The notation $\bullet t$ is used to denote the set of input places for a transition t . At the same time, a place p is called an output place of transition t iff there exists a directed arc from t to p , and the notation $t \bullet$ is

used to denote the set of output places for a transition t . Similarly, $\bullet p$ is used to denote the set of input transitions for a place p , and $p\bullet$ is used to denote the set of output transitions for a place p . Formally, the *pre-* and *post-sets* of a transition $t \in T$ are defined respectively as $\bullet t = \{p | Pre(p, t) > 0\}$ and $t\bullet = \{p | Post(p, t) > 0\}$, and the *pre-* and *post-sets* of a place $p \in P$ are defined respectively as $\bullet p = \{t | Post(p, t) > 0\}$ and $p\bullet = \{t | Pre(p, t) > 0\}$.

$C = POST - PRE$ is a $n \times m$ incidence matrix for the net N with n places and m transition. The vector $X \geq 0$ is called *T-semiflow* if $C.X = 0$, and the vector $Y \geq 0$ is called *P-semiflow* if $Y^T.C = 0$.

Liveness is one of the most important Petri nets properties. A Petri net N is called live iff for every reachable state M' and every transition t , there is a state M'' reachable from M' which enables t .

Boundness is another property for Petri net. A Petri net N is bounded iff for each place p there is a natural number n such that for every reachable state the number of tokens in p is less than n . The net is called *safe* iff for each place the maximum number of tokens does not exceed 1.

A subclass of classical Petri nets class is *Workflow Nets*. A workflow net is a Petri net with a clear starting and ending point; the start place i and the end place o . It is used as a tool for the representation, validation and verification of workflow procedures, see Figure (1.17).

Every other place or transition is on a path between the start place and the end place. For workflow nets, the initial marking is a marking where there is one token in the initial place of the process. The final marking is a marking with a token in the final place of the process. A strong correctness criterion of workflow nets is the soundness property. Workflow Net is sound if i) the token in the source place ends in the sink place (consumed by the sink place) after a sequence of firing and there is no other token in the Nets.

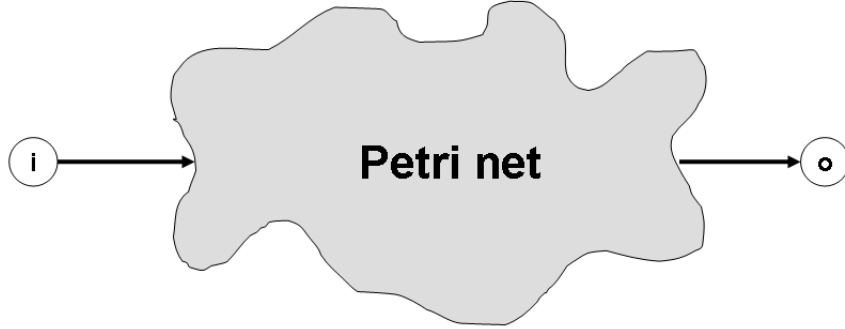


Figure 1.17: Workflow Net

ii) there is no dead transition in the workflow Nets. Van der Aals and al. have defined in [Aals 00] an extended Workflow nets \overline{N} , which represents an extended Petri nets for a workflow Nets N with additional transition that linking the source and the sink places. They prove that workflow Net N is sound iff \overline{N} is live and bounded. This definition will be used to verify our workflow process.

A strong correctness criterion of workflow nets is the soundness property. A process is sound when for each marking that can be reached from the initial marking, the final marking is reachable.

However, using classical Petri to describe COFIDIS contracts workflow tend to be complex and extremely large and does not allow modeling data or time. For this end, one can use any of the many extensions of the classical Petri net. In this thesis, we are interested in only two extensions. (i) The colored Petri nets (CP-nets or CPN), used to model data, (ii) Timed Petri nets used to represent time. A Petri net extended with color or time is called a *high-level Petri net*. In the next subsection, we present these Nets in an informal manner.

1.4.2.1 Colored Petri nets

In colored Petri nets, tokens often represent objects in the modeled system. These objects in general have attributes. If a credit demand arrives to a credit company, the demand is modeled by a token in the Petri net, and we want other information such as, client name, identification number, and amount. These attributes can not be easily represented by a token in a classical Petri net. By using colored Petri nets, each token has a value referred to as 'color', and transitions determine the values of the produced tokens depending on the basis of the values of the consumed tokens. This means, each transition describes the relation between the values of the consumed token 'input tokens' and the values of the produced token 'output tokens'. Preconditions on the tokens values can be set and the transitions will fire only if these preconditions are satisfied. For more details on colored Petri nets, readers are referred to [Jens 03]

1.4.2.2 Timed Petri nets

In timed Petri nets time is used to describe system durations and delays (temporal behavior). Time can be introduced in Petri nets in many ways. It can be associated to tokens, places, and transitions or to any combination of them. In this work, we are interested by time associated to transition. For this type class of Petri nets, for each transition t we define a lower bound d and an upper bound D of time, $[d, D]$ with $d \leq D$. The lower bound d must be ≥ 0 and the upper bound D must be $\leq +\infty$. When the transition t is enabled, it can fire only when d time units are passed and before D time units are reached. For example, in Figure (1.18), If one token arrives at $P0$ at time 3 and another token arrives at $P1$ at time 5, then the transition $t0$ can fire only after 8 unit of time. If a transition has no explicit temporal

constraint, then $[0, +\infty]$ is assumed by default.

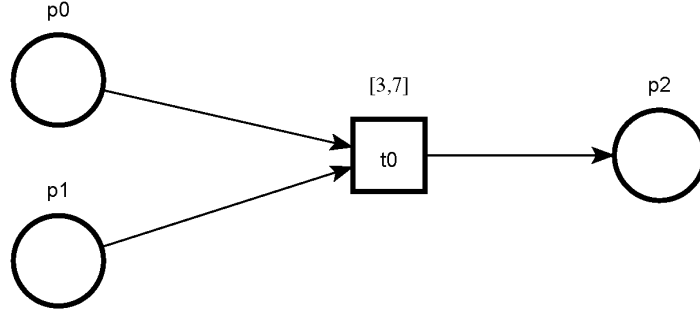


Figure 1.18: Timed Petri nets

1.5 Modeling and operational analysis for the current workflow at COFIDIS

This section, we demonstrate the first three phases of the proposed framework on real workflow reengineering process (COFODIS mail flow). The fourth and the fifth steps will be discussed in details later in the next sections.

In order to ease workflow analysis and comprehension, we model the current workflow "*the as-is workflow*" at COFIDIS that we have described in section 1.2 using BPMN, see Figure 1.19. This model describes in details the different phases that mails have to pass in the current workflow system to handle the daily received mails. As we can see from Figure 1.19, each mail must be opened and classified by type of contracts. After that, contracts are sent to the decision makers in order to assignment the contracts. When a collaborator receives a contract, he checks if the contract contains all the documents needed to correctly handle the contract. If any document is missed, collaborator contact contracts sender to reclaim missed document. At the

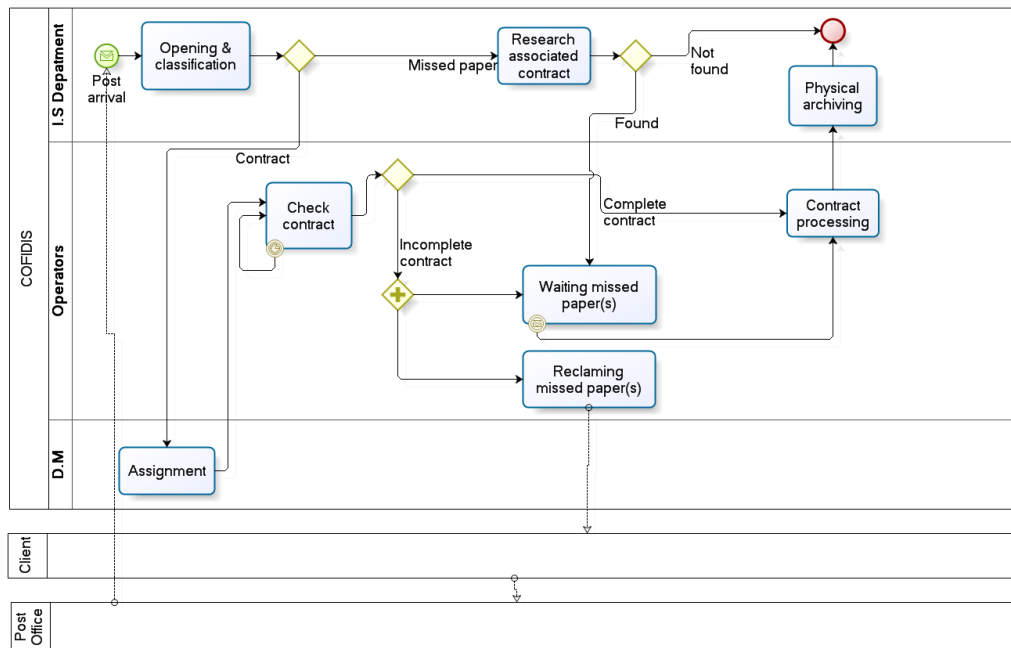


Figure 1.19: Current mails workflow within COFIDIS

same time, the collaborator keeps the contracts on his desk waiting for the missed document. From this process is easily to notice that joining received missed documents to the corresponding contracts is a hard task and time consuming, since there is no trace for the contract after it has been assigned. More, the current contracts flow is performed completely in manual manner. Thus, losing some contracts components or even the entire contract is very probable, especially for those incomplete contracts. And the worst thing in the current workflow is that decision maker does not have any information about the contract process progress within the different phases of contracts handling.

Another important problem appears concern collaborators capacities. Although the competence per contract type for each collaborator are defined in the competence matrix, in reality collaborators may spend less or more time

than expected for each contract. Especially for incomplete contracts that will need properly less time than expected if they were completed ones. Thus, if the percentage of the daily incomplete contracts is (α) , then the percentage of complete contracts will be $(1 - \alpha)$.

As a result, collaborators real capacities are not fixe and vary during the day. This means, contracts assignment that has been done in the morning is no more optimal (if it was supposed to be optimal) during the day.

Another important problem is contracts assignment policy. This assignment policy is not optimal, but hoped to be near to optimal. It depends on manager experience. Bad contracts assignment may force managers to call unnecessary temporary workers to process all received contracts. Each unnecessary worker represents a considerable additional charge for the enterprise. Moreover, current contracts assignment is a static and irreversible. Once contracts are assigned, no later modifications can be realized in order to cope with any eventual work necessity or events e.g. the eventual departure of some collaborators for personal reasons. Finally, incomplete contracts kept by collaborators on their desk may represent additional load if the missed documents of incomplete contract arrive. This additional load is not counted in the current assignment method.

In order to overcome the previous mentioned problems and optimize contract handling process, we propose to replace current paper flow system by a new automated paperless workflow system. Figure 1.20 demonstrates in details the proposed workflow.

To ensure contracts integrality and traceability, received contracts with theirs attached documents are scanned at their arrival to form one integral electronic pack. Each electronic pack is then given a unique id (barcode). This id will be used to trace and follow contract progress from its arrival

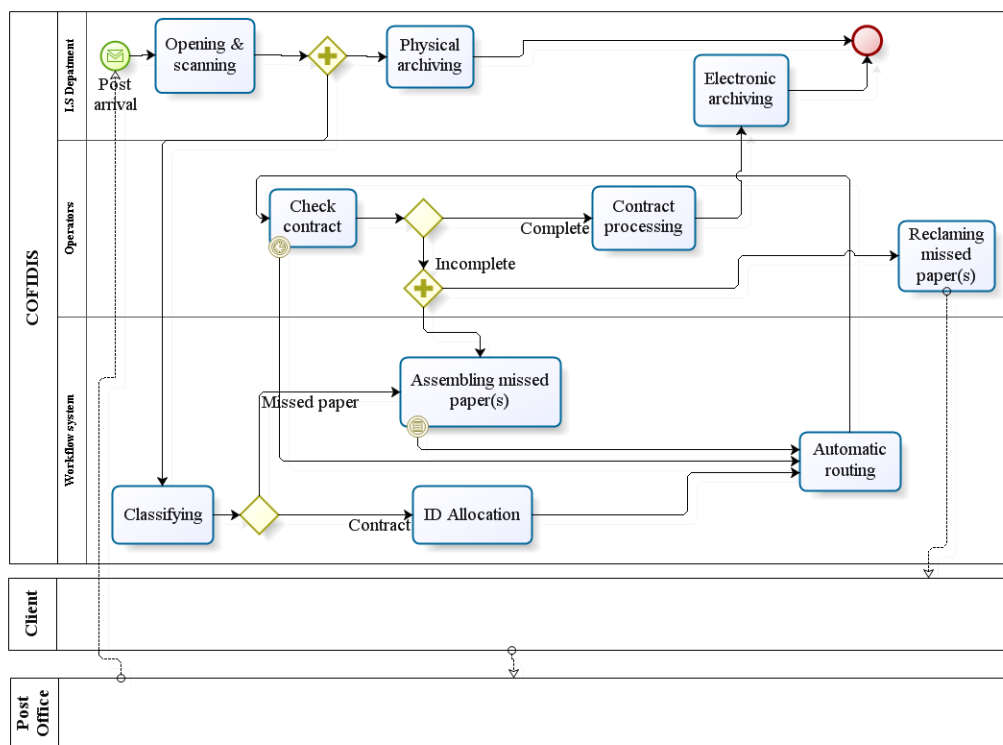


Figure 1.20: New proposed mails workflow

through handling process and ending by archiving. When some documents are missed for a contract, enterprise collaborator contacts concerned client by mailing him a list of missed documents. This mail contains pack id in which missed documents must be integrated. If a received mail contains new contract, it enters the assignment process. If a mail contains missed paper with a pack id, it is assembled with the corresponding pack automatically, otherwise enterprises worker search manually the corresponding pack by using enterprise information system and the client information in the missed document. This process is time consuming and may delay some contracts assemblage for one or more day.

When an incomplete contract is received, it is sent for being assembled and waits the arrival of missed documents. History for each contract is kept for further use. This history contains helpful information such as the collaborator that has already handled the contract in the past. Once missed documents are assembled to their pack, the contract is injected in the assignment process. Each incomplete contract can be reassigned to the same collaborator that has already started processing it, or assigned to a new different collaborator. In the first case, expected processing time registered in the collaborator competence matrix is reduced since the collaborator has already studied this contract. In the second case, collaborators competences remain intact as this will be the first time they will process the contract.

In the next two sections, we continue demonstrating the fourth and fifth phases of our framework by using always COFIDIS contracts flow example. These sections will deal with validating and optimization of BPMN models.

1.6 Workflow analysis

In this section, we discuss and explain how to employ Petri nets to analyze, verify and validate BPMN model. This additional phase can reveal serious problems and system deficiencies that were hidden by BPMN model. Finding system deficiencies and problems will help system designers in solving and improving them in the modeling phase and before system implementation. This means, less modifications and debugging time, and so, reduced cost.

In literature, many efforts exist in the transformation of BPMN model to the corresponding Petri nets or one of its extensions. Raedts and al[Raed 07] transform BPMN model to the an extended Petri nets known as YASPER (Yet Another Smart Process EditoR). This transformation is realized in order to verify soundness property. However, no others important analysis are possible directly. In stead, authors propose to transform YASPER to classical Petri nets and then perform the wished analysis. Another attempts were done using a new promising extension to Petri nets called YAWL (Yet Another Workflow Language)[Aals 05, Deck 08, Ye 08]. YAWAL is a new workflow modeling language with formal semantics that extend Petri nets. It was developed by Eindhoven University of Technology and Queensland University of Technology. However, verification YAWL model is computationally more complex than the corresponding Petri nets.

The most remarkable work on the transformation of BPMN to Petri nets is that one done by Dijkman and al in[Dijk 07]. The authors extend BPMN transformation to Petri nets that was already initiated in[Ouya] to ensure a core subset of BPMN is deadlocks and livelocks free before transforming it to BPEL. The authors succeeded in transforming most BPMN objects such as subprocess and exception handling. Message flow and initial state of BPMN model have been also included in their paper. Still, the mapping does not

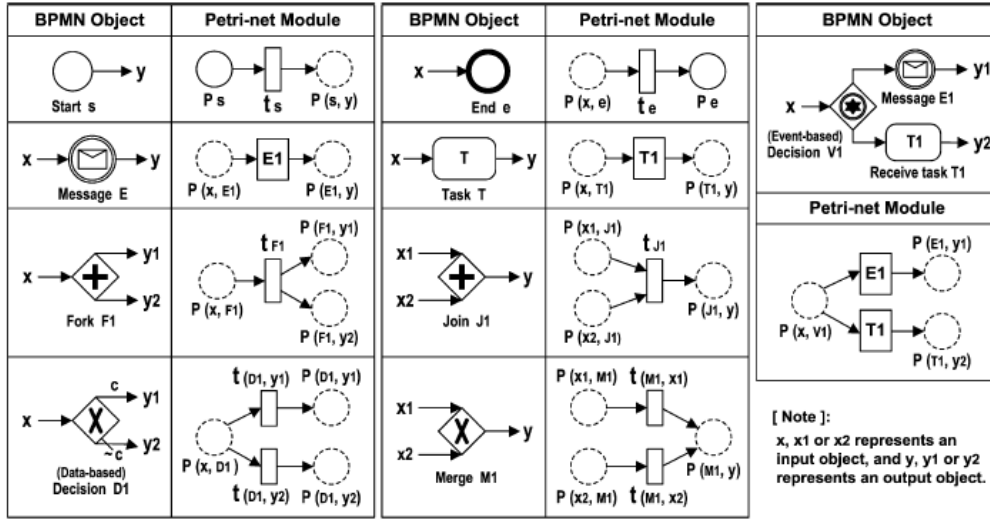


Figure 1.21: The mapping of task, events, and gateways to Petri-net as proposed by Dijkman

cover some workflow patterns such as parallel mutli-instance due to Petri nets limitation. In this paper, we choose to use the classical Petri nets, since our BPMN model does not contain any of workflow patterns that are not covered by the Petri nets.

In Figure 1.21, we can see the main BPMN objects that had been transformed by Dijkman and al in[Dijk 07].

In this work, we are concerned in both behavioral properties such as Reachability, liveness and boundedness and in the structural property the P-semiflow. Behavioral properties permit testing and analyzing all possible scenarios (markings) that the new system may reach from the initial state. As a result, serious problems in the designed system can be revealed in the modeling phase. These problems are in general produced either by human mistakes or by bad interpretation of BPMN specification. System reachability is important to ensure that every possible task designed in the new

workflow system will be executed if the corresponding conditions and circumstances are held. Still, it is not sufficient only to ensure system reachability without guaranty that the system is live (deadlock free). Liveness is important to ensure that the new workflow system will never arrive at a point where it can not be advanced (blocked). Finally, the boundedness property and its particular case the safeness is important to verify that the model will handle only one taken each time.

The Figure 1.23, gives an example of a BPMN model with some errors and the corresponding Petri nets result from the transformation proposed in [Dijk 07]. This example represents the process of funding a project proposed by one of enterprise clients. To accept funding this project, two conditions must be held. The first condition is to convince the enterprise by the project marked study, and the second condition is to have available financial resources at the enterprise to fund the project. It is clear in this example that if one of the two conditions is validated to true and the other is validated to false the process will be blocked (deadlock). The example is a simple and can be verified manually in short time. In business real world, a process is in general bigger and more complicated, which requires more time and effort to verify any of the past mentioned properties. However, current transformation will result in unreachable marking after process termination. To avoid this problem, the end places of the model are connected to the initial place in order to insure process continuity. And in this case only the real unreachable marking can be verified. See Figure 1.23 In our example, we use the same transformation to validate and verify the new workflow system. The new Petri nets model is created by transforming the proposed BPMN model in Figure 1.20 to the corresponding Petri nets. For simplicity reasons, the corresponding Petri nets for the model in Figure 1.20 only represent the

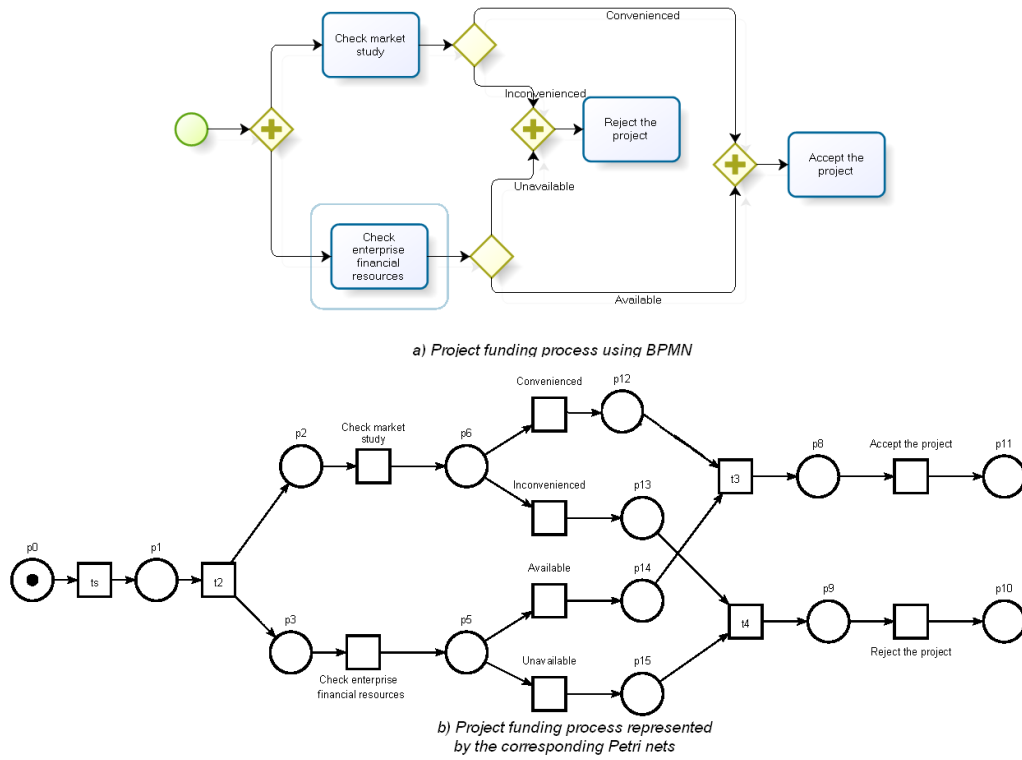


Figure 1.22: Funding project process and the corresponding Petri nets model

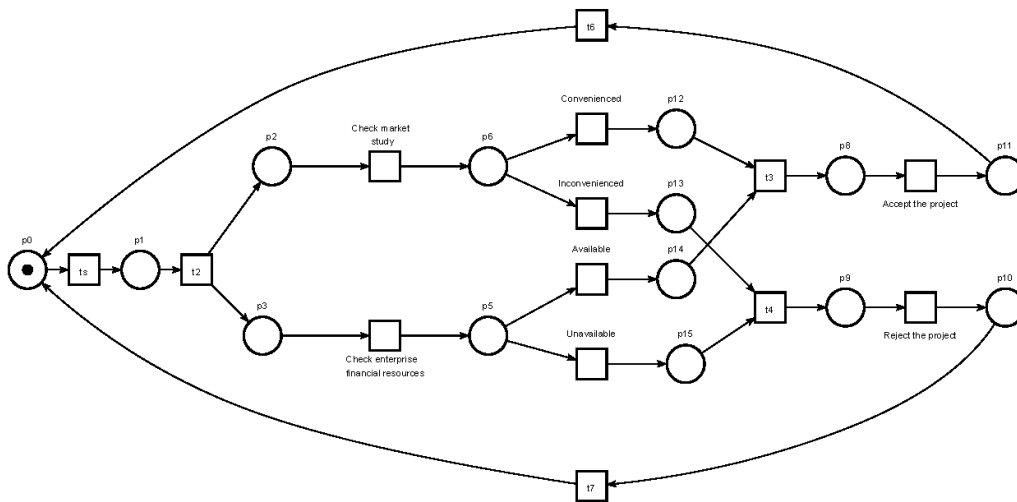


Figure 1.23: Modified Petri nets model to represent continuous funding project process

parts of BPMN model after task called *Classifying*. We have also avoided all external parts of the BPMN that COFIDIS does not control such as the send of incomplete document. In other words, we have supposed that each incomplete contract will be surly assembled and injected in the Petri net as a new contract. At the same time and to ensure resources availability, an operator can handle only one contract at the time, we have added an additional place p_{11} which will not effect process execution. To test process soundness we added the linking transition to link sour and sink places. Finally notice that this transformation represent only contract handling for one operator and there is no need to represent all enterprise operators, since this will be the same representation. See Figure 1.24 Figure 1.24.(a), simulates the pro-

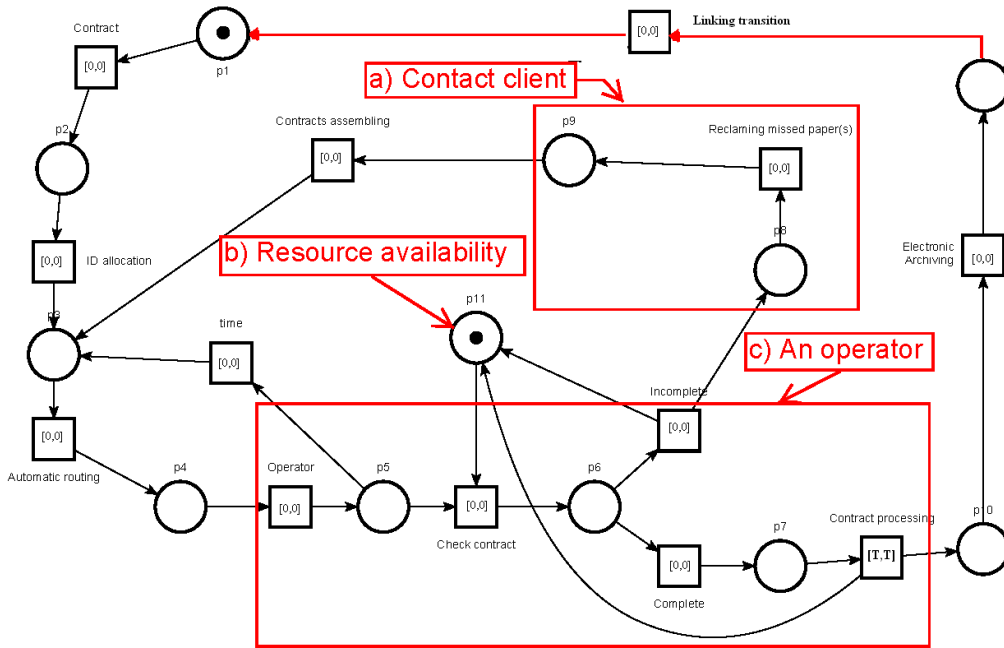


Figure 1.24: Petri nets model for the proposed workflow. a) Handling a contract by a collaborator. b) Client response.

cess of reclaiming missed paper(s), whereas Figure 1.24.(b), ensure resource

availability, Figure 1.24.(c) represents an operator.

Petri nets different properties are tested for the produced Petri nets model using TINA⁹[Bert 04]. Figure 1.25 shows the results obtained from performing behavioral and structural analysis. From the Figure, we see that the proposed model is live which ensure that the model is deadlock free. The model is also bounded and invariant which means the number of generated tokens (processed contracts) is fixe by operators for each given time.

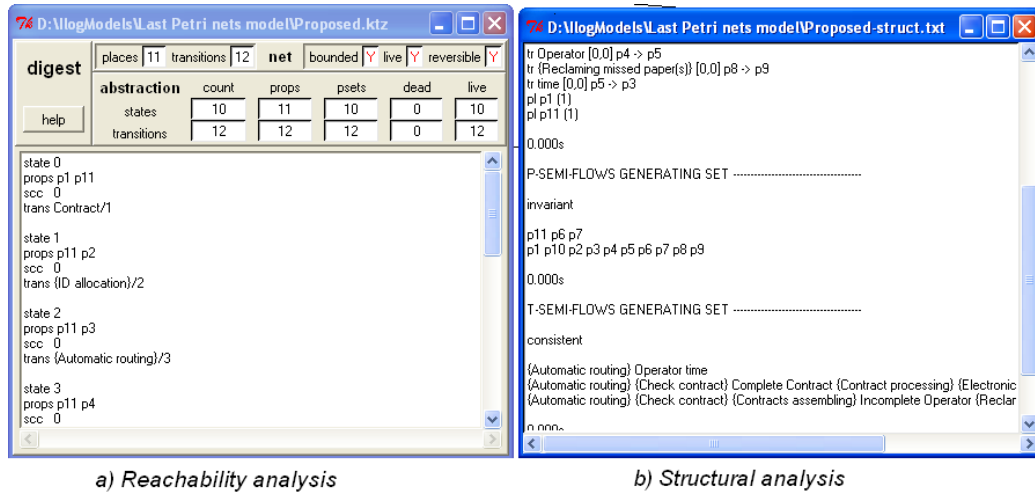


Figure 1.25: Behavioral and structural analysis

1.7 Workflow optimization

In the last section, we have shown how to use classical Petri net to simulate and verify some important properties of a workflow system represented by BPMN model. Although verifying workflow correctness is a good thing, it will be better if we can also ensure that it works efficiently. Classical Petri nets are limited when they are used to represent different type of data

⁹Toolbox for editing and analyzing Petri nets and time Petri nets

(tokens). Thus, its capacity to perform performance analysis is also limited. Representing additional information in a model may dramatically change its performance results.

For this reason, in this subsection we propose to transform the BPMN model to colored Petri nets[Jens 03]. This transformation will permit simulating and representing additional information and data in our system. As a result, perform more performance analysis on the business process to clarify its bottle-necks.

In our problem, the additional information that could be presented in the colored Petri nets includes contracts types; collaborator matrix competence (need time to handle each contract type); and complete/incomplete contracts. In colored Petri nets each color can be used to represent contract type, contract description (complete/incomplete). At the same time, each collaborator transition that used to model contract processing can define a different delay for each colored token. And this will represent collaborator competences regarding contract type. We have used the famous CPN tool[Jens 07] to represent and realize our performance analysis. To distribute contracts, we simulate the same process that is used currently by COFIDIS. This distribution is done randomly without any concern to collaborator competence. To test current contract flow at COFIDIS, we simulate the contracts workflow with a load of 500 contracts of 10 different quantities per type and for ten collaborators. The load of every collaborators was checked at different moment of time in the day, at the morning (time=0), at the end of work day (time=80) and finally the time for each collaborator to handle his entire daily load was also noted.

Simulation results show clearly that the current contract assignment policy is the main factor that obstacles the improvement of current contract

workflow at COFIDIS. Result shows that even though collaborators start their day with nearly equal load of contracts, at the end of the day the difference in collaborators loads are very large. This situation persists even with multiple executions of the simulator for the same sample of contracts. This is a normal result as the assignment process is performed randomly. The only thing that changes with different executions of the simulator is the persons with highest or lowest load. Even with new samples with new quantities of contract per type, we observe the same situation. Figure 1.26 shows a part of the colored Petri nets with different loads of three collaborators (Col 5, Col 6 and Col 7) after passing 80 unit of time since starting contracts handling process. As we can notice collaborator 5 and 6 have respectively 27 and 29 contracts in their buffers waiting to be handled. This represents nearly the half of the number of the contracts in the buffer for the collaborator 7 with 57 contracts.

The following table, Table 1.3, demonstrates the simulation results for the ten collaborators in the same sample. It demonstrates the number of contracts to handle per collaborator at two times, at time=0 (contract processing is started) and at time=80 (the work hours is terminated).

To calculate the needed unit of times to handle all received contracts using current assignment algorithm, we keep executing the simulator until there is no more contract in collaborators buffers. Table 1.4 shows the total unit of times needed to handle all contracts when using two different contracts assignment for the same sample of contracts. From the Table 1.4, we can notice that the current assignment process is far a way to be optimal or even constants. Not only, every contracts assignment needs different number of work unit time but also company collaborators must exceed theirs daily work hours (80 unites of time) to handle all contracts. Thus, bad contracts

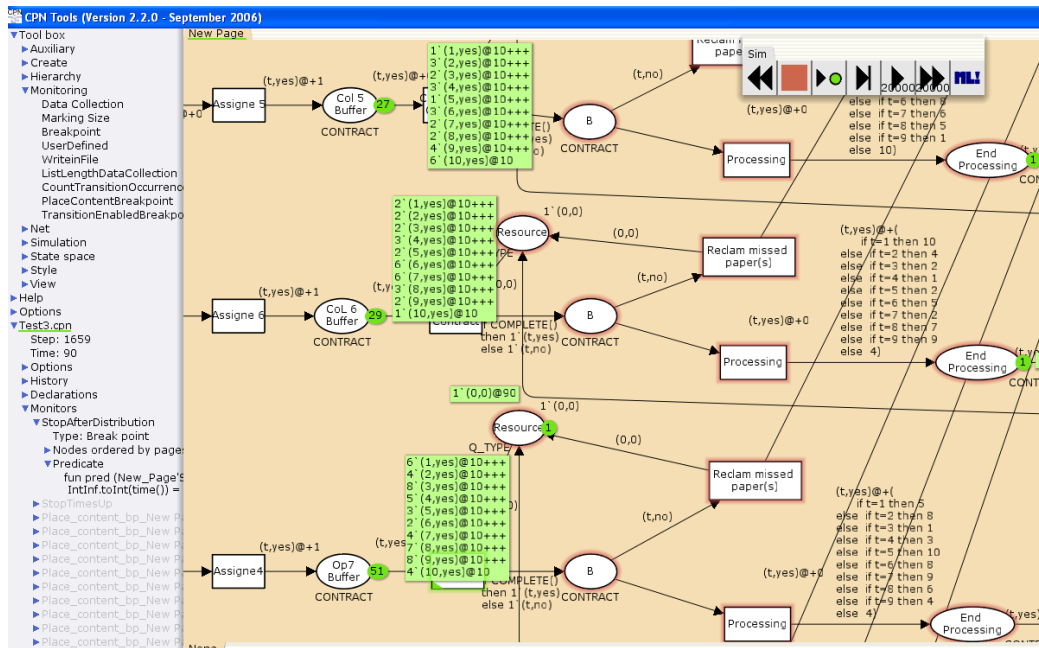


Figure 1.26: A part of the colored Petri nets that simulate contracts workflow at time stamp 80.

Col_ID	Number of contracts to be handled	
	time=0	time=80
Col_1	47	25
Col_2	56	34
Col_3	49	36
Col_4	39	19
Col_5	45	27
Col_6	61	51
Col_7	55	42
Col_8	49	38
Col_9	49	32
Col_10	54	34

Table 1.3: Results extracted from simulating contracts assignment process using colored Petri nets.

assignment could lead to duplicate the number of work unit time according to the number of work unit if the contracts were assigned optimally. And that means to decision makers either to postpone some contracts or hire temporary workers which forms additional unjustified costs for the company.

Col_ID	Unit times needed to handle all contracts in collaborators' buffers	
	Assignment 1	Assignment 2
Col_1	223	224
Col_2	223	247
Col_3	324	369
Col_4	183	244
Col_5	225	280
Col_6	199	197
Col_7	324	228
Col_8	351	388
Col_9	287	219
Col_10	232	335
Total	2571	2731

Table 1.4: Result from two different contracts assignments for the same contracts sample. Different contracts assignments needed different number of work time units.

In order to solve this problem and ensure contract assignment optimality, contracts assignment process must rely either on proofed exact algorithm or on a robust mathematical formulation. This formulation must be capable to cover all the aspects of the problem and give the solution in reasonable time. In this thesis, we have chosen the later choice since we can profit essentially from the Petri nets formulation of the problem and induce some important information.

The advantage of using Petri nets is its ability to be used in the analysis of the different flow in a workflow. More precisely; the analysis of Petri nets P-semiflow presented by Campos in [Camp 90][Sifa 80]. This analysis does not permit only the identification of workflow shortcoming and bottlenecks, but it also can help in inducing some useful mathematical formulation. Thus, we can these formulations with additional information from enterprise information system, to optimize contracts flow.

In[Camp 90], Campos present the formulation (1.1) to be used in analyzing Petri nets P-semiflow.

$$\left\{ \frac{Y_i^T \cdot W^- \cdot Z_t^{(f)}}{Y_i^T \cdot M_o} \right\}, \quad \forall \text{ } P - \text{semiflow representing collaborator } i. \quad (1.1)$$

Where

- Y_i^T : P-semiflow for collaborator i
- W^- : Weight of the arc to transition from its input place
- $Z_t^{(f)}$: Represents the weighted workload vector of transition with their corresponding routing ration (normalized with feedback transition TF)
- M_o : The initial marking

For this purpose, at first we have presented the different operators that are working in the company and the competence matrix that determine the needed time for each operator by contract type. This is can be done by duplicating the part c in the Figure 1.24 $N \times M$ times, where N is the number of operators and M is the number of different types of contracts. For each contract type related to a certain operator, the corresponding *Contract processing type* transition is associated to processing time in the operator competence matrix. Figure 1.27 demonstrates an example of Petri nets resulted

from unfolding colored Petri nets for two operators and for two contract type. For facility in this example we use two contract types with two colors *RED* and *BLUE*.

After these transformations, Campos's formulation can be applied and collaborators daily loads can be induced with the help of additional information from company registration as seen in (1.2).

$$\sum_{j=1}^J X_{ij} \times T_{ij} + \sum_{k=1,}^K C_{ik} \times (a_{ik} \times T_{new_{ik}} + \sum_{l=1, l \neq i}^I a_{lk} \times T_{old_{lk}}), \quad \forall i \in \{1, 2, \dots, I\} \quad (1.2)$$

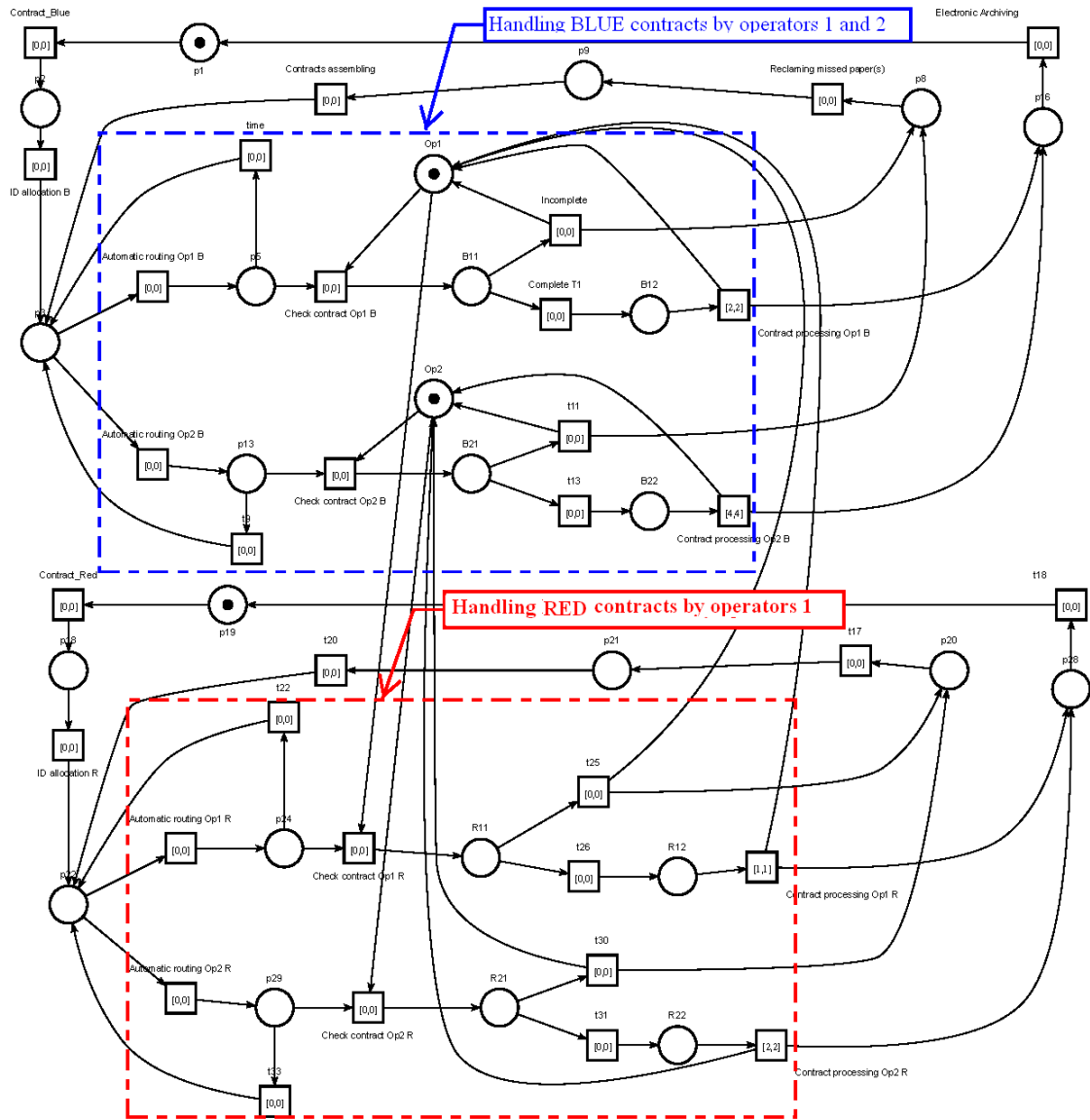
Where

- X_{ij} : The number of contracts of type j assigned to worker i ,
- T_{ij} : Needed time for primary, secondary workers i to handle contract type j for the first time,
- K : The number of assembled contracts from the day before and had not been handled yet,
- A_{IK} : two dimensional matrix representing the historic of assembled contracts k , where

$$- a_{ik} = \begin{cases} 1 & \text{if contract } k \text{ was treated by collaborator } i, \\ 0 & \text{otherwise.} \end{cases}$$

- C_{ik} : Binary variable, where

$$- C_{ik} = \begin{cases} 1 & \text{if assembled contract } k \text{ is assigned to} \\ & \text{collaborator } i, \\ 0 & \text{otherwise.} \end{cases}$$



- $Tnew_{ik}$: needed time to treat assembled contract k if it is assigned to the same collaborator i that has already treated it,
- $Told_{ik}$: needed time to treat the assembled contract k if it is assigned to a new collaborator i

Since the formulation in (1.1) demonstrates collaborator i possible load, this charge must not exceed collaborator daily capacity CAP_i , which is determined in the morning of each day. As a result formulation (1.2) becomes the left hand side of the inequality in (1.3). This inequality must be respected at each time contracts are assigned.

$$\sum_{j=1}^J X_{ij} \times T_{ij} + \sum_{k=1,}^K C_{ik} \times (a_{ik} \times Tnew_{ik} + \sum_{l=1, l \neq i}^I a_{lk} \times Told_{lk}) \leq Cap_i$$

$$\forall i \in \{1, 2, \dots, I\}$$
(1.3)

This constraint now can be used to optimize contracts assignment process and as a result minimize the number of called temporary workers needed to handle all received contract. The formulation of all problem aspect has generated a new assignment problem[Shra 08b]. This new assignment problem was not discussed before in literature. In the next chapter we represent the new problem with all its dimensions and represent it as linear programming problem. This linear programming problem can then be solved by using any mathematical solver such as Cplex¹⁰ solver which uses a branch and bound method to guaranty solution optimality. Related problems to the new

¹⁰Cplex : an optimization software package produced by ILOG, which uses an advanced mathematical programming and constraint-based optimization techniques to find problem optimal solution. <http://www.ilog.com/>

problem are also discussed. Similarities and differences are explained. And proposition to use the new problem in decision taking is presented as we have demonstrated in[Shra 08a].

1.8 Conclusion

In this chapter, we have presented a new framework for workflow reengineering. The proposed framework uses the most recent and promising language to model and execute workflow, the Business Process Execution Language (BPEL) and the Business Process Modeling Notation (BPMN). It uses BPMN to model the *as-is* and *to-be* workflow and it uses BPEL to implement the target workflow application as web service. The choice of BPMN is justified by its capacity and richness to represent different workflow patterns and different business process. BPMN is a comprehensive notation for both managers and system analysts and it eases the exchange of ideas between them to have a fine and deep operational analysis for the system. This permits to identify system weakness and draw backs from operational point of view. Thus, a new efficient and optimized workflow can be designed using BPMN. The target BPMN model is then can be used to generate BPEL code. Different approaches and techniques used to generate BPEL code from BPMN model have been briefly introduced. These approaches have passed considerable steps toward complete and automatic code generation. They have been grouped and implemented in one open source tool called BPMN2BPEL. The tool takes as an input a BPMN model that conforms to a particular XML format and outputs BPEL process. Thus, it is sufficient only to model the wished workflow using BPMN to obtain the final workflow implementation.

However, BPMN flexibility may leads to undesired properties, such as deadlocks and livelock. These properties will be directly reflected in the final code implementing the target workflow. More, BPMN is process oriented notation, and it is limited when it come to represent data. Thus, to overcome these problems, two additional phases are applied to BPMN representing *to-be* model before being adopted to generate BPEL code. A verification and

validation phase and an optimization phase. These phases are realized by the transformation of BPMN model to classical and colored Petri nets. Classical Petri net are used to validate and verify model behavioral properties, whereas colored Petri nets are used to simulate and realize a performance analysis for the new model. When ever a problem is detected in the Petri net model, the BPMN model is remodeled to avoid this problem. The final BPMN model is then used to generate BPEL execution code. Petri net is a formal language and has taken a lot of attention and consideration where many tools are consecrated to verify their undesired properties. Additional advantage of using Petri nets is the ability to perform performance analysis in which mathematical formulation can be induced and so mathematically formulate the problem. This formulation can then be solved by any mathematical solvers to have exact optimal solution. Real case study has been used to demonstrate the different phases of the proposed frameworks. And new assignment problem is appeared within the new workflow.

Chapter 2

Business process optimization and help in decision taking

2.1 Introduction

In chapter1, we have presented a new framework for workflow reengineering process. The new framework has been applied to real case study which is the mail workflow at COFIDIS. In this chapter, we present our second participation in the project GOCD that concerns contracts assignment optimization at COFIDIS. This problem raised when applying the optimization phase of the new framework for COFIDIS. In the next section we briefly introduce related assignment problem in literature. Similarities and differences of these problems to the assignment problem at COFIDIS are clarified. In section2.3, the new assignment problem is presented. The mathematical representation for this problem is demonstrated in details. In section 2.4, we present our approach to exploit and integrate the new assignment problem in the proposed decision making tool. This followed by the simulation and test result. We terminate by our conclusion and perspectives.

2.2 Related Problems

In literature, we find many assignment problems with different formulation and constraints. In this section we are interested in only two assignment problems that relate contract assignment problem at COFIDIS. Resolution methods for these problems are discussed and similarities and differences to our problem are underlined.

2.2.1 Bin packing problem BPP

Bin packing problem (BPP) is one of the most famous and widely studied assignment problems. It is well known for being one of the combinatorial NP-hard problems [M Ga 79]. Its simplest form described as a set of bins $Y = \{y_1, y_2, \dots, y_n\}$ of equal capacity c , and a list of objects $I = \{1, \dots, m\}$. All objects have the same weight w regarding all bins. The objective is to find the minimum number of bins to pack all the objects without exceeding bins capacities. A binary mathematical representation for the problem is:

$$\text{Minimize} \quad z = \sum_{i=1}^n y_i \quad (2.1)$$

$$\text{Subject to} \quad \sum_{j=1}^m wx_{ij} \leq cy_i, \quad \forall i \in \{1, 2, \dots, n\} \quad (2.2)$$

$$\sum_{j=1}^m x_{ij} = 1, \quad \forall i \in \{1, 2, \dots, n\} \quad (2.3)$$

$$y_i, x_{ij} \in \{0, 1\} \quad (2.4)$$

Where

$$y_i = \begin{cases} 1 & \text{if bin } i \text{ is used;} \\ 0 & \text{otherwise.} \end{cases}$$

and

$$x_{ij} = \begin{cases} 1 & \text{if object } j \text{ is assigned to bin } i; \\ 0 & \text{otherwise.} \end{cases}$$

Many researches were realized to find approximated solution for this problem [Brow 79, Csir 98, L Ep 07], and more few researches were destined to find exact solution [Mart 90, chol 97]. The most famous approximated algorithm for the BPP are:

- *Next-fit*: The algorithm starts by filling the first bin. When the bin is full, the algorithm opens new bin and starts filling it. The process continues until there are no more elements in our list. There is no concern in this algorithm to verify if the next element in the list may fit to any previous bins.
- *First-fit*: The algorithm puts the current element in the list in the first open bin that can hold this element. If there is no capable open bin, a new bin is opened to place the current element.
- *Best-fit*: The algorithm puts the current element in the list in the opened bin with the smallest free space that can hold this element. If there is no capable open bin, a new bin is opened to place the current element.
- *Worst-fit*: The algorithm puts the current element in the list in the opened bin with the largest free space that can hold this element. If there is no capable open bin, a new bin is opened to place the current element.

BPP variants include, two dimensional BPP [Chun 82, Berk 87, Lodi 99, Lodi 02, JPuc 07, Poly 07] and three dimensional BPP [Mart 00, Miya 07].

In two dimensional bin packing problem, objects have two dimensions (criteria) that must be respected when packing the object. A good example of two dimensions BPP is when packing objects with a width and a length for each. In three dimensions BPP, objects have three dimensions that must be respected when packing these objects. In most literatures, three dimensions BPP objects are represented by boxes with different volumes. In fact, BPP can have as many dimensions as we wish. These dimensions can be either physical or imaginal dimensions.

Another well studied variant of BPP is the extendable bin packing problem [Dell 98, Coff 06]. In this BPP type, bins have initial capacities that can be extended later. The extension is applied if the original bins capacities were incapable to pack all objects. This variant can be useful in presenting work hours and to determine the exact needed over timed hours.

The previous variants of BPP can be either on-line [GGal 95, L Ep 07, Csir 98] or off-line [Dawa 01, Dell 98] bin packing problem. In the on-line BPP, object information is known once that the object is packed and there is no concern to objects sequences. In other words, objects are packed according to their arrival. Whenever an object is packed, it can not be repacked later. Packing objects in production chain is a good example on on-line bin packing. In the contrary, the off-line bin packing problem, complete information about all objects is available before starting packing process. Objects can be packed and repacked as will as packing process did not terminate. This allows a better exploitation for bins capacities.

2.2.2 Generalized assignment problem GAP

Another famous assignment problem is the generalized assignment problem (GAP). The GAP is a generalization of the Multi-knapsack problem

[Mart 90]. In GAP problem, a set of objects $J=\{1, 2, \dots, m\}$ with cost (weight) w_j and profit p_{ij} must be assigned to a set of agents (bins) $Y=\{y_1, y_2, \dots, y_n\}$ with capacities $CAP=\{cap_1, cap_2, \dots, cap_n\}$. Each object can be allocated to any but only one agent. Objects treatment requests resources which change, depending on the object and the agent treating it. Each agent can have different capacity. The objective is to maximize the profit without exceeding agent's capacities. A mathematical representation of the problem is the following:

$$\text{Maximize} \quad z = \sum_{i=1}^n \sum_{j=1}^m p_{ij} x_{ij} \quad (2.5)$$

$$\text{Subject to} \quad \sum_{i=1}^m w_{ij} x_{ij} \leq cap_i, \quad \forall i \in \{1, 2, \dots, n\} \quad (2.6)$$

$$\sum_{j=1}^m x_{ij} \leq 1, \quad \forall i \in \{1, 2, \dots, n\} \quad (2.7)$$

$$x_{ij} \in \{0, 1\} \quad (2.8)$$

Where

$$x_{ij} = \begin{cases} 1 & \text{if object } j \text{ is assigned to agent } i; \\ 0 & \text{otherwise.} \end{cases}$$

Recent works on approximated algorithms to solve GAP problem can be found in [Nuto 06, Cohe 06] and for exact method [Save 97, Ross 75]. Readers are referred to [Catt 92] for a survey on algorithms used to solve GAP problem.

2.2.3 The suitability of BPP and GAP to represent COFIDIS contracts assignment problem

In studying the two presented problems, we notice the following. The two problems have different objectives and different formulation. BPP searches to minimize the number of used bins to pack all objects in the object list without any consideration to profit, whereas GAP problem considers profit for each packed object and the allocation of all objects is not considered as a constraint. This means it is possible to have an optimal solution in GAP problem without distributing all objects. More, in BPP the objects have an equal value whatever was the bin used to pack them, which is not the case in the GAP problem, the profit of an object depends on the object and on the agent treating that object.

Comparing the previous BPP and GAP problem description, with COFIDIS contract assignment problem, we see that COFIDIS assignment problem corresponds to both the GAP and BPP problem. Each collaborator at COFIDIS represents a bin with a capacity (daily work ours), and each contract represents an object. Processing time for each contract defined in the competence matrix is the weight for the contract regarding the collaborator that will handle it.

However, COFIDIS assignment problem corresponds to GAP problem in searching optimal treatment time for company workers. Still, GAP problem searches optimal solution without any concern to the number of used bins. This means it is possible to have optimal solution that assigns contracts to temporary workers despite the existence of available free time to company collaborators. However, COFIDIS wants to assign contracts to their collaborators and to fill their capacities before calling any additional temporary worker.

Similarly, using BPP to distribute COFIDIS contracts does not answer COFIDIS requirements. BPP does not consider objects profits. More, when distributing COFIDIS contracts using BPP, it is probable to have optimal solution that assigns contracts to secondary bins and excludes some primary bins. This is because BPP does not distinguish between bins representing company collaborator "*Primary bin*" from those bins representing temporary workers "*Secondary bin*".

As we can see neither BPP nor GAP problem answers COFIDIS assignment problem needs. For that, in the next section we present and detail new assignment problem that can answer COFIDIS assignment problem.

2.3 The Generalized assignment problem with identified first-used bins (GAPIFB)

In order to overcome BPP and GAP weakness to present COFIDIS contract assignment problem, we present in this section our new assignment problem. The new problem will be called the Generalized Assignment Problem with Identified First-used Bins (GAPIFB). GAPIFB regroups characteristics from well known assignment problems, the Bin Packing Problem and the Generalized Assignment Problem. The new problem can be defined as a set of tasks (objects) that must be assigned to a set of bins (agents). Each task size is not fixed and depends on the bin that will be used to pack the task, according to a predefined competence matrix. Bins can be of two types, primary bin (to present company collaborators) and secondary bins (to present temporary workers). The use of secondary bin is allowed only when the use of all primary bins are incapable to pack all tasks. The objective function is to minimize the number of used secondary bins.

GAPIFB mathematical formulation

In order to formulate the new GAPIFB problem, we use a set of binary and integer variables instead of using completely binary variable to present the problem. We use binary variable U_i only to represent both primary workers and secondary workers. Where U_i takes the value 1 if the worker i is used, otherwise it takes 0. And since our objects consist of contracts of several types, we use an integer variables X_{ij} that represent the number of contracts of the same type j assigned to worker i . Notice that J represents contracts types number. In this way, we reduce total variables number that is normally used when formulating BPP. In fact, we pass from linear problem with binary

variables to linear problem with integer variables.

For example, an instance of the problem with N primary workers, M available secondary workers, J contract types and NC contracts, needs $(N + M) + (N + M) * NC$ binary variables when using binary representation ($N + M$ variable to present agents and $(N + M) * NC$ to present contracts). Whereas grouping contracts by type requires $(N + M) + (N + M) * J$ variables ($N + M$ variables to present agents and $(N + M) * J$ to present the assigned contracts). The following is a generic formulation for the new assignment problem. Any additional specifications for this model to adapt the real case at COFIDIS are not covered here for simplicity reasons. Thus the mathematical formulation to represent GAPIFB is the following:

$$Min \sum_{i=1}^I U_i \quad (2.9)$$

Subject to

$$\sum_{j=1}^J X_{ij} \times T_{ij} \leq CAP_i \times U_i, \quad \forall i \in \{1, 2, \dots, I\} \quad (2.10)$$

$$\sum_{i=1}^I X_{ij} = QT_j, \quad \forall j \in \{1, 2, \dots, J\} \quad (2.11)$$

$$\sum_{i=1}^N U_i = N, \quad (2.12)$$

$$U_i \in \{0, 1\} \quad \forall i \in I, \forall \quad (2.13)$$

$$X_{ij} \in \mathbb{Z}^+, \quad \forall i \in \{1, 2, \dots, I\}, \forall j \in \{1, 2, \dots, J\} \quad (2.14)$$

Where

- N : Is the number of primary bins (company Collaborators),

- M : Is the number of available secondary bins (temporary workers),
- I : Is the number of primary bins and available secondary bins, $I = N + M$,
- U_i : Boolean variable to present primary, secondary bin i , where
 - : $\forall U_i \in \{U_1, U_2, \dots, U_N\}$, U_i represents primary bins,
 - : $\forall U_i \in \{U_{N+1}, U_{N+2}, \dots, U_I\}$, U_i represents secondary bins,
- CAP_i : Is primary, secondary bins capacities,
- J : Is the number objects types (number of contracts type),
- T_{ij} : Needed time (cost) for primary, secondary workers i to handle task of type j ,
- X_{ij} : The number of task of type j assigned to primary or secondary bin i .

Constraint (2.16) ensures that the capacity of agents is not violated. Constraint (2.17) ensures that all tasks are allocated and each task is assigned to only one agent. To ensure the use of all primary agents (which was a the main problem in classical BPP), we add constraint (2.18). This constraint forces the solver to search solution where $U_i=1, \forall i \in \{1, 2, \dots, N\}$, this set contains only primary agents and N is the number of company primary agents.

2.4 GAPIFB and decision making tool

As we have said before, one of the major GOCD projects aims is to install a new decision making tool at COFIDIS. The purpose of this tool is to optimize contracts handling process for both short-term (daily received contracts) and for the mid-term (received contracts for a predefined period). In the short-term, decision makers are concerned by the following issues.

- Is company resources are capable to handle all received contract?
 - If yes, what contracts assignment they must follow?
 - If not, what is the minimum number of temporary workers are needed to handle all received contracts;
- Can they choose another objective function to distribute contracts?
- How can they optimize company resources in real time, to cope with eventual events, such as,
 - The sudden departure of some employers,
 - The increase of collaborators free capacity caused by handling incomplete contracts or assembled contract as we have seen in 1.5.

The last point is very essential, since collaborators free capacities change during the day as we have explained in the operational analysis in 1.5. This change is result from handling incomplete contracts which take less time than completed ones and from handling complete contracts that have more or less time than it was expected in the competence matrix. Thus, even though contracts assignment was optimal in the morning, it will not be so as time passes. And it is probable that collaborators will have free capacity to handle

In the mid-term, decision makers would like to know if they can postpone some contracts in over loaded days to those days with weak load (under-loaded days) to avoid hiring unnecessary workers in the current day.

In the following we present an approach to handle contract handling optimization for both short-term and mid-term duration. The approach depends mainly on an interactive decision making tool. This tool will allow decision makers to simulate contracts assignment process and see the impact of their decisions in real time. In both approaches, *minimizing the number of called temporary workers is the major objective*. This is very important, since humane resources are the main financial consumer for any services company. Each approach will be presented as an integer linear problem, where there exist several advanced algorithms for solving it such as, a branch and bound, branch-and-cut and branch-and-price.

2.4.1 Short-term approach

For the short-term approach, we propose a two stages decision making approach. In the first stage, the GAPIFP formulation is used to know if company current resources are capable to handle daily received contracts, and if not, the formulation will determine in exact the minimum number of temporary workers needed to handle all received contracts. In the second stage, contract assignment can be optimized according different criteria, where a set of objective functions to assign contracts is proposed to the decision maker. These new objectives are limited by the number of temporary workers that the manager has decided to hire in the first stage. Thus decision makers have the possibility to see and evaluate the direct effect of their choices concerning contracts assignment. Figure (2.2) demonstrates decision making process for short-term approach.

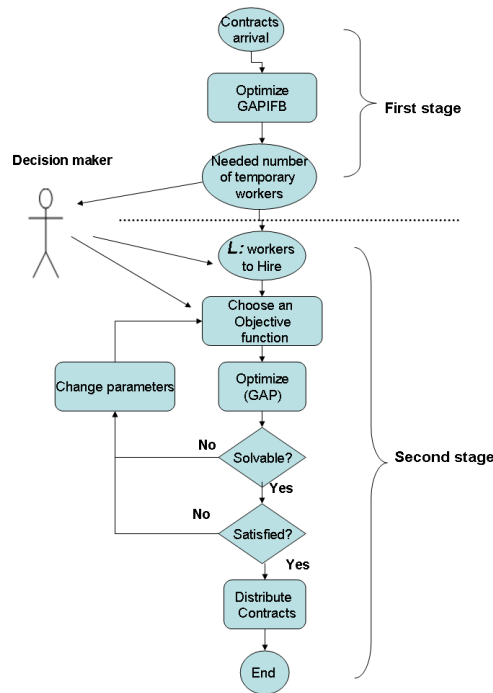


Figure 2.2: Interactive decision-making process

As the Figure in (2.2) demonstrates, at the end of the first stage, company decision maker has a complete idea about company situation whether it is overloaded or under loaded. If it is overloaded, the exact number " EN " of secondary workers needed to treat all contracts will be given to decision maker. Decision maker then decide whether to hire secondary workers or not. If the decision was to hire temporary workers, he must decide if either to hire the totality needed secondary workers or to hire certain number defined as " L "? In the second stage, the final choice of the used objective function is left to decision makers. One objective can be to give high treatment priority to contracts that can not be delayed or to contracts considered as profitable to the company. Another objective could be to treat important contracts types uniquely by company collaborators as they have the best experience and skills. The fairness of collaborators loads can be significance objective

from the social vision of point. To maximize the rate of profitability by collaborators can be an interesting objective from economic vision. Notice that the second objective is always limited by the number of temporary workers that the decision makers decide to take. The following represents the notations used to formulate the first stage:

- N : The number of primary workers (company collaborators),
- M : The number of available secondary workers (temporary workers),
- I : The number of primary workers and secondary workers, $I = N + M$,
- $U_i(t)$: Boolean variable to present primary, secondary worker i at time t , where
 - : $\forall U_i(t) \in \{U_1(t), U_2(t), \dots, U_N(t)\}$, $U_i(t)$ represents primary workers,
 - : $\forall U_i(t) \in \{U_{N+1}(t), U_{N+2}(t), \dots, U_I(t)\}$, $U_i(t)$ represents secondary workers,
- $QT_j(t)$: Quantity of contract type j at time t ,
 $QT_j(t) \in \{QT_1(t), QT_2(t), \dots, QT_J(t)\}$,
- T_{ij} : Needed time for primary, secondary workers i to handle a contract of type j ,
- K : Number of assembled contracts during the day before and had not been handled yet,
- A_{IK} : Two dimensional matrix representing the historic of assembled contracts, where

$$- a_{ik} = \begin{cases} 1 & \text{if contract } k \text{ was handled by worker } i, \\ 0 & \text{otherwise.} \end{cases}$$

- C_{ik} : binary variable, where

$$- C_{ik} = \begin{cases} 1 & \text{if assembled contract } k \text{ is assigned to worker } i, \\ 0 & \text{otherwise.} \end{cases}$$

- $Tnew_{ik}$: Needed time to handle assembled contract k if it is assigned to the same worker i that has already handled it,
- $Told_{ik}$: Needed time to handle the assembled contract k if it is assigned to a new worker i ,
- $Prs(t)$: Set contains company workers present at time t ,
- $Abs(t)$: Set contains company workers absent at time t .

The adapted GAPIFB mathematical formulation to minimize the number of temporary workers is the following:

$$Min \sum_{i=1}^I U_i(t) \tag{2.15}$$

Subject to

$$\begin{aligned} & \sum_{k=1,}^K C_{ik} \times (a_{ik} \times Tnew_{ik} + \sum_{l=1, l \neq i}^I a_{lk} \times Told_{lk}) + \\ & \sum_{j=1}^J X_{ij} \times T_{ij} \leq Cap_i(t) \times U_i(t), \quad \forall i \in \{1, 2, \dots, I\} \end{aligned} \tag{2.16}$$

$$\sum_{i=1}^I X_{ij} = QT_j(t), \quad \forall j \in \{1, 2, \dots, J\} \tag{2.17}$$

$$\sum_{k=1}^K C_{ik} = K, \forall i \in \{1, 2, \dots, I\} \quad (2.18)$$

$$\sum_{i=1}^N U_i(t) = |Prs(t)|, \quad (2.19)$$

$$U_i(t) = 0, \quad \forall U_i(t) \in Abs(t), \quad (2.20)$$

$$U_i \in \{0, 1\} \quad \forall i \in I, \quad (2.21)$$

$$X_{ij} \in \mathbb{Z}^+, \quad \forall i \in \{1, 2, \dots, I\}, \forall j \in \{1, 2, \dots, J\} \quad (2.22)$$

The first stage is computed at the morning of each day and before handling any contract (at time equal 0). Constraint (2.16) ensures that the capacity of agents is not violated. Constraints (2.17) and (2.18) ensures that all contracts are allocated and each contracts is assigned to only one agent. To ensure the use of all primary workers (which was a problem in classical BPP), we add constraints (2.19) and (2.20). These constraint forces the solver to search solution where $U_i(t)=1, \forall i \in Prs(t)$. $|Prs(t)|$ is the cardinality of the set $Prs(t)$ (the number of element in the set). This set contains only primary worker that are available at time t .

The second stage is formulated as GAP problem and the decision of manager from the first stage is used to determine the number of hired temporary workers. In this stage, L is the number of secondary worker the manager decide to hire.

Although, the two stage are executed at the morning of each day, the second stage can also be executed at any time t of the day, for example when

workers accumulated free capacity is attained the predefined threshold as we demonstrated in Figure 2.1.

As we said before many objectives can be used in the second stage, but for facility we chose here one objective which is to minimize the total treatment time for all agents.

$$\text{Min} \sum_{i=1}^{N+L} \sum_{j=1}^J X_{ij} \times T_{ij} + \sum_{i=1}^{N+L} \sum_{k=1}^K C_{ik} \times (a_{ik} \times T_{new_{ik}} + \sum_{l=1, l \neq i}^{N+L} a_{lk} \times T_{old_{lk}}) \quad (2.23)$$

$$\sum_{k=1}^K C_{ik} \times (a_{ik} \times T_{new_{ik}} + \sum_{l=1, l \neq i}^I a_{lk} \times T_{old_{lk}}) + \sum_{j=1}^J X_{ij} \times T_{ij} \leq \text{Cap}_i(t) \times U_i(t), \quad \forall i \in \{1, 2, \dots, I\} \quad (2.24)$$

$$\sum_{i=1}^{N+L} X_{ij} = Q T_j(t), \quad \forall j \in \{1, 2, \dots, J\} \quad (2.25)$$

$$\sum_{k=1}^K C_{ik} = K, \quad \forall i \in \{1, 2, \dots, I\} \quad (2.26)$$

$$U_i(t) = 0, \quad \forall U_{it} \in \text{Abs}_t \quad (2.27)$$

$$\sum_{i=1}^N U_i(t) = |\text{Prs}_t| \quad (2.28)$$

Constraint (2.24) ensures that workers capacities will not be violated. Constraints (2.25) and (2.26) ensure that all received contracts will be distributed. Note that the equality in constraints (2.25) and (2.26) are correct only when the number of exact temporary workers needed to handle all received contract equal the number of really hired temporary workers $EN=L$.

When $EN > L$, then constraint (2.25) become as shown in (2.29). Where α is a real number $\in [0,1]$ that represents the percentage of contract of type j that must be handled. This percentage can be changed every time the problem is not solvable.

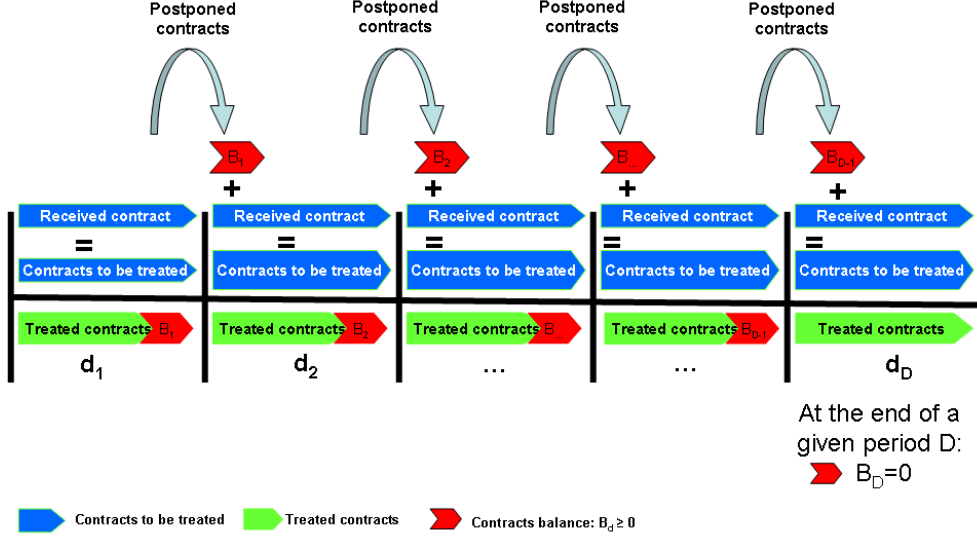
$$\sum_{i=1}^{N+L} X_{ij} = QT_j(t) \times \alpha_j, \quad \forall j \in \{1, 2, \dots, J\} \quad (2.29)$$

2.4.2 Mid-term approach

In the short-term approach, we consider received contracts for only one day only. And we suppose that the daily received contracts must be treated at their arrival day.

In reality, decision maker can decide to postpone some contracts and not to call any temporary workers to treat the overloaded contracts. This decision could be taken, for example, when the financial resources do not permit to employ temporary workers, or when these contracts are not of great importance. On the other hand, COFIDIS is not always overloaded, and received contracts can be less than COFIDIS capacity, which result in loss of work-hours in under loaded days. Add to that, capable workers can be absent when the company receives their favor contracts.

In this section we improve the previous approach by considering contracts flow over a given period D instead of only one day. We suppose that each contract type can be delayed for some days before being treated. This delay allows decision makers to postpone some contracts that are received in overloaded days to those days which are under loaded or even to those days when capable workers are present. We argue that, this will reduce the number of called temporary workers in overloaded days, and at the same time, will ensure entire exploitation of company resources in under loaded days. Figure 2.3 demonstrates this process.


 Figure 2.3: Postponed contracts flow over D days

To achieve this, quantities and types of received contracts over a given period D must be known in advance. The prediction process can be done using existing mathematical model that use regression analysis such as *Poisson regression analysis*. In Poisson regression, historical count data is used to predict expected contracts quantities and types for a given year period. This data can be easily obtained from COFIDIS historical records for the last years. For more information reader is referred to [Came 98].

In our model, we assume a fixed number for both company workers and available temporary workers during a given period D . To meet work condition reality, where some company workers can be absent in some days, we define two sets, *Prs* set and *Abs* set. The first set contains company workers present at day $d=1$ to $d=D$, whereas the second (*Abs* set), contains company workers absent in the same period.

The formulation and the used notation for the improved model is defined

in the following

- D : Given period in days $D=\{d_1, d_2, \dots, d_D\}$,
- Abs : Primary workers absent set, $U_{di} \in Abs$, if primary worker i is absent in day d
- Prs : Primary workers present set, $U_{di} \in Prs$, if primary worker i is present in day d
- N : Number of primary workers (company workers),
- M : Number of available secondary workers (temporary workers),
- I : Number of primary worker and available secondary workers, $I = N + M$,
- U_{di} : Boolean variable. It presents if primary/secondary worker i , is used in day d ,
- QT_{dj} : Expected received quantity of contract type j in day d ,
- CAP_{di} : Primary/secondary workers i capacities in day d ,
- X_{dij} : Number of contracts of type j assigned to primary/secondary worker i in day d , integer variable.

$$Min \sum_{d=1}^D \sum_{i=1}^I U_{di} \quad (2.30)$$

Subject to

$$\sum_{d=1}^D \sum_{i=1}^N U_{di} = |Prs| \quad (2.31)$$

$$U_{di} = 0, \quad \forall U_{di} \in Abs \quad (2.32)$$

$$\sum_{j=1}^J X_{dij} \times T_{ij} \leq CAP_{di} \times U_{di}, \quad \forall i \in \{1, 2, \dots, I\}, \forall d \in \{1, 2, \dots, D\} \quad (2.33)$$

$$\sum_{d=1}^D \sum_{i=1}^I X_{dij} = \sum_{d=1}^D QT_{dj}, \quad \forall j \in \{1, 2, \dots, J\} \quad (2.34)$$

$$\sum_{\alpha=1}^d \sum_{i=1}^I X_{\alpha ij} \leq \sum_{\alpha=1}^d QT_{\alpha j}, \quad \forall d \in \{1, 2, \dots, D\}, \forall j \in \{1, 2, \dots, J\} \quad (2.35)$$

$$U_{di} \in \{0, 1\} \quad \forall i \in \{1, 2, \dots, I\}, \forall d \in \{1, 2, \dots, D\} \quad (2.36)$$

$$X_{dij} \in \mathbb{Z}^+, \quad \forall d \in \{1, 2, \dots, D\}, \forall i \in \{1, 2, \dots, I\}, \forall j \in \{1, 2, \dots, J\} \quad (2.37)$$

Objective function (2.30), minimizes the total number of used workers whither they are primary workers or temporary workers, which is not what we are looking for. To overcome this problem, we add constraint (2.31) to search only solutions containing all primary workers present in the period $[1, D]$. $|Prs|$ in (2.31) represents the cardinality (number of elements) in the Prs set. Constraint (2.32) guarantees that no primary worker i in day d is chosen, if he is absent in that day (exist in the Abs set). Constraint (2.33), ensures that the capacity of worker i in day d will not be violated. To be sure that all received contracts in the period $[1, D]$ will be assigned, we add constraint (2.34).

To show that the contracts can be postponed before the date D (D excluded), constraint (2.35) is used, which is induced from inequality (2.38).

$$X_{d\beta j} \leq \sum_{\alpha=1}^d QT_{\alpha j} - \left(\sum_{\alpha=1}^{d-1} \sum_{i=1}^I X_{\alpha ij} + \sum_{\substack{i=1 \\ i \neq \beta}}^I X_{dij} \right) \quad (2.38)$$

2.4.3 Simulation and test results

In order to test and evaluate our two approaches, we use CPLEX¹ solver which uses branch and bound method to compute optimal solutions for both approaches, short-term and mid-term². Considered results in the short-terms are only for the first stages i.e. the results for the GAPIFB formulation. There is no interest to present second stage results, as they are qualitative issue left to decision makers.

In the short-term approach, we use real samples registered in COFIDIS historical records. The results of our approach are then compared to the results obtained by COFIDIS current used method. Our approach is capable to detect overloaded and under loaded situation. In overloaded situation, it gives the exact number of needed temporary workers to treat all received contracts, which is always less than or equal the number realized by COFIDIS current used method. No company worker is excluded from any of the proposed solutions. This is a major problem when solving the problem by a classical BPP formulation. Using BPP formulation could exclude some company workers from the solution, if the use of temporary workers gives better results. The execution time within the size of our problem, is ordered by milliseconds, and this is very satisfactory to COFIDIS decision makers.

For confidentiality reasons, the used samples are generated randomly. Each sample simulates the received contracts over a period of twelve days. The number of company worker and available temporary workers is assumed to be fixed over the twelve days and to be 100 and 10 consecutively. For each

¹CPLEX : an optimization software package produced by ILOG, which uses an advanced mathematical programming and constraint-based optimization techniques to find problem optimal solution. <http://www.ilog.com/>

²Tests were held on Intel Dual Core T7200 2.00GHz machine, 2Go of RAM

Days	Execution time (s)	Received contracts	Company workers	Available temporary workers	Needed temporary workers
Day 1	0.031	5000	100	10	10
Day 2	0.016	4700	100	10	6
Day 3	0.016	4200	100	10	0
Day 4	0.016	4600	100	10	7
Day 5	0.031	4400	100	10	1
Day 6	0.047	4900	100	10	9
Day 7	0.015	4800	100	10	9
Day 8	0.016	4600	100	10	3
Day 9	0.032	4300	100	10	1
Day10	0.015	4400	100	10	0
Day11	0.015	4400	100	10	1
Day12	0.016	4300	100	10	0

Table 2.1: Simulation results of the short-term approach for the first stage

day, contracts quantities and their types are generated randomly, and varied between 4200 and 5200 contracts per day. Company workers competence matrix is also generated randomly. We suppose that all temporary workers have the same competence by contract type. Simulation results of the short-term approach and for the first stage with a period of 12 days can be seen in Table 2.1.

Zeros in the column, *Needed temporary workers*, implies that these days are under loaded and there is no need to call any temporary worker.

With the current assignment method at COFIDIS, it is probable to call unnecessary temporary worker in these days as the optimality is not guaranteed. This can be seen clearly by examining the generated sample for the days 4 and 8. In these days, we receive the same quantity of contracts (4,600 contracts), but in day 4 we need four additional temporary workers than what we need in day 8. This is justified by the differences in quantities for each contract type from one day to another.

To verify our mid-term approach credibility, we use the same samples that we have used in Table 2.1. We define four periods of time that the received contracts can be postponed to, when the company is overloaded. These periods are defined by three, six, nine and twelve days. Optimal solution for each period can be seen in Table 2.2 in the first line. The corresponding cumulative sum of the obtained solutions for the same period using short-term approach can be seen in the second line of the same table.

From Table 2.2, we see that 25% of used temporary workers in short-term approach could be saved by using our proposed mid-term approach, which is significant to company decision makers. Profit average over ten random generated samples and for four period of times are demonstrated in Table 2.3.

Defined period in days	[1-3]	[1-6]	[1-9]	[1-12]
Needed temporary workers for a period using mid-term approach	12	28	37	35
Corresponding cumulative sum of obtained solutions for the same period using short-term approach	16	33	46	47
Difference	4	5	9	12
Profit	25.00%	15.15%	19.57%	25.53%

Table 2.2: First stage results for the short-term approach and the mid-term approach

Period	Profit average
[1-3]	14.82%
[1-6]	14.55%
[1-9]	17.43%
[1-12]	19.32%

Table 2.3: Average profit of using mid-term approach over 10 random samples

2.5 Conclusion

In this chapter, we have presented and defined a new real world assignment problem. The new problem regroups characteristics from both well known problems, Bin Packing Problem (BPP) and Generalized Assignment Problem (GAP). We have called our problem *Generalized Assignment Problem with Identified First-used Bins (GAPIFB)*. As BPP, the new GAPIFB problem consists of allocating a set of different tasks to a set of bins with different capacity for each bin. On the other hand, tasks' weights in the new problem change from one bin to another according to a predefined competence matrix, as it is defined in the GAP problem. The objective function in the new problem is to find the minimum number of needed secondary bins to pack all existing tasks, if the use of primary bins was not sufficient.

The originality in the new problem is the existence of two types for bins, primary bins and secondary bins. Primary bins must be used and filled at first, before any use of any other secondary bins.

This problem represents a real assignment problem obtained from COFIDIS Credit Company, where tasks represent contracts and bins represent company collaborators and temporary workers that treat these contracts.

The problem has been formulated as an integer linear problem (ILP) with help of the results obtained from the performance analysis in the optimization phase of the framework proposed in the last chapter. The optimal solution was then calculated using CPLEX solver, which uses an advanced constraint-based optimization techniques to find problem optimal solution. Problem execution time is ordered by milliseconds. Our model shows the capacity to detect overloaded and under loaded situations. Secondary bins are used only and only if tasks load exceeds primary bins capacities. In both, under loaded and overloaded situation, all primary bins appear in the optimal solution.

This forms a real problem if we try to solve our problem by classical BPP form. Classical BPP form does not distinguish bins types when it searches the optimal solution, which makes it possible to exclude some primary bins if the use of secondary bin gives best result. In addition, BPP defines constant weight per task type, whatever the used bin was, which is not the case in GAPIFB. Simulation results show that GAPIFB model ensures optimal tasks distribution.

We propose to integrate the new assignment problem in decision making tool. This tool helps decision makers in taking their decisions for both short-term "daily decision" and mid-term "decision over several days". In the short-term, the proposed approach allows a real time optimization for contracts assignment.

Real samples from COFIDIS have been used to test the proposed approaches for both terms, short-term and mid-term. Simulation results show that our approaches are more efficient than the current used method at COFIDIS. With short-term approach, COFIDIS decision maker knows in advance the exact number of temporary worker that they will need to treat all received contract. This forbids unnecessary call to temporary workers.

The proposed mid-term approach allows COFIDIS to save up 25% of temporary workers when considering contract distribution over a predefined period. The only problem in this approach is to predict correctly received contracts quantities and theirs types over this period. We believe this is not a problem, since COFIDIS has large historical data on the received contracts types and quantities over years. These data can be used with any known prediction method to estimate the expected received contract over a given year period.

Since Cplex execution time to solve the GAPIFB and within its current

size is in millisecond, there is no interest to develop our own exact solution method. However, a dedicated resolution method to solve GAPIFB can form a rich subject to be explored.

Chapter 3

Conclusion and perspectives

This thesis has been realized in the context of the GOCD project, to install a new dematerialized workflow system and helping decision tool. Through this project, new problems have been emerged leaving wide horizon for other researches.

In the first part of this work, we have presented and discussed a new framework to model, verify, optimize and implement workflow systems. The proposed framework uses the most recent and promising language to model and execute workflow, the Business Process Execution Language (BPEL) and the Business Process Modeling Notation (BPMN). It uses BPMN to model the *as-is* and *to-be* workflow and it uses BPEL to implement the target workflow application as web service. The choice of BPMN is due to its capacity and richness to represent different workflow patterns and different business process. BPMN is a comprehensive notation for both managers and system analysts and it eases the exchange of ideas between them to have a fine and deep operational analysis for the system. This permits to identify system weakness and draw backs from operational point of view. Thus, a new efficient and optimized workflow can be designed using BPMN. The

target BPMN model is then can be used to generate BPEL code. Different approaches and techniques used to generate BPEL code from BPMN model have been briefly presented. These approaches have passed considerable steps toward complete and automatic code generation.

However, BPMN flexibility may leads to undesired properties, such as deadlocks and livelock. These properties will be directly reflected in the final code implementing the target workflow. More, BPMN notation was designed to model business process, and little consideration was concentrated to represent data. To overcome these problems, two additional phases are applied to the BPMN model representing the *to-be* workflow model before being adopted to generate BPEL code. A verification and validation phase and an optimization phase. These phases are realized by the transformation of BPMN model to classical and colored Petri nets. Classical Petri net are used to validate and verify model behavioral properties, whereas colored Petri nets are used to simulate and realize a performance analysis for the new model. When ever a problem is detected in the Petri net model, the BPMN model is remodeled to avoid this problem. The final BPMN model is then used to generate BPEL execution code. Petri nets are a formal language and it has taken a lot of attention and consideration. And many tools were consecrated to verify their undesired properties. Additional advantage of using Petri nets is the ability to perform performance analysis in which mathematical formulations can be induced. These formulations can then be used to solve the problem with any mathematical solvers. Real case study has been used to demonstrate the different phases of the proposed frameworks. And new assignment problem within the new workflow has been emerged. The formulation and the employment of the new problem in a decision making tool is the second contribution of this thesis.

The new problem is a real world assignment problem where contracts represent a list of objects, and company collaborators and temporary worker represent a group of bins to pack these objects. The new problem has been called *Generalized Assignment Problem with Identified First-used Bins (GAPIFB)*. As BPP, the new GAPIFB problem consists of allocating a set of different tasks to a set of bins with different capacity for each bin. On the other hand, tasks' weights in the new problem change from one bin to another according to a predefined competence matrix, as it is defined in the GAP problem. The objective function in the new problem is to find the minimum number of needed secondary bins to pack all existing tasks, if the use of primary bins was not sufficient. The originality in the new problem is the existence of two types for bins, primary bins and secondary bins. Primary bins must be used and filled at first, before any use of any other secondary bins.

This new problem has been formulated mathematically and its entire dimensions have been covered. This formulation was realized with the help of the performance analysis realized in the optimization phase of the proposed framework.

It has been formulated as an integer linear problem (ILP). The optimal solution was then calculated using CPLEX solver, which uses an advanced constraint-based optimization techniques to find problem optimal solution. Problem execution time, within the size of the problem at COFIDIS, is ordered in milliseconds. The mathematical model for the new problem has proved that it was capable to detect overloaded and under loaded situations. Secondary bins are used only, and only if, tasks load exceeds primary bins capacities. In both, under loaded and overloaded situation, all primary bins appear in the optimal solution. This forms a real problem if we try to solve our problem by classical BPP formulation. Classical BPP formulation does

not distinguish bins types when it searches the optimal solution, which makes it possible to exclude some primary bins if the use of secondary bin gives best result. In addition, BPP defines constant weight per task type, whatever the used bin was, which does not represent the case in the GAPIFB. Simulation results show that GAPIFB model ensures optimal tasks distribution.

We propose to integrate the new assignment problem in decision making tool. This tool helps decision makers in taking their decisions for both short-term "daily decision" and mid-term "decision over several days". In the short-term, the proposed approach allows a real time optimization for contracts assignment where contract assignment process is controlled and monitored all long the day.

The proposed mid-term approach allows COFIDIS to get advantages when company capacity is under loaded. Thus, distributing the loads in overloaded day over these day where the company is under loaded.

Real samples from COFIDIS have been used to test the proposed approaches for both terms, short-term and mid-term. Simulation results show that our approaches are more efficient than the current used method at COFIDIS. With short-term approach, COFIDIS decision maker knows in advance the exact number of temporary worker that they will need to treat all received contract, which forbids unnecessary call to temporary workers.

Mid-term approach shows the ability to save up 25% of temporary workers when considering contract distribution over a predefined period. The only problem in this approach is to predict correctly received contracts quantities and their types over this period. We believe this is not a problem, since COFIDIS has large historical data on the received contracts types and quantities over years. These data can be used with any known prediction method to estimate the expected received contract over a given year period.

This work is only the first step in the reengineering process for company's workflow. Generalizing this approach is our next objective.

Future works on the transformation of BPMN to BPEL using Model Driven Architecture (MDA) seems to be a very promising research domain. This will permit the implementation of a workflow by different BPLE workflow engine.

Although the execution time to solve GAPIFB problem is ordered in milliseconds, we believe that increasing problem size or its structure will yield in exploding the execution time to find optimal solution. And since GAPIFB problem is unique in its characteristic, dedicated exact or approximated resolution methods to solve this problem represent rich domains for concerned researchers to be developed and explored.

Bibliography

- [Aals 00] W. M. P. van der Aalst and A. H. M. ter Hofstede. “Verification of Workflow Task Structures: A Petri-Net-Based Approach”. *Information Systems*, Vol. 25, No. 1, pp. 43 – 69, 2000.
- [Aals 05] W. van der Aalst and A. ter Hofstede. “YAWL: yet another workflow language”. *Information Systems*, Vol. 30, No. 4, pp. 245 – 275, 2005.
- [Aals 06] W. M. P. van der Aalst, M. Dumas, A. H. M. ter Hofstede, and N. Russell. *Business Process Management*, Chap. On the Suitability of BPMN for Business Process Modelling, pp. 161–176. Springer Berlin / Heidelberg, 2006.
- [Akeh 04] D. Akehurst. “Validating BPEL Specifications using OCL”. Tech. Rep., University of Kent, 2004.
- [Arki 05] A. Arkin, S. Askary, B. Bloch, F. Curbera, Y. Goland, N. Kartha, C. Liu, S. Thatte, P. Yendluri, and A. Yiu. *Web Services Business Process Execution Language Version 2.0*. WS-BPEL TC OASIS, 2005.
- [Bare 06] L. Baresi, A. Maurino, and S. Modafferi. “Towards Distributed BPEL Orchestrations”. In: *Third Workshop on Software Evo-*

- lution through Transformations: Embracing the Change (SE-TRa'06). Electronic Communications of the EASST*,, 2006.
- [Berk 87] J. O. Berkey and P. Y. Wang. “Two-Dimensional Finite Bin-Packing Algorithms”. *The Journal of the Operational Research Society*, Vol. 38, pp. 423–429, 1987.
- [Bert 04] B. Berthomieu, P. Ribet, and F. Vernadat. “The tool TINA – Construction of abstract state spaces for petri nets and time petri nets”. *International Journal of Production Research*, Vol. 42, pp. 2741–2756, 2004.
- [Bord 04] B. Bordbar and A. Staikopoulos. “On Behavioural Model Transformation in Web Services”. *Conceptual Modelling for Advanced Application Domain, Shanghai, China*, pp. 667–678, 2004.
- [Brow 79] D. J. Brown. “A lower bound for on-line one-dimensional bin packing algorithms”. Tech. Rep., Illinois Univ at Urbana-Champaign applied computation theory group, 1979.
- [Came 98] A. Cameron and P. Trivedi. *Regression analysis of count data*. Cambridge University Press, 1998.
- [Camp 90] J. Campos. *Performance Bounds for Synchronized Queueing Networks*. PhD thesis, Universidad de Zaragoza, 1990.
- [Catt 92] D. G. Cattrysse and L. N. V. Wassenhove. “A survey of algorithms for the generalized assignment problem”. *European Journal of Operational Research*, Vol. 60, pp. 260–272, 1992.
- [chol 97] A. choll, R.K., and C. Jürgens. “Bison: A fast hybrid procedure for exactly solving the one-dimensional bin packing problem”.

- Computers & Operations Research*, Vol. 24, No. 7, pp. 627 – 645, 1997.
- [Chun 82] F. R. K. Chung, M. R. Garey, and D. S. Johnson. “On Packing Two-Dimensional Bins”. *SIAM Journal of Algebraic and Discrete Methods*, Vol. 3, pp. 66–76, 1982.
- [Coff 06] E. Coffman and G. S. Lueker. “Approximation algorithms for extensible bin packing”. *Journal of Scheduling*, Vol. 9, pp. 63–69, 2006.
- [Cohe 06] R. Cohen, L. Katzir, and D. Raz. “An efficient approximation for the Generalized Assignment Problem”. *Information Processing Letters*, Vol. 100, No. 4, pp. 162 – 166, 2006.
- [Csir 98] J. Csirik and G. J. Woeginger. *On-line packing and covering problems*, Chap. 7, pp. 147 – 177. Springer Berlin / Heidelberg, 1998.
- [Dawa 01] M. Dawandea, J. Kalagnanam, and J. Sethuramanc. “Variable Sized Bin Packing With Color Constraints”. *Electronic Notes in Discrete Mathematics*, Vol. 7, pp. 154–157, 2001.
- [Deck 08] G. Decker, R. Dijkman, and M. Dumas. “Transforming BPMN Diagrams into YAWL Nets”. *Business Process Management*, Vol. 5240/2008, pp. 386–389, 2008.
- [Dell 98] P. Dell’Olmo, H. Kellerer, M. Speranzac, and Z. Tuza. “A 13/12 approximation algorithm for bin packing with extendable bins”. *Information Processing Letters*, Vol. 65, pp. 229–233, 1998.

- [Dijk 07] R. Dijkman, M. Dumas, and C. Ouyang. “Formal semantics and analysis of BPMN process models using Petri nets”. Tech. Rep., 7115, Queensland University of Technology, Brisbane, 2007.
- [GGal 95] G. Galambos and G. Woeginger. “On-line bin packing - A restricted survey”. *Mathematical Methods of Operations Research*, Vol. 42, pp. 25–45, 1995.
- [Gine 07] P. Giner, V. Torres, and V. Pelechano. “Bridging the Gap between BPMN and WS-BPEL. M2M Transformations in Practice”. *3rd International Workshop on Model-Driven Web Engineering*, Vol. 261, 2007.
- [Jens 03] K. Jensen. *Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use*. Springer, 2nd edition, 2003.
- [Jens 07] K. Jensen, L. Kristensen, and L. Wells. “Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems”. *International Journal on Software Tools for Technology Transfer (STTT)*, Vol. 9, pp. 213–254, 2007.
- [JPuc 07] J. Puchinger and G. Raidla. “Models and algorithms for three-stage two-dimensional bin packing”. *European Journal of Operational Research*, Vol. 183, pp. 1304–1327, 2007.
- [LEp 07] R. v. S. L. Epstein. “Online bin packing with resource augmentation”. *Discrete Optimization*, Vol. 4, pp. 322–333, 2007.
- [Lodi 02] A. Lodi, S. Martello, and M. Monaci. “Two-dimensional packing problems: A survey”. *European Journal of Operational Research*, Vol. 141, pp. 241–252, 2002.

- [Lodi 99] A. Lodi, S. Martello, and D. Vigo. “Heuristic and metaheuristic approaches for a class of two-dimensional bin packing problems”. *INFORMS Journal on Computing*, pp. 345–357, 1999.
- [M Ga 79] D. J. M. Garey. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, New York, 1979.
- [Mart 00] S. Martello, D. Pisinger, and D. Vigo. “The Three-Dimensional Bin Packing Problem”. *OPERATIONS RESEARCH*, Vol. 48, pp. 256–267, 2000.
- [Mart 90] S. Martello and P. Toth. *Knapsack problem algorithms and computer implementation*. John Wiley and Sons, 1990.
- [Miya 07] F. Miyazawa and Y. Wakabayashi. “Two- and three-dimensional parametric packing”. *Computers & Operations Research*, Vol. 34, pp. 2589–2603, 2007.
- [Mura 89] T. Murata. “Petri nets: Properties, analysis and applications”. *Proceedings of the IEEE*, Vol. 77, No. 4, pp. 541–580, Apr 1989.
- [Nuto 06] Z. Nutov, I. Beniaminy, and R. Yuster. “A $(1-1/e)$ -approximation algorithm for the generalized assignment problem”. *Operations Research Letters*, Vol. 34, No. 3, pp. 283 – 288, 2006.
- [OMG 06] O. M. G. (OMG). *Business Process Modeling Notation Specification, Final Adopted Specification*. February 2006.
- [Osyc 85] A. Osyczka. “Multicriteria optimization for engineering design”. In: J. S.Gero, Ed., *Design Optimization*, pp. 193–227, Academic Press, 1985.

- [Ouya] C. Ouyang, M. Dumas, W. van der Aalst, and A. ter Hofstede. “From business process models to process-oriented software systems: The BPMN to BPEL way”. Tech. Rep., 06-27, BPMcenter (2006).
- [Ouya 06a] C. Ouyang, M. Dumas, S. Breutel, and A. ter Hofstede. “Translating standard process models to BPEL”. *Advanced Information Systems Engineering*, pp. 417–432, 2006.
- [Ouya 06b] C. Ouyang, M. Dumas, A. ter Hofstede, and W. van der Aalst. “From BPMN Process Models to BPEL Web Services”. *Proceedings of the IEEE International Conference on Web Services*, 2006.
- [Ouya 06c] C. Ouyang, W. van der Aalst, M. Dumas, , and A. ter Hofstede. “Translating BPMN to BPEL”. Tech. Rep., BPM-06-02, BPMCenter.org, 2006.
- [Ouya 07] C. Ouyang, M. Dumas, A. ter Hofstede, and W. van der Aalst. “Pattern-based translation of bpmn process models to bpel web services”. *International Journal of Web Services Research (JWSR)*, 2007.
- [Petr 62] C. Petri. *Kommunikation mit Automaten*. PhD thesis, Institut für instrumentelle Mathematik, Bonn, 1962.
- [Poly 07] S. Polyakovsky and R. MHallah. “An agent-based approach to the two-dimensional guillotine bin packing problem”. *European Journal of Operational Research*, 2007.
- [Pran 08] D. Prandi, P. Quaglia, and N. Zannone. *Coordination Models and Languages*, Chap. Formal Analysis of BPMN Via a Translation

- into COWS, pp. 249–263. Vol. 5052 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, 2008.
- [Puhl 06] F. Puhlmann and M. Weske. *Business Process Management*, Chap. Investigations on Soundness Regarding Lazy Activities, pp. 145–160. Springer Berlin / Heidelberg, 2006.
- [Raed 07] I. Raedts, M. Petkovic, Y. Usenko, J. der Werf, J. Groote, and L. Somers. “Transformation of BPMN Models for Behaviour Analysis”. In: J. C. Augusto, J. Barjis, and U. Ultes-Nitsche, Eds., *MSVVEIS*, pp. 126–137, INSTICC PRESS, 2007.
- [Reck 06] J. Recker and J. Mendling. “On the Translation between BPMN and BPEL: Conceptual Mismatch between Process Modeling Languages”. In *Proceedings 18 th International Conference on Advanced Information Systems Engineering*, pp. 521–532, 2006.
- [Ross 75] G. Ross and R. Soland. “A branch and bound algorithm for the generalized assignment problem”. *Mathematical Programming*, Vol. 8, pp. 91–103, 1975.
- [Save 97] M. Savelsbergh. “A Branch-And-Price Algorithm for the Generalized Assignment Problem”. *Operations Research*, Vol. 45, pp. 831–841, 1997.
- [Shra 08a] A. Shraideh, H. Camus, and P. Yim. “Two stages optimization problem: New variant of Bin Packing Problem for decision making”. In: *Computer Science and Information Technology, 2008. IMCSIT 2008. International Multiconference on*, pp. 921–925, Oct. 2008.

- [Shra 08b] A. Shraideh and H. C. P. Yim. “New generalized assignment problem with identified first-use bins”. In: *VI ALIO/EURO Workshop on Applied Combinatorial Optimization*, Facultad de Ciencias Exactas y naturales, Universidad de Buenos Aires, December, 2008. 2008.
- [Shra 09] A. Shraideh, H. Camus, and P. Yim. “Business Process Optimization by Workflow Analysis”. In: *22nd International Conference on Computer Applications in Industry and Engineering (CAINE)*, San Francisco, CA, USA, 2009.
- [Sifa 80] J. Sifakis. *Net Theory and Applications*, Chap. Performance evaluation of systems using nets, pp. 307–319. Springer Berlin, 1980.
- [Whit 04a] S. White. “Introduction to BPMN”. Tech. Rep., Available at www.bpmn.org, 2004.
- [Whit 04b] S. White. “Workflow Patterns with BPMN and UML”. Tech. Rep., IBM, January 2004.
- [Whit 05] S. White. “Using BPMN to Model a BPEL Process”. *BPTrends*, Vol. 3(3), pp. 1–18, 2005.
- [Wohe 05] P. Wohed, W. van der Aalst, M. Dumas, A. ter Hofstede, and N. Russell. “Pattern-based analysis of BPMN - an extensive evaluation of the control-flow, the data and the resource perspectives.”. Tech. Rep., BPM Center Report BPM-05-26, BPMcenter.org, 2005.
- [Wong 08] P. H. Wong and J. Gibbons. *Formal Methods and Software Engineering*, Chap. A Process Semantics for BPMN, pp. 355–374.

Vol. 5256 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, 2008.

- [Ye 08] J. Ye, S. Sun, W. Song, and L. Wen. “Formal Semantics of BPMN Process Models Using YAWL”. In: *Intelligent Information Technology Application, 2008. IITA '08. Second International Symposium on*, pp. 70–74, Dec. 2008.