

**UNIVERSITA' DEGLI STUDI DI NAPOLI
"FEDERICO II"**



**DOTTORATO DI RICERCA IN ECONOMIA E MANAGEMENT DELLE
AZIENDE E DELLE ORGANIZZAZIONI SANITARIE**

**STRUMENTI PER LA MODELLAZIONE
E L'ANALISI DEI PROCESSI SANITARI**

Relatore

Ch.mo Prof. Alessandro Pepino

Ch.ma Prof.ssa Maria Triassi

Candidato

Marcella Rovani

Sommario

1. INTRODUZIONE.....	4
2. I SISTEMI DI BUSINESS PROCESS MANAGEMENT	7
2.1 Definizioni	7
2.2 La modellazione dei processi tramite Workflow	11
2.3 Architettura dei Sistemi di Workflow Management	14
2.4 Le Reti di Petri	22
3. STUDIO DEI FLUSSI SANITARI MEDIANTE WORKFLOW	38
3.1 Il Simulatore Woped	38
3.2 Case Study: Il Flusso di Gestione Centralizzata degli acquisti di farmaci e dispositivi medici	55
4. BUSINESS PROCESS MANAGEMENT E PROCESS MINING	69
4.1 Introduzione	69
4.2 Caratteristiche del Process Mining	70
4.3 Le tecniche di Process Mining	72
4.4 Gli strumenti per il Process Mining	77
5. MODELLAZIONE DEI PROCESSI SECONDO IL LINGUAGGIO DICHIARATIVO.....	82
5.1.Introduzione	82
5.2 Modellazione dichiarativa dei processi: il tool Declare.....	84

6. PROCESS MINING DICHIARATIVO NEL SETTORE SANITARIO: UNA NUOVA PROPOSTA METODOLOGICA	90
6.1 Premesse.....	90
6.2 Metodologia	93
6.3 Case Study: il trattamento chirurgico del criptorchidismo	96
7. CONCLUSIONI.....	106
Bibliografia	108

1. INTRODUZIONE

A seguito della diffusione e del rafforzamento - nello scenario italiano ed internazionale - delle politiche manageriali di tipo aziendalistico delle Organizzazioni Sanitarie, [1] [2], é stata rivolta sempre maggiore attenzione all'esigenza di attuare una gestione efficace ed efficiente dei processi clinici da realizzare a tutela della salute [3].

Fenomeni tipici dei Paesi Occidentali, quali l'invecchiamento della popolazione, la crescita della domanda dei servizi sanitari, unitamente all'evoluzione tecnologica ed alla crescente sensibilizzazione verso l'appropriatezza della cura e la sicurezza dei pazienti, hanno reso il contesto sanitario un mercato dinamico in rapida evoluzione. In tale panorama le Organizzazioni Sanitarie, come gli Ospedali, tendono sempre più ad incrementare la produttività e a ridurre i costi di gestione delle loro attività, garantendo al contempo elevati livelli di assistenza alla popolazione [3].

La gestione ed il monitoraggio dei processi clinici hanno pertanto assunto un ruolo fondamentale nel contesto del management sanitario. Il sistema sanitario è infatti caratterizzato da processi molto complessi e altamente flessibili, che comportano il coinvolgimento di un numero significativamente elevato di figure professionali e la registrazione di grossi volumi di dati: basti pensare che per un gruppo di pazienti nella stessa condizione clinica possono essere richiesti diversi esami, un diverso ordine di esecuzione delle attività o l'intervento di particolari figure specialistiche a seconda della singolarità dei casi trattati [4].

La crescente innovazione tecnologica ha inoltre determinato un'incessante introduzione di nuovi programmi sanitari, aumentando, di conseguenza, le possibilità di scelta tra diverse tipologie di trattamento. E' stata pertanto focalizzata l'attenzione sul metodo strutturato e multidimensionale di valutazione delle tecnologie denominato *Health Technology Assessment* [5], e sulla tecnica di aggregazione dei dati nota come MetaAnalisi, che consente la stima quantitativa degli outcome derivanti dall'impiego delle tecnologie biomediche e in generale dai flussi clinici [6] [7].

Dalle ricerche condotte e da numerosi riscontri, sia in ambito nazionale che internazionale, emerge in modo chiaro che un corretto studio di HTA richiede necessariamente una razionalizzazione ed informatizzazione dei processi clinici, al fine di ottenere una stima corretta degli outcome connessi ai flussi di interesse e dei relativi costi e risorse, ragionando dunque sia in termini di efficacia che di efficienza. [8]. Uno studio di valutazione delle tecnologie condotto in presenza di flussi clinici non informatizzati comporta, infatti, un enorme dispendio di energie per la raccolta e l'analisi dei dati, i quali non sempre sono disponibili in tempo reale; risulta inoltre impossibile stimare l'efficacia di una

tecnologia o di un processo in un contesto che potrebbe essere di per sé inefficiente o inefficace per carenze legate al flusso di lavoro.

E' emersa, quindi, la necessità di approfondire lo studio degli strumenti di analisi e di progetto con particolare riferimento all'implementazione di sistemi informativi dedicati alla simulazione ed alla reingegnerizzazione dei flussi di lavoro che caratterizzano i processi clinici [9].

Il progetto di ricerca condotto ha pertanto avuto ad oggetto l'approfondimento delle tecniche di analisi e di progetto dei flussi di lavoro e di applicazione dei Sistemi di Workflow Management nel settore sanitario.

I Sistemi di Workflow Management comprendono la vasta area di software progettati per codificare, rappresentare e simulare flussi di lavoro al fine di ottimizzarli e automatizzarli, con l'obiettivo ultimo di raggiungere un incremento dell'efficienza attraverso un migliore controllo del processo, una maggiore flessibilità e una più alta qualità dei servizi offerti [10].

Un sistema di Workflow correttamente realizzato è in ogni momento in grado di fornire una vista sullo stato corrente e mostrare la sequenza di eventi che hanno portato a quel determinato stato. [11] Tale prerogativa, pertanto, può spingere ad adottare un tipo di sistema piuttosto che un altro a seconda del caso applicativo, soprattutto in ambito sanitario, in cui sono necessarie l'integrazione e la coordinazione in un contesto multiutenza. [8] .

Nel presente lavoro di ricerca e' stato pertanto effettuato uno studio delle diverse tipologie di software esistenti per la gestione dei workflow. Tale ricerca è stata affiancata dalla contestuale verifica, mediante esempi applicativi, delle potenzialità di tali software nel rispondere alle numerose e particolari esigenze di analisi dei processi sanitari, con particolare riferimento alla definizione delle attività costituenti i processi medesimi, delle risorse, dei tempi e dei costi ad essi associati, e al monitoraggio online ed offline.

I primi strumenti analizzati sono stati simulatori di Workflow basati sulla teoria delle reti di Petri [12], principalmente il simulatore Woped (www.woped.org). Le reti di Petri sono un formalismo - matematico e grafico allo stesso tempo - che consente di modellare un processo definendone le attività, assegnando a ciascuna di esse risorse (persone, beni etc.) e durata, e specificando le regole che determinano l'evoluzione del flusso. Basandosi sul formalismo delle reti di Petri, il tool open source Woped ha consentito di disegnare ed analizzare alcuni processi di interesse, un esempio dei quali attiene alla centralizzazione degli acquisti di farmaci e dispositivi medici all'interno della Regione Campania [13]. In particolare, una volta disegnato il modello (sulla base di informazioni raccolte mediante osservazione diretta del processo), ne è stato simulato il comportamento in differenti condizioni operative, mettendo in luce i punti deboli e ipotizzando strategie migliorative.

Mediante il simulatore Woped, è stato inoltre analizzato il workflow del Centro di Ipertensione dell'AOU Federico II, [14] con l'individuazione dei 'colli di bottiglia' che determinano la generazione delle code e l'allungamento delle liste di attesa e dei tempi per l'accesso alle cure da parte dei cittadini. L'individuazione quantitativa delle problematiche alla base delle liste di attesa ha consentito di individuare nella riorganizzazione del personale una possibile strategia di riduzione delle code.

Se le tecniche di simulazione dei Workflow consentono di valutare quantitativamente gli aspetti legati alle risorse, ai tempi ed ai costi di un determinato flusso di lavoro nelle modalità in cui esso appare all'osservazione diretta, l'esigenza di monitorare l'esecuzione concreta e puntuale dei processi all'interno delle strutture sanitarie ha spinto ad approfondire anche le tecniche di Process Mining. [15]. Le tecniche di Process Mining consentono di estrarre, a partire dai dati registrati all'interno dei Sistemi Informativi Sanitari (sotto forma di 'event log'), informazioni sui processi eseguiti all'interno di una particolare unità informativa, tipicamente il singolo reparto o centro terapeutico-diagnostico. In tal modo è possibile migliorare la conoscenza dei processi e rendere più veritiera la loro rappresentazione secondo la logica dei workflow.

L'analisi degli event log assume inoltre un ruolo non secondario nella verifica della compliance dei percorsi di cura messi in atto all'interno delle unità operative con le prescrizioni delle Linee Guida Nazionali ed Internazionali. Nel corso dello studio, in particolare, sono stati analizzati i flussi di lavoro attuati per la cura di alcune patologie urinarie, quali il criptorchidismo e il cancro alla vescica. Mediante lo studio delle European Guidelines of Urology, sono stati disegnati i modelli 'guidelines-based' dei processi, successivamente validati con chirurghi esperti del settore. Utilizzando il tool per Process Mining ProM (www.promtools.org), tali modelli sono stati poi confrontati con i log di eventi registrati all'interno del Sistema Informativo del Reparto di Urologia dell'Ospedale di Zwolle (The Netherlands). Tale confronto ha consentito di individuare tipologie di deviazioni del processo reale da quello prescritto, e di correlarle alla tipologia di paziente (es: maschio/femmina, fumatore, diabetico), aprendo interessanti prospettive per la particolarizzazione dei percorsi di cura e per l'individuazione di modalità di gestione delle eccezioni terapeutiche.

Nel presente lavoro di tesi vengono approfondite le tecniche di analisi dei processi citate, con la descrizione dettagliata di alcuni casi di studio.

In particolare, i capitoli II e III si soffermano sui sistemi di Workflow Management e sulle reti di Petri, includendo la descrizione di un caso di studio. I capitoli IV e V sono dedicati rispettivamente al Process Mining ed al linguaggio dichiarativo. Nel capitolo VI viene proposta l'adozione di una nuova metodologia di analisi dei processi sanitari. Il capitolo VII conclude il presente lavoro.

2. I SISTEMI DI BUSINESS PROCESS MANAGEMENT

2.1 Definizioni

L'approccio al Business Process Management è storicamente scaturito dall'osservazione che ciascun prodotto di una qualsiasi organizzazione derivi dall'impiego di una determinata quantità di risorse a seguito del lavoro di un determinato numero di persone. E' necessario, pertanto, applicare tecniche di governo di tali processi al fine di gestire in maniera efficiente le risorse umane e materiali. In tale scenario i Sistemi Informativi giocano un ruolo chiave, dal momento che la maggior parte delle attività svolte all'interno di una particolare organizzazione richiede un flusso di informazioni chiaro e disponibile in tempo reale.

Come riportato da [16], si definisce *Business Process* l'insieme delle attività svolte congiuntamente all'interno di un ambiente tecnico-organizzativo; tali attività sono nel loro insieme finalizzate al raggiungimento di un obiettivo comune. Ciascun business process è gestito da una singola organizzazione, ma può interfacciarsi con processi messi in atto da altre organizzazioni.

In tale ottica, diverse definizioni in Letteratura sono state date al Business Process Management.

Una prima definizione è stata data da Horak et Al. [17] Essa afferma che *il Business Process Management è un approccio sistematico e strutturato per analizzare, migliorare, controllare e gestire i processi di business con lo scopo di migliorare la qualità dei prodotti e dei servizi.*

Un'altra definizione di BPM è quella data da Weske e van der Aalst [18], secondo cui *un BPM è un sistema il cui scopo è quello di supportare i processi di business utilizzando metodi, tecniche e software per progettare, mettere in esecuzione, controllare e analizzare i processi operativi che coinvolgono esseri umani, organizzazioni, applicazioni, documenti e altre fonti di informazioni.*

L'ultima definizione che citeremo è quella data da Gartner [19], secondo cui *un sistema BPM è un sistema composto da servizi e strumenti che supportano in modo esplicito la gestione del processo di business (analisi, definizione, esecuzione, monitoraggio e amministrazione).*

Dall'unione delle tre definizioni di Business Process Management è possibile ricavarne una che descrive in modo completo un sistema BPM: il Business Process Management è una disciplina gestionale che utilizza un approccio sistematico e strutturato con il fine di supportare la gestione esplicita di un processo di business utilizzando metodi, tecniche e strumenti, che coinvolgono esseri umani, organizzazioni, applicazioni, documenti e altre fonti di informazione, con lo scopo di raggiungere gli obiettivi di business dell'organizzazione allineando i processi di business a questi obiettivi.

Il *Business Process Management* include dunque concetti, metodi e tecniche a supporto della modellazione, della gestione, della configurazione, dello sviluppo, del monitoraggio e dell'analisi dei processi eseguiti. La base del business process management è rappresentata dalla modellazione dei processi come relazioni tra attività che coinvolgono diverse risorse: una volta che il processo è stato definito, è possibile analizzarlo ed ottimizzarlo.

In passato, la gestione dei processi era manuale, guidata dall'esperienza del personale e dalle procedure trascritte su carta. Nel corso del tempo, le imprese hanno iniziato a dotarsi di Sistemi Informativi a supporto delle attività svolte, traendone numerosi benefici, fino all'introduzione dei Business Process Management Systems.

Un *Business Process Management System* (BPMS) è un sistema software esplicitamente progettato per gestire l'esecuzione dei processi di business [17]

Nonostante infatti l'applicazione delle tecniche di BPM abbia considerevoli vantaggi anche solo limitatamente all'analisi dei modelli cosiddetti 'penna e carta', come i diagrammi di flusso, la vera rivoluzione si ha quando tali applicazioni mirano all'automazione dei processi medesimi [20]. Nel contesto del Business Process Automation, al quale ci si riferisce solitamente come **Workflow Management**, una precisa descrizione di un processo è finalizzata a guidarne l'esecuzione con una determinata performance.

Nell'esecuzione di un processo, il lavoro è assegnato a determinate risorse, che possono essere sia esseri umani che soluzioni software che svolgono azioni automatizzate; tali risorse eseguono le attività o task ad esse assegnate in un determinato tempo, con determinati costi. Definendo attività e risorse, è possibile quantificare il carico di lavoro di ciascuna risorsa e la sua disponibilità ad eseguire ulteriori compiti. In molti casi, è utile classificare le risorse in base ai ruoli da esse svolte nel processo così da definire l'assegnazione dei compiti in base ai ruoli, come accade all'interno delle organizzazioni pubbliche e private. Come conseguenza della esplicita definizione delle attività costituenti il processo e della loro sequenza temporale, così come della categoria di risorse deputata all'esecuzione di tali attività, è possibile controllare i processi ed adattarli tempestivamente ai cambiamenti del contesto circostante, come fluttuazioni del mercato ed innovazioni legislative.

La realizzazione di un sistema di business process management passa attraverso una serie di stadi che si ripetono ciclicamente, secondo quello che viene definito **Business Process Lifecycle** [21], rappresentato in Figura 1, che definisce il ciclo di vita del sistema dalla sua progettazione, alla sua evoluzione.

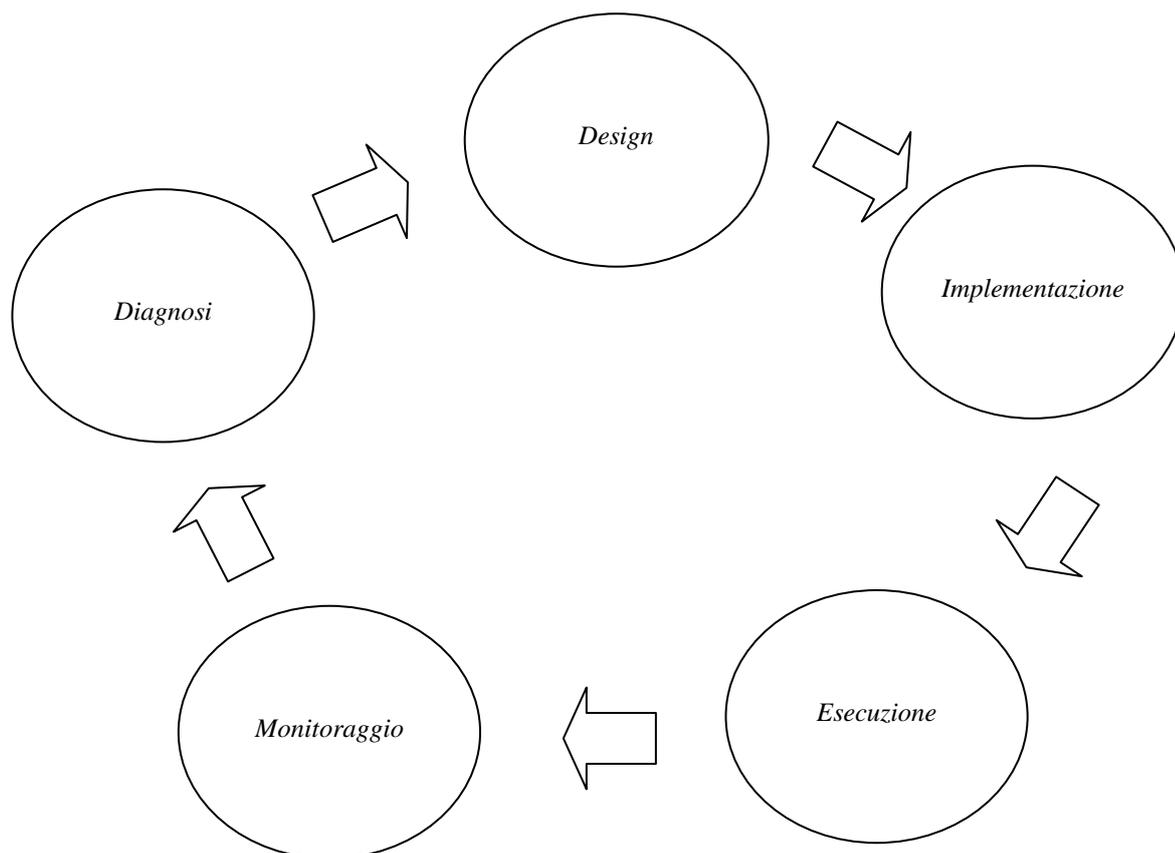


Fig.2.1.1: BPM Lifecycle

Come mostrato in fig.1, la prima fase del Lifecycle prevede il disegno del processo. Viene pertanto eseguita un'indagine volta ad identificare le caratteristiche del processo stesso, insieme all'individuazione delle tipologie di linguaggio che più si adattano alla sua rappresentazione. L'output di questa fase è la formalizzazione di un modello di processo, che viene validato con il supporto degli stakeholders (nel caso di processi sanitari medici, infermieri, farmacisti etc) e di strumenti quali interviste, questionari, checklists.

Una volta che il modello è stato disegnato, è necessario implementarlo.

L'implementazione di un processo può richiedere o meno il supporto di particolari strumenti software. In alcune organizzazioni, infatti, ci si limita a definire verbalmente un insieme di regole e procedure alle quali il personale addetto all'esecuzione del processo dovrà attenersi per garantirne il completamento. Qualora siano previsti software dedicati a supporto dell'esecuzione del processo, essi devono essere in questa fase opportunamente configurati. La configurazione prevede anche l'integrazione con eventuali software esistenti per l'esecuzione di task elementari inglobate nel

processo; tale integrazione può rivelarsi più o meno complessa a seconda dell'architettura del sistema informativo esistente.

Terminata la configurazione, è necessario effettuare delle simulazioni per individuare e risolvere eventuali problemi di runtime e di interfaccia dei software eventualmente installati con quelli preesistenti. A seconda dei casi, è possibile che in questo step venga prevista anche un'attività di training del personale in merito all'impiego dei software installati.

A questo punto inizia l'esecuzione del processo, finalizzata al raggiungimento di predeterminati obiettivi. Tipicamente il processo inizia con un particolare input, quale ad esempio, nel contesto sanitario, la richiesta di effettuare un esame. Nei moderni sistemi di workflow management, è possibile monitorare online il processo medesimo: in ogni istante è infatti possibile conoscere quale attività è in esecuzione e quale risorsa la sta svolgendo. Il monitoraggio è fondamentale per apportare prontamente azioni correttive e/o fornire informazioni in merito allo stato del processo ad eventuali supervisori e/o responsabili.

Il monitoraggio, sia che venga eseguito in modalità online (ovvero durante l'esecuzione del processo) che offline (al termine del processo), è fondamentale a fornire i dati necessari per la diagnosi, durante la quale vengono effettuate dettagliate analisi in merito ai tempi, alle risorse e ai costi del processo eseguito. Diverse tecniche, come il Process Mining, che si basa sull'analisi dei log di eventi registrati all'interno dei Sistemi Informativi [22] [15], oppure gli strumenti di simulazione [23] consentono di estrapolare utili informazioni individuando eventuali opportunità di ottimizzazione del flusso di lavoro. Tale opportunità si traduce in una revisione del modello del processo stesso, con una nuova ripetizione del ciclo.

L'importanza ed il significato del BPM Lifecycle è fondamentale legata a due aspetti. Il primo è che i processi sono spesso complessi e richiedono continue evoluzioni ed adattamenti dei modelli che li descrivono. La seconda deriva dalla difficoltà di standardizzare i modelli per particolare categoria di processo, dal momento che la stessa tipologia di flusso di lavoro può esibire aspetti diversi a seconda del contesto in cui viene eseguito. Ciò rende necessario prevedere un approccio flessibile e ciclico nell'analisi dei processi.

Come risultato del BPM Lifecycle ed in particolare dei risultati delle attività diagnostiche relative all'esecuzione del processo, si introduce il concetto di **Business Process Re-engineering**, BPR. Si intende per Business Process Re-engineering un intervento organizzativo di profonda revisione dei procedimenti operativi al fine di conseguire un miglioramento radicale della loro esecuzione [24]. Lo stimolo verso un intervento di business re-engineering può essere originato dai risultati di un benchmarking, da un'analisi di customer satisfaction, da un mutamento del quadro operativo, dalla

ridefinizione degli obiettivi aziendali, dall'evidenza di parametri critici che indichino la necessità o l'opportunità di migliorare l'efficacia e l'efficienza di un processo aziendale, dalla necessità di introdurre nuove metodologie di lavoro, o da altre situazioni legate al contesto specifico. Lo scopo del reengineering non è il raggiungimento di miglioramenti marginali, ma piuttosto sostanziali del processo. Il mezzo con cui giungere ad un miglioramento sostanziale non consiste soltanto nella modifica degli schemi esistenti, ma in una totale riprogettazione dei flussi di lavoro. La progettazione infatti è di primaria importanza, in quanto un processo non correttamente pianificato, anche se eseguito da personale esperto e motivato, non potrà risultare performante: in questo panorama si introduce la struttura a feedback del BPM Lifecycle, che prevede il ritorno alla modellazione a seguito della diagnosi, secondo un ciclo continuo ed iterativo.

2.2 La modellazione dei processi tramite Workflow

Come accennato nel paragrafo precedente, un particolare aspetto da tenere in considerazione qualora ci si appresti a gestire un processo mediante gli strumenti di Business Process Management, è la necessità di monitorarne l'esecuzione. Nelle aziende dotate di un sistema di gestione della qualità, ad esempio, in accordo alla norma ISO 9001¹, i processi aziendali devono essere misurabili e monitorabili nel tempo mediante l'utilizzo di indicatori di prestazione chiave, detti KPI 'Key Performance Indicators'.

I programmi che si occupano della rilevazione degli indicatori di prestazione chiave forniscono dei resoconti sintetici sull'operatività dei processi; il livello di dettaglio degli indicatori può andare dal livello globale della società al singolo operatore/macchina. Per conseguire tale risultato, l'utilizzo di sistemi di Workflow Management è una delle tecniche più diffuse nell'ambito del Business Process Management.

Con l'espressione di 'Sistemi di Workflow Management' (WMGS) si intendono tutti quegli strumenti che favoriscono ed implementano strategie di collaborazione [25]. L'impiego di tali strumenti è stato ufficialmente sostenuto e promosso dalla Workflow Management Coalition (WfMC), organizzazione internazionale no-profit fondata nel 1993 e composta da venditori sviluppatori, analisti e ricercatori. La missione del WfMC è stata promuovere l'uso dei workflow attraverso l'istituzione di standard software per la terminologia, interoperabilità, connettività tra prodotti di workflow.

¹ <http://www.iso.org/iso/home.html>

La WfMC definisce i Workflow Management System come ‘Sistemi che definiscono, creano e gestiscono l’esecuzione di workflow attraverso l’uso di applicazioni software’. Essi sono in grado di interpretare gli schemi di workflow, interagire con gli users ed invocare l’uso di strumenti e applicazioni automatizzate [26]. In pratica un sistema di gestione di Workflow (WfMS) è un sistema software che supporta l’esecuzione coordinata di semplici attività, assegnandole a esecutori umani o automatizzati, per raggiungere un obiettivo prefissato di un processo di business [11]. Il termine **workflow** può dunque essere tradotto in italiano come ‘*gestione elettronica dei processi lavorativi*’. In modo pratico, un workflow definisce i compiti che ciascun attore del processo è tenuto a portare a termine, i tempi da rispettare, e lo scambio di informazioni necessarie a ciascun attore per la realizzazione del proprio compito.

Da quanto spiegato finora, emerge chiaramente che all’impiego dei sistemi di workflow per la gestione di processi di business possono essere associati dei benefici tangibili quali la riduzione dei costi, l’aumento della produttività e la riduzione dei tempi di esecuzione. Ci sono poi alcuni vantaggi derivanti dall’impiego dei workflow che non sono direttamente misurabili, ma non per questo di secondaria importanza, e sono i cosiddetti ‘benefici non tangibili’, che consistono in un miglioramento della qualità dei servizi ed in una gestione più efficace delle informazioni, insieme ad un maggiore supporto alle decisioni e alla pianificazione.

Le caratteristiche di un workflow che concernono la descrizione degli aspetti che sono rilevanti per il controllo ed il coordinamento delle attività durante la loro esecuzione, così come i rapporti tra le attività stesse, prendono il nome di **specifiche**.

A seconda di tali specifiche, i workflow possono dividersi in tre categorie [27]:

1. Workflow produttivi:

I workflow produttivi sono orientati alla produzione su larga scala e all’implementazione dei processi critici di un’azienda, coinvolgendo pertanto un elevato numero di utenti e di sistemi. Essi gestiscono un gran numero di attività simili, ripetitive e prevedibili, al fine di ottimizzare la produttività intesa ad esempio come pezzi prodotti per unità di tempo. I workflow produttivi sono orientati ad ottenere un alto livello di automazione, per trarre il massimo vantaggio dall’assenza di componente umana, necessaria spesso solo per gestire eventuali eccezioni. Essi gestiscono processi altamente strutturati con pochissime variazioni. Questo tipo di workflow si trova ad affrontare molti casi nell’arco di una giornata; un classico esempio sono i flussi assicurativi.

2. Workflow amministrativi

I workflow amministrativi si caratterizzano per la loro facilità di definizione e di modifica. Essi implementano infatti flussi di lavoro che evolvono a seconda delle necessità e dell'esperienza acquisita durante l'utilizzo: un esempio è il workflow relativo alla gestione di uffici comunali, dove le regole procedurali devono seguire l'evoluzione delle normative ministeriali. Per questa tipologia di workflow la flessibilità è molto più importante della produttività, e pertanto tipicamente essi consentono la gestione di un numero di istanze ogni ora minore di almeno uno o due ordini di grandezza rispetto ai workflow produttivi. Essi descrivono processi 'case-driven', vale a dire processi che seguono una procedura ben definita; un percorso alternativo per un caso è possibile solo se questo è stato precedentemente predefinito.

3. Workflow collaborativi

I workflow collaborativi si focalizzano sulla comunicazione e sullo scambio di informazioni fra gruppi di lavoro. Essi sono tipici dei contesti in cui non è richiesta un'alta produttività, quanto piuttosto si rendono necessari strumenti di confronto e condivisione.

Diversamente dalle altre tipologie di workflow, essi non si basano su un flusso che procede in direzione di uno stadio finale, ma prevedono una serie di iterazioni della stessa attività, o addirittura ripercorrono quelle precedenti fino al raggiungimento di un accordo. I workflow collaborativi utilizzano spesso la posta elettronica per consegnare e negoziare il lavoro; sono pertanto caratterizzati da bassa produttività e massima flessibilità. Essi sono adatti per la gestione di processi meno rigidi e prevedono una relazione tra le risorse più vicina a quella della collaborazione tra gruppi di lavoro.

4. Workflow ad hoc

Il workflow ad hoc è una tipologia di workflow che viene definito a seconda del particolare processo considerato, il quale richiede un workflow apposito, realizzato di volta in volta. Esso viene utilizzato per modellare processi la cui struttura non è definita completamente a priori: consente pertanto lo svolgimento di determinate attività in maniera dinamica, anche quando il processo risulta già essere in esecuzione.



Fig.2.2.1: Tipologie di Workflow

Oltre al raggiungimento di elevatissimi livelli di interoperabilità, i workflow consentono di ottenere l'automazione del processo, con una notevole riduzione dei tempi e degli errori nell'esecuzione delle attività, un miglioramento della qualità del servizio offerto dall'organizzazione e del controllo sui processi stessi. Dal punto di vista dell'esecuzione, supportano la comunicazione, l'analisi e la comprensione dei fenomeni legati al ciclo di vita dei processi; consentono anche elevati livelli di monitoraggio, fornendo elementi per valutazioni sulla misura della performance e del processo e per il supporto decisionale.

2.3 Architettura dei Sistemi di Workflow Management

Come illustrato nel paragrafo precedente, un workflow è una struttura che include le attività da svolgere e le risorse deputate alla loro esecuzione, coordinando e supervisionando lo svolgimento del lavoro [21].

All'interno di un workflow, ogni fase di lavoro è attivata da una richiesta da parte di un utente e termina fornendo un'informazione da restituire all'utente stesso, o da trasferire ad altre fasi del

processo in esecuzione. Appositi canali di comunicazione garantiscono il trasferimento di tali informazioni, ed ognuna delle attività individuate in risposta alla richiesta attivata può essere eseguita non appena si rendono disponibili i dati necessari per il suo svolgimento.

I soggetti di un workflow sono risorse aziendali che eseguono il lavoro associato ad una particolare attività.

Le risorse possono essere classificate in:

- Risorse Umane: un singolo individuo o gruppi di persone che condividono un insieme di compiti da svolgere;
- Applicazioni Software;
- Hardware specifici.

I Workflow si basano sull'applicazione di tecnologie standard utilizzate nell'ambito dei Web Services [28], ovvero - come definito dal *World Wide Web Consortium*² - sistemi software progettati per supportare l'interoperabilità tra diversi elaboratori su una medesima rete: tipici esempi sono l'XML ed il WSDL. L'XML (eXtensible Markup Language) è un metalinguaggio in grado di modellare agevolmente una rilevante quantità di aspetti del mondo informatico. Il WSDL (Web Service Description Language) è il linguaggio di descrizione delle interfacce dei servizi ed è pensato per l'automatizzazione delle interazioni [29].

Per comprendere nella pratica l'architettura di un sistema di workflow management, è necessario prima di tutto definire gli elementi che caratterizzano un workflow.

L'obiettivo primario di un sistema di workflow è il trattamento dei casi. Ciascun caso è relativo al particolare processo che deve essere eseguito. Un processo è costituito da un certo numero di task da attuare a da e un insieme di condizioni che determinano una selezione quantitativa (temporale) e qualitativa (ad esempio nel caso di attività alternative) nella esecuzione di due o più task.

Due task che devono essere eseguite in stretto ordine costituiscono una **sequenza**: la task successiva nella sequenza non può essere eseguita finché la precedente non è terminata.

Il processo può prevedere inoltre, in un qualsiasi stadio della sua esecuzione, un costrutto condizionale che determina, in funzione di determinate variabili, quale task debba essere eseguita in alternativa tra diverse specificate opzioni. Ancora, alcune task possono essere eseguite in parallelo o in un ordine arbitrario perché tra loro non c'è una relazione di causalità. Affinché il processo giunga ad una conclusione, tuttavia, in un determinato punto tutti i rami di esecuzione parallela devono essere sincronizzati e convergere in un unico ramo che conduce allo stop del processo.

² <http://www.w3.org/>

Ciascun caso, relativo al particolare processo da eseguire, costituisce una **istanza** di uno schema di processo: se più casi sono trattati nello stesso modo appartengono allo stesso case type, cioè allo stesso processo. Un caso ha un ciclo di vita limitato, che si conclude con il termine del processo.

A ciascun caso sono tipicamente associati tre elementi:

- 1) Un insieme di **case attribute**, cioè un range di variabili locali (es: nel caso in cui il caso sia un paziente da sottoporre ad un esame diagnostico, case attribute possono essere il sesso e l'età)
- 2) Le **condizioni**, ovvero le opzioni di flusso determinate dagli attributi (es: ai pazienti in una determinata fascia di età viene effettuato un esame, ai pazienti in un'altra fascia due esami).
- 3) Il **contenuto** del caso, cioè gli oggetti, concreti o virtuali ad esso relativi.

Una task è l'unità elementare di esecuzione di un processo. Le task sono entità astratte che devono essere successivamente concretizzate nei casi: una **work item** è la istanza di una task all'interno di un caso.

Con il termine **attività** si intende l'esecuzione concreta di un work item. Le attività sono le uniche entità che possono modificare lo stato; il sistema di workflow può al più individuare lo stato corrente del processo. L'insieme di work item che in ogni momento possono diventare attività in esecuzione costituisce la cosiddetta **worklist**.

Una work item diventa un'attività quando una risorsa, assegnata all'esecuzione di quella particolare task, la prende in carico. Le risorse, dette anche attori, rappresentano quindi i mezzi di produzione che possono eseguire le task; esse possono essere persone fisiche, macchine, mezzi di trasporto o applicazioni. In un sistema reale, non tutte le risorse sono adibite all'esecuzione di tutte le task, ma ciascuna di esse prende in carico esclusivamente le tipologie coerenti con le proprie competenze. Pertanto, in un sistema di workflow, le risorse possono essere suddivise in classi, dette **ruoli**. Ogni ruolo identifica una capacità necessaria ad eseguire una o più task specifiche.

Le risorse che possono eseguire la stessa tipologia di task hanno lo stesso ruolo. Una risorsa può inoltre avere molteplici capacità e svolgere più ruoli nell'esecuzione di uno stesso processo.

Indipendentemente dalle caratteristiche specifiche di ogni singolo processo, la Workflow Management Coalition definisce un'architettura comune per la realizzazione dei Workflow Management Systems: tutti i sistemi di workflow, infatti, esibiscono alcune caratteristiche comuni che forniscono una base per lo sviluppo di capacità di integrazione ed interoperabilità tra diversi prodotti. Tale architettura di riferimento si basa sulla definizione di tre aree di funzionalità:

1. Funzionalità in fase di progettazione: riguardano la definizione e la modellazione del workflow, per quanto attiene ad esempio agli stati, alle proprietà dei casi, alle attività e alle

relazioni tra esse, consentendo la realizzazione di numerosi modelli per molteplici tipologie di processo.

2. Funzionalità in fase di esecuzione: sono relative alla gestione del workflow durante l'esecuzione, con particolare riferimento alla successione delle attività. In particolare, l'esecuzione e l'istradamento del processo sono gestiti da un modulo fondamentale, il **Workflow Enactment Service**, cuore dell'intero sistema di workflow management. Tale modulo consente, inoltre, l'adattabilità del workflow al processo, ovvero fa in modo che il flusso di lavoro venga modificato per ogni singolo caso istanziato.
3. Funzionalità di interazione con le risorse: siano esse utenti umani o tool informatici, sono deputate all'assegnazione dei work-item.

Le Figure 2.3.1 e 2.3.2 mostrano i principali componenti ed interfacce di un sistema di workflow management. Il cuore è rappresentato dal **Workflow Enactment Service**, che assicura che le giuste attività siano eseguite nell'ordine giusto e dalle persone corrette. Il Workflow Enactment Service può essere composto da molti motori di workflow o **Workflow Engine**: l'uso in parallelo di più motori di workflow può aumentare la scalabilità dell'intero sistema. Il Workflow Engine svolge le seguenti funzioni: creare nuovi casi e rimuove quelli conclusi, consentire l'attuazione dei meccanismi di routing, gestire gli attributi dei casi, sottoporre ciascun work item alla risorsa corretta, basandosi sulla classificazione dei ruoli, monitorare la consistenza o 'soundness' del flusso, gestire gli eventi di annullamento delle attività eventualmente generati dalle risorse.

Il motore di workflow supporta l'esecuzione del flusso così come esso è stato definito in fase di design. La definizione dei processi e la classificazione delle risorse, a secondo del caso di riferimento, è realizzata mediante appositi tools genericamente noti come **Process Definition Tools**. Tali strumenti permettono da un lato di modellare sequenze, paralleli, scelte alternative, attraverso delle apposite interfacce grafiche, dall'altro di definire gli attributi dei casi ed i tempi di esecuzione di ciascuna attività; verificano, inoltre, la correttezza sintattica dei processi definiti. Alcuni strumenti più evoluti effettuano anche una verifica strutturale del workflow, tramite la quale è possibile individuare l'eventuale presenza di task impossibili da raggiungere con i meccanismi di routing implementati o in presenza di loop infiniti, mettendo in atto dei meccanismi di warning. L'analisi strutturale infatti è finalizzata ad evitare la definizione di workflow in cui, al termine del flusso, ci possano essere attività ancora in esecuzione o nella worklist.

Le risorse coinvolte nell'esecuzione del processo entrano in contatto con il workflow attraverso la **Workflow Client Application**. Ogni risorsa ha una propria worklist contenente l'elenco delle attività che può eseguire in base al ruolo che svolge; naturalmente, si tratta di attività che aspettano

di essere prese in carico nel contesto del processo in esecuzione: la selezione di un elemento della worklist equivale alla notifica che la risorsa sta iniziando l'esecuzione di una particolare task. Esistono due modalità di assegnazione delle attività, denominate rispettivamente push e pull; in modalità push il motore assegna le attività alle singole risorse, forzandole ad eseguirle; in modalità pull un work-item è assegnata ad un gruppo di risorse all'interno del team, cioè quelle che ricoprono i ruoli deputati all'esecuzione di quella particolare attività; a questo punto una qualsiasi risorsa facente parte di questo gruppo può valutare quando e se prendere in carico l'attività. Oltre a presentare ad ogni risorsa la lista di work-item che essa può eseguire, il work-list handler può ordinare tali attività in base alle proprietà che hanno nello sviluppo del flusso.

Essendo particolarmente complessi, gli aspetti di gestione delle risorse e di supervisione del workflow vengono gestiti da appositi strumenti, gli **Administration and Monitoring Tools**. In particolare, gli **Operational Management Tool** si occupano della gestione del workflow, per quanto concerne l'aggiunta e la rimozione di risorse, oltre che la modifica delle informazioni riguardanti le risorse medesime. I **Recording and Reporting Tool**, invece, registrano e memorizzano dati inerenti l'esecuzione dei workflow, come il tempo medio di completamento di una istanza e la percentuale di istanze completate in un determinato tempo. Tali informazioni sono fondamentali per valutare i livelli di prestazione ed i colli di bottiglia di un determinato flusso.

Schematicamente, si può dunque considerare un sistema di Workflow Management System come composto di quattro elementi [23]:

1. **Designer**: consente la progettazione dei flussi di lavoro, con le attività e le risorse che li costituiscono;
2. **Server**: favorisce l'interazione dei flussi tra utenti e sedi diverse via web, fax, mail o sms;
3. **Worklist**: deputato alla gestione delle richieste e dell'interazione degli utenti con i flussi;
4. **Status**: per rilevare statistiche sui carichi di lavoro e sullo smistamento dei flussi di approvazione.

Inoltre, un Workflow Management System (WfMS) è definito attraverso tre aree funzionali³ [30] [31]:

1. **Build-time functions**: responsabili della definizione e della modellazione dei processi di workflow e delle attività che li costituiscono;
2. **Run-time control functions**: responsabili dei processi di workflow in un ambiente operativo e della sequenzializzazione delle attività gestite;

³ <http://www.wfmc.org/model.htm>

3. **Run-time interactions with Human Users and Information Technology:** tools applicativi che si interfacciano con l'utente e permettono di eseguire e gestire le attività.

La figura 2.3.1 schematizza le fasi relative alla progettazione di un workflow management system.

La realizzazione di un sistema di workflow ha inizio con lo studio del processo da definire e modellare. Una volta che il flusso di lavoro è stato modellato, la fase successiva definisce il suo comportamento evolutivo, rappresentandolo con soluzioni software che si occupano della creazione e del controllo delle istanze operazionali, ovvero di richieste di attivazione del flusso, tramite la schedulazione delle attività e l'invocazione di appropriate risorse umane e tecnologiche. L'interazione con tali risorse viene infine gestita nell'ambito della terza fase, che si occupa delle soluzioni specifiche per gestire il trasferimento del controllo, l'evoluzione dello stato di processi e attività, il trasferimento dei dati e le interfacce.

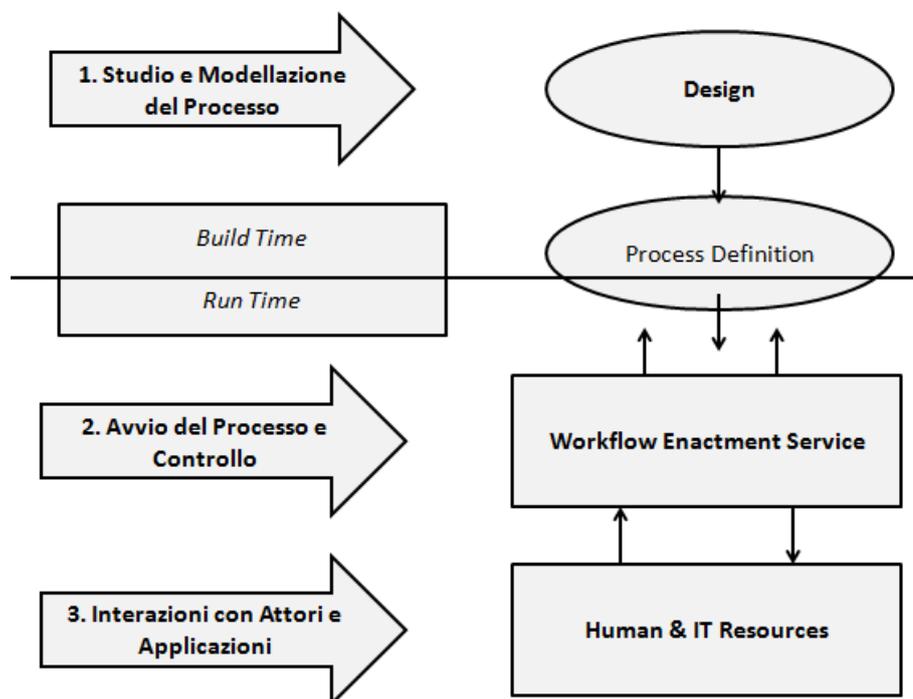


Fig.2.3.1: Fasi di progettazione di un Workflow Management System

Nel modello di Workflow sviluppato dalla Workflow Management Coalition⁴, esistono 5 interfacce ciascuna con caratteristiche diverse (WfMC-Hollingsworth). Queste sono mostrate nella figura sottostante (fig.2.3.2):

⁴ <http://www.wfmc.org/model.htm>

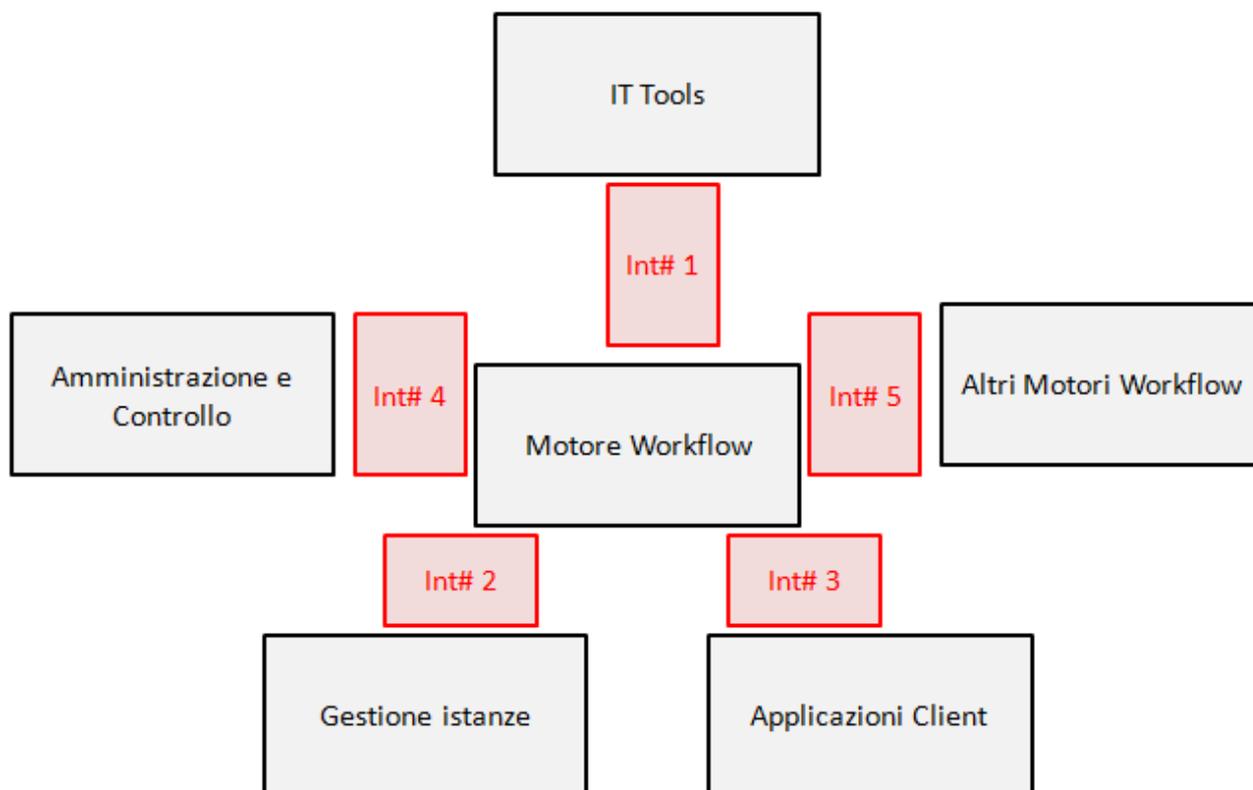


Fig.2.3.2: Interfacce del Motore di Workflow

L'interfaccia #1 è detta **Process Definition Tool Interface**. Essa fornisce uno standard per la definizione dei processi gestiti tramite il motore di Workflow.

L'interfaccia# 2, chiamata **Invoked Application Interface**, fornisce uno standard di accesso alle funzioni di Workflow e ai dati delle applicazioni integrate; consente, inoltre, l'accesso ai dati specifici delle applicazioni direttamente dal Workflow System.

L'interfaccia #3 è detta **Workflow Client Application Interface**: essa fornisce le funzionalità per il dialogo tra il motore di Workflow ed il client standard per la gestione delle attività proceduralizzate.

L'interfaccia #4, **Administration and Monitorig**, fornisce uno standard - secondo quanto codificato dall'Audit Data Specification del WfMC - di accesso ai dati dei processi di Workflow in tempo reale. I dati di tipo vengono infatti depositati in un apposito database di servizio.

L'interfaccia #5, denominata **Workflow Interoperability Interface between two different Workflow Enactmen Services**, fornisce uno standard di interscambio tra diversi motori di Workflow.

Come mostrato in fig 2.3.2, il supporto fisico allo sviluppo di un Workflow è garantito dai Workflow Enactment Services (WES), che consistono di uno o più motori capaci di creare, gestire ed eseguire le istanze di workflow.

Una volta modellato, il workflow può essere in stato di running – quando è stata accolta un'istanza e si stanno eseguendo una o più attività per soddisfarla – o suspended, quando non ci sono istanze per le quali sono state allocate le attività alle risorse.

Un workflow può essere costituito da attività singole - che non è utile decomporre ulteriormente - oppure da **sottoprocessi**, ovvero parti del flusso di lavoro che comprendono più attività e hanno specifici attributi in termini di obiettivo, input e output, contribuendo però nel contempo al raggiungimento dell'obiettivo più generale dell'intero flusso di lavoro.

Per quanto riguarda i dati memorizzati all'interno di un workflow management system, essi vengono classificati in 3 tipologie:

1. Workflow Control Data: interni e controllati dal WfMS;
2. Workflow Relevant Data: utilizzati dal WfMS per determinare le transazioni di stato delle istanze di processo;
3. Workflow Application Data: sono dati specifici di ogni istanza di processo, quindi non accessibili dal WfMS.

I dati gestiti internamente al sistema di workflow management, ovvero le singole e specifiche conoscenze, per essere effettivamente utilizzate devono essere archiviate nel sistema e contenere meta-informazioni che ne permettano il reperimento. Alla base di questo approccio c'è il concetto del cosiddetto **Knowledge Management**. Con il termine Knowledge management si intende la gestione della conoscenza aziendale [32]: esso include l'insieme di metodologie e strumenti tecnologici che consentono di raccogliere e capitalizzare la conoscenza esplicita ed implicita di tutte le persone che formano un'organizzazione, consentendone la formale creazione, la modifica e la distribuzione. In questa accezione rientrano: documenti, dati, fatti, giudizi, metodologie, algoritmi, know-how, best-practices.

2.4 Le Reti di Petri

Il successo di un sistema di workflow si basa sulla qualità dei modelli di flussi di lavoro che lo caratterizzano. Esistono al giorno d'oggi diversi strumenti per modellare ed analizzare i workflow; la maggior parte di questi è basata sulla teoria delle reti di Petri.

Una rete di Petri è un modello astratto e formale atto a rappresentare la dinamica di un sistema che esibisce attività asincrone e concorrenti [33].

Le reti di Petri sono state introdotte nel 1962 da Carl Adam Petri. Esse costituiscono uno strumento per la modellazione di processi ed in particolare per la modellazione delle comunicazioni e delle interazioni fra processi paralleli. Si tratta di un linguaggio grafico che presenta notevoli vantaggi, in quanto risulta semplice da utilizzare, ma, al contempo, prevede un alto rigore formale, permettendo così la definizione di processi in maniera chiara ed inequivocabile. Per questo motivo le reti di Petri prevedono l'utilizzo di definizioni precise eliminando di fatto ambiguità, incertezze e contraddizioni; inoltre, grazie alle forti basi matematiche, forniscono un formalismo che permette l'utilizzo di tecniche analitiche.

Mediante l'utilizzo delle reti di Petri viene superato un limite molto restrittivo derivante dall'utilizzo dei diagrammi di flusso. Questi ultimi, infatti, presentano una notevole serializzazione delle attività descritte: ad esempio, due rami distinti di un diagramma di flusso non possono essere mai percorsi contemporaneamente, rappresentando quindi due strade distinte: la scelta di un ramo piuttosto di un altro viene presa utilizzando un criterio rigidamente deterministico. Con le reti di Petri, invece, non solo è possibile rappresentare e descrivere globalmente un processo, ma vi è anche la possibilità di seguirne l'evoluzione, permettendo di visualizzare lo stato in cui si trova in un certo istante la rete in maniera dinamica. Esse costituiscono uno strumento espressivo che si presta alla modellazione ed analisi di sistemi concorrenti, dando al modello di un sistema il rigore formale necessario per eliminare ogni tipo di ambiguità al fine di effettuare analisi quantitative e qualitative oltre che verifiche sul comportamento del sistema.

Le reti di Petri (rdP) possono essere viste come un'estensione degli automi, in quanto possono rappresentare sistemi ad infiniti stati con un numero finito di nodi di un grafo, anche se sono strutture estremamente più compatte.

Esse sono impiegate essenzialmente come strumento ausiliario di analisi o come strumento diretto di progetto. Nel primo caso, si modella il sistema oggetto di studio con una rete di Petri e se ne analizza il comportamento per verificare la correttezza del progetto e la sua corrispondenza con il sistema. Nel secondo caso, l'intero processo di specifica e progetto viene svolto con reti di Petri.

Occorrono, quindi, tecniche di traduzione delle reti in sistemi adatti per un'implementazione diretta.

Oltre alla rappresentazione grafica, semplice e di agevole interpretazione, alla base matematica che fornisce la possibilità di applicare diverse tecniche di analisi formale grazie alle quali è possibile verificare le proprietà fondamentali di una rete e analizzarne il comportamento, bisogna considerare che ogni porzione di una rete di Petri rappresenta una specifica attività. E' pertanto possibile ottenere una rappresentazione modulare del processo: se esso è composto da più sottosistemi che interagiscono tra loro, è in pratica possibile rappresentare ciascun sottosistema come una semplice sottorete e poi, mediante operatori di rete, unire le varie sottoreti per ottenere il modello del processo complessivo. Inoltre, le reti di Petri permettono di rappresentare esplicitamente il concetto di concorrenza, cioè di attività che possono venire svolte parallelamente.

Date le proprietà sopra esposte, le reti di Petri si prestano in modo particolare per rappresentare comportamenti complessi come:

1. la sincronizzazione;
2. il succedersi asincrono di eventi;
3. operazioni concorrenti e parallele;
4. conflitti;
5. mutua esclusività e stocasticità;
6. ciclicità;
7. condivisione di risorse.

Nella pratica, una Rete di Petri è un grafo che consiste di **posti**, **transizioni** ed **archi** che li collegano [12]. Gli archi di input collegano i posti con le transizioni, mentre gli archi di output collegano le transizioni con i posti (fig.2.4.1).

Una rete P/T (posti/transizioni) è un grafo bipartito, orientato e pesato. I due tipi di vertici sono detti: posti (rappresentati da cerchi) e transizioni (rappresentati da barre o da rettangoli). Sono possibili solo collegamenti tra nodi diversi.

Si definisce *pre-set* di un nodo della rete di Petri l'insieme di nodi immediatamente a monte del nodo stesso, mentre il *post-set* è l'insieme di nodi immediatamente a valle [34]. Si indica il pre-set di x con $\bullet x$ e il post-set con $x\bullet$.

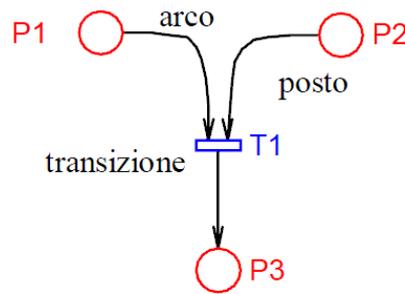


Fig.2.4.1: Rete posto/transizione

Al fine di descrivere l'evoluzione dinamica di una rete, bisogna introdurre i concetti di marcatura (marking) e peso: la marcatura è riferita al fatto che ogni posto contiene un numero intero non negativo di token (o gettoni), rappresentati con pallini neri. La funzione peso si riferisce al fatto che ad ogni arco è associato un numero intero non negativo, detto peso. Una rete di Petri in cui tutti i pesi degli archi sono unitari viene anche detta *ordinaria*. La marcatura di tutti i posti della rete definisce lo stato globale della rete, che evolve come conseguenza dello scatto delle transizioni associate agli eventi. L'evoluzione di una rete di Petri è determinata dal verificarsi di uno o più eventi che hanno la possibilità di accadere nel sistema [12].

Lo stato della rete indica una sua configurazione in un determinato istante dell'esecuzione del processo descritto dalla rete stessa. Lo stato viene rappresentato graficamente collocando dei token nei vertici: i token, quindi, indicano lo stato di avanzamento delle operazioni svolte nei vari step del processo. In altri termini possiamo dire che una rete di Petri evolve passando attraverso una serie di stati che rappresentano gli step del processo.

Come descritto da [35], elementi costitutivi di una rete di Petri sono i posti, le transizioni, gli archi, la marcatura, i pesi e le capacità, come rappresentato in fig.2.4.2.

I posti generalmente rappresentano le risorse o lo stato di un'operazione. Nel primo caso, la presenza dei token nei posti indica la disponibilità della risorsa in oggetto; il posto, anche detto *place*, può anche indicare una locazione geografica, un mezzo di comunicazione etc. Quando il posto rappresenta lo stato di un'operazione, la presenza del token nel posto indica che l'operazione è in corso.

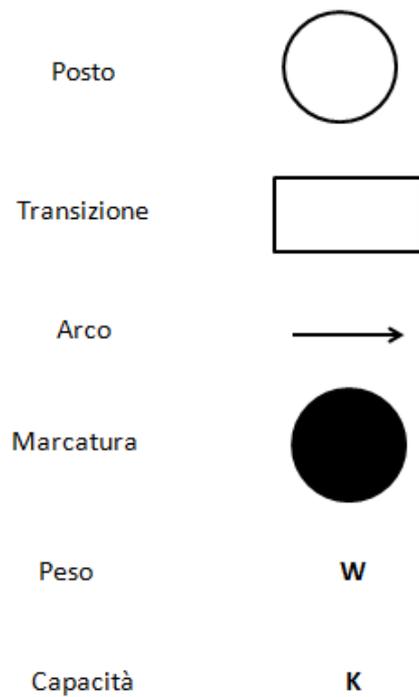


Fig.2.4.2: Elementi costitutivi di una rete di Petri

Le transizioni generalmente rappresentano attività, oppure inizio e/o fine di attività.

I token (gettoni) possono essere rappresentazioni dei seguenti elementi:

- un oggetto fisico, un prodotto, una persona;
- un'informazione, un messaggio, un segnale, un rapporto;
- una collezione di oggetti, un camion con dei prodotti, un magazzino, un file di indirizzi;
- un indicatore di stato, lo stato di un processo, lo stato di un oggetto etc.;
- un indicatore di una condizione: la presenza del gettone indica che una certa condizione è verificata.

Abbiamo definito una rete di Petri come un grafo bipartito, orientato e pesato. Un grafo è una struttura relazionale composta da un insieme finito di oggetti (sul piano più strettamente matematico un insieme finito di punti) detti *nodi* e da un insieme di relazioni (geometricamente segmenti di retta o di curva) tra coppie di oggetti detti *archi*. L'importanza che assume questa struttura logica è da attribuire alla constatazione che le strutture reali possono essere schematizzate utilizzando i grafi [36], da una rete stradale (dove i nodi sono gli incroci e gli archi le strade) ad un programma di calcolo (dove i nodi rappresentano le istruzioni ed esiste un arco tra due nodi se le relative istruzioni

possono essere eseguite in successione) oppure una struttura dati (dove i nodi rappresentano i dati semplici e gli archi i legami tra i diversi dati, realizzabili tramite puntatori).

In generale, gli archi rappresentano una relazione fra una coppia di nodi; se tale coppia è ordinata, cioè se gli archi hanno una testa o nodo di arrivo ed una coda o nodo di partenza, il grafo si dice *orientato*, dove (i, j) indica un arco diretto dal nodo i al nodo j .

In un grafo orientato, presa la coppia di nodi (i, j) , i è detto predecessore di j e j è detto successore di i . Nel caso in cui tutte le coppie di nodi presenti nel grafo non siano ordinate il grafo si dice *non orientato*. In un grafo orientato si hanno due tipi di cammino: il *cammino non orientato o a catena*, che non pone vincoli all'orientamento degli archi, o il *cammino semplice*, che richiede che la sequenza di archi sia tale che la testa di un arco coincida con la coda del successivo.

Un grafo tale per cui per ogni coppia di nodi esiste un cammino che li unisce è detto *grafo connesso*. Nel caso di grafi orientati, se esiste un cammino per ogni coppia di nodi si dice che il grafo è fortemente connesso, mentre se esiste solo un cammino non orientato fra ogni coppia di nodi il grafo si dice debolmente connesso.

Un grafo (orientato o non orientato) si dice *completo* se per ogni coppia di nodi esiste un arco che li congiunge.

Infine, si definisce *grafo completo pesato* un grafo tale per cui ogni nodo è collegato a tutti i rimanenti nodi con un arco e ad ogni arco è associato a un valore.

Oltre alla comoda rappresentazione grafica, le reti di Petri sono dotate anche di una rappresentazione matematica relativamente semplice. Tale rappresentazione, detta *matriciale* o *algebrica*, può essere utile per eseguire analisi automatiche della rete, al fine di verificare il soddisfacimento di alcune proprietà di base.

La rappresentazione algebrica delle reti di Petri si basa sui concetti di matrice d'ingresso, matrice d'uscita, matrice d'incidenza, vettore marcatura, sequenza di scatti e vettore delle occorrenze.

Inoltre, con tale rappresentazione si può descrivere l'evoluzione di una rete di Petri in termini di equazione di stato, cioè in modo molto simile a come si descrive l'evoluzione di un qualunque sistema dinamico [37].

Per dare una definizione matematica dei concetti appena elencati, si definisce una rete come una sestupla (S, T, F, W, K, M_0) , dove:

- S è l'insieme degli eventi (finiti) $S = \{S_i\}, i = 1, 2, \dots, |S| \quad |S| < \infty$
- T è l'insieme delle condizioni (finito e disgiunto da S)
 - o $T = \{T_k\}, k = 1, 2, \dots, |T| \quad |T| < \infty, \quad S \cap T = \emptyset$

- F è la relazione di flusso (indica quali coppie (s, t) o (t, s) sono connesse da un arco e dice quindi il verso dell'arco) $F \subseteq (S \times T) \cup (T \times S)$
 - W è la funzione peso (associa ad ogni arco un numero intero positivo) $W: F \rightarrow N - \{0\}$
 - K è la funzione capacità (associa una capacità ad ogni posto) $K: S \rightarrow N$
 - M_0 è la funzione marcatura iniziale (indica quanti token ci sono all'inizio di ogni posto) $M_0: S \rightarrow N$

Si definisce *matrice d'ingresso* I la matrice che ha S righe (una cioè per ogni posto della rete) e T colonne (una per ogni transizione della rete). Il suo elemento di posizione (k, j) vale quanto il peso dell'arco che va dal posto k alla transizione j se quest'arco c'è, altrimenti il suo valore è zero. La matrice I è dunque definita come $I_{|P|, |T|}$, con

$$I(k, j) = \begin{cases} W(s_k, t_j) & \forall (s_k, t_j) \in F; \\ 0 & \forall (s_k, t_j) \notin F. \end{cases}$$

Si definisce *matrice d'uscita* O la matrice che ha S righe (una cioè per ogni posto della rete) e T colonne (una per ogni transizione della rete). Il suo elemento di posizione (k, j) vale quanto il peso dell'arco che va dalla transizione k al posto j se quest'arco c'è, altrimenti vale zero. La matrice O è dunque definita come $O_{|P|, |T|}$, con

$$O(k, j) = \begin{cases} W(t_k, s_j) & \forall (t_k, s_j) \in F; \\ 0 & \forall (t_k, s_j) \notin F. \end{cases}$$

Si definisce *matrice d'incidenza* C la matrice che ha S righe (una cioè per ogni posto della rete) e T colonne (una per ogni transizione della rete). Essa è definita come $C = O - I$.

Data una rete con marcatura M , si definisce il *vettore marcatura* m come il vettore colonna di dimensione S le cui componenti sono valori interi non negativi che rappresentano il numero di marcature contenute in ogni posto della rete. Il vettore m è pertanto definito come

$$m = [m_1 m_2 \dots m_{|S|}]', \text{ con } m_i = M(s_i), i = 1, 2, \dots, |S|.$$

Una sequenza di scatti abilitata in una marcatura M_0 è una sequenza di transizioni $t_i, 1 \leq i \leq n, \in N - \{0\}$, tali che t_1 è abilitata in M_0 e lo scatto di t_i porta in una marcatura in cui t_{i+1} è abilitata. Il

vettore delle occorrenze s , associato ad una sequenza di scatti S , è un vettore colonna di dimensioni T , la cui componente generica s_i è pari al numero delle occorrenze della transizione t_i nella sequenza S .

Dalle definizioni e osservazioni precedenti, è possibile giungere ad una formulazione molto compatta dell'evoluzione di una rete di Petri, in termini del tutto simili a quella di un qualunque sistema dinamico. Ad una rappresentazione di questo tipo si dà il nome di *equazione di stato*, in quanto serve a calcolare lo "stato successivo" (cioè la marcatura) di una rete nota la marcatura precedente e l'evento (lo scatto di una transizione) avvenuto.

Si supponga infatti che M_0 sia la marcatura corrente di una data rete con matrice di incidenza C e si supponga inoltre che sia possibile applicare una certa sequenza di scatti S , con vettore delle occorrenze s . Sia M_1 la marcatura raggiunta dopo l'applicazione della sequenza S . Si può facilmente osservare che l'equazione:

$$M_1 = M_0 + C_s$$

,descrive l'evoluzione della rete in quanto definisce lo stato attuale (la marcatura M_1), noto lo stato iniziale (la marcatura iniziale M_0) e noti gli eventi avvenuti dall'inizio ad ora (la sequenza S , che ovviamente deve essere ammissibile). L'equazione ottenuta è un'equazione lineare ed ha una fortissima analogia con la legge di evoluzione dei sistemi dinamici.

Sulla base di quanto finora esposto, si possono definire cinque tipi di analisi diverse che possono essere applicate sulle Reti di Petri:

1. **Analisi enumerativa:** Metodologia basata sulla costruzione del grafo di raggiungibilità, di seguito definito. La maggior controindicazione in questo tipo di analisi è l'elevata complessità computazionale per la costruzione del grafico (state explosion problem [35]).
2. **Analisi per trasformazione:** metodologia basata sull'applicazione di regole che semplificano la struttura della rete di Petri senza alterarne le proprietà da analizzare; in questo modo si potrà analizzare una rete di Petri più semplice ma equivalente.
3. **Analisi strutturale:** Metodologia basata su tecniche di algebra lineare, che fanno uso della matrice di incidenza e dell'equazione di stato della rete di Petri, o su tecniche di analisi della struttura a grafo bipartito della rete di Petri. Esistono due tipologie di tecniche di analisi strutturale: le tecniche di *algebra lineare* e di *programmazione lineare*, basate sull'equazione di stato delle reti di Petri. Per certe tipologie di proprietà (boundedness, conservatività, reversibilità), questa prima classe di metodologie fornisce il risultato in un tempo minore rispetto alla tecnica di analisi enumerativa.

4. **Tecniche di analisi di grafi**, basate sulla considerazione della rete di Petri come grafo orientato bipartito; esse sono in genere applicabile solamente in relazione a specifiche classi di reti di Petri ordinarie.
5. **Analisi per simulazione**: Metodologia basata sull'esecuzione del token game. La simulazione viene soprattutto usata per comprendere il funzionamento del sistema rappresentato dalla rete di Petri.

Un altro concetto importante da definire quando si parla delle reti di Petri, è il grafo di raggiungibilità.

Il grafo di raggiungibilità di una rete di Petri (N, M_0) , è il grafo $R(N, M_0)$, in cui ciascun nodo rappresenta un marking raggiungibile da M_0 e ciascun arco rappresenta il firing (l'attivazione) di una transizione.

$$\mathcal{R}(N, M_0) = \{M_k : \exists \sigma : M_0 [\sigma > M_k]\}$$

Nel grafo di raggiungibilità esiste un arco etichettato da t_j che parte dal nodo che rappresenta il marking M_i e arriva nel nodo che rappresenta il marking M_h se e solo se M_h è raggiungibile da M_i attraverso il firing della transizione t_j , cioè se $M_i [t_j > M_h]$.

Il grafo di raggiungibilità può essere finito o infinito.

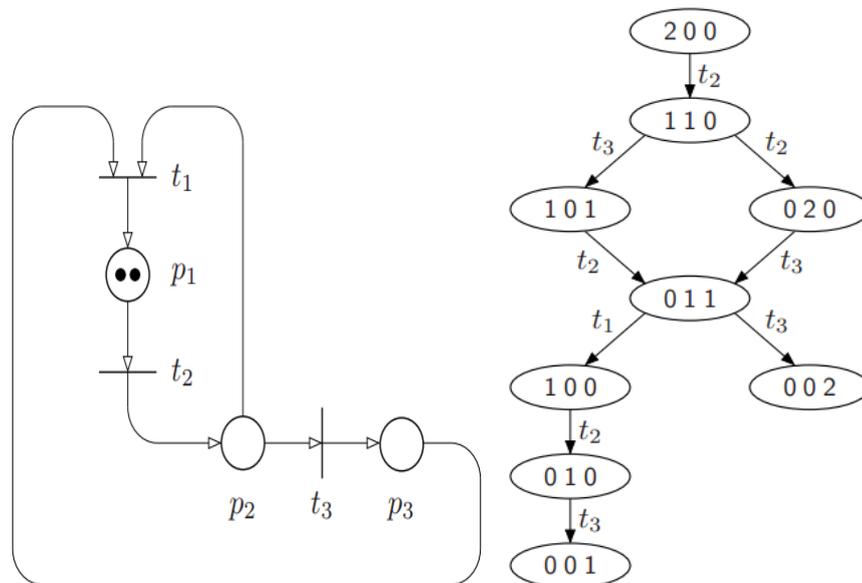


Fig.2.4.2: Esempio di rete di Petri con il corrispondente grafo di raggiungibilità

La Procedura di costruzione di una rete di Petri si basa su una serie di step, di seguito elencati [38]

1. **Identificazione di tutte le operazioni e di tutte le risorse del sistema:** a partire dalla descrizione non formalizzata del sistema, si devono individuare tutti gli eventi, operazioni, processi, risorse, condizioni, percorsi di trasporto, ecc., che caratterizzano il sistema.
2. **Identificazione di tutte le relazioni:** tutte le connessioni tra operazioni e processi devono essere prese in considerazione, insieme alle politiche di routing nel caso di operazioni eseguibili da più risorse; le risorse devono essere distinte in risorse condivise e non condivise e, nel caso di risorse condivise, devono essere definite le politiche di allocazione delle stesse sulle varie operazioni.
3. **Modellazione della rete di Petri:** vengono definiti i posti e le transizioni che rappresentano gli eventi, le operazioni, i processi, ecc. definiti al punto 1; i posti e le transizioni vengono posizionati secondo le relazioni identificate al punto 2; vengono identificati lo stato iniziale del sistema e di conseguenza viene il numero corretto di token in ogni posto. In presenza di transizioni temporizzate è necessario associare, sulla base delle caratteristiche del sistema, l'opportuna temporizzazione delle stesse.
4. **Modifica della rete di Petri:** l'ultimo step consiste nella verifica, tramite l'esecuzione del token game, del corretto funzionamento della rete di Petri. Se il comportamento della rete è coerente con le caratteristiche del sistema rappresentato la procedura si conclude, altrimenti si modifica la rete fino al raggiungimento del comportamento desiderato.

Le Proprietà di una rete di Petri [12] possono essere sintetizzate come di seguito:

1. **Raggiungibilità (reachability):** una marcatura M^* si dice raggiungibile a partire da una data marcatura M se esiste almeno una sequenza di transizioni tali che attivaandole a partire da M si ottenga M^* . Questa proprietà è importante in quanto ci dice che è possibile ottenere una data marcatura (cioè un determinato stato) a partire da un'altra.
2. **Ripetitività (repetitivity):** la ripetitività di una sequenza di transizioni assicura che essa possa essere ripetuta infinite volte.
3. **Limitatezza (boundedness):** poiché il numero di gettoni indica anche il numero di variabili, è importante accertare che il numero di quest'ultimi, che può venirsi a trovare in un determinato posto a seguito dell'evoluzione della rete, non superi una soglia limite.
4. **Binarietà o sicurezza (safeness):** in una rete binaria ogni posto può contenere al più un token.

5. **Conservatività (conservativeness)**: poichè i gettoni vengono usati per rappresentare le risorse, è importante che si conservino. Inoltre si può sottolineare che la conservatività strutturale implica la proprietà di limitatezza strutturale, ma non è vero il contrario.
6. **Vivezza (liveness)**: una rete si dice viva se e solo se tutte le sue transizioni sono vive. Se una transizione t è viva, vuol dire che esiste una sequenza di transizioni che porta da M_0 ad M^* in cui essa è abilitata. In una rete viva tutte le transizioni possono scattare infinite volte.

Le proprietà delle reti di Petri che hanno un'importanza particolare nell'analisi e nella gestione dei processi discreti sono la boundedness, safeness, la liveness e la reversibilità. La boundedness e la safeness implicano l'assenza di overflow: in particolare, per quanto riguarda i posti che rappresentano, garantiscono che il numero di istanze in attesa non aumenti all'infinito. La liveness implica l'assenza di deadlock: questa proprietà garantisce quindi che l'esecuzione possa proseguire in maniera corretta e che tutti i processi possano avere luogo. La reversibilità implica un comportamento ciclico del sistema: questa proprietà è molto importante nei sistemi di produzione discreta in quanto è alla base del cosiddetto 'error recovery': se il sistema ha comportamenti indesiderati, ed è reversibile, allora è possibile ritornare allo stato iniziale (che si suppone essere un buono stato) e ricominciare il processo come pianificato.

Un evento è caratterizzato da due attributi:

- la possibilità che esso accada → *abilitazione di una transizione*
- l'effetto del suo accadimento sul sistema → *scatto di una transizione*

Per ogni transizione si identificano i posti di ingresso (input places) e i posti di uscita (output places) [11].

Si ha l'abilitazione di una transizione se si verificano alcune condizioni che dipendono dal peso degli archi in ingresso alla transizione e dalla marcatura dei posti in ingresso alla transizione. In particolare l'abilitazione di una transizione dipende dalla marcatura corrente (che rappresenta lo stato della rete).

Sinteticamente possiamo dire che una transizione t è abilitata in una certa marcatura M se ogni posto a monte di t ha un numero di gettoni almeno pari al peso dell'arco che lo collega a t .

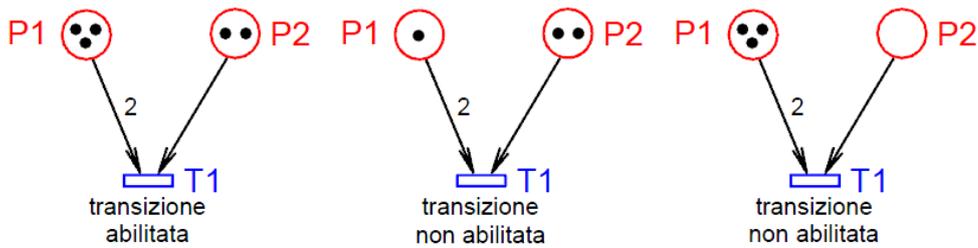


Fig.2.4.3: Esempio di transizioni abilitate e non abilitate

Lo scatto di una transizione abilitata determina il ‘consumo’ di token nei posti a monte e la generazione di token nei posti a valle, determinando così una modifica nella marcatura della rete. Tuttavia, è più corretto pensare che i gettoni tolti dai posti di pre-set scompaiano e che si creino gettoni nei posti di post-set. E’ possibile che più transizioni abilitate si attivino contemporaneamente purché non siano in conflitto tra loro.

In generale il numero di gettoni eliminati può essere diverso dal numero di quelli creati. Lo scatto di una rete è un fenomeno locale ed indivisibile. La regola di scatto non è sufficiente a determinare compiutamente l’evoluzione di una rete perché, in una generica marcatura, può esserci una situazione di conflitto, ovvero può accadere che più transizioni risultino abilitate allo scatto, e, chiaramente, se si sceglie di farne scattare una piuttosto che un’altra l’evoluzione futura della rete non è la stessa. Va sottolineato inoltre che il criterio di scelta di una transizione è un criterio del tutto non deterministico. Il criterio garantisce il rispetto della località dell’evoluzione del sistema, vale a dire l’indipendenza degli eventi. Una volta che una transizione abilitata scatta, per decidere quale sarà la futura transizione abilitata a scattare si deve attuare una nuova valutazione della rete, in quanto la marcatura creata dallo scatto della precedente transizione può aver abilitato nuove transizioni e aver disabilitato alcune di quelle abilitate in precedenza.

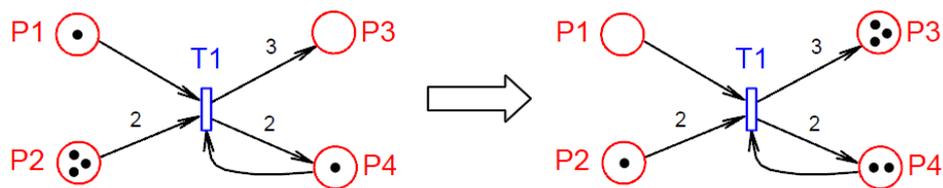


Fig.2.4.4: Esempio di scatto di una transizione

Esistono diversi tipi di transizioni, di seguito illustrate:

1. Transizioni in sequenza:

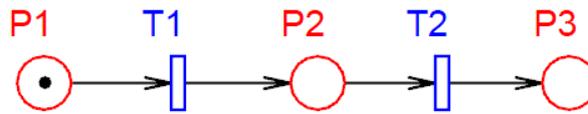


Fig.2.4.5: Transizione in sequenza

Due transizioni T_1 e T_2 si dicono in sequenza e T_1 precede T_2 in una data marcatura.

2. Transizioni in conflitto:

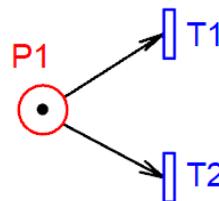


Fig.2.4.5: Transizione in conflitto

Due transizioni T_1 e T_2 si dicono in conflitto strutturale se hanno almeno un posto d'ingresso in comune. Due transizioni in conflitto strutturale, sono in conflitto effettivo in una marcatura M se sono entrambe abilitate, ma lo scatto di una, disabilita l'altra [39]

3. Transizioni in concorrenza strutturale

Due transizioni T_1 e T_2 si dicono in concorrenza strutturale se non condividono nessun posto d'ingresso. Due transizioni in concorrenza strutturale sono in concorrenza effettiva in una marcatura M se sono entrambe abilitate, vale a dire che lo scatto di una non disabilita l'altra.

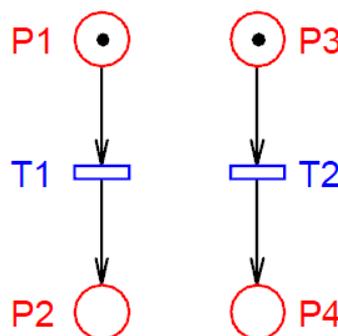


Fig.2.4.6: Transizione in concorrenza strutturale

4. Transizioni in sincronizzazione e inizio concorrenza

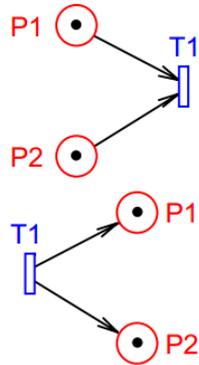


Fig.2.4.7: Transizione in sincronizzazione ed in inizio concorrenza

Questo tipo di struttura è legato al concetto di concorrenza: due sequenze concorrenti sono attivate dallo scatto di una transizione del secondo tipo e la fase di concorrenza termina con lo scatto di una transizione del primo tipo [40].

Qualora ci si appresti a modellare un flusso di lavoro mediante le reti di Petri, bisogna tenere presente che esistono tipi di Reti di Petri differenti, il cui comportamento deve essere interpretato evidenziando alcune importanti differenze [41]:

1. **Reti condizioni/eventi:** Con questo tipo di reti vengono rappresentati sistemi e processi in cui gli eventi si realizzano solo quando tutte le condizioni che li precedono direttamente sono soddisfatte. La realizzazione di un evento porta a soddisfare delle condizioni successive facendo in modo che la dinamica della rete proceda nel suo iter.
2. **Reti posti/transizioni:** rispetto alle reti condizioni/eventi, le reti posti/transizioni ammettono la presenza di più di un token nei posti ed anche il numero di token che transitano sugli archi della rete è stabilito sulla base di una capacità non negativa assegnata agli archi stessi; inoltre è possibile assegnare una capacità o un peso intero non negativo agli archi della rete: in tal caso ad ogni passo della rete, lungo un arco, passano non uno, ma un numero di token pari alla capacità dell'arco.
3. **Macchina a stati finiti e grafi marcati:** una macchina a stati finiti è un caso particolare di una rete di Petri P/T. In questa rete ogni transizione è limitata ad avere un solo posto in ingresso ed un solo posto in uscita. Una macchina a stati può generare conflitti, ma non può

modellare la creazione e la distruzione di gettoni necessaria per rappresentare la concorrenza e la sincronizzazione.

4. **Grafo marcato:** un grafo marcato è una rete di Petri ordinaria in cui ogni posto è limitato ad avere una sola transizione di ingresso ed una sola transizione di uscita. Un grafo marcato può modellare la concorrenza e la sincronizzazione, ma non può rappresentare conflitti.
5. **Rete a libera scelta (free choice):** una rete a scelta libera è una rete di Petri tale che, per ogni arco da un posto ad una transizione, o quel particolare posto è l'unico in ingresso alla transizione (quindi non c'è sincronizzazione), oppure quella transizione è l'unica transizione in uscita da quel posto (quindi non ci sono conflitti). Mediante questa tipologia di rete è pertanto possibile rappresentare sia conflitti che concorrenza, con alcune limitazioni. In particolare, si vuole evitare che delle transizioni in conflitto tra loro possano non essere simultaneamente abilitate (per effetto di posti a monte che si svuotano).

Le macchine a stati e grafi marcati sono reti a scelta libera, ma non vale il viceversa.

6. **Rete a scelta estesa (extended free choice Petri net):** in tale tipologia di rete, se due posti hanno una transizione di uscita in comune, allora essi hanno le stesse transizioni d'uscita. Una rete a scelta libera è anche una rete a scelta libera estesa ma non vale il viceversa.
7. **Rete a scelta libera asimmetrica (asymmetric free choice Petri net):** è una rete di Petri tale che se due posti hanno delle transizioni di uscita in comune, l'insieme di transizioni di uscita di un posto contiene quello dell'altro. Una rete a scelta libera estesa è anche una rete a scelta libera asimmetrica ma non vale il viceversa.

Le reti P/T (posti/transizioni) hanno dei limiti [42], per superare i quali nel corso del tempo sono state proposte delle estensioni.

Uno dei principali problemi rilevati, è quello riguardante il cosiddetto problema del 'test a zero', ovvero la possibilità di abilitare una transizione in corrispondenza all'assenza di token in un dato posto di ingresso: solo se la rete è limitata, infatti, il test a zero è possibile tramite l'introduzione di posti complementari.

Tra le possibili estensioni definite per una rete di Petri, in accordo ad esempio a [43], oppure a [44] evidenziamo:

1. **Reti con archi inibitori:** consentono di condizionare l'abilitazione di una transizione all'assenza di token nella transizione precedente.
2. **Reti colorate:** in una rete colorata non c'è un unico tipo di token: ci sono token di diversi "colori", cioè di diversi tipi. I token colorati intervengono nell'abilitazione delle transizioni,

in quanto gli archi sono etichettati: per far scattare una transizione occorrono necessariamente token di un certo tipo.

3. **Reti con priorità:** in tali reti, la scelta di quale fra le transizioni abilitate debba scattare è deterministica e precisamente basata sulla priorità di tali transizioni. E' sempre possibile trasformare una rete con priorità in una rete con archi inibitori e viceversa.

Le Reti di Petri classiche non modellano il tempo, danno solo un ordinamento temporale parziale fra gli eventi. Le reti di Petri temporizzate, introdotte come estensione, si dividono in deterministiche e stocastiche.

4. **Reti temporizzate deterministiche:** i valori o gli intervalli temporali sono definiti in modo deterministico: esempi sono le Time Petri Nets [45] e le Timed Petri Nets [46]. In una Time Petri Nets ogni transizione è etichettata con due istanti di tempo t_{min}, t_{max} . Per poter scattare, una transizione deve rimanere abilitata almeno per il tempo t_{min} ; se al tempo t_{max} essa è ancora abilitata, deve scattare. In una timed Petri net ogni transizione ha una durata t ; la transizione deve scattare immediatamente appena è abilitata, e lo scatto non è istantaneo, ma dura il tempo t . Si tratta di un modello utile per effettuare valutazioni di prestazioni di sistemi ciclici, in quanto fornisce la possibilità di calcolare il tempo minimo di ciclo; per reti limitate, consente la validazione delle specifiche e i check di vitalità.
5. **Reti temporizzate stocastiche:** in tale tipo di rete, i valori o gli intervalli temporali sono definiti in modo probabilistico. Rientrano in questa categoria le Stochastic Petri Nets [47] e le Generalized Stochastic Petri Nets [48]. Nelle Stochastic Petri Nets ogni transizione ha un tempo di scatto non prefissato, ma distribuito esponenzialmente: ad ogni transizione è associata una variabile aleatoria a valori reali non negativi, che esprime il tempo che intercorre fra l'istante di abilitazione della transizione e l'istante di scatto; se più transizioni sono abilitate, scatta quella con tempo di scatto minore. Con le Generalized Stochastic Petri Nets le transizioni sono immediate o temporizzate. Le transizioni immediate modellano quelle attività con tempi di esecuzione trascurabili rispetto agli altri.
6. **Interval timed coloured Petri net (ITCPN):** la caratteristica principale di tale tipo di reti è che il tempo è associato direttamente ai token presenti nella rete. Si crea un timestamp che indica l'istante di tempo in cui il token è utilizzabile. C'è una differenza tra il tempo di esecuzione (firing) e il tempo di timestamp: quest'ultimo rappresenta infatti un ritardo di esecuzione, definito da un intervallo specifico [49].

Riepilogando quanto descritto finora, la temporizzazione è dunque realizzata associando ad ogni transizione un ritardo che corrisponde al tempo che deve trascorrere fra la sua abilitazione e il

successivo scatto. In riferimento all'abilitazione della transizione, si possono verificare due condizioni: la prima è quella in cui la transizione rappresenta un evento istantaneo, pertanto la temporizzazione consiste nell'associare un intervallo di tempo alla transizione indicando in questo modo che l'evento deve avvenire non prima e non dopo dei margini di tempo predefiniti. La seconda situazione è quella in cui la transizione è un evento non istantaneo, che quindi richiede un certo tempo per verificarsi. I token vengono rimossi dai posti delle pre-condizioni nel momento in cui la transizione risulta abilitata e vengono collocati nelle post-condizioni solo allo scadere del tempo richiesto dalla transizione per essere completata.

La variabile tempo viene introdotta all'interno delle reti di Petri con l'obiettivo principale di poter studiare le *performance* di un sistema [50], ovvero per studiare e valutare l'analisi quantitativa delle prestazioni.

Le reti di Petri temporizzate presentano una serie di vantaggi, in quanto consentono di modellare la stocasticità intrinseca dei sistemi reali, possono essere utilizzate sia come modelli logici che prestazionali, sono in grado di riprodurre le caratteristiche proprie dei sistemi ad eventi discreti quali priorità e sincronizzazione, si realizzano mediante strumenti informatici dedicati.

Le reti di Petri possono essere utilizzate nelle applicazioni di Workflow Management [51]. Esistono infatti, come illustrato nel capitolo successivo, diversi tool che consentono la modellazione e l'analisi dei flussi di lavoro secondo tecniche basate sulla teoria di Petri.

3. STUDIO DEI FLUSSI SANITARI MEDIANTE WORKFLOW

3.1 Il Simulatore Woped

Le reti di Petri possono essere utilizzate nelle applicazioni di Workflow Management [51].

Numerosi tool consentono al giorno d'oggi di modellare ed analizzare i processi mediante le reti di Petri; il sito⁵ fornisce una interessante overview di quelli ad oggi più diffusi. E' importante sottolineare che non tutti i tool elencati in⁶ consentono la modellazione di un flusso di lavoro con la relativa analisi di performance, ma molti si fermano al design, con scopi prevalentemente didattici di descrizione dei processi.

Nel presente lavoro di ricerca diversi tool basati sulle reti di Petri sono stati esaminati per individuare quale di essi si prestasse maggiormente all'analisi dei processi sanitari. L'esigenza era individuare uno strumento che consentisse la definizione delle attività caratterizzanti i flussi di lavoro, – con le relative tempistiche e le risorse deputate alla loro esecuzione - e che permettesse di simulare il comportamento del flusso mettendone alla luce caratteristiche di performance.

Dopo aver effettuato alcune prove di rappresentazione di flussi semplici mediante i tool disponibili, si è deciso di utilizzare, per analizzare alcuni flussi di interesse, il *Workflow Petrinet Designer Woped*.

Woped⁷ è un software open-source, sviluppato dalla University of Cooperative Education Karlsruhe in Germania. Esso usa il linguaggio delle Petri Net Markup Language, detto PNML [52], permettendo di creare, modificare, simulare e analizzare reti di Petri P/T. Il suo primo sviluppo risale al 2003, mentre la prima pubblicazione, ad opera di Freytag [53] è del Marzo 2005. La caratteristica principale di WoPeD è la semplicità, l'interazione e l'immediatezza visiva che consentono al sistema di essere un ottimo strumento per l'insegnamento, la ricerca e per un facile e-learning ed e-publishing [54]. WoPeD ha principalmente scopi didattici, sostenendo ricercatori, docenti e studenti sul tema delle reti di Petri e sulla gestione dei flussi di lavoro. E' strettamente legato alla notazione proposta da Van der Aalst [51] [11] che permette di visualizzare sia la struttura che la dinamica delle reti di flusso, permettendo in questo modo di entrare approfondirne i principi di base, i concetti chiave, le notazioni, le proprietà.

⁵ lavorohttp://www.informatik.uni-hamburg.de/TGI/PetriNets/tools/quick.html

⁶ lavorohttp://www.informatik.uni-hamburg.de/TGI/PetriNets/tools/quick.html

⁷ http://woped.dhbw-karlsruhe.de/woped/

L'architettura di WoPeD si basa su un Model-View-Controller (MVC), ovvero su un modello architetturale vista/controllo in cui si distinguono tre ruoli chiave:

1. **Model**, che fornisce i metodi per accedere ai dati utili all'applicazione;
2. **View**, che si occupa dell'interazione con gli utenti, oltre alla visualizzazione dei dati;
3. **Controller**, che riceve i comandi dall'utente e li attua, modificando così lo stato degli altri due componenti.

Il modello dei dati è rappresentato dallo standard PNML. Il Petri Net Markup Language, mediante le sue proprietà di leggibilità, universalità e modulabilità, riesce a rispondere ad una delle esigenze più importanti degli strumenti che utilizzano reti di Petri, vale a dire la possibilità di importare e/o esportare flussi realizzati con reti di Petri provenienti da tool differenti.

Lo standard PNML definisce un grafo etichettato avente due tipi di nodi: posti e transizioni; oltre a questi troviamo oggetti, etichette (sottoforma di annotazioni o attributi), archi, pagine, riferimenti ai nodi, che concretizzano i concetti esposti nel paragrafo precedente.

WoPeD memorizza dunque le reti come PNML, un formato basato sull'XML, che può essere condiviso con molti altri strumenti che utilizzano la teoria delle reti di Petri.

Una delle caratteristiche di WoPeD è la semplicità: esso, infatti, non solo permette l'utilizzo della classica notazione delle reti di Petri P/T, ma offre la possibilità di utilizzare anche supporti provenienti dalla teoria classica del workflow, come gli operatori **Xor split/join**, **And split/join**, **trigger** e **assegnazione delle risorse**; inoltre permette di creare una gerarchia dei processi interni al flusso realizzato, mediante la creazione di **sottoprocessi**. In questo modo, anche le reti più complesse, possono essere rappresentate graficamente e analizzate in dettaglio.

L'interfaccia utente di WoPeD presenta le seguenti funzionalità: Menubar, Toolbar, Drawing area e Status bar, come mostrato in figura 3.1.1.

La finestra principale della barra Menubar presenta le seguenti voci di menu:

- **File**: permette di aprire un nuovo editor, di salvare, cancellare, esportare o caricare un file stampare, chiudere il programma.
- **Edit**: permette di copiare, tagliare e incollare gli elementi presenti nella rete.
- **Analyze**: consente di avviare le simulazioni, dal token game alle analisi quantitative, di capacità e di semantica, nonché di visualizzare il reachability graph.
- **View**: consente di effettuare lo zoom (in/out); di raggruppare gli elementi; di linearizzare e modificare l'orientamento del modello creato.
- **Options & Help**: consente di accedere alla configurazione e modificare parametri come lingua, colori, etc. e di consultare il manuale d'uso con esempi di reti.

- **Community:** facilita lo scambio di informazioni attraverso il collegamento ai social network.

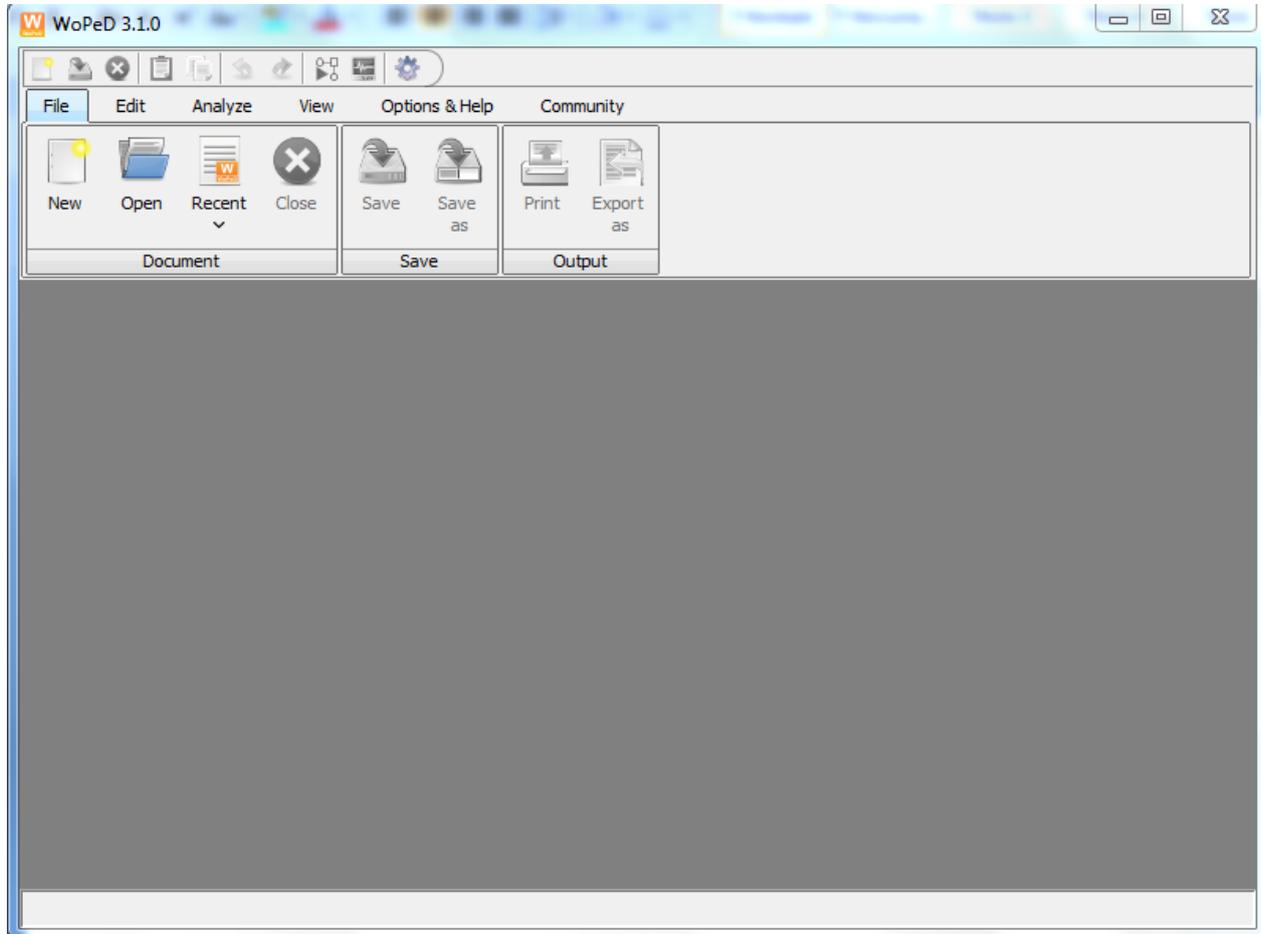


Fig.3.1.1: Interfaccia Utente di Woped

Il Toolbar consente di eseguire materialmente operazioni quali: l'apertura di un nuovo editor, zoom, linearizzazione del modello, inserimento di elementi all'interno della rete, etc.



Fig.3.1.2: Woped Toolbar

Per quanto riguarda la Drawing Area, WoPeD offre la possibilità di avere una schermata multi-documento, cioè consente di visualizzare contemporaneamente un numero arbitrario di finestre dell'editor, ciascuna delle quali è in grado di contenere un particolare modello di rete di Petri.

La barra di stato *status bar* della finestra principale mostra una barra delle applicazioni per ogni editor. Ogni finestra di editor aperta ha una propria barra che mostra alcuni dati statistici sul numero di elementi della rete, la direzione corrente del modello, il fattore di zoom e un'icona che indica se la rete è stata o meno salvata.

Fondamentali per il funzionamento del tool Wooped, sono gli elementi di sintassi.

Per poter disegnare una rete con WoPeD, si utilizzano i tradizionali elementi della teoria delle reti di Petri, vale a dire posti, transizioni, archi e token. Ulteriori elementi risultano poi fondamentali per una corretta implementazione e analisi della rete: le transition AND-split, XOR-split, AND-join, e XOR-join, i sottoprocessi, i trigger ed il resource mapping.

La **AND-split** è una transizione derivata dalle teoria del workflow che esplica il concetto del parallelismo. Essa presenta un solo ingresso e due o più uscite ed il suo comportamento è il seguente: se c'è un token in ingresso, quando scatta la transizione, ogni posto di uscita avrà un token. Allo stesso modo la **AND-join** concretizza il concetto di parallelismo, infatti è utilizzata per chiudere rami paralleli. Presenta due o più ingressi e una sola uscita. Se più token sono presenti nei posti d'ingresso, scattando la transizione, si ritorna ad avere un solo token in uscita.

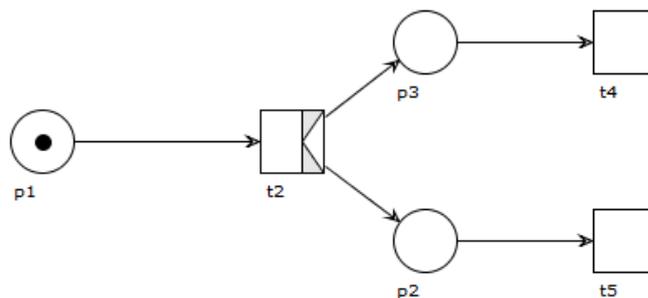


Fig.3.1.3: AND-Split

Le **XOR**, split e join, sono transizioni che vengono utilizzate nel momento in cui si deve compiere una scelta, o quando c'è la presenza e la necessità di una interazione. In particolare la XOR-split presenta un posto in ingresso e due o più uscite. Può essere il punto di partenza o il punto finale di un processo che prevede alternative. Se vi è almeno un token su un posto di ingresso, il numero di token viene diminuito di un'unità e un gettone si sposta quindi nel posto di uscita selezionato.

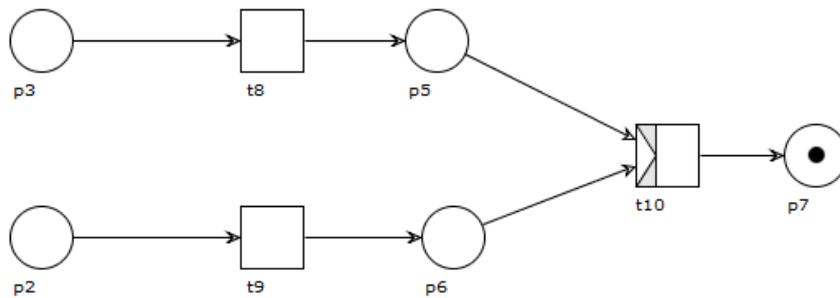


Fig.3.1.3: AND-join

La transizione **XOR-join** viene utilizzata quando si deve chiudere un percorso che presenta più alternative. Questa ha due o più posti di ingresso e uno solo di uscita. Se vi è almeno un gettone in uno dei posti in ingresso, questo viene rimosso e portato nel posto di uscita.

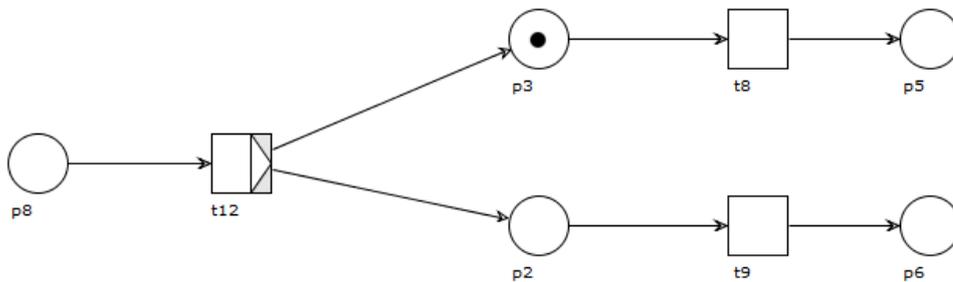


Fig.3.1.3: XOR-split

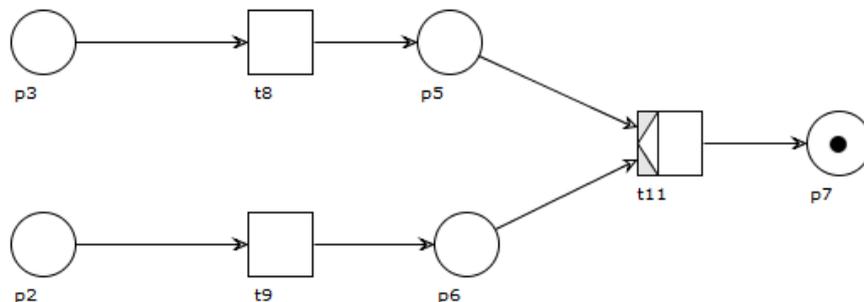


Fig.3.1.4: XOR-join

Le transizioni descritte possono essere definite come transizioni *elementari*, in quanto possono essere ulteriormente combinate per creare task complesse, come XOR-split-join transitions, AND-

split-joins, XOR-join-AND-split, AND-join-XOR-split. Tali transizioni vengono utilizzate nei processi che prevedono interazioni, alternative e scelte. La AND-split-join transizione fa sì che, se vi è almeno un token in ogni posto in ingresso, allo scattare della transizione si abbia un decremento di un'unità nei posti di ingresso e un incremento dello stesso ordine nei posti in uscita. Con la XOR-split-join viene selezionato un posto in ingresso che perde un token, e si sceglie poi quale dei posti di uscita debba essere incrementato di un'unità. Con la XOR-join-AND-split si sceglie da quale dei posti di ingresso prelevare il gettone, mentre i due o più posti in uscita incrementeranno di un'unità. La And-join-XOR-split prevede che vi sia almeno un token in uno o più posti di ingresso che andranno a diminuire di un'unità: il token tolto in ingresso sarà indirizzato e il nel posto di uscita selezionato.

Qualora si debbano modellare e gestire reti ampie ed articolate, si può ricorrere all'impiego dei **sottoprocessi**. All'interno del processo globale viene in pratica creata una rete ulteriore, con posti, transizioni e archi; anche il sottoprocesso presenta un ingresso ed un'uscita. Viene utilizzato per creare una gerarchia all'interno del flusso generale ed è, in molti casi, l'unico modo che si ha per poter analizzare nel dettaglio reti molto complesse. All'interno di un sottoprocesso, vi possono poi generare ulteriori sottoprocessi. In questo modo è possibile realizzare una sorta di gerarchia seguendo due possibili approcci: l'approccio top-down o l'approccio bottom-up [44]. L'approccio top down parte dal livello più alto, ovvero il modello del disegno globale, fino a giungere ai livelli più bassi dove ci sono solo posti e transizioni; il bottom up, invece, si sviluppa in senso opposto, partendo dal livello più basso (posti e transizioni) andando a salire(fino ad ottenere il modello globale. Un altro grande vantaggio nell'utilizzo di sottoprocessi sta nel fatto che si ha la possibilità di riutilizzare processi già precedentemente definiti. In particolare, se un sottoprocesso ricorre più volte, basterà definirlo una volta sola.

E' possibile creare i sottoprocessi dal menu Edit di Woped, come mostrato in figura 3.1.5, in cui sono rappresentate anche le icone delle transizioni illustrate precedentemente.

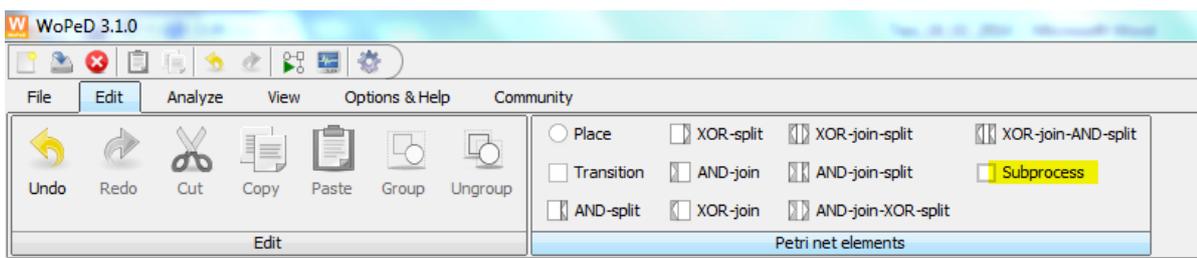


Fig.3.1.5: Sottoprocesso

All'origine di ogni transizione, vi è generalmente un **trigger**. Il ruolo e l'utilizzo dei trigger è molto importante, in quanto essi non modificano il workflow, ma agiscono sul loro ambiente di esecuzione. La traduzione letterale di trigger, ovvero 'innesco', rende bene l'idea della funzione: quando infatti si associa un trigger ad una transizione, significa che quest'ultima per attivarsi necessita di un evento specifico, solitamente un evento esterno rispetto allo scenario di riferimento.

WoPeD consente di rappresentare quattro tipologie di trigger [53]:

1. **Time trigger:** il simbolo usato per indicare il time trigger è un clock. Lo scatto della transizione così indicata dipende dal raggiungimento o meno di un determinato istante di tempo precedentemente fissato.
2. **Message trigger:** la transizione è abilitata solo dopo l'arrivo di un messaggio che è indipendente dal processo modellato e al quale la transizione fa riferimento. La transizione dipenderà, quindi, da un evento esterno. Il suo simbolo è una busta.
3. **Resource trigger:** l'abilitazione o meno della transizione dipende dalla disponibilità, in quel momento, di una o più risorse precedentemente allocate, ovvero assegnate all'esecuzione di una particolare attività. Il simbolo usato è quello di una freccia rivolta verso il basso.
4. **No trigger:** può essere assegnato a tutte le transizioni presenti nella rete, qualora non ci siano particolari vincoli che impediscono alla transizione di scattare, naturalmente sempre tenendo conto delle normali regole della teoria di Petri (ad esempio la necessità di avere almeno un gettone in una postazione di ingresso).

In Fig.3.1.6. vengono rappresentati rispettivamente il message trigger, il resource trigger ed il time trigger.

In molti casi l'esecuzione di una transizione dipende dalla disponibilità di risorse specifiche o di risorse presenti all'interno di un gruppo. WoPeD consente l'assegnazione di un ruolo e di un gruppo a ciascuna risorsa; inoltre, a seconda delle esigenze, è possibile definire un *compound roles* ed un *compound groups*, vale a dire ruoli e gruppi 'principali' ossia che contengono ad esempio due o più ruoli precedentemente definiti.

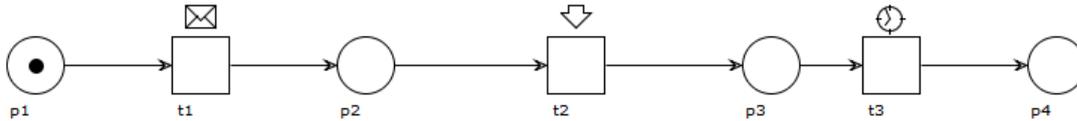


Fig.3.1.6: Rappresentazione dei trigger in Woped

L'interfaccia **Resource Mapping** è fondamentale quando nel progetto di una rete vi sono molte task che, per essere eseguite, necessitano dell'intervento di specifiche risorse, umane e non. Inoltre, come già accennato, si ha la possibilità di raggruppare più risorse e di definire ruoli selettivi e specifici. Questa mappatura delle risorse viene gestita all'interno di un'apposita finestra di dialogo, rappresentata in figura seguente:

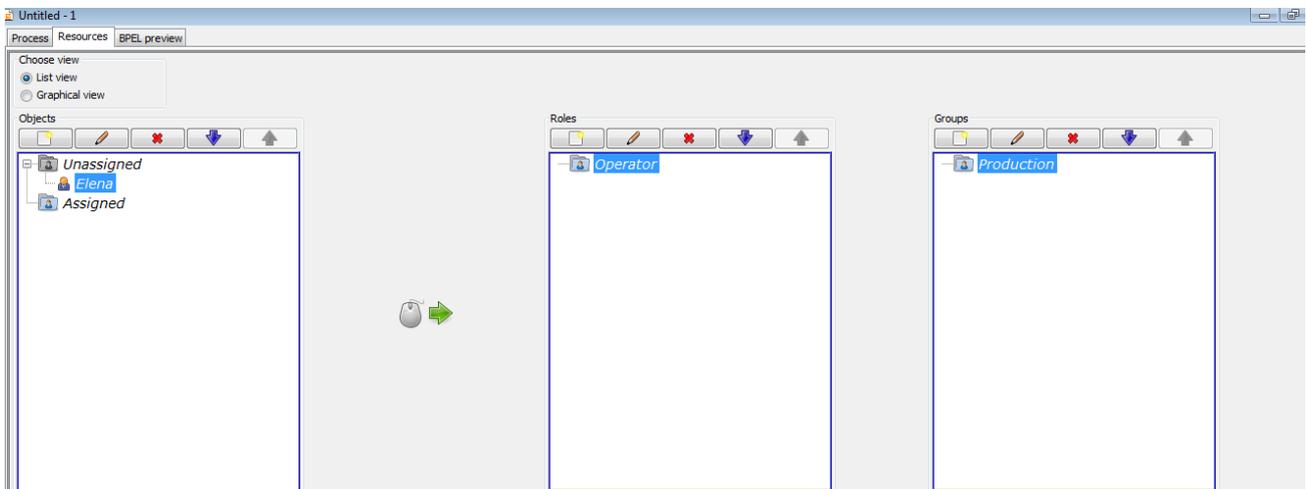


Fig.3.1.6: Resource Mapping Interface

Mediante l'utilizzo di uno specifico algoritmo, Woped consente di trasformare automaticamente il disegno di una rete in un editor BPEL. Come già accennato, il BPEL (Business Process Execution Language) è un linguaggio basato sullo standard XML per la definizione di processi aziendali [54]. Il BPEL nasce storicamente dalla fusione di due tecnologie sviluppatesi indipendentemente all'inizio degli anni 2000: WSFL di IBM e XLANG di Microsoft. Esso origina dall'esigenza di disporre non solo di un linguaggio di programmazione utile ai tecnici del software, ma anche di un

formalismo utilizzabile dai manager e dagli esperti dei domini applicativi, che possa realizzare una piattaforma comune per la collaborazione fra diverse aree disciplinari.

Utilizzando il BPEL un programmatore può descrivere le attività aziendali, rappresentandole come processi che, a loro volta, raggruppano le singole funzionalità secondo precise regole aziendali, o Business Rules. La definizione di un processo BPEL include, tra l'altro, la definizione delle relazioni tra il processo ed i suoi partner, rappresentate dai cosiddetti partner link. Uno stesso processo può essere in pratica istanziato più volte, in modo che ogni istanza provveda a soddisfare una specifica richiesta pervenuta e che più richieste possano essere processate contemporaneamente: il processo di esecuzione di un esame diagnostico, ad esempio, può essere eseguito n volte per n pazienti e la richiesta dell'esame innesca l'avvio del processo; ogni paziente che richiede l'esame rappresenta la singola istanza di processo.

Affinché l'esecuzione di ogni istanza avvenga in modo corretto può essere necessario inserire nei messaggi scambiati tra processo e partner delle informazioni di correlazione, che permettono di associare ciascun messaggio all'istanza corretta a cui è destinato o da cui proviene [55].

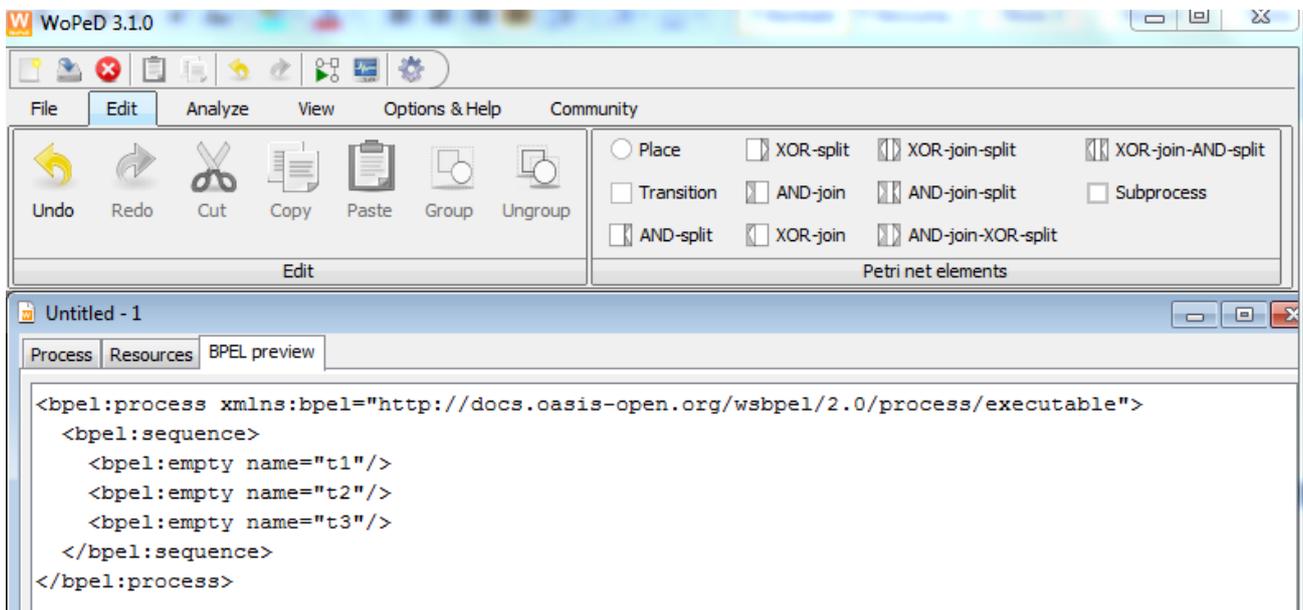


Fig.3.1.8: BPEL preview

Una volta creato il disegno del flusso di lavoro, WoPeD dà la possibilità effettuare una simulazione animata della sua esecuzione; tale simulazione è chiamata **token game**. Come mostrato nella figura 3.1.9, l'interfaccia permette di navigare in avanti, indietro, all'interno dei sottoprocessi, uno step alla volta o con più step consecutivi, ripercorrendo dunque l'intero processo.

Naturalmente, quando è attiva la funzione token game la rete non può essere modificata. Il software permette la registrazione e la riproduzione delle sessioni di simulazione, nonché il loro salvataggio e la possibilità di una successiva consultazione.

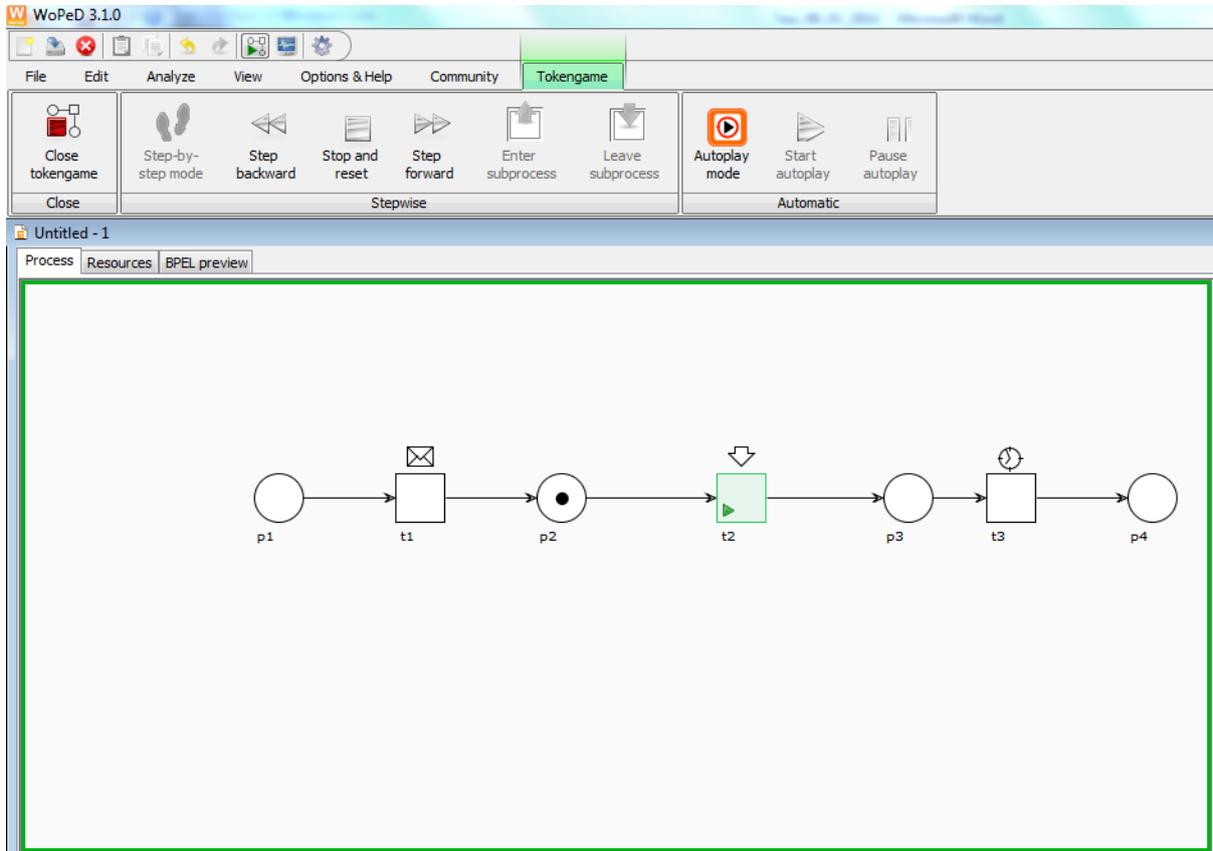


Fig.3.1.9: Token Game

Come accennato nel paragrafo precedente, un concetto importante nell'ambito delle reti di Petri è il reachability graph. La "raggiungibilità" degli stati di una rete di Petri, può essere rappresentata con un grafo orientato, i cui punti rappresentano gli stati e gli archi le transizioni tra due stati. La costruzione del suddetto grafico avviene in questo modo: viene preso lo stato iniziale, ad esempio M_0 , e si esplorano tutte le possibili transizioni che possono aver luogo a partire da esso. Si passa poi alla valutazione delle transizioni e degli altri stati ad esse connessi.

WoPeD crea un'unica finestra aggiuntiva per tutti i grafi di raggiungibilità, condivisa da più editor qualora siano attive più sessioni; naturalmente nella finestra comparirà il grafo riferito all'editor evidenziato. La modalità di visualizzazione del grafico può variare a seconda che si preferisca visualizzarlo in scala di grigi, oppure a colori.

WoPeD consente inoltre di valutare una notevole varietà di proprietà quali le free-choice situation, soundness, well-structuredness: alcune di queste sono analizzate attraverso l'applicazione di specifici algoritmi, altre utilizzando il tool Woflan [56].

L'analisi Woflan fornisce una rappresentazione delle proprietà esaminate: essa mostra una lista degli errori presenti sulla rete e individua le possibili sorgenti di criticità. Tali valutazioni possono essere effettuate anche a livello locale, cioè in riferimento ai singoli sottoprocessi presenti nel modello generale.

In accordo a [56], l'analizzatore Woflan (WORkFLOW ANalyzer), è suddiviso in tre parti:

1. **Parser:** si basa sul presupposto teorico che la struttura di una rete di Petri debba essere caratterizzata da un unico punto di ingresso e un unico punto di uscita e che ogni nodo della rete debba avere uno specifico percorso, con un input ed un output. Dalla lettura della rete, il parser avverte l'utente qualora questa non soddisfi questo requisito fondamentale.
2. **Analysis routines:** tale valutazione viene utilizzata come base per le successive analisi, in quanto fornisce all'utente una serie di informazioni come la presenza di task senza risorse, punti morti, task non chiuse, errato utilizzo degli elementi di sintassi ed inoltre fornisce un'analisi su una delle principali proprietà strutturali di una rete di Petri, ovvero la soundness [57].
3. **User interface:** per presentare i risultati dell'analisi, Woflan ha una propria interfaccia grafica. Quest'ultima consente l'analisi di più processi contemporaneamente, che possono selettivamente essere evidenziati dall'utente.

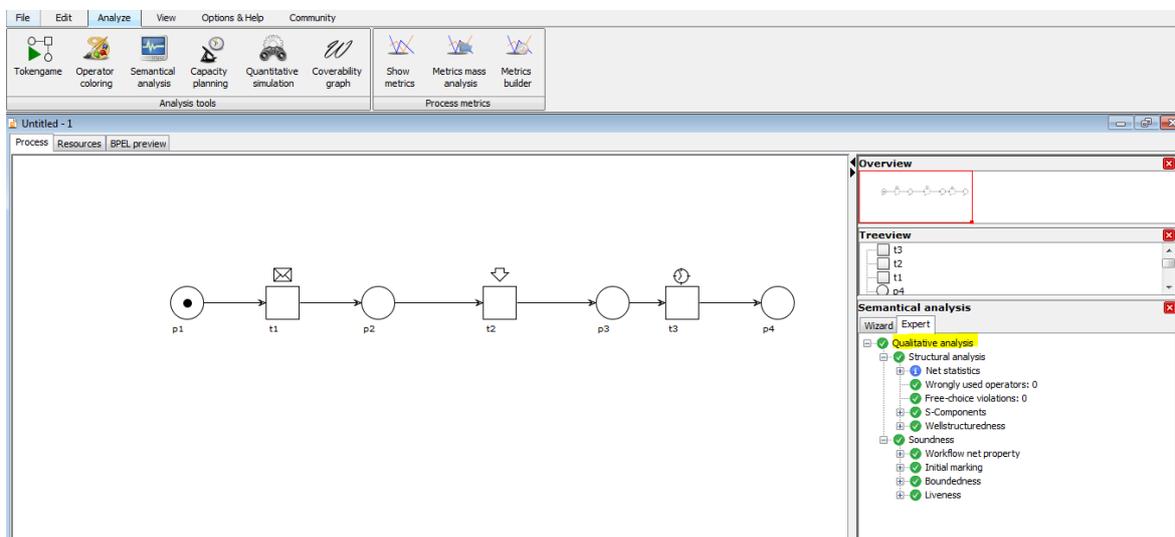


Fig.3.1.10: Analisi Qualitativa

Oltre all'analisi qualitativa, WoPeD consente di effettuare anche un'analisi quantitativa del modello di rete creato. Mediante la simulazione dell'esecuzione del flusso di lavoro disegnato, è possibile conoscere, per ciascuna transizione, il tempo di servizio, il tempo effettivo di esecuzione, i casi conclusi e i tempi di attesa. Inoltre, è possibile anche conoscere nel dettaglio il numero di casi serviti, la lunghezza massima delle code, il numero di casi non serviti ed ancora presenti all'interno della rete quando la simulazione del processo viene arrestata. In aggiunta, può essere calcolata la percentuale di utilizzo di ogni singola risorsa presente nella rete.

Per poter effettuare questo tipo di valutazioni, ogni qual volta si modella la rete, bisogna specificare le seguenti informazioni:

1. tempi medi di servizio di ciascuna transizione;
2. valori di probabilità sugli archi in uscita da xor-split;
3. assegnazione delle risorse, e dei relativi ruoli e gruppi ogni task che prevede l'utilizzo di risorse.

La figura 3.1.11 la schermata per definire le proprietà di una task. Inizialmente essa viene identificata con un nome; ad essa viene poi associata la tipologia del trigger che ne abilita lo svolgimento; se per la sua esecuzione è previsto l'utilizzo di una risorsa, se ne specificano il ruolo ed il gruppo. Si definisce infine il tempo medio di esecuzione della task in oggetto.

Quando nel flusso di lavoro sono presenti delle XOR-split, ai fini di una corretta analisi risulta necessario inserire dei valori di probabilità su ogni arco in uscita da questa transizione.

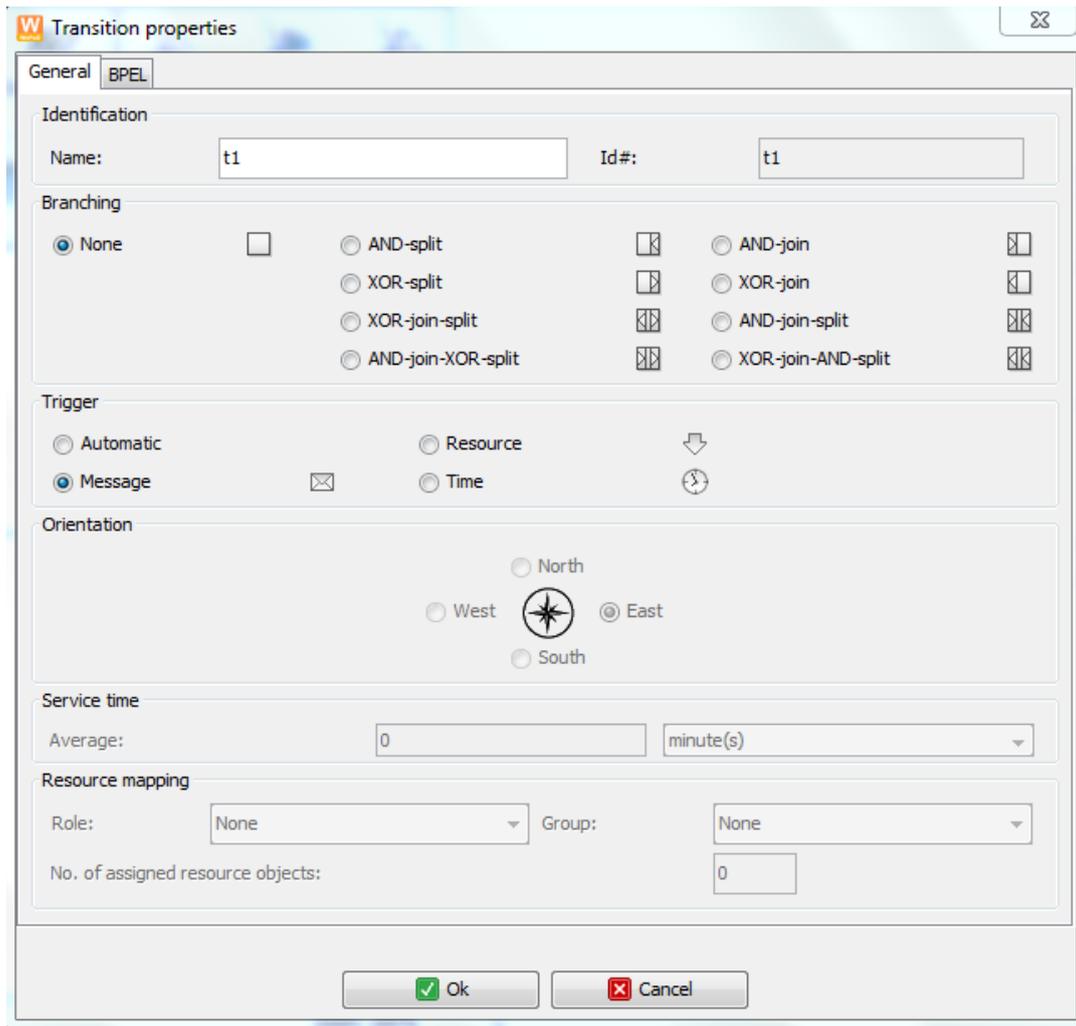


Fig.3.1.11: Proprietà di una transizione

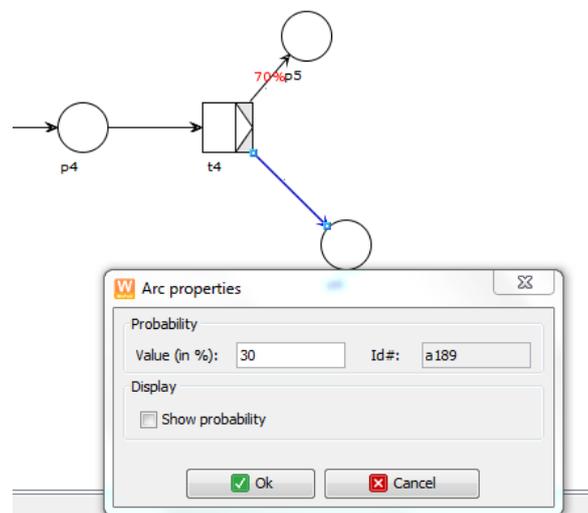


Fig.3.1.12: Definizione delle probabilità sugli archi

Nella schermata è possibile definire il valore, naturalmente compreso tra 0 e 100%, e si può scegliere se mostrarlo sulla rete oppure lasciarlo implicitamente fissato.

Un'importante caratteristica di WoPeD risiede nella possibilità di utilizzare unità di tempo differenti per le analisi quantitative, ovvero l'unità di tempo standard può essere modificata attraverso la specifica finestra di dialogo time model.

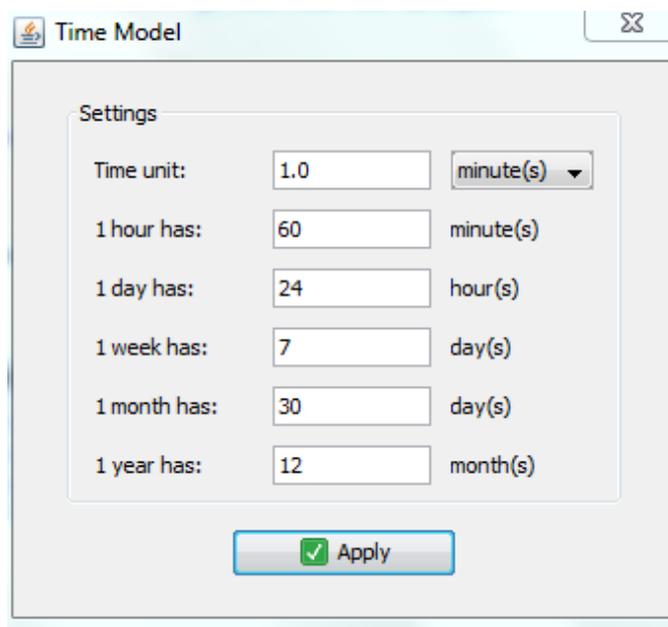


Fig.3.1.13: Time model window

WoPeD consente di effettuare due tipi differenti di analisi quantitativa: la **simulazione** e l'**analisi di capacità**.

Preliminare alla simulazione è la verifica della 'solidità' o soundness della rete: qualora dovessero presentarsi delle anomalie, l'utente verrà avvertito con opportuni messaggi di errore. L'avvio di una simulazione, inoltre, richiede la definizione e l'esecuzione di diversi passaggi di configurazione. Fondamentale risulta la definizione di parametri quali il numero medio di casi o istanze che arrivano in un determinato intervallo di tempo (λ casi) e la lunghezza del periodo sulla quale si vuole effettuare la simulazione. Naturalmente, come si può ben immaginare, aumentando la lunghezza del periodo miglioreranno i risultati ottenuti in termini statistici, a discapito, però, del tempo di calcolo. In Fig. 3.1.14 viene rappresentata la schermata dell'analisi quantitativa, con i parametri da settare. Con il parametro 'Queueing discipline' si definisce la tipologia di coda, ovvero la modalità di immagazzinamento dei token; la scelta è orientata verso due differenti approcci: FIFO (First in - first

out), secondo cui il primo token ad entrare è anche il primo ad uscire; LIFO (Last in - First out), secondo cui l'ultimo oggetto inserito è il primo a presentarsi in uscita.

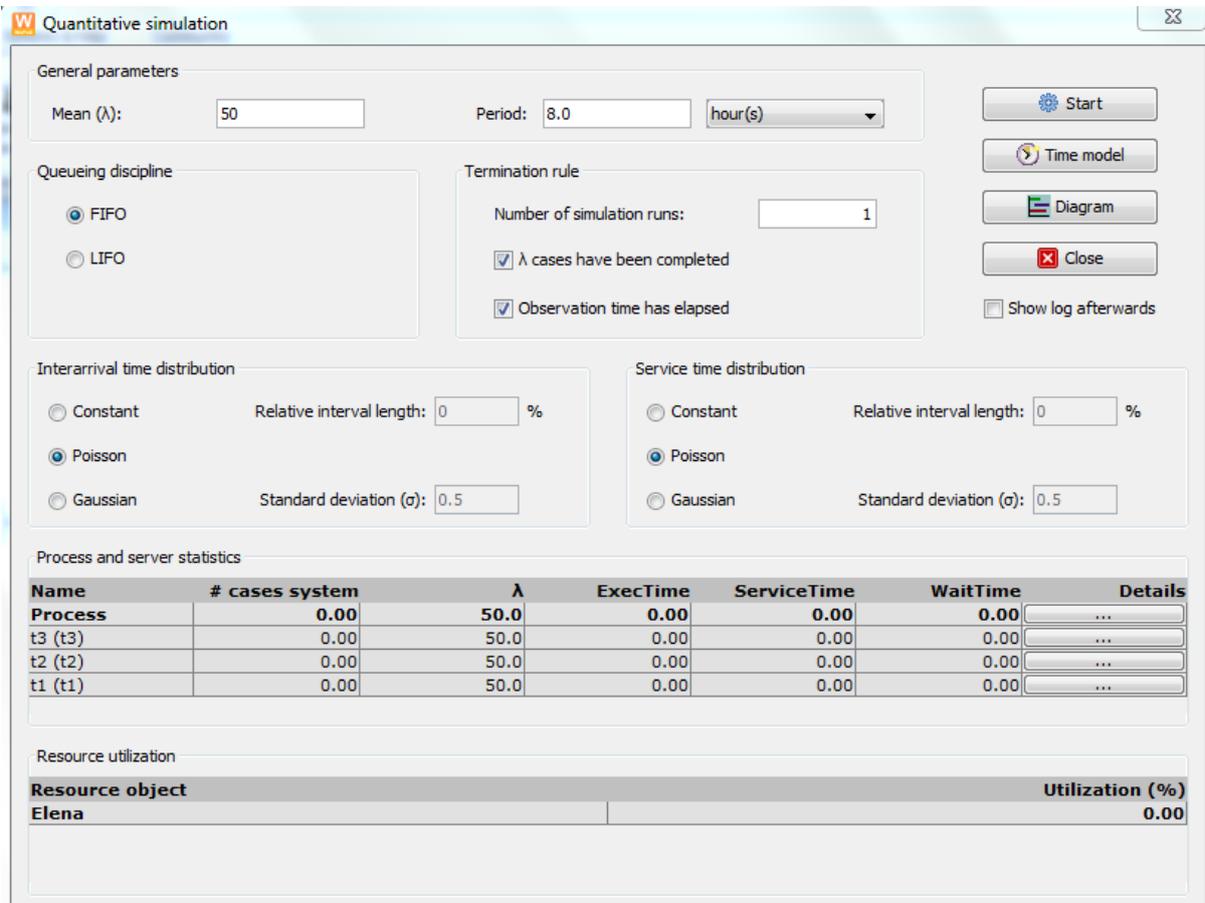


Fig.3.1.14: Quantitative Simulation

Con 'termination rule', si definiscono le regole affinché il flusso possa ritenersi concluso: o quando tutti i λ casi sono terminati o quando il periodo di osservazione è trascorso; in questa fase si imposta anche il numero di simulazioni da effettuare. Si ricorda che l'analisi che WoPeD restituisce è sempre data da una media aritmetica ricavata dalle simulazioni precedentemente effettuate, per cui più il numero di simulazioni è alto, più attendibile sarà il risultato restituito.

Il successivo step di configurazione prevede la definizione del tipo di distribuzione dei tempi di arrivo. L'analisi viene fatta utilizzando delle opportune distribuzioni di probabilità (si ricorda che una distribuzione di probabilità è un modello matematico che collega i valori di una variabile alle probabilità che tali valori possano essere osservati): Costante; Poisson e Gaussiana [58].

Altre funzionalità della finestra di dialogo *Simulazione* sono la possibilità di poter esportare i file dell'analisi con estensione .csv e di aprire una finestra dove visualizzare il protocollo come testo,

tradizionalmente salvato in formato .xml. Inoltre, con l'icona *Diagram* è possibile visualizzare un diagramma che fornisce la rappresentazione grafica della ripartizione delle risorse con i rispettivi tempi di simulazione.

Nella parte in basso della figura, relativa ai risultati, nella prima colonna sono indicati i nomi delle transizioni presenti nel processo; di seguito si visualizza il numero dei casi ancora presenti nelle al momento dello stop della simulazione; sono poi indicati il numero di λ (casi) trattati da ogni task e nelle successive tre colonne sono specificate le relative tempistiche: dapprima viene indicato il tempo effettivo impiegato per eseguire quello specifico compito, poi il 'service time' - che indica il tempo prefissato determinato nella fasi di analisi e progettazione del flusso - ed infine troviamo il tempo di attesa, che altro non è che la differenza tra le due tempistiche precedenti.

Il **tempo di attesa** risulta fondamentale, perché dalla sua valutazione è possibile individuare la presenza o meno di code che rallentano la normale esecuzione del flusso di lavoro.

L'ultima colonna, infine, descrive in maniera più dettagliata le singole task, indicando ad esempio il numero di casi che vengono eseguiti senza attesa, il numero di casi completati, la massima lunghezza della coda, il massimo tempo d'attesa, il numero di casi iniziati quando la simulazione è stata stoppata.

L'ultima parte della schermata di simulazione mostra l'elenco delle risorse utilizzate nella rete, e la relativa percentuale di utilizzo (0 nel caso in esame, a soli scopi illustrativi).

Come accennato in precedenza, Woped consente di effettuare anche un'**analisi di capacità**, che fornisce i dettagli relativi ad ogni task. Avviando l'analisi, si osservano infatti le caratteristiche di ogni transizione: il tempo di servizio, le istanze eseguite, il tempo di esecuzione di ogni caso, le istanze eseguite nell'intero periodo di osservazione; inoltre si ha anche una valutazione sul tempo complessivo che ogni gruppo di risorse impiega per eseguire tutti i compiti assegnati, nonché l'indicazione del numero minimo di risorse necessarie, valutato in base alla percentuale di utilizzo impostata (di solito si sceglie di impostare una percentuale di utilizzo dell'80%).

Anche per l'analisi di capacità, operazione preliminare è la configurazione dei parametri di analisi, quali il periodo di osservazione ed il numero medio di casi in arrivo.

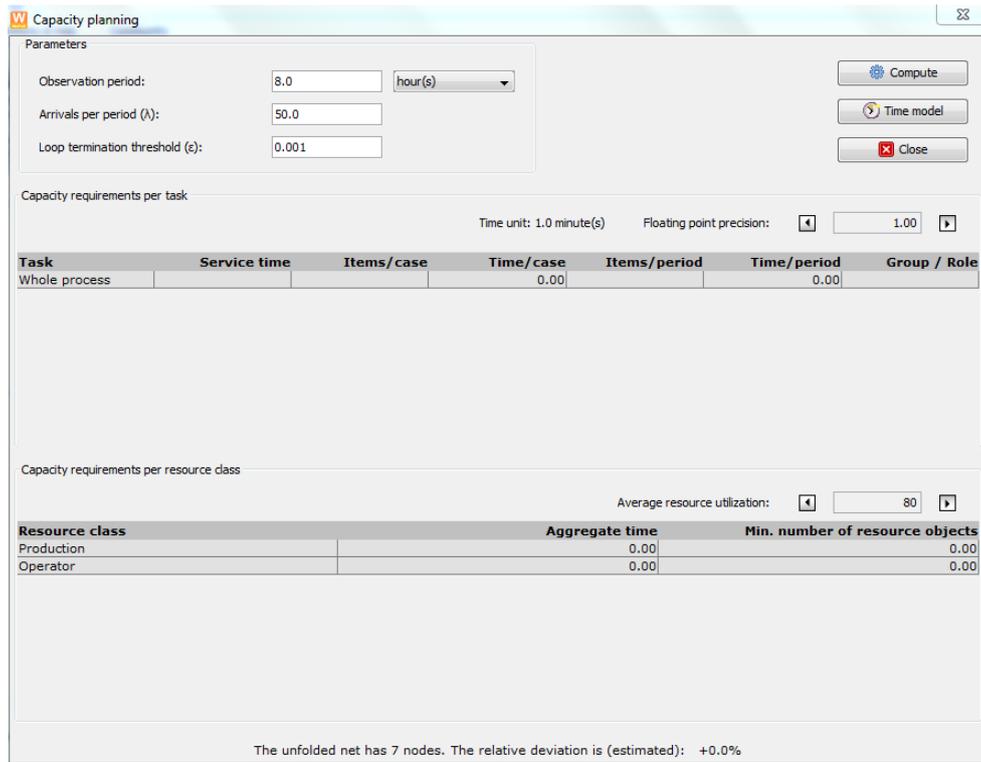


Fig.3.1.14: Capacity Planning

Un ulteriore parametro viene definito in questa fase, indicato con ϵ . Ad ogni nodo è in pratica assegnato un valore φ_k che andrà man mano a diminuire ogni qual volta si ripeterà un'azione su quel particolare nodo: poiché i cicli di ripetizione in linea di principio possono presentarsi infinite volte, occorre un condizione necessaria di interruzione: $\varphi_k^* p < \epsilon$, dove p è la probabilità indicata sul bordo di uscita. I risultati risultano essere migliori per un ϵ piccolo, tuttavia quando i calcoli si presentano estremamente complessi, molto spesso l'analisi si interrompe creando molti overflow.

Come si evince in figura, i risultati ottenuti sono schematizzati in due tabelle. Nella prima, (*task*) vengono evidenziate le frequenze di esecuzione delle singole task (nulle nella figura, a soli scopi illustrativi). Nella seconda tabella, invece (*resource class*) viene indicato il tempo totale di impiego di ogni classe di risorse; viene inoltre specificato il numero minimo di risorse necessarie per l'esecuzione del flusso di lavoro. Per questo tipo di valutazione di norma viene scelta una percentuale media di utilizzazione delle risorse pari all'80%, ma naturalmente questo parametro può essere modificato in base alle esigenze del particolare contesto analizzato.

Per comprendere le modalità di impiego del tool Woped per l'analisi dei flussi di lavoro nel contesto sanitario, verrà descritto, nel seguente paragrafo, l'impiego di tale strumento per l'analisi di un particolare case study di interesse, ovvero la gestione centralizzata degli acquisti di farmaci e dispositivi medici [13].

3.2 Case Study: La gestione centralizzata degli acquisti di farmaci e dispositivi medici

Negli ultimi 20 anni, le Organizzazioni Sanitarie Europee stanno progressivamente integrando a livello regionale i propri flussi di lavoro, al fine di garantire adeguati servizi e rispondere alle necessità espresse dalla popolazione, nonostante la limitatezza delle risorse finanziarie disponibili.

In particolare, numerosi enti amministrativi nazionali e regionali puntano ad integrare e centralizzare, sotto la supervisione di un organismo pubblico, tutte le attività tra di loro assimilabili che vengono comunemente svolte, con maggiori spese, da pochi organismi periferici denominati nel seguito del paragrafo 'Public Health Companies' (PHC), che sono in genere organizzazioni pubbliche responsabili dell'erogazione di servizi sanitari, e di altre attività correlate, riguardanti un particolare distretto all'interno dell'area nazionale/regionale [59] [60].

Un organismo centralizzato per il procurement di beni e servizi, tipicamente di natura pubblica, consente di ottenere risparmi significativi prestando al contempo attenzione alla buona qualità: pertanto, negli ultimi anni si è assistito ad una progressiva centralizzazione degli acquisti di farmaci e dispositivi medici tanto nelle regioni italiane quanto in Europa [61] [62].

Gli Enti incaricati di centralizzare le attività, nella maggior parte dei casi, sono in particolare responsabili di scegliere i fornitori più competitivi, al fine di incrementare le quantità e ridurre i costi; le PHC restano di solito incaricate della gestione della supply chain, con particolare riferimento alla verifica della merce ricevuta ed al pagamento dei fornitori. In alcuni casi, quando le PHC non sono in grado di rispettare le condizioni contrattuali con i fornitori perché pesantemente indebitate, gli organismi centralizzanti prendono in carico anche parte della supply chain (ordini, fatturazione pagamenti) per conto delle PHC, per garantire un puntuale pagamento dei fornitori.

Al fine di approfondire i meccanismi caratterizzanti la gestione centralizzata degli acquisti di farmaci e dispositivi medici, è stata focalizzata l'attenzione su una realtà regionale in cui un'agenzia appositamente incaricata – chiamata nel seguito Centrale Acquisti - gestisce l'intero processo di approvvigionamento. I flussi di lavoro caratterizzanti la gestione centralizzata sono stati modellati attraverso una rete Petri in Woped, al fine di analizzare quantitativamente vantaggi ed aspetti critici di tale approccio.

Nel caso di studio considerato, la Centrale Acquisti è responsabile dell'approvvigionamento di farmaci e dispositivi medici per conto di aziende sanitarie pubbliche distribuite sul territorio regionale, denominate appunto 'Public Healthcare Companies' (PHC).

A livello operativo, i flussi di lavoro possono essere schematizzati come in Figura 3.2.1 :

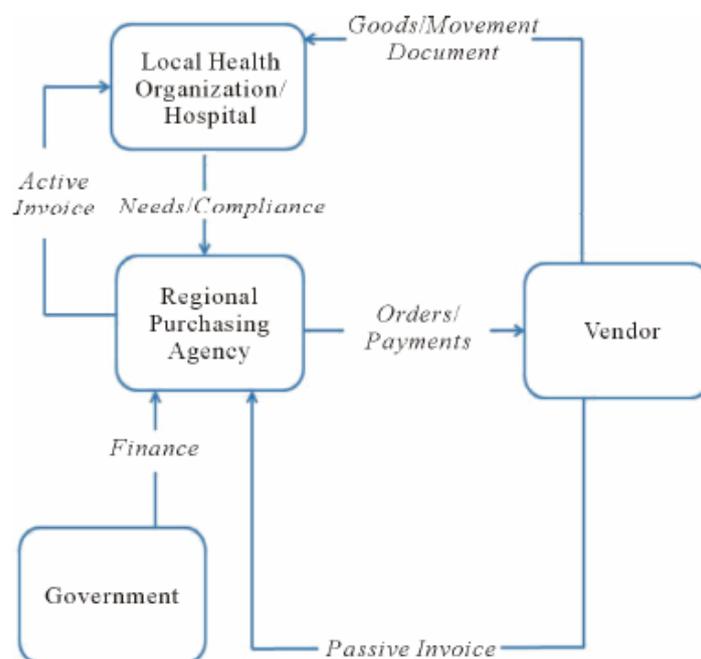


Fig.3.2.1: Flusso connesso alla centralizzazione degli acquisti in una realtà regionale

Periodicamente la Centrale Acquisti riceve dalle PHC informazioni in merito ai loro fabbisogni di farmaci e dispositivi medici (tipologia, quantità). Si occupa pertanto di indire delle procedure di gara e valutare le offerte proposte dai Fornitori dei prodotti richiesti, scegliendo poi la più conveniente. A questo punto le PHC – per conto della Centrale Acquisti ed attraverso un sistema informativo opportunamente configurato – inviano gli ordini ai Fornitori che sono stati selezionati in fase di gara, sulla base degli accordi riguardanti i prezzi e le condizioni di fornitura (tempi, trasporto). I Fornitori spediscono dunque le merci ordinate con i relativi documenti di trasporto alle PHC, mentre inviano le corrispondenti fatture (fatture passive) direttamente alla Centrale Acquisti. La Centrale Acquisti provvede al pagamento dei fornitori ed emette le corrispondenti fatture attive alle PHC. Le PHC non effettuano alcun pagamento nei confronti della Centrale Acquisti, ma sono responsabili della ricezione e della logistica legata alla movimentazione della merce, così come della verifica della correttezza dei documenti di trasporto che è indispensabile per il pagamento della fattura passiva al fornitore.

Sulla base di specifici accordi con l'Amministrazione Regionale, il pagamento della fatture attive alla Centrale Acquisti da parte delle PHC viene effettuato mediante uno specifico meccanismo di prelievo dai fondi che l'Amministrazione Regionale periodicamente mette a disposizione di

ciascuna PHC; tale meccanismo rappresenta un elemento essenziale di garanzia per la Centrale Acquisti, e, di conseguenza, per i Fornitori.

Tale specifico modello operativo, che differisce da quelli adottati dalle Centrali Regionali Italiane, genera una duplicazione del flusso amministrativo. Infatti, per un singolo ordine ad un determinato Fornitore, vengono generate due fatture – una passiva dal Fornitore alle Centrale ed una attiva dalla Centrale alla PHC - che devono essere individualmente processate dalla Centrale Acquisti. Tale peculiarità rappresenta un aspetto critico che impone che il flusso documentale e la sua gestione, dall'ordine al pagamento della fattura, sia il più automatizzato possibile. Di conseguenza, la Centrale Acquisti deve dotarsi di un appropriato sistema informativo amministrativo in grado di gestire, per un considerevole volume di documenti (corrispondenti a circa 200.000 ordini ogni anno) la grande eterogeneità di tipologie di prodotti e di accordi contrattuali, che variano da Fornitore a Fornitore.

Il Sistema Informativo deve gestire l'intero workflow amministrativo; rispetto ad un tradizionale sistema informativo manageriale [63], esso deve anche prevedere un set specifico di funzioni legate alle particolarità della supply chain regionale.

Dalla figura 3.2.1 si evince l'alto grado di integrazione richiesto tra i sistemi informativi delle singole PHC e quello della Centrale Acquisti; devono infatti essere predisposti appositi moduli di integrazione tra i sistemi informativi locali delle PHC ed il sistema centralizzato. Le PHC, infatti, devono non solo essere in grado di interfacciarsi con la Centrale per quanto concerne gli ordini, ma devono anche garantire un'adeguata gestione delle loro attività interne, attraverso un proprio sistema informatico indipendente. Pertanto, deve essere realizzata un'integrazione tra il sistema informativo della Centrale ed i sistemi indipendenti delle PHC, che preveda:

1. L'update del database relativo agli ordini, che mantenga l'allineamento tra gli ordini effettuati dalle PHC ed i corrispondenti ordini generati all'interno del Sistema Informativo Regionale.
2. L'update del database dei documenti di trasporto, per garantire l'allineamento tra i documenti gestiti dalle PHC e quelli corrispondenti nel Sistema Centralizzato, in modo da consentire alla Centrale Acquisti il pagamento delle fatture passive al Fornitore.

Da un punto di vista tecnico, l'integrazione tra il sistema informativo centralizzato e quelli locali delle PHC è possibile mediante un semplice scambio FTP [64] , malgrado le tecnologie Web Services e SOA potrebbero dimostrarsi più efficienti [65]. Il sistema informativo dovrebbe consentire alle PHC di effettuare via Web delle proposte di ordini, che rappresentano le richieste delle PHC alla Centrale Regionale, la quale deve dunque dare una conferma dell'ordine stesso prima

che esso venga consegnato al fornitore. In genere, tali attività vengono svolte da staff amministrativo delle PHC con la supervisione di un farmacista/amministrativo dipendente della Centrale Acquisti. Completato l'ordine, la Centrale Acquisti - in quanto detentrica del contratto di fornitura - lo trasmette (automaticamente tramite fax o mail) al fornitore; naturalmente lo stesso ordine, attraverso il sistema integrato, viene esportato al sistema informativo della PHC richiedente.

Gli operatori della PHC, alla ricezione della merce, devono provvedere ad effettuare all'interno del proprio sistema informativo il data entry di tutti i dettagli relativi ai prodotti ricevuti; questi dati vanno allo stesso modo esportati verso il sistema informativo della Centrale Acquisti.

Quest'ultima deve inoltre registrare le fatture passive inviate dai Fornitori. Come mostrato in figura 3.2.2, in generale tale attività prevede un semplice data entry di alcune informazioni chiave.

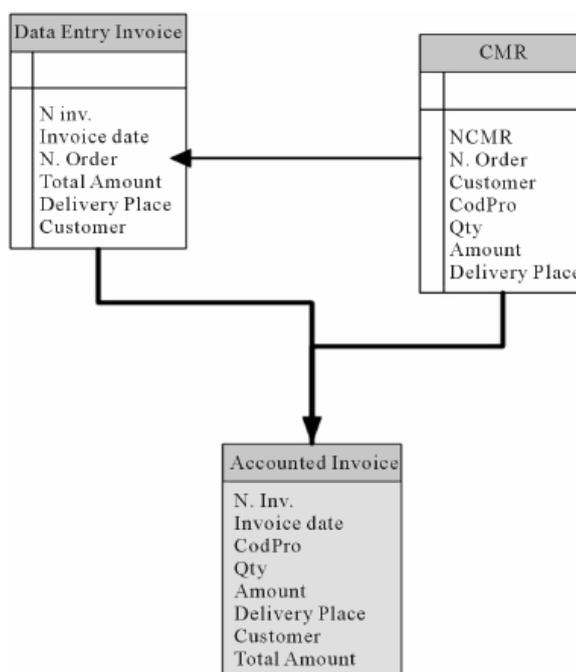


Fig.3.2.2: Data Entry necessario alla generazione delle fatture

La generazione delle fatture attive richiede tutti i dettagli contenuti nelle righe dell'ordine. Ciò significa che, dato l'elevato volume di documenti, (circa 200.000 documenti all'anno), il data entry manuale delle fatture passive genera un consistente carico di lavoro per gli operatori ed una potenziale perdita di qualità dei dati, rendendo il modello non propriamente funzionale agli scopi. L'introduzione della fattura elettronica in Italia potrebbe risolvere il problema, tuttavia la maggior parte dei Fornitori non è ancora abilitata alla sua generazione. Una possibile soluzione per evitare il data entry manuale è la ricostruzione dei dati della fattura passiva utilizzando le informazioni contenute nel documento di trasporto, informazioni che vengono normalmente registrate dalle PHC alla ricezione dei prodotti ordinati. Come si evince dalla figura 3.2.2, sulla base del numero d'ordine

e del codice Fornitore, sarebbe possibile identificare – attraverso la tabella dei documenti di trasporto (indicata in figura come CMR) – i dettagli relativi alla fattura passiva. Tale meccanismo consente di generare automaticamente una ‘fattura d’acconto’, riga per riga coerente con la fattura passiva, evitando il data entry manuale. Ovviamente si rende necessario un certo numero di controlli per verificare l’esatta corrispondenza tra tale fattura ed il documento di trasporto.

Sfortunatamente, tuttavia, accade spesso che, quando la fattura passiva viene registrata all’interno del sistema informativo, i dati riguardanti il documento di trasporto non risultano ancora disponibili perché non sono stati ancora inseriti nei database delle PHC; in aggiunta, l’importo (unit cost), o la descrizione della merce sulla fattura possono essere diversi da quelli riportati nel documento di trasporto, a causa di un disallineamento tra i database. In questi casi, i dati della fattura passiva devono necessariamente essere registrati manualmente; piccole differenze dovute all’arrotondamento numerico possono in ogni caso essere consentite nel rispetto dei limiti consentiti dalla legge: ciò consente di evitare, in un numero significativo di circostanze, il data entry manuale, rendendo pertanto maggiormente efficiente il flusso operativo.

Sulla base delle fatture di acconto il sistema informativo dovrebbe generare, automaticamente, le corrispondenti fatture attive, da inviare alle PHC, così che possano essere detratte dai rispettivi budget. L’ultimo step del ciclo prevede una comparazione tra le fatture passive ed i documenti di trasporto già inseriti all’interno del sistema informatico, al fine di autorizzare il pagamento del fornitore. In questa fase, che è sotto la responsabilità della Centrale Acquisti, quest’ultima dovrebbe verificare, riga per riga, l’esatta corrispondenza tra i prodotti ordinati e quelli consegnati e fatturati. Tale verifica è senza dubbio onerosa, ma nel caso in cui la fattura passiva sia generata automaticamente, anche il check può essere reso automatico.

Risulta quindi chiaro che, al fine di ridurre il carico di lavoro ed i tempi operativi, ed anche di migliorare la qualità globale del processo, è fondamentale ottenere un perfetto allineamento dei codici prodotto utilizzati al momento dell’effettuazione dell’ordine dalle PHC ed i codici prodotto del fornitore. In poche parole, il corretto ed efficace management dell’intera supply chain del farmaco – ordine, documento di trasporto, fattura, pagamento – richiede un accurato allineamento dei codici registrati nei sistemi informatici delle PHC con gli stessi riportati negli ordini e nelle fatture, in modo da rendere immediato il confronto tra fatture, ordini e documenti di trasporto.

Di conseguenza, è fondamentale assicurare la consistenza tra i dati contenuti nelle seguenti tabelle:

- Codici Prodotto delle Organizzazioni Sanitarie
- Codici Prodotto dei Fornitori
- Codici di farmaci e dispositivi medici

- Contratti
- Ordini
- Documenti di Trasporto
- Fatture

Ogni disallineamento tra tali tabelle costituisce una potenziale causa di confusione ed ambiguità nel confronto tra le righe dell'ordine e della fattura: da tali ambiguità derivano le difficoltà nelle autorizzazioni ai pagamenti ed i conseguenti ritardi nel completamento del flusso.

Da quanto descritto, risulta chiaro che i vari step della supply chain devono fare i conti con un certo numero di punti di forza e di debolezza che derivano dalla struttura del flusso di lavoro e dai diversi attori coinvolti nel processo.

Al fine di approfondire i diversi step della supply chain, e di individuare uno strumento di simulazione di un tale flusso organizzativo che potesse essere utilizzato per ulteriori casi di studio, il processo è stato modellato utilizzando le reti di Petri [12].

Come riportato in Letteratura [51], e illustrato nel paragrafo precedente, le reti di Petri consentono di modellare un processo specificando la sequenza di attività che lo costituiscono ed assegnando a ciascuna di esse la relativa durata e le risorse deputate alla loro esecuzione. Una volta disegnato il modello del processo, è possibile simularne l'esecuzione per riprodurre il suo comportamento in differenti condizioni, al fine di evidenziarne le criticità.

La figura seguente mostra la rete di Petri realizzata, utilizzando il simulatore Woped [53] per descrivere la supply chain di interesse. Senza pretese di completezza, essa evidenzia gli aspetti critici del flusso di lavoro connesso alla centralizzazione di farmaci e dispositivi medici da parte della Centrale Acquisti.

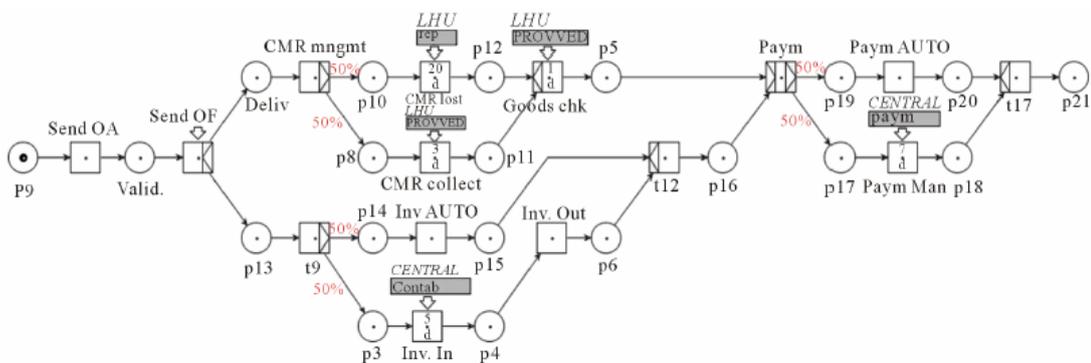


Fig.3.2.3: Gestione centralizzata di farmaci e dispositivi medici – rete di Petri

In accordo alle regole alla base del funzionamento delle reti di Petri, il singolo token rappresenta, all'interno del workflow, il documento di ordine/trasporto/fattura. In particolare, si assume che un singolo token corrisponda ad un totale di 200 ordini/documenti di trasporto/fatture. In tali circostanze il flusso inizia con la generazione dell'ordine da parte della PHC (attraverso il sistema informativo regionale centralizzato). Tali ordini vengono indicati come OA. Essi vengono dunque trasmessi dalla Centrale Acquisti ai Fornitori attraverso la generazione di un documento ulteriore indicato come OF (Fig.3.2.4).

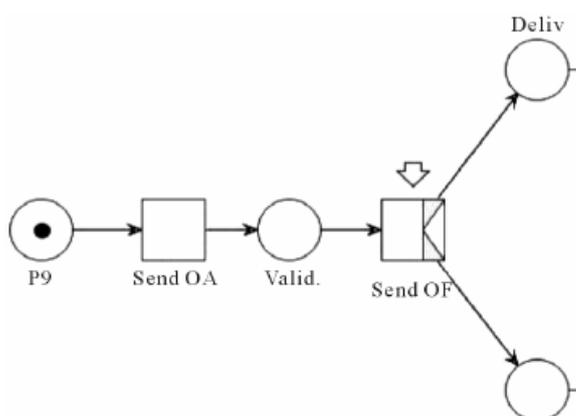


Fig.3.2.4: Fase di invio dell'ordine

Ricevuto l'ordine, i Fornitori consegnano i prodotti, con i relativi documenti di trasporto, direttamente alle PHC. Queste ultime, pertanto, devono effettuare le attività logistiche di distribuzione dei prodotti ai singoli reparti ordinanti al loro interno e verificare la conformità della merce ricevuta. Una volta che l'ordine è stato inviato al Fornitore, il workflow prevede una AND split per indicare l'esecuzione di due rami di attività in parallelo: il primo - mostrato in figura 3.2.5 - legato alle attività della Centrale Regionale per la gestione delle fatture attive e passive; il secondo - mostrato in figura 3.2.6 - relativo alle attività svolte dalle PHC, in particolare alla gestione degli aspetti logistici che includono la verifica dei documenti di trasporto.

Il secondo ramo (fig 3.2.6) prevede inoltre, sulla base di ciò che realmente accade, due alternative. In un primo caso i prodotti ed i documenti di trasporto vengono correttamente e rapidamente gestiti (ramo CRM collect); nell'altro caso, invece, i documenti di trasporto – a causa di inefficienze intrinseche della rete logistica delle PHC – seguono un percorso alternativo in cui le medesime attività vengono svolte in un tempo significativamente maggiore (ramo CMR lost). I due rami convergono poi in un'unica AND join che simula la verifica con esito positivo della corrispondenza necessaria al pagamento della fattura (Paym.chk).

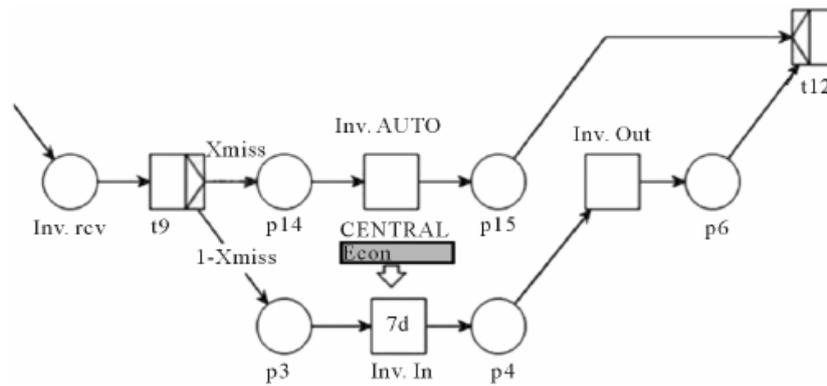


Fig.3.2.5: Gestione delle fatture

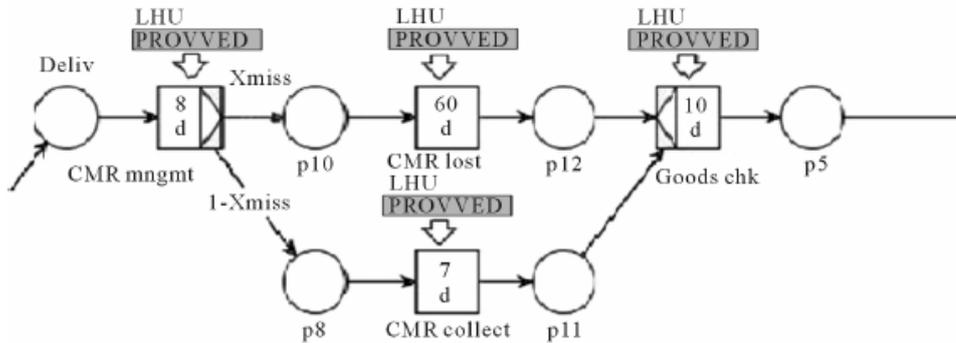


Fig.3.2.6: Gestione dei documenti di trasporto

Riassumendo, il processo di fatturazione prevede sequenzialmente la registrazione delle fatture passive inviate dai fornitori alla Centrale Acquisti e la generazione delle corrispondenti fatture attive dalla Centrale Acquisti alle PHC che hanno effettuato l'ordine. Il modello prevede che tutte queste attività possano essere generate sia in modalità manuale che automatica; nel primo caso la registrazione delle singole righe della fattura deve essere effettuata attraverso data entry manuale; nel secondo caso, assumendo che sia l'ordine che il documento di trasporto siano disponibili in formato elettronico, la ricostruzione delle righe della fattura può essere effettuata automaticamente. Pertanto, la corretta gestione dei documenti di trasporto impatta sull'intero processo a due livelli: direttamente per quanto riguarda la verifica della corrispondenza con l'ordine; indirettamente attraverso la possibilità di automatizzare la registrazione delle fatture attive e passive nel sistema informativo.

Allo stesso modo, l'ultimo segmento del modello simula la fase di autorizzazione del pagamento: essa può essere svolta manualmente – attraverso la verifica manuale riga per riga del documento di trasporto associato all'ordine – o automaticamente grazie alla ricostruzione della fattura descritta precedentemente (fig.3.2.7).

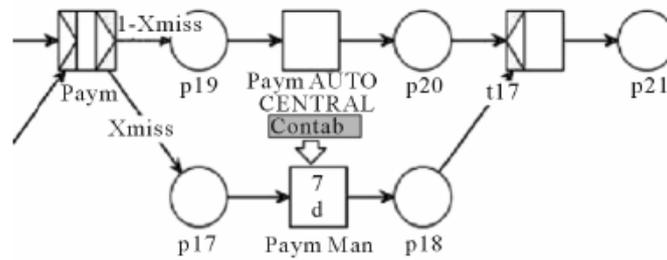


Fig.3.2.7: Pagamento

Dal momento che la gestione automatica delle fatture dipende dal numero di documenti di trasporto registrati in tempo, le diverse situazioni operative sono rappresentate, all'interno del modello, attraverso una serie di XOR che indirizzano su basi statistiche i token in diversi rami, a seconda della percentuale di documenti di trasporto che sono stati o meno correttamente gestiti (Xmiss).

Per simulare l'esecuzione del workflow, ciascuna transizione della rete di Petri deve essere configurata correttamente con la precisa identificazione delle risorse deputate alle attività.

Per la configurazione del modello proposto - come mostrato in tabella 3.2.1 - i dati da misurare nel sistema reale sono stati, pertanto, i seguenti:

1. Xmiss: la percentuale di documenti gestiti con significativo ritardo, che convenzionalmente si è assunta pari a 20 giorni contro i 3 previsti in una situazione fisiologica;
2. N. Inv.: il numero di risorse umane dello staff della Centrale Regionale responsabili della gestione delle fatture e dei pagamenti;
3. T. Inv.: il tempo richiesto per registrare 200 fatture passive dall'unità di personale: tale valore è convenzionalmente fissato pari a 5 giorni/unità staff;
4. N. Paym: numero di risorse responsabili della gestione di fatture e pagamenti;
5. T. Paym: tempo richiesto per l'autorizzazione del pagamento di 200 fatture passive dall'unità di personale; tale valore è convenzionalmente fissato pari a 7 giorni/unità di personale;
6. N.cmr: numero di unità della PHC responsabili della gestione dei documenti di trasporto;

7. T. cmr 1: tempo medio di gestione dei documenti di trasporto (canale ottimo): tale dato è convenzionalmente fissato pari a 7 giorni/unità di personale per 200 fatture/documenti di trasporto;
8. T. cmr 2: tempo medio di gestione dei documenti di trasporto (canale con ritardo): tale dato è convenzionalmente fissato pari a 20 giorni/unità di personale per 200 fatture/documenti di trasporto;
9. T. cmr 3: tempo medio di verifica della corrispondenza tra i documenti: tale dato è generalmente pari a 1 giorno/unità di personale per 200 fatture/documenti di trasporto.

E' importante sottolineare che i valori delle variabili 1,2,3,4 e 5 sono stati stimati sulla base dell'esperienza diretta all'interno della Centrale Acquisti considerata; i valori delle variabili 6, 7, 8 e 9 derivano dalle indicazioni delle PHC coinvolte nello specifico caso di studio [13]. Ad ogni modo, una stima più accurata delle variabili considerate non influirebbe sulla struttura del modello. In tabella 3.2.1 vengono elencati i valori assegnati ai suddetti parametri di configurazione.

	Min Value	Max Value	Fixed Value
Xmiss.	10	90	
N.inv.	4	9	
N.cmr			12
N.Paym	4	6	
T.Inv			5 days
T.Paym.			5 days
T.cmr1			7 days
T.cmr2			20 days
T.cmr3			1 day

Tab.3.2.1: Parametri di configurazione

I parametri selezionati sono stati scelti partendo dal presupposto che sia interessante analizzare il comportamento del modello come funzione dei parametri sui quali la Centrale Acquisti ha responsabilità. E' stato considerato dunque che la Centrale Acquisti possa modificare i valori di

N.inv o N.Paym aumentando sia il personale di registrazione delle fatture che quello adibito ad autorizzare i pagamenti. In alternativa, sarebbe anche possibile modificare Xmiss attraverso delle azioni mirate sulle PHC cercando di sensibilizzarle verso una corretta gestione dei documenti di trasporto. Per quanto riguarda gli altri parametri, risulta piuttosto complesso pensare di modificarli, in quanto sono sotto la completa amministrazione delle PHC.

Usando il tool Woped, con la configurazione indicata nel menu dedicato all'impostazione dei parametri, è stata effettuata un'analisi mirata ad individuare i punti di forza e di debolezza che caratterizzano il modello disegnato. Tale analisi ha messo in luce alcune inefficienze, in particolare relative alla gestione dei documenti di trasporto, che influenzano l'intero flusso operativo.

In particolare, nel corso dell'analisi è stata stimata la durata dell'intero flusso, che può essere considerata una stima del tempo richiesto per il pagamento, 'P delay', che rappresenta uno degli aspetti prioritari per una Centrale Acquisti che si assuma il ruolo di gestire i pagamenti dei fornitori.

Il trend di 'P delay' è stato analizzato come funzione delle seguenti variabili:

1. percentuale di documenti di trasporto non disponibili 'X miss'
2. numero di risorse deputate alla gestione delle fatture e dei pagamenti 'N. paym.' e 'N. inv.'

La tabella 3.2.2 evidenzia i parametri richiesti dal software per la simulazione, in riferimento al presunto volume di lavoro.

Valore di ciascun token	200
Numero di token in una simulazione	1000
Numero di fatture	200,000
Periodo di osservazione	1 anno
Unità di tempo	1 giorno

Tab.3.2.2: Parametri di simulazione

In tabella 3.2.3 ed in figura 3.2.8 sono riportati i risultati di alcune simulazioni effettuate considerando diversi valori di documenti di trasporto non disponibili 'X miss'.

Come ci si aspettava, i tempi di pagamento sono risultati drasticamente ridotti al diminuire della percentuale di X.miss, come conseguenza del miglioramento dell'automazione del workflow. Tale aspetto è rappresentato nel modello dai rami identificati con 1-X miss.

X miss (%)	P.delay (giorni)
90	2279,05
80	590,67
70	251,84
60	165,84
50	103,19
40	71,29
30	70,22
20	60,65
10	74,63

Tab.3.2.3: Ritardo nel pagamento in funzione di Xmiss

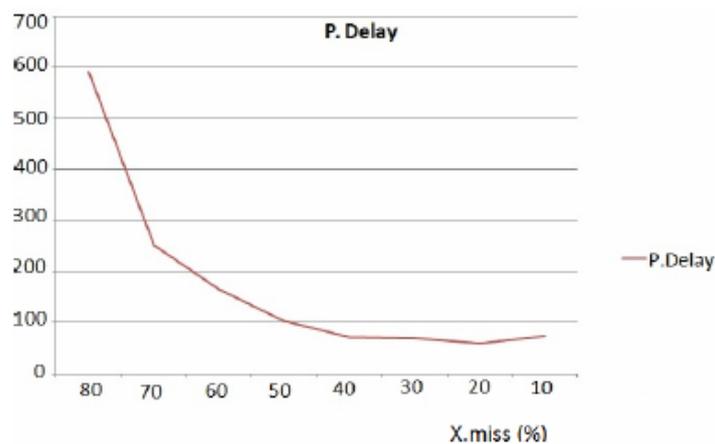


Fig.3.2.8 Ritardo nel pagamento in funzione di Xmiss

Bisogna considerare, tuttavia, che sulla base dell'esperienza reale non è possibile ottenere una diminuzione di Xmiss al di sotto del 50%: pertanto, è stata effettuata una seconda analisi in cui alla variabile Xmiss in uscita dalla XOR è stato assegnato il valore di 50%.

E' stata dunque eseguita una simulazione impostando i parametri come riportato in tabella 3.2.4:

	P.delay (giorni)	N.inv.	N.paym.	Tot Risorse
Xmiss=50%	112	4	4	8
	100	5	4	9
	87	6	4	10
	76	7	4	11
	67	8	4	12
	66	9	4	13
	43	8+1	4+1	13
	33	7+2	4+2	13
	32	6+3	4+3	13

Tab.3.2.2: Parametri di simulazione (con Xmiss=50%)

Assegnato il valore pari al 50% alla variabile X.miss, il tempo di pagamento Pdelay è stato calcolato dapprima aumentando il numero di risorse deputate alla gestione delle fatture N. inv. da 4 a 9, poi supponendo di condividere le stesse risorse nelle operazioni legate all'autorizzazione dei pagamenti (così da mantenere fisso il numero totale delle risorse pari a 13). Con tale impostazione, è stato ottenuto un tempo di ritardo per il pagamento delle fatture pari a 32 giorni, che risulterebbe più che soddisfacente nella prassi di globale gestione delle attività.

Naturalmente, tali risultati forniscono soltanto un'indicazione di quanto il ritardo nei pagamenti sia legato all'ammontare delle risorse e non considerano eventuali ritardi addizionali dovuti alla gestione dei rapporti con le banche o con le amministrazioni regionali. I risultati delle simulazioni risultano tuttavia consistenti con l'esperienza reale.

Lo studio condotto ha consentito di constatare che l'istituzione della Centrale Acquisti garantisce significativi risparmi nell'erogazione dei servizi e nell'acquisto di farmaci e dispositivi medici.

La gestione della supply chain richiede, d'altra parte, un sistema informativo specificamente progettato per la gestione del flusso di lavoro.

A causa della condivisione di responsabilità tra la Centrale Regionale e le PHC, tuttavia, ci sono alcuni aspetti problematici da tenere necessariamente in considerazione. Prima di tutto, è necessario che i rispettivi Sistemi Informativi siano altamente integrati, al fine di garantire il trasferimento in tempo reale di documenti, così come un perfetto allineamento delle informazioni, al fine di evitare inconsistenze tra ordini - documenti di trasporto - fatture che avrebbero dei negativi risvolti sulla liquidazione dei fornitori.

Si può inoltre affermare che la simulazione condotta ha mostrato come sia importante una corretta gestione della logistica legata alla consegna della merce al fine di assicurare un tempestivo data entry dei documenti di trasporto nel sistema informativo: infatti, il ritardo dei pagamenti legato ad inefficienze nella gestione dei documenti di trasporto ($X_{miss} \geq 50\%$) risulta inaccettabile. Pertanto, la gestione del flusso di lavoro richiede uno stretto controllo da parte della Centrale sulle PHC, le quali sono responsabili sia della ricezione che della verifica di conformità della merce. Dato un numero medio di documenti di trasporto registrati in ritardo (X_{miss} pari a 50%) e data una produttività mediamente pari ad 800 fatture al mese per ciascuna unità di personale, (incluso la gestione di fatture e pagamenti) risulterebbe possibile, con approssimativamente 13 risorse, (tra cui almeno 9 dovrebbero svolgere sia la registrazione delle fatture che l'autorizzazione ai pagamenti) gestire un flusso di 200,000 fatture all'anno, garantendo un tempo medio per i pagamenti di circa 30 giorni.

Sebbene numerose informazioni concernenti la struttura del modello derivino dall'esperienza sul campo, tale modello potrebbe essere utilizzato per l'analisi di ulteriori casi di studio. Sfortunatamente, alcuni fattori aventi un certo grado di aleatorietà devono essere tenuti in considerazione, come ad esempio il fatto che la fatturazione elettronica, a causa di legislative nazionali, risulta residuale rispetto alla fatturazione ordinaria. Inoltre, bisogna considerare che il turnover del personale e gli aspetti legati al training rischiano di rallentare le tempistiche stimate.

Per quanto riguarda il metodo applicato, si può concludere che esso ha consentito sia la modellazione che l'analisi del flusso di lavoro oggetto di studio, evidenziandone le criticità e portando all'individuazione di concrete ipotesi di ottimizzazione del flusso medesimo. In particolare, partendo dal modello proposto, importanti considerazioni sono state avanzate in merito alla mole di lavoro da gestire ed alla percentuale e alla modalità di allocazione delle risorse disponibili.

4. BUSINESS PROCESS MANAGEMENT E PROCESS MINING

4.1 Introduzione

Come spiegato nell'Introduzione, nelle moderne aziende i sistemi informativi e le tecnologie informatiche sono diventate sempre più importanti a causa della grossa mole di dati presenti quotidianamente registrati [66].

Ai fini di una corretta gestione dell'azienda, i Sistemi Informativi devono essere perfettamente integrati all'interno dell'organizzazione aziendale, il che comporta una perfetta conoscenza dei processi operativi dell'azienda stessa. In questa accezione i sistemi informativi possono rientrare nella più ampia categoria dei sistemi BPM, ovvero di Business Process Management [67].

In particolare, i Sistemi Informativi registrano i dati relativi all'esecuzione delle attività in particolari **Log di eventi** [15], che possono essere visti come una cronologia di ciò che accade all'interno della particolare organizzazione. I dati registrati possono essere utilizzati per ottenere una chiara descrizione dei processi aziendali in esecuzione. In accordo a [68], ciò accade anche all'interno delle Organizzazioni Sanitarie, in cui i Sistemi Informativi supportano sempre di più le attività di management e controllo dei processi.

Una delle aree emergenti del Business Process Management è rappresentata dal **Business Activity Monitoring**, BPA [69], che estende le funzionalità dei sistemi informativi alla diagnosi dei processi a partire dalla rilevazione dei dati memorizzati sotto forma di log di eventi, che vengono impiegati per ricavare i modelli dei processi eseguiti e rilevarne i punti di forza e di debolezza. Questo approccio è conosciuto come **Process Mining** [70].

Come dettagliato in [15], le tecniche di Process Mining consentono di estrarre conoscenza dai log che vengono registrati dai sistemi informativi odierni. Grazie a tali tecniche è possibile dedurre, monitorare e migliorare i processi in una grande varietà di domini applicativi.

Se le tecniche di workflow design consentono di modellare un processo basandosi sulle informazioni che derivano dalla Letteratura, dalle interviste al Personale coinvolto nel processo medesimo o dall'osservazione diretta, le tecniche di process mining consentono di ricavare modelli dei processi a partire da dati inerenti la loro esecuzione concreta e puntuale.

Nel settore sanitario, in particolare - in cui i processi sono in rapida evoluzione e caratterizzati da un elevato numero di variabili - tale aspetto assume un'importante rilevanza. Le tecniche di process mining si rivelano pertanto un importante strumento per migliorare la conoscenza dei processi stessi e monitorarli. Il presupposto del process mining, infatti è di scoprire, monitorare e migliorare **processi reali**, estraendo informazioni sulla loro concreta esecuzione. Il termine **reali** nasce dalla

necessità di distinguere un comportamento solamente ipotizzato o supposto – sulla base di diverse informazioni - da quello effettivo del sistema. Se la prima fase della ricerca proposta si è soffermata pertanto sull'analisi dei processi sanitari a partire da modelli realizzati 'a mano', l'attenzione è stata successivamente posta sulle possibili applicazioni delle tecniche di Process Mining al settore sanitario.

4.2 Caratteristiche del Process Mining

Al giorno d'oggi il Process Mining rappresenta un'area di ricerca abbastanza viva, in quanto quasi tutti i tipi di sistemi informativi (WFM, ERP, CRM, SCM e B2B) producono dei log contenenti informazioni relative allo svolgimento delle attività aziendali [20].

Il punto di partenza per qualsiasi tecnica di Process Mining è quindi un log degli eventi (event log o semplicemente log). Tutte le tecniche di Process Mining assumono che sia possibile registrare sequenzialmente eventi in modo che ogni evento si riferisca ad una determinata attività, cioè ad un passo ben definito di un processo, e sia associato ad un particolare case, cioè una determinata istanza di processo. I log possono contenere anche ulteriori informazioni circa gli eventi, quali le risorse - ossia le persone o i dispositivi che eseguono o che danno inizio ad un'attività - i timestamp, o altri attributi associati ad un evento, come il sesso o l'età di un paziente. Gli eventi registrati possono essere di vario tipo: da un paziente che richiede l'esecuzione di un esame, ad un dottore che regola un apparecchio per i raggi X, ad un comune cittadino che fa richiesta per una patente di guida, a un contribuente che sottomette una dichiarazione dei redditi, a un viaggiatore che riceve il numero di un biglietto elettronico. La sfida è cercare di sfruttare i dati in modo significativo, per modellare i processi reali e fornire suggerimenti per una migliore allocazione delle risorse deputate alla loro esecuzione, identificare colli di bottiglia, prevedere problemi, registrare violazioni di regole, raccomandare azioni correttive.

Bisogna sottolineare che, al fine di applicare in modo corretto le tecniche di Process Mining, è necessario tenere in considerazione alcuni aspetti fondamentali.

Prima di tutto, si deve tenere presente che gli **eventi** vanno trattati come entità di prima classe: essi costituiscono infatti il punto centrale di ogni attività di Process Mining. Gli eventi non sono necessariamente memorizzati all'interno di log, ma possono essere memorizzati anche in tabelle relazionali, messaggi, archivi di posta elettronica ed altre sorgenti informative, e da questi estratti sotto forma di event log [71]. La qualità di un risultato di una tecnica di Process Mining è

strettamente legata alla qualità dei dati di ingresso: di conseguenza, i log sono di primaria importanza nell'ambito dei sistemi informativi che supportano i processi da analizzare.

Un secondo aspetto da considerare è che le tecniche di process mining devono essere applicate con il chiaro fine di rispondere a determinate domande: senza un obiettivo concretamente definito è infatti molto difficile estrarre dati ed eventi significativi.

Inoltre, ogni modello di processo è costituito da una struttura portante, chiamata flusso di controllo. I costrutti di base per la modellazione del flusso di lavoro, chiamati '*workflow pattern*' - supportati da tutti i linguaggi convenzionali quali BPEL, diagrammi UML etc - sono la sequenza, il parallelismo (AND split/ join), i punti di scelta (XOR split/join) e i cicli. Al di là dei pattern di base è desiderabile supportare anche gli OR split/join poiché permettono di dare una rappresentazione compatta alle decisioni inclusive e ai punti di sincronizzazione parziale.

Un altro aspetto da tenere in considerazione è che il modello di processo estratto dal log degli eventi può coprire vari aspetti: organizzativo, temporale, legato ai dati, alle risorse etc. Per tale motivo, l'individuazione della relazione tra gli eventi nel log e gli elementi nel modello rappresenta il punto di partenza per diversi tipi di analisi: gli eventi vanno precisamente collegati alle istanze del processo (*event correlation*). In alcuni casi stabilire questa relazione può non essere banale, poiché, ad esempio, un evento potrebbe riferirsi a due attività differenti oppure tale collegamento potrebbe non essere chiaro. Queste ambiguità devono essere rimosse grazie alla conoscenza del dominio applicativo al fine di interpretare correttamente i risultati del Process Mining.

Bisogna inoltre tener presente che i modelli estratti dai log di eventi forniscono viste della realtà, le quali costituiscono un'astrazione utile del comportamento catturato nel log. Esistono molteplici viste utili, scelte in base alle richieste dei vari *stakeholder*. Di fatto, i modelli derivati dai log possono essere interpretati come '*mappe*', che possono focalizzarsi su diversi aspetti (flusso di controllo, flusso dei dati, tempi, risorse, costi, ecc.) e mostrarli a diversi livelli di granularità e precisione, in base al livello di scelte (strategiche, tattiche, operative) da effettuare deciso dai vari stakeholder. L'interpretazione dei modelli come mappe consente inoltre di enfatizzare determinati passi o aspetti del processo (utilizzando ad esempio colorazioni differenti in base all'importanza) o addirittura eliminare le attività del processo ritenute poco significative. Queste considerazioni mostrano che è fondamentale scegliere la giusta rappresentazione e dimensionarla in base agli utilizzatori finali.

Altro aspetto di vitale importanza è la considerazione del Process Mining come processo continuo: data la natura dinamica dei processi, non è consigliabile vedere il Process Mining come un'attività eseguita una sola volta. L'obiettivo infatti non è la creazione di modelli fissi, ma quello di infondere vita nei modelli di processo così da incoraggiare utenti e analisti nell'osservarli di giorno in giorno.

In quest'ottica il Process Mining deve essere considerato come un processo continuo capace di fornire informazioni utili su diverse scale temporali: minuti, ore, giorni, settimane, mesi, anni etc.

Alla luce di quanto affermato, è possibile sostenere che il Process Mining consente varie prospettive di utilizzo, di seguito sintetizzate:

- **Prospettiva del flusso di controllo:** si focalizza sull'ordine delle attività. L'obiettivo di questa prospettiva consiste nel trovare una buona caratterizzazione di tutti i possibili percorsi in un processo. Il risultato, tipicamente, è espresso in termini di una rete di Petri o di altri formalismi (come per esempio EPC, BPMN o diagrammi UML delle attività).
- **Prospettiva dell'organizzazione:** si focalizza sulle informazioni che riguardano le risorse presenti all'interno dei log, ovvero attori (persone, sistemi, ruoli o dipartimenti) del processo, con le rispettive relazioni. L'obiettivo è quello di strutturare l'organizzazione classificando le persone in base ai ruoli che svolgono e alle unità organizzative, oppure di costruire una rappresentazione della social network dell'organizzazione.
- **Prospettiva dell'istanza:** si concentra sulle proprietà di un case. Ovviamente, un case può essere caratterizzato dal suo percorso nel processo oppure dagli attori che operano nello stesso. Tuttavia, i case possono essere definiti anche attraverso valori assunti da altri tipi di dati. Ad esempio, se un case rappresenta un paziente che deve effettuare un esame, potrebbe essere interessante conoscerne il sesso o l'età.
- **Prospettiva del tempo:** è legata a quando un evento è accaduto e alla sua frequenza. Quando agli eventi sono associati dei timestamp, è possibile individuare colli di bottiglia, misurare i livelli di un servizio, monitorare l'uso delle informazioni temporali, delle risorse, e predire il tempo restante per il completamento di un'istanza.

4.3 Le tecniche di Process Mining

In Fig.4.3.1 viene illustrata una famosa rappresentazione delle caratteristiche, delle applicazioni, e delle tipologie di process mining, che verranno dettagliate nel seguito di questo paragrafo.

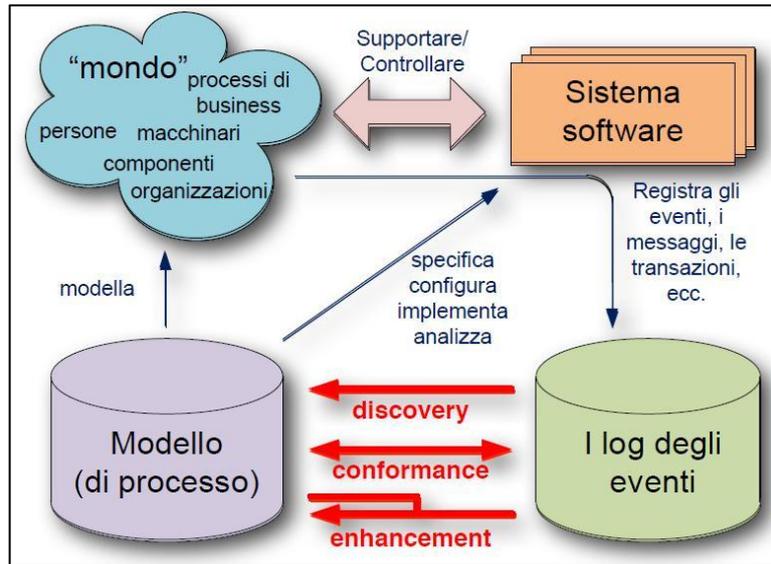


Fig.4.3.1: Process Mining Framework

Fondamentale è chiarire il concetto di event log. In generale, un **log di eventi** è una fonte di dati in cui sono stati memorizzati tutti gli eventi che caratterizzano un certo sistema, con lo scopo di poterne tracciare il comportamento e analizzare errori o comportamenti non corretti. Tipicamente vengono memorizzati eventi sequenziali che si riferiscono ad una ben definita attività e possono essere riferiti ad un determinato caso. Un caso è una particolare istanza di un processo, ossia una particolare sequenza di eventi. La teoria classica del Process Mining richiede che ogni log contenga le seguenti informazioni:

1. Identificativo ID del caso;
2. Identificativo ID dell'evento;
3. Collocazione temporale di un evento all'interno del caso (timestamp).

Per fornire una definizione formale di log di eventi, si consideri un insieme A di eventi, da cui si estragga una traccia S , ossia una sequenza ben definita di eventi. Sia L log di eventi l'insieme delle tracce S . Si definiscano una traccia $S_1 = \langle A, B, C, D \rangle$, che rappresenta il percorso seguito nel caso 1, una traccia $S_2 = \langle A, C, B, D \rangle$ che rappresenta un secondo caso, chiamato caso 2, e una terza traccia $S_3 = \langle A, E, D \rangle$ che rappresenta il caso 3. I seguenti casi possono essere caratterizzati da una determinata molteplicità. In generale un log degli eventi L caratterizzato dalle tracce descritte, potrà essere dunque definito come:

$$L = [(A, B, C, D)^m; (A, C, B, D)^n; (A, E, D)^p],$$

dove m , n e p rappresentano la molteplicità rispettivamente delle tracce S_1 , S_2 e S_3 .

Un esempio di un frammento di log di eventi è mostrato in figura 4.3.2.

Case ID	Event ID	Properties	
		Timestamp	Activity
1	7785621	30-11-2011:08.27	First Hospital Admission
	7785624	2-12-2011:13.24	Preoperative Screening
	7785625	4-12-2011:8.30	Laparoscopic Gastrectomy
	7785631	4-12-2011:13.00	Nursing
2	7785631	1-12-2011:11.00	Preoperative Screening
	7785634	2-12-2011:13.24	Laparoscopic Gastrectomy
	7785638	2-12-2011:16.35	Nursing
	7785640	3-12-2011:13.00	Laparoscopic Gastrectomy
	7785641	3-12-2011:15.00	Nursing
	7785661	4-12-2011:9.00	First Hospital Admission
3	7785654	7-12-2011:10.00	First Hospital Admission
	7785642	8-12-2011:13.24	Laparoscopic Gastrectomy
	7785655	9-12-2011:16.35	Nursing

Fig.4.3.2: Frammento di log di eventi

Come anticipato, la qualità dei risultati di una tecnica di Process Mining è strettamente legata alla qualità dei dati di ingresso contenuti nei log, che deve essere esaminata preliminarmente all'analisi. Esistono alcuni criteri, di seguito elencati, per valutare la qualità dei dati a disposizione [15]:

- **Attendibilità:** riguarda la possibilità di affermare con certezza che gli eventi registrati siano effettivamente accaduti e che gli attributi degli eventi siano corretti;
- **Completezza:** una volta fissato un preciso contesto, nessun evento dovrebbe mancare;
- **Semantica:** deve essere ben chiara la definizione di ogni evento;
- **Sicurezza:** i dati legati agli eventi devono essere registrati rispettando criteri di privacy e sicurezza.

In base al soddisfacimento di tutti o di alcuni dei suddetti criteri, i log di eventi vengono classificati in 5 livelli: eccellenti, ottimi, buoni, scarsi e poveri.

All'interno dei log, fondamentali sono gli eventi, ovvero le generiche attività che compongono i processi. La registrazione degli eventi prevede i seguenti elementi:

1. Ogni evento è univocamente definito in una determinata fase del processo;
2. Ogni evento è riferito ad un case, cioè ad una istanza del processo;
3. Per ogni evento è specificato un originator, cioè il personale o le risorse impiegate per lo svolgimento dell'attività;
4. Per ogni evento è indicato un timestamp, cioè un riferimento temporale indicante il momento in cui viene effettuata l'attività;

5. Per ogni evento possono essere presenti degli attributi, in grado di fornire informazioni aggiuntive sulle caratteristiche delle attività svolte.

Gli eventi all'interno del log devono necessariamente essere caratterizzati da una sequenzialità, devono cioè essere ordinati. L'ordinamento, spesso identificato come una sequenza temporale grazie alla indicazione dei timestamp, consente di ricavare importanti informazioni sulle relazioni causali presenti tra le diverse attività.

Si definisce quindi **flusso** un insieme di eventi caratterizzati da una sequenzialità. Nell'ambito del Process Mining il concetto di flusso è paragonabile al concetto di case o istanza del processo. Un'istanza del processo rappresenta una singola esecuzione del processo analizzato; le istanze dello stesso processo sono indipendenti tra loro, per cui è possibile che siano in esecuzione simultaneamente. Ogni istanza del processo è univocamente identificata da un codice che può cambiare a seconda del processo analizzato (ad esempio può essere un numero d'ordine, un codice fiscale, il numero di una cartella clinica, ecc.).

Le principali applicazioni del Process Mining includono:

1. Deduzione (automatica) di processi (**Discovery**);
2. Estrazione di un modello di processo a partire da un log;
3. Verifica di conformità (**Conformance Checking**);
4. Monitoraggio di eventuali discrepanze tra un modello e un log;
5. Individuazione di reti sociali (Social Network) e organizzative;
6. Costruzione automatica di modelli di simulazione;
7. Estensione e revisione di modelli (**Enhancement**);
8. Predizione delle possibili future evoluzioni di un'istanza di processo;
9. Raccomandazioni su come operare sulla base di dati storici.

In riferimento all'obiettivo di ottenere un modello che rappresenti il processo in esame, il Discovery, la Conformance Checking e L'Enhancement sono gli aspetti di maggiore interesse. La fig. 4.3.3 rappresenta come essi siano tra loro strettamente collegati: le tecniche di discovery consentono infatti di ricavare, a partire dal log di eventi, un modello del processo. Le tecniche di conformance checking matchano il modello realizzato con il log stesso, al fine di verificare la compliance tra i due. Le tecniche di enhancement, infine, puntano ad ottenere un modello migliorativo rispetto a quello inizialmente ricavato. E' importante sottolineare che in genere il log a disposizione per un determinato processo viene splittato in due: un primo frammento viene impiegato per la fase cosiddetta di 'training', ovvero di discovery del modello; il secondo frammento è usato invece per il 'test', ovvero in fase di conformance e di enhancement.

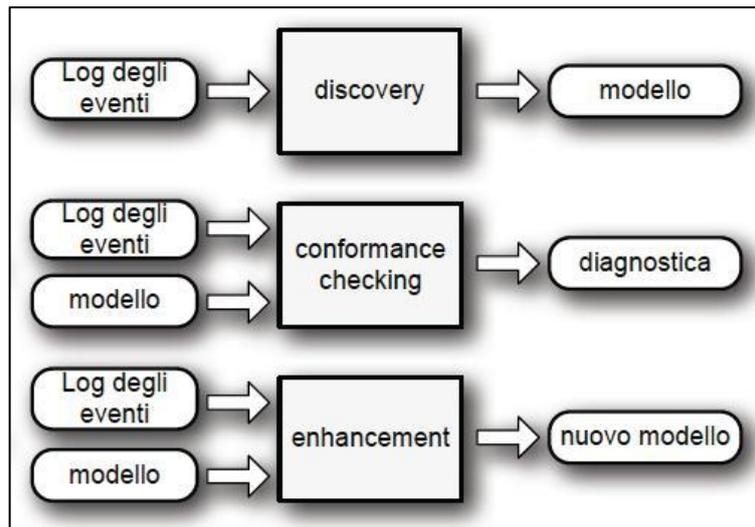


Fig.4.3.3: Principali tecniche di Process Mining

Uno dei primi obiettivi del Process Mining è dunque la realizzazione di un modello grafico che descriva in maniera esaustiva ciò che realmente accade durante l'esecuzione di un processo: la tecnica che consente l'identificazione di tale modello è chiamata **Model Discovery** [72]. Solitamente, come spiegato in [72], i modelli realizzati a partire dagli event log sono rappresentati secondo la logica dei workflow; la tipologia di modello più ricorrente è la Rete di Petri, definita nel capitolo 3.

Altra tecnica di Process Mining è la **Conformance Checking** [73]. Con questa tecnica, un modello di processo preesistente è confrontato con le informazioni, relative allo stesso processo, estratte dal log. La conformance checking può essere applicata per verificare la conformità tra modelli e realtà (rappresentata dai log di eventi), dati in input un log e un modello. L'output consiste in informazioni diagnostiche che mostrano le differenze tra il modello ed il log. Tale tecnica può essere applicata a diversi tipi di modelli: procedurali, organizzativi, dichiarativi, regole di business, leggi, etc. e, come accennato precedentemente, può essere un utilissimo ausilio anche per la valutazione della qualità dei modelli derivati tramite le tecniche di discovery.

Altra applicazione del Process Mining è l'**Enhancement** [15]. In tal caso, l'idea è quella di estendere o migliorare un modello di processo esistente utilizzando le informazioni contenute nei log. Mentre la conformance checking misura quanto un modello è allineato con ciò che accade nella realtà, l'Enhancement si propone di cambiare o estendere il modello preesistente. Ad esempio, considerando i timestamp in un event log, è possibile estendere il modello per mostrare colli di

bottiglia, tempistiche nell'erogazione di un servizio, tempi di produttività e frequenze. L'output di tali tecniche è quindi una versione estesa e migliorata del modello di input.

4.4 Gli strumenti per il Process Mining

Nel corso degli ultimi venti anni, sono stati sviluppati moltissimi strumenti, sia accademici (EMiT, Little Thumb, InWoLve, Process Miner, MinSoN) sia commerciali (ARIS PPM, HP BPI, ILOG JViews) per l'estrazione di informazioni dai log degli eventi (come i “*log transazionali*” dei sistemi ERP o gli “*audit trails*” dei sistemi WFM) al fine di effettuare analisi di Process Mining [74].

Tali strumenti, per quanto performanti a livello di funzionalità sviluppate, utilizzavano diversi formati per la lettura e/o la registrazione dei file log, oltre ad avere modalità differenti nella presentazione dei risultati: era dunque estremamente complicato impiegarli per effettuare analisi su uno stesso set di dati e comparare risultati ottenuti. Era altresì problematico interfacciare strumenti differenti per effettuare analisi congiunte.

Per superare questo tipo di problemi, è stata realizzata una piattaforma estensibile in grado di supportare una grande varietà di tecniche di Process Mining sotto forma di plug-in, sviluppati in linguaggio Java e liberamente scaricabili. La piattaforma, denominata ProM⁸ [75], è distribuita tramite licenza CPL (Common Public Licence) ed è quindi open source, permettendo così a Ricercatori e Sviluppatori di contribuire al suo sviluppo implementando costantemente nuovi plug-in.

La schermata di avvio del tool per Process Mining ProM è rappresentata in figura 4.4.1.

La funzione Import, visibile in alto a destra, consente di iniziare qualunque analisi si voglia applicare importando il log di eventi e/o un modello XML di un processo.

⁸ <http://www.promtools.org/prom6/>

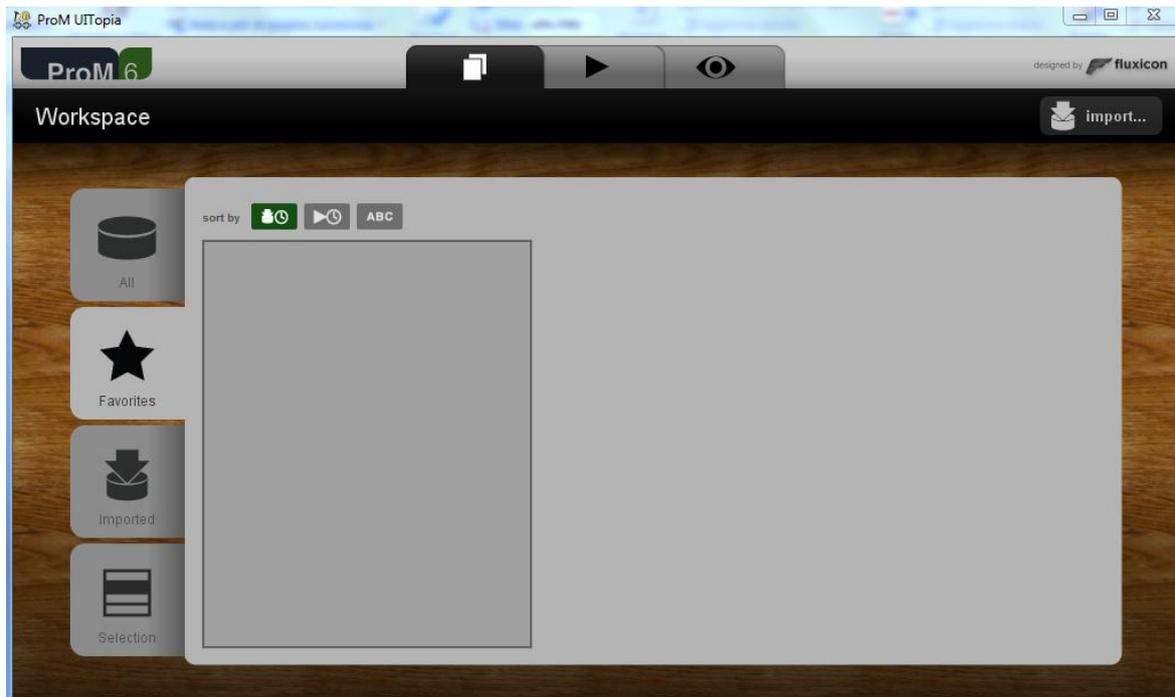


Fig.4.4.1: Tool per Process Mining ProM

Dal momento che ogni sistema informativo registra il log in un formato differente, è stato sviluppato un formato standard per i file log da importare in ProM, in cui è possibile convertire i file originali. Il file utilizzato è in pratica un XML il cui formato è stato deciso in base alla struttura dei log uscenti dai sistemi ERP e/o WFM.

La struttura del file XML che compone il log degli eventi utilizzato dal software ProM è rappresentata in figura 4.4.2. L'elemento iniziale è chiamato *WorkflowLog* e contiene due elementi facoltativi, *Data* e *Source*, ed un certo numero di elementi obbligatori, detti *Process*. L'elemento *Data* permette la registrazione arbitraria di una serie di dati testuali, tramite una lista di elementi o *Attribute*. L'elemento *Source* può essere usato per registrare informazioni riguardo il sistema informativo dal quale il log è stato prelevato. L'elemento *Process* si riferisce ad uno specifico processo registrato nel log dal sistema informativo; dato che tali sistemi controllano diversi processi, nel log potremmo avere più elementi di tipo *Process*. Per ogni processo, possiamo avere più process instance, cioè più casi rappresentativi delle varie esecuzioni del processo. All'interno delle singole istanze del processo abbiamo gli eventi, registrati come *AuditTrailEntry*. Ogni *AuditTrailEntry* è associato ad un'attività (*WorkflowModelElement*), caratterizzata da uno stato (*EventType*), ad esempio start, complete, schedule, ecc. Per ogni attività possono essere registrate anche informazioni

facoltative che rendono più completo il log: tra queste troviamo il tempo di esecuzione dell'attività (*Timestamp*) e le eventuali risorse impiegate per il suo svolgimento (*Originator*).

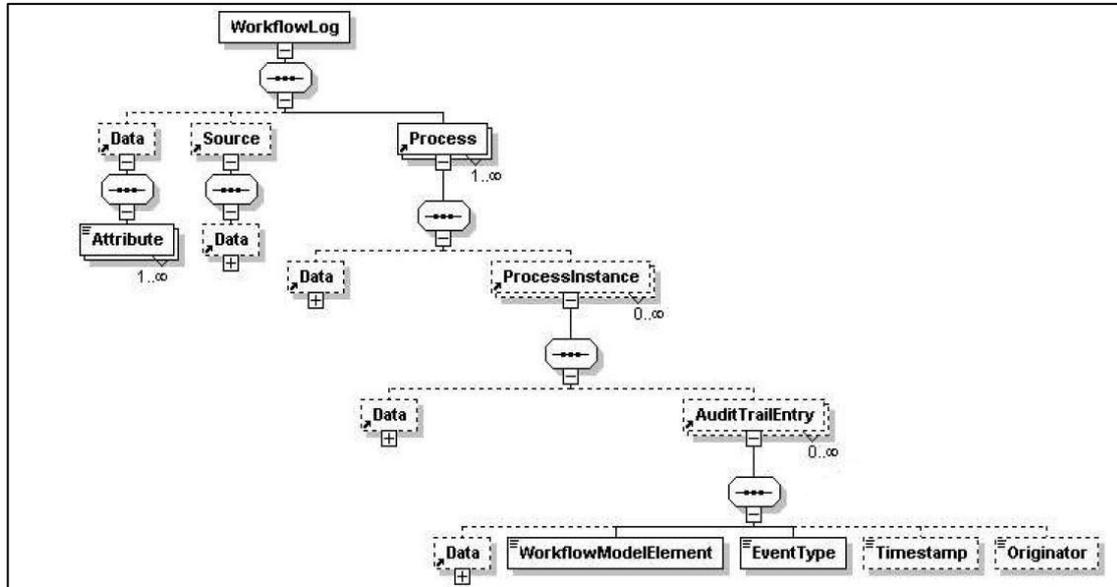


Fig.4.4.2: Formato del File MXML

La piattaforma consente l'interazione tra un numero elevatissimo di plug-in, che tipicamente rappresentano l'implementazione di un particolare algoritmo di mining. Una volta sviluppato il plug-in, esso può essere facilmente implementato nella piattaforma senza doverla modificare (cioè senza dover ricompilare il codice), ma semplicemente aggiungendo il nome del plug-in al file .in al quale la piattaforma fa riferimento.

In Fig.4.4.3 viene schematizzata la struttura della piattaforma, con il log e i diversi plug-in.

Come anticipato, il tool inizia le analisi con la lettura del file XML tramite il componente **Log Filter**, in grado di interfacciarsi con un set molto grande di dati e di ordinare gli eventi presenti in ogni istanza del processo in base al relativo timestamp, ancora prima di effettuare il mining. Qualora il log non sia provvisto di timestamp, il Log Filter mantiene invariato l'ordine delle attività così come compare nel file XML in ingresso. Anche gli **Import Plugin** possono fungere da punto di partenza, consentendo l'upload di vari modelli che variano dalle reti di Petri fino a vere e proprie formule logiche. I **Mining Plugin**, partendo dai dati presenti nel log, consentono di effettuare il mining, ricavando così un modello di processo a partire dai dati a disposizione.

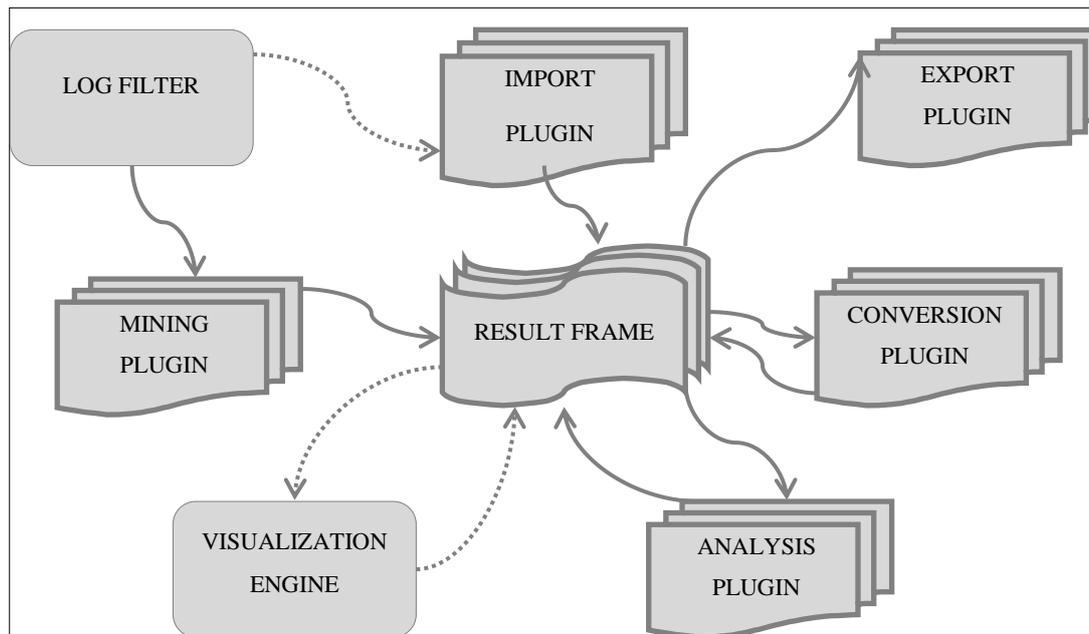


Fig.4.4.2: Architettura della piattaforma ProM

Tipicamente, i risultati del mining danno origine ad una serie di visualizzazioni (*Visualization Engine*), come reti di Petri, EPC, Social Network, che possono poi essere direttamente analizzate con gli *Analysis Plugin*, e/o convertite in altri formati (ad esempio da una rete di Petri ad un modello EPC, o viceversa), tramite i *Conversion Plugin*.

Si individuano cinque tipologie principali di plug-in:

1. **Algoritmi di Mining:** il software implementa ed utilizza molti algoritmi di mining per la costruzione dei modelli del processo. Tra questi troviamo:
 - **α -algorithm:** estrae una rete di Petri dal log di eventi;
 - **Tshingua- α algorithm:** tale algoritmo è strettamente correlato all' α -algorithm, ma utilizza un approccio differente, in quanto utilizza i timestamp presenti nel log per costruire una rete di Petri;
 - **Heuristics miner:** basa la scoperta dei modelli del processo sulle relazioni causali intercorrenti tra le singole attività;
 - **Genetic algorithm:** simula il processo evolutivo dei sistemi biologici, consentendo la definizione di nuovi modelli partendo dalle informazioni contenute nei modelli precedentemente scoperti. L'output fornito è una rete euristica, che può essere convertita in una rete EPC o in una rete di Petri;
 - **Multi-phase mining:** utilizza una serie di algoritmi eseguiti in successione producendo una serie di rappresentazioni intermedie.

- **Social Network miner:** utilizza le informazioni sugli originator del log per generare grafici ed informazioni sulle interazioni tra le diverse risorse impiegate nell'esecuzione del processo.
2. **Export Plugin:** consentono il salvataggio delle reti e delle informazioni ottenute dall'analisi del log in diversi formati (PNML, CSV, ecc.), per l'utilizzo in altri applicativi.
 3. **Import Plugin:** consentono l'importazione di grafici e dati da altri software (ad esempio reti EPC prodotti dal software ARIS PPM).
 4. **Analysis Plugin:** consentono di effettuare analisi specifiche su dati e grafici ottenuti. Tra questi troviamo:
 - **Case data extraction:** consente di estrarre dati numerici sugli attributi associati ai diversi casi del processo, per poi esportarli in formato CSV;
 - **LTL Checker:** verifica se il log soddisfa alcune regole logiche (Linear Temporal Logic);
 - **Conformance Checker:** permette l'analisi di una rete di Petri al fine di verificare se il log è correttamente riprodotto nel modello;
 - **Performance analysis:** analizza i timestamp presenti nel log per individuare i possibili colli di bottiglia del flusso di lavoro;
 - **Trace clustering:** permette la creazione di sub-log contenenti gruppi omogenei di casi.
 5. **Conversion:** consente la conversione tra differenti formati (per esempio da una rete di Petri ad una rete EPC, e viceversa).

Dopo questa panoramica generale sul tool per Process Mining ProM, il capitolo 7 illustrerà una concreta applicazione dei concetti fin qui esposti, attraverso la descrizione di un particolare caso di studio.

5. MODELLAZIONE DEI PROCESSI SECONDO IL LINGUAGGIO DICHIARATIVO

5.1.Introduzione

Come illustrato nel precedente paragrafo, attraverso le tecniche di process mining è possibile migliorare l'analisi ed il monitoraggio dei processi reali [15].

Tale tipologia di monitoraggio, tuttavia, mette spesso in luce che i processi reali, soprattutto quelli inerenti il contesto sanitario, possono essere molto più variabili di quanto previsto. Il punto di partenza delle tecniche di process mining è costituito dai log di eventi: ciascun evento in un log si riferisce ad una attività (ossia uno step preciso all'interno del processo) ed è legato ad un particolare caso (ossia una particolare istanza di processo, rappresentata ad esempio dal percorso del singolo paziente). Le attività riferite ad una istanza di processo sono ordinate nel tempo e possono essere viste come lo svolgimento della particolare istanza, acquisendo nel complesso la denominazione di traccia. I log di eventi possono poi contenere informazioni aggiuntive come le risorse (persone o strumenti) che eseguono le attività, il timestamp di ciascun evento, o attributi riferiti a ciascun caso (esempio caratteristiche particolari dei pazienti: sesso, età etc).

Tipicamente, tre tecniche principali di process mining vengono applicate per analizzare i log di eventi: process discovery, conformance checking e model enhancement, illustrati nel paragrafo precedente.

Il model discovery restituisce un modello del processo a partire da un log di eventi, secondo diverse tecniche. Numerose applicazioni, tuttavia, mostrano che applicando tecniche di discovery basate sulla logica sequenziale delle reti di Petri, possono essere ricavati dei modelli di processo di facile comprensione qualora il modello in esame sia semplice, procedurale e caratterizzato da un elevato grado di sequenzialità. Qualora invece il processo oggetto di studio presenti un alto grado di variabilità e non si presenti come strutturato e costituito da una sequenza ripetitiva e ben definita delle attività, le tecniche di discovery basate sulle reti di Petri possono generare modelli di difficile interpretazione, denominati in gergo 'Spaghetti like' – fig. 5.1.1 - [22].

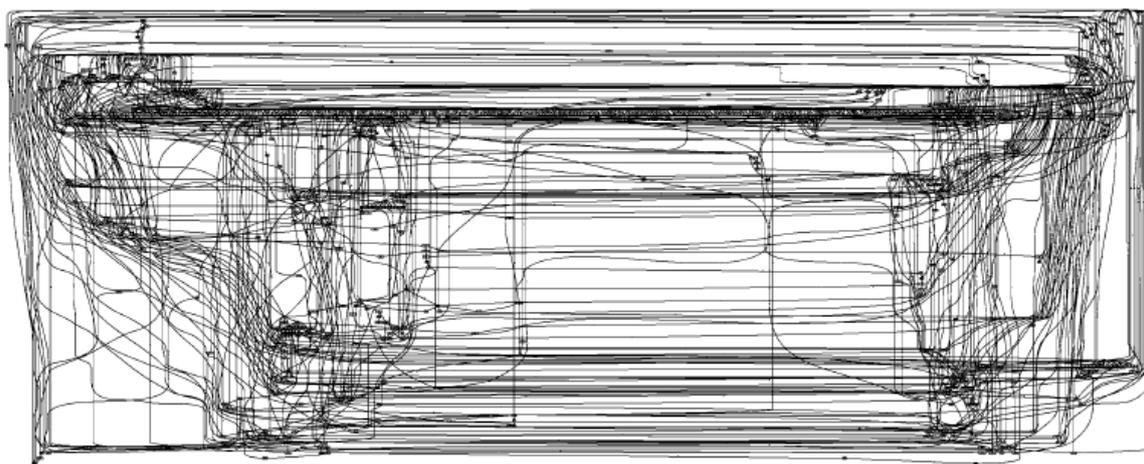


Fig.5.1.1: Esempio di modello 'Spaghetti like'

Modelli come quello mostrato risultano, evidentemente, di difficile comprensione e non si prestano facilmente ad essere analizzati.

Il rischio di ottenere modelli del tipo 'spaghetti', è ancora più forte nel settore sanitario rispetto ad altri. Come sintetizzato da Rebuge ed Al. [76], infatti, i processi diagnostico/terapeutici sono fortemente dinamici, complessi, interdisciplinari e ad hoc a seconda del caso clinico. Inoltre, a causa della complessità del contesto in cui si sviluppano, coinvolgono numerose variabili che possono essere diversamente gestite a seconda della condizione dello specifico paziente.

Da quanto esposto è emersa la necessità di studiare linguaggi di rappresentazione alternativi al linguaggio procedurale caratterizzante le reti di Petri. Sono state pertanto approfondite le tecniche dichiarative di rappresentazione dei processi.

Come suggerito da Muylar e Al. [77], i linguaggi dichiarativi consentono una maggiore flessibilità nella modellazione dei processi rispetto a quelli imperativi. Questi ultimi sono infatti linguaggi che prevedono di specificare tutte le possibili modalità di esecuzione di un determinato processo, producendo modelli 'chiusi', ovvero rigidi nella sequenza delle attività che li compongono. Se i linguaggi imperativi si prestano a descrivere processi ben strutturati in ambienti stabili, possono tuttavia condurre alla rappresentazione di modelli non leggibili e complessi in ambienti turbolenti e fortemente variabili. Di converso, i linguaggi dichiarativi non modellano l'esecuzione dei processi includendo tutti i comportamenti possibili, ma soltanto quelli non consentiti. In questo modo, consentono la rappresentazione di diversi comportamenti, dal momento che nell'esecuzione del processo tutto è consentito a meno che non sia stato esplicitamente vietato [78]. Ciò è ottenuto mediante una rappresentazione del processo come un set di regole, in modo tale che esso possa essere eseguito in numerosi modi possibile purché tali regole non vengano violate. Data la

dinamicità dei processi sanitari, si è pensato di rappresentarli mediante l'impiego del linguaggio dichiarativo, utilizzando il tool Declare [79], i cui principi base sono illustrati nel seguente paragrafo.

5.2 Modellazione dichiarativa dei processi: il tool Declare

Declare⁹ è un linguaggio dichiarativo introdotto da M. Pesic e W. Van der Aalst [80] e supportato da un apposito tool che include un designer, un workflow engine, un gestore di worklist (worklist handler) e diverse funzioni implementate per analizzare il processo.

Il linguaggio di Declare è caratterizzato da un set di templates, definiti come entità astratte che definiscono classi parametrizzate di regole specificate attraverso la semantica della Logica Temporale Lineare (LTL) [81]. A ciascun template è associata una corrispondente ed univoca rappresentazione grafica.

I template possono essere classificati in quattro gruppi: template di esistenza, di relazione, di relazione negativa e di alternativa. I template di esistenza coinvolgono una sola attività e definiscono la cardinalità o la posizione consentita di una attività in una istanza di processo. I template di relazione e di relazione negativa definiscono dipendenze/dipendenze negative tra due attività. I template di alternativa definiscono relazioni alternative tra le attività.

I Constraint sono delle istanze concrete dei template ed acquisiscono la rappresentazione grafica del template corrispondente.

Un modello Declare è un set di constraint che legano le attività di un processo.

Di seguito viene mostrata la rappresentazione grafica della schermata principale del tool Declare, e dei principali template con il relativo significato secondo la logica dichiarativa.

⁹ <http://www.win.tue.nl/declare/>

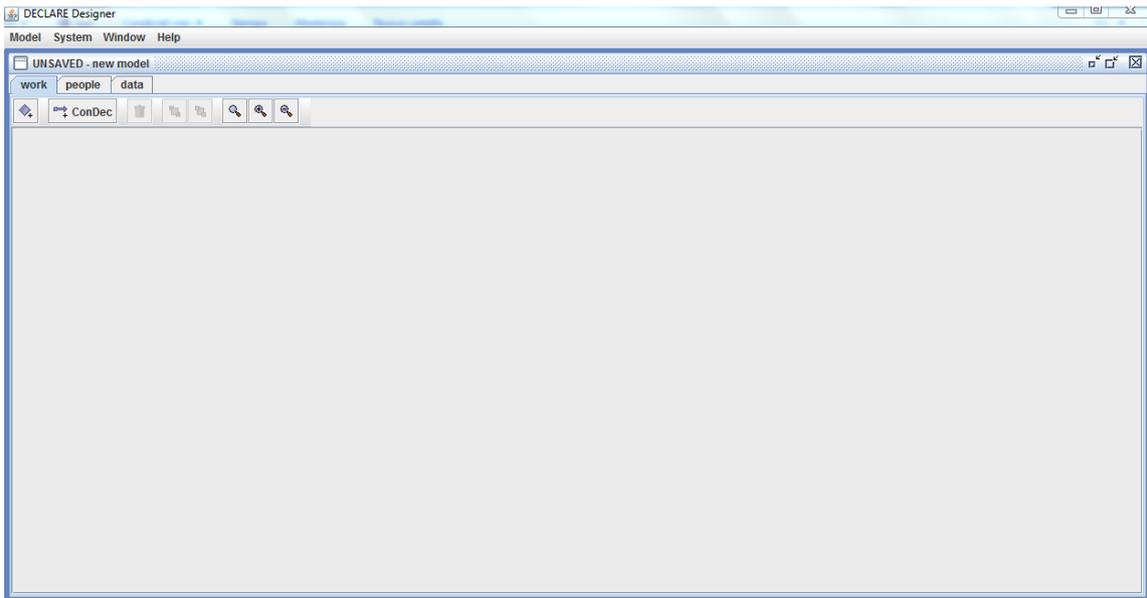


Fig.5.2.1: Il Declare Designer

Template di esistenza

Existence: indica che l'attività A è eseguita almeno una volta in uno dei casi che compongono il log.

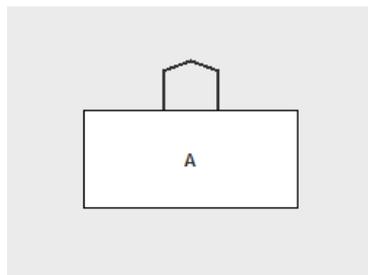


Fig.5.2.2: Existence

Absence: indica che l'attività A non viene mai eseguita all'interno del log di eventi.

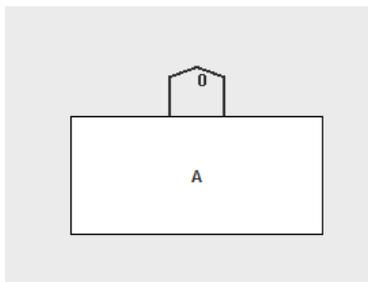


Fig.5.2.3: Absence

Template di Relazione

Responded Existence: indica che se un'attività A è eseguita (almeno una volta) all'interno di una traccia, allora anche l'attività B deve essere eseguita all'interno della stessa traccia.



Fig.5.2.4: Responded Existence

Co-existence: se un'attività A è eseguita almeno una volta in una determinata traccia, allora anche B deve essere eseguita prima o dopo A.

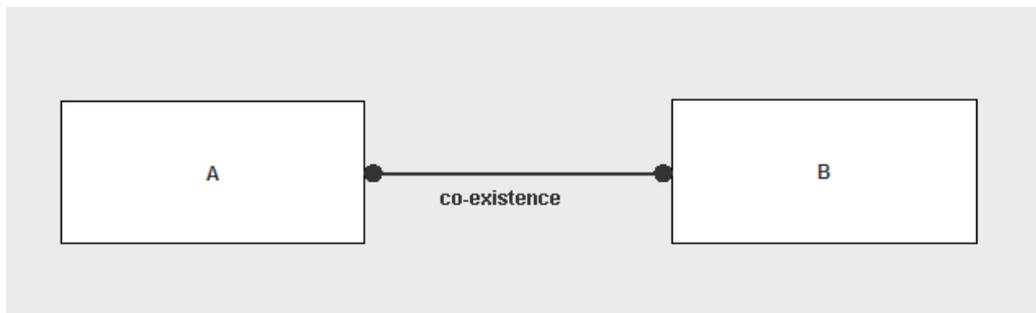


Fig.5.2.5: Coexistence

Response: se una attività A viene eseguita, B deve essere eseguita dopo di essa.

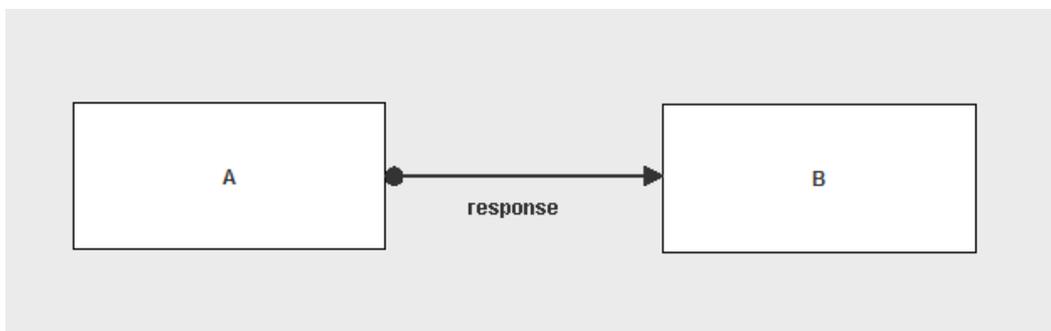


Fig.5.2.6: Response

Precedence: L'attività B deve essere preceduta da A e non può verificarsi finchè A non è stata eseguita.

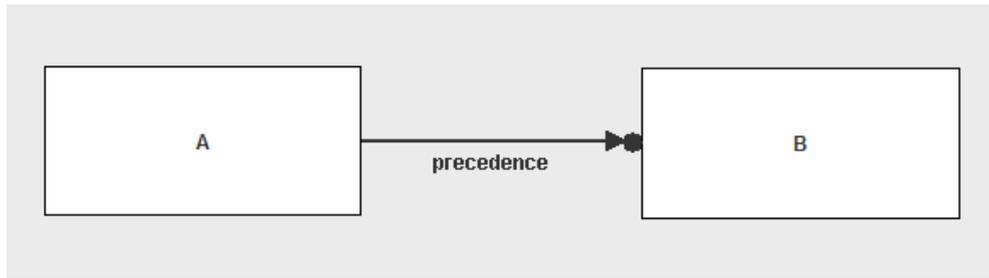


Fig.5.2.7: Precedence

Succession: Dopo A deve essere eseguita almeno una volta B. B deve inoltre essere preceduta da A e non può essere eseguita finché non è stata eseguita A.

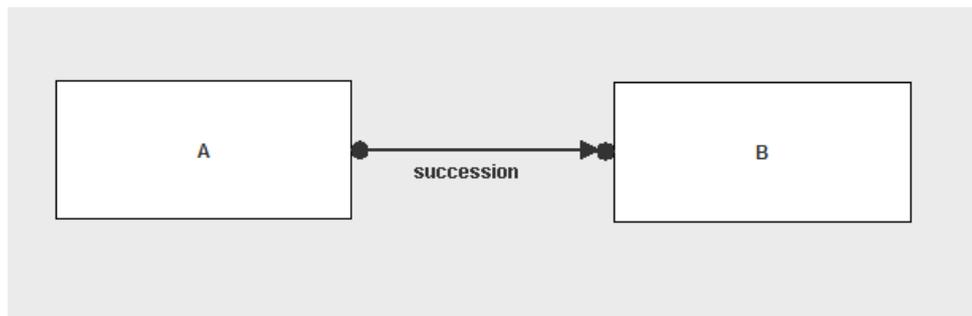


Fig.5.2.8: Succession

Varianti dei template di relazione sono quelli di relazione alternata o diretta: i primi prevedono l'intersorsi di altre attività tra A e B nel soddisfacimento delle relazioni descritte; i secondi prevedono che A e B siano 'vicine', ovvero non vengono eseguite altre attività cronologicamente interposte. Un esempio è dato dalla chain response in figura seguente:

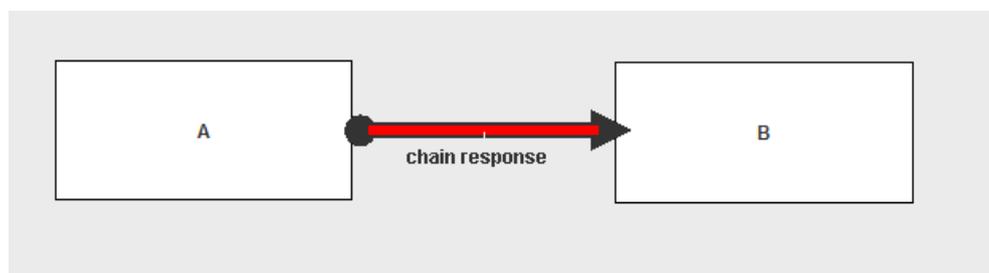


Fig.5.2.9: Chain Response

Tale template indica che dopo A viene immediatamente eseguita B.

Template di Relazione Negativa

Not coexistence: Se A viene eseguita, B non può essere eseguita e viceversa.

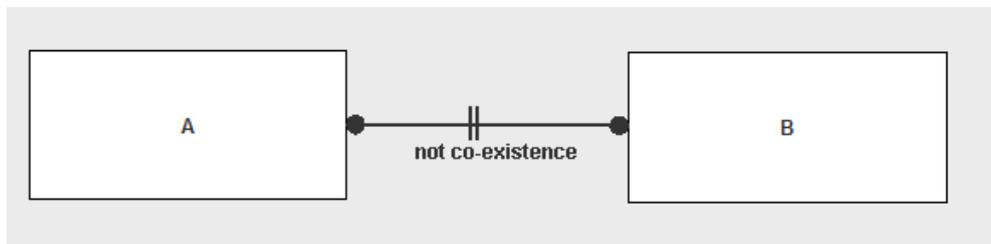


Fig.5.2.10: Not Coexistence

Not succession: Prima di B non può essere eseguita A e dopo A non può essere eseguita B.

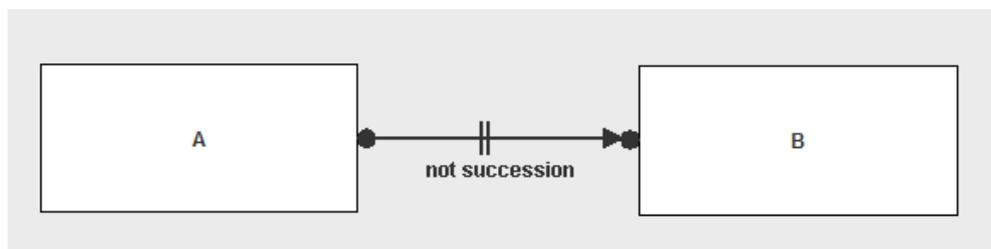


Fig.5.2.11: Not Succession

Template di Alternativa

Choice: Almeno una attività tra A e B deve essere eseguita.

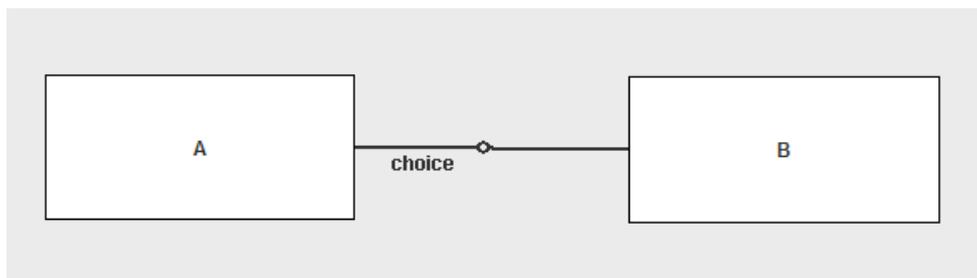


Fig.5.2.12: Choice

Exclusive Choice: una delle due attività a scelta tra A e B può essere eseguita, ma non entrambe.

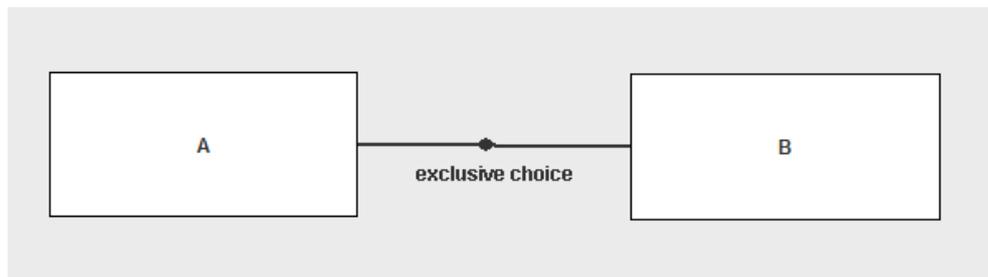


Fig.5.2.13: Exclusive Choice

I modelli realizzati in Declare prevedono dunque la definizione delle attività che costituiscono il processo e dei constraint che ne definiscono le relazioni. Essi consentono pertanto di rappresentare soltanto le regole alla base del processo, e non tutte le sue possibili esecuzioni, dando la possibilità di ottenere modelli strutturalmente più semplici e di più agevole interpretazione.

Illustrati i principi base del Process Mining e del linguaggio dichiarativo, nel seguente capitolo viene proposta una metodologia che prevede la loro combinazione nell'analisi dei processi sanitari, con riferimento ad un particolare caso di studio.

6. PROCESS MINING DICHIARATIVO NEL SETTORE SANITARIO: UNA NUOVA PROPOSTA METODOLOGICA

6.1 Premesse

I processi sanitari si articolano in processi strettamente medici – riguardanti le attività diagnostico/terapeutiche svolte per curare i pazienti – ed in processi organizzativi, principalmente legati alla gestione degli aspetti gestionali/burocratici [68]. I processi sanitari di carattere organizzativo vengono generalmente gestiti nel rispetto di procedure ben definite, mentre i processi diagnostico-terapeutici devono essere abbastanza flessibili da adattarsi alla variabilità del contesto sanitario ed allo spazio di decisionale dei medici nell'esercizio della loro professione. Come sintetizzato da Rebuge ed Al. [76], i processi diagnostico/terapeutici sono infatti fortemente dinamici, complessi, interdisciplinari e modellati a seconda del caso clinico che si affronta. Inoltre, a causa della complessità del contesto in cui si sviluppano, coinvolgono numerose variabili che possono essere diversamente gestite a seconda della condizione dello specifico paziente.

I processi diagnostico-terapeutici sono generalmente gestiti in accordo alle Linee Guida, con l'obiettivo di tradurre la ricerca in azioni [82] con un elevato grado di efficacia ed efficienza [83]. D'altro canto è stato spesso riscontrato, nel monitoraggio dell'esecuzione dei processi, che esiste una sorta di 'gap' tra tali raccomandazioni e l'effettiva pratica clinica [84] [85], nel senso che le attività effettivamente eseguite differiscono da quelle prescritte dalle Linee Guida.

L'esistenza di un gap tra le prescrizioni dell'Evidence Based Medicine e l'effettiva pratica clinica rappresenta al giorno d'oggi un aspetto rilevante nel contesto del Management Sanitario [85] [86]. Tale gap è stato evidenziato in numerosi campi, come quello dermatologico [87], psichiatrico [88], urologico [89]. In [90], gli Autori hanno investigato le ragioni per le quali i medici non sempre mettono in pratica le 'best evidences': mediante interviste strutturate, essi hanno rilevato che le esperienze personali, gli studi, la formazione, relazioni particolari con i singoli pazienti, aspetti logistici e pratici hanno una forte influenza sulle decisioni cliniche, determinando scostamenti della pratica quotidiana dai processi prescritti. In alcuni casi, le decisioni cliniche possono anche essere influenzate dai costi di una particolare terapia e dalla condizione socioeconomica del paziente [91]. In più, in alcune circostanze le linee guida possono non rispondere in pieno alle esigenze di trattamento di un particolare paziente [92]: ad esempio, un paziente può presentare allergie che gli impediscono l'assunzione dei medicinali prescritti solitamente per curare la patologia da cui è affetto; gli interventi chirurgici, anche se raccomandati dalle linee guida per il trattamento di alcune

patologie, possono essere controindicati per pazienti 'critici'; patologie quali diabete, difetti di coagulazione, problemi cardiaci, possono richiedere ulteriori indagini prima di poter procedere con la terapia prescritta, causando un rallentamento del percorso di cura, e così via. In numerosi casi, pertanto, il gap che emerge è paziente-correlato, poiché i dottori devono adeguarsi alle particolari esigenze richieste dallo specifico caso clinico.

D'altra parte, ci sono anche alcune deviazioni che possono essere attribuite alla difficoltà di mettere in pratica e diffondere i risultati della ricerca clinica, trattandosi di un percorso che richiede un tempo variabile e spesso lungo. Pertanto, si può ritenere che esistono alcune deviazioni che possono influire positivamente sul percorso di cura, altre che invece possono compromettere il recupero del paziente. Di conseguenza, è necessario individuare tali deviazioni ed identificarne le cause così da definire quali di esse possono essere classificate come positive – e pertanto devono essere prese in considerazione per migliorare le prescrizioni delle linee guida – e quali invece devono essere evitate e corrette nella quotidiana pratica clinica. Per poter affrontare la problematica in termini quantitativi e contribuire concretamente all'analisi di tali deviazioni, è necessario disporre di informazioni precise e puntuali in merito all'esecuzione dei processi. In [8], gli Autori sottolineano come numerosi Studi abbiano dimostrato gli effetti positivi dell'impiego dei Sistemi Informativi nel monitorare i processi sanitari. Numerosi altri studi, hanno d'altra parte evidenziato la validità delle tecniche di Process Mining nel soddisfare i medesimi propositi [93], dal momento che consentono di acquisire una profonda conoscenza dei processi passati così da evitare errori nel futuro [94].

Pertanto, una delle sfide nel contesto dell'analisi dei processi sanitari è l'analisi di tale problematica, attraverso delle tecniche che consentano di:

1. misurare quantitativamente il gap tra il processo prescritto e quello svolto nella pratica
2. identificare le incoerenze ricorrenti e maggiormente significative
3. produrre risultati che possano essere facilmente comprensibili per i medici, così da poter individuare con il loro supporto le cause delle deviazioni riscontrate.

Tali deviazioni possono infatti essere, in alcuni casi, legate ad errori nell'esecuzione dei processi, che vanno corretti e monitorati; in altri casi, invece, non è da escludere che possono rappresentare migliorie del processo stesso dovute all'esperienza dei medici, da tenere in considerazione ai fini della ricerca e dell'aggiornamento delle linee guida.

Per eseguire uno studio di questo tipo, tuttavia, devono essere tenuti in considerazione numerosi aspetti, legati alla nota e ampiamente menzionata complessità del settore sanitario.

Prima di tutto, occorre modellare i processi tenendo conto della loro dinamicità: modifiche delle terapie, delle medicine, dei protocolli, possono invalidare un processo in corso e richiedere

l'esecuzione di azioni riparatorie volte a modificare il processo stesso [95]. E' pertanto necessario adoperare un linguaggio di modellazione dei processi che sia in qualche modo flessibile ed adattabile ai cambiamenti. Come suggerito da Muylar e Al. [77], e spiegato nel capitolo 6 di questo lavoro, i linguaggi dichiarativi consentono una maggiore flessibilità nella modellazione dei processi rispetto a quelli imperativi. Questi ultimi sono infatti linguaggi che prevedono di specificare tutte le possibili modalità di esecuzione di un determinato processo, producendo modelli 'chiusi', ovvero rigidi nella sequenza delle attività che li compongono. Se i linguaggi imperativi si prestano a descrivere processi ben strutturati in ambienti stabili, possono tuttavia condurre alla rappresentazione di modelli non leggibili e complessi in ambienti turbolenti e fortemente variabili. Di converso, i linguaggi dichiarativi non esprimono tutti i comportamenti possibili nell'esecuzione dei processi, ma soltanto quelli non consentiti. In questo modo, consentono la rappresentazione di diversi comportamenti, dal momento che nell'esecuzione del processo tutto è consentito a meno che non sia stato esplicitamente vietato [78]. Ciò è ottenuto mediante una rappresentazione del processo come un set di regole, in modo tale che esso possa essere eseguito in numerosi modi possibili purché tali regole non vengano violate. Data la dinamicità dei processi sanitari, si è pensato di rappresentarli mediante l'impiego del linguaggio dichiarativo, utilizzando il tool Declare Designer, descritto nel precedente capitolo [79].

Un secondo aspetto da considerare qualora ci si appresti a modellare un processo sanitario con il fine di analizzarlo, è che se si considerano esclusivamente le Linee Guida come parametro di riferimento, si corre il rischio di disegnare modelli troppo astratti per essere comparati con la realtà: essi possono infatti essere affetti dalla soggettività del disegnatore, che può avere diverse competenze/specializzazioni ed interpretare in base alla propria formazione quanto riportato all'interno delle linee guida: uno stesso processo, pertanto, può assumere diverse rappresentazioni in diversi contesti. Inoltre, dal momento che la pratica clinica è in costante evoluzione grazie ai miglioramenti ottenuti giorno dopo giorno, dopo un certo periodo di tempo un modello disegnato per un particolare processo corre il rischio di diventare obsoleto e deve essere ridisegnato in accordo alla pratica corrente. Quanto esposto consente di comprendere che alcune deviazioni nell'esecuzione del processo possono essere giustificate dal fatto che i modelli disegnati necessitano di essere continuamente ridefiniti in accordo alla evoluzione della pratica clinica. D'altra parte, alcune deviazioni sono dovute a veri e propri errori nell'esecuzione delle attività: tali errori devono essere monitorati e prevenuti perché influenzano negativamente la qualità delle cure.

Infine, poiché sia l'efficacia che l'efficienza dei trattamenti necessitano di essere costantemente migliorate [96], le analisi di performance risultano estremamente complesse ogni qualvolta si voglia

analizzare un processo sanitario. Ad esempio, in caso di processi di trattamento di patologie urgenti, l'intervallo di tempo tra l'ammissione del paziente in ospedale e l'esecuzione dell'intervento chirurgico è un parametro critico che può compromettere il recupero del paziente e necessita pertanto di essere monitorato.

Per rispondere alle esigenze descritte, si possono utilizzare le tecniche di process mining. Come spiegato nel capitolo 4, le tecniche di process mining consentono di migliorare la conoscenza di un processo estraendo informazioni dagli event log estratti dai Sistemi Informativi Sanitari. Come ultimo step del percorso di ricerca descritto in questo lavoro di tesi, si è deciso pertanto di combinare le tecniche di process mining con l'adozione del linguaggio dichiarativo per analizzare i processi sanitari e rispondere alle problematiche sopra riassunte.

In particolare, la metodologia proposta si ripropone di investigare le deviazioni tra il processo 'ottimo', modellato con il linguaggio dichiarativo in base alle prescrizioni delle Linee Guida, ed il processo concretamente svolto nella quotidiana pratica clinica in una determinata unità operativa (reparto, laboratorio, centro di riabilitazione etc). L'identificazione di tali deviazioni può essere utilizzata sia per aggiornare le Linee Guida in accordo alle deviazioni 'positive' della corrente pratica clinica, oppure per identificare deviazioni 'patologiche' dovute ad una scorretta attuazione delle prescrizioni che può danneggiare il paziente. Parallelamente all'identificazione delle deviazioni suddette, anche la dimensione temporale costituisce un parametro da valutare qualora si analizzi un processo clinico. Il caso di studio presentato riguarda il percorso diagnostico-terapeutico relativo alla cura del criptorchidismo. Il modello dichiarativo del processo è stato confrontato con i log di eventi estratti dal Sistema Informativo del Reparto di Urologia dell'Ospedale di Zwolle (The Netherlands). Il processo è stato poi analizzato dal punto di vista temporale.

.

6.2 Metodologia

Gli elementi fondamentali della metodologia proposta sono schematizzati in figura 6.2.1.

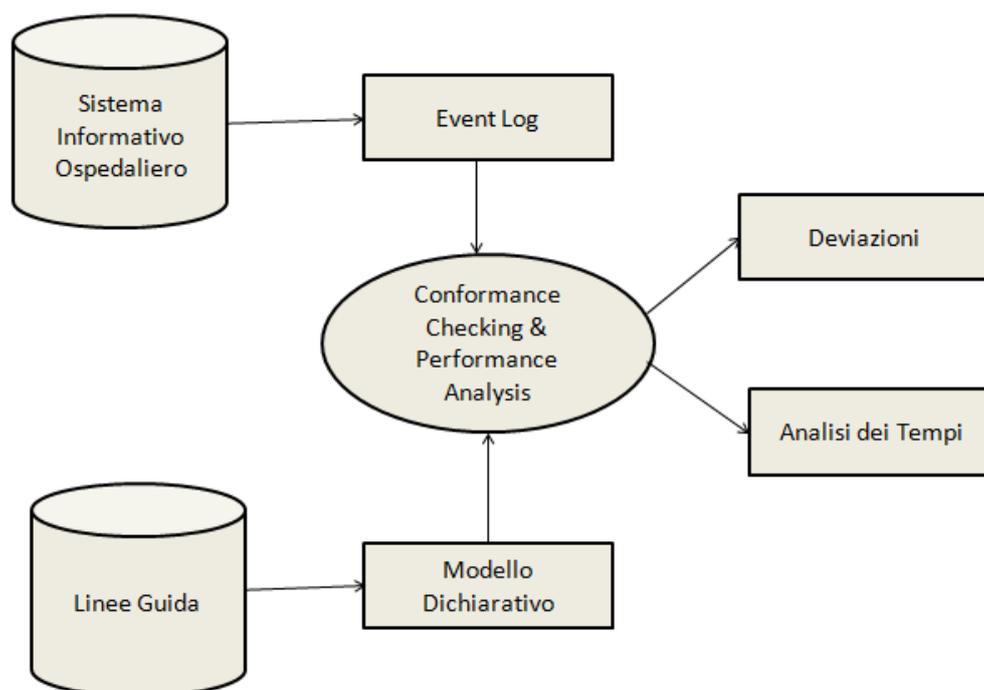


Fig.6.2.1: Metodologia per l'analisi dei processi sanitari

La prima fase consiste nella realizzazione del modello dichiarativo del processo, sulla base delle prescrizioni illustrate all'interno delle Linee Guida. Dal momento che tale rappresentazione è basata su principi teorici definiti e stabiliti dalla Letteratura Scientifica, tale modello verrà di seguito denominato 'de jure model'. Nell'ultimo decennio, numerosi approcci finalizzati alla rappresentazione delle linee guida attraverso modelli formali sono stati testati nel contesto della simulazione di sistemi decisionali a supporto dei medici. [77] [97] [98]. Il limite di molti di questi modelli riguarda la rappresentazione dei processi in relazione alla loro variabilità: come motivato nel precedente paragrafo, si è deciso pertanto di utilizzare il linguaggio dichiarativo mediante il tool Declare descritto nel capitolo 5. In particolare, il Declare Designer è stato individuato come strumento principe per rappresentare il processo oggetto di studio come un insieme di attività e di regole che definiscono le relazioni tra esse.

D'altra parte, il concreto svolgimento di un processo di trattamento all'interno di una struttura ospedaliera, può essere analizzato utilizzando le informazioni memorizzate all'interno dei Sistemi Informativi Sanitari [68], i quali tipicamente registrano le attività svolte durante l'esecuzione del processo in un event log. Un event log contiene informazioni riguardo attività e casi (istanze di processo); esso può inoltre contenere informazioni temporali (timestamps) e relative alle risorse

coinvolte nell'esecuzione del processo. Le tecniche di Process Mining [15] analizzano tali event log consentendo di acquisire conoscenza dei processi oggetto di studio. Nella metodologia proposta, in particolare, vengono impiegate tecniche di conformance checking e time dimension analysis.

Una volta realizzato il modello 'de jure', le tecniche di conformance checking vengono applicate per misurare l'aderenza del modello disegnato in base alle linee guida con gli event log che riflettono la esecuzione 'de facto' del processo. Esse consentono pertanto di individuare le deviazioni tra le linee guida e il processo concreto; a partire dalle deviazioni identificate, esse individuano anche come il processo reale possa essere modificato per rispecchiare meglio le prescrizioni fornite dalle linee guida mediante un allineamento delle tracce.

In particolare, la metodologia proposta si basa – per quanto concerne la conformance checking - sull'impiego di due particolari plugin del tool per Process Mining Prom¹⁰: il Declare Checker ed il Declare Analyzer, implementati presso l'Eindhoven University of Technology.

Il Declare Analyzer riceve in input un modello realizzato con Declare ed un event log, ed individua le reciproche corrispondenze e deviazioni a livello di ogni singola traccia. In particolare, esso individua quando un constraint viene attivato nel log di eventi, e quando è soddisfatto oppure violato; esso indica anche i casi in cui il soddisfacimento di un constraint entra in conflitto con un altro presente nel modello, ovvero la presenza di un 'conflitto'. Ogni volta che nel log un constraint è rispettato, si parla di 'fulfillment'. Quando la traccia non corrisponde a quanto specificato nel modello si genera una 'deviazione'. Il Declare Analyzer è in grado di fornire dati relativi al numero di attivazioni, fulfillment, violazioni e constraint sia dal punto di vista del totale calcolato considerando l'intero log analizzato, sia per ogni singola traccia/istanza di processo. Dal momento che ogni istanza di processo in un log di eventi si riferisce al processo di trattamento di un singolo paziente, considerando le informazioni fornite dal Declare Analyzer a livello del singolo caso, ovvero della singola istanza di processo, è possibile identificare le deviazioni e motivarle alla luce delle caratteristiche del particolare paziente.

Il Declare Checker, ricevendo in input un modello Declare ed un event log, stima la 'fitness', ovvero la corrispondenza globale tra i due; esso fornisce inoltre indicazioni in merito a come le istanze del processo che non corrispondono al modello 'de jure' possono essere modificate per eliminare le non conformità, consentendo l'identificazione di soluzioni riparatorie nell'esecuzione del processo. Il Declare Checker si compone di due diversi plugin: il Declare Replayer ed il Declare Diagnoser. Il primo riceve come input il modello Declare del processo ed il log di eventi e misura per ciascuna istanza di processo – ovvero per ciascuna traccia dell'event log - la fitness dell'istanza stessa con il

¹⁰ <http://www.promtools.org/prom6/>

modello, in un range da 0 (worst case) a 1 (best fitness). Se vengono identificate delle tracce per le quali la fitness è minore di 1, il plugin identifica un allineamento tra la traccia ed il modello, mostrando come la traccia stessa debba essere modificata per essere allineata con il modello.

A partire dai risultati del Declare Replayer, il Declare Diagnoser genera una sorta di mappa per dare una visione globale delle divergenze tra il modello atteso e quello reale. In particolare, esso genera un modello uguale al modello Declare 'de jure', in cui i constraint e le attività sono affiancati da un numero, variabile da 0 a 1, che rappresenta il rispettivo grado di conformità. Il grado di conformità di una attività nel modello è una indicazione del numero di volte in cui l'attività stessa è stata rimossa o aggiunta all'interno delle tracce del log esaminato per realizzare l'allineamento tra tutte le tracce ed il modello atteso. Il grado di conformità di un constraint nel modello è un'indicazione del numero di volte in cui tale constraint è stato violato all'interno del log. Nella mappa prodotta dal Declare Diagnoser, attività e constraint del modello de jure vengono rappresentate con un colore dal rosso al verde, indicando un grado di conformità che varia tra 0 (rosso) e 1 (verde).

Terminata la conformance checking, il processo viene analizzato dal punto di vista temporale. Per effettuare tale tipo di analisi a partire da un modello dichiarativo del processo, è stato utilizzato il plugin in ProM 'extend a Declare model with Time Information'. Tale plugin, ricevuti in input il modello Declare e l'event log, misura il tempo di attivazione di ciascun constraint indicato nel modello utilizzando il timestamps nell'event log. In particolare, esso fornisce una stima del tempo medio, minimo e massimo rispetto alla totalità delle tracce, con le relative deviazioni standard. Tali misure forniscono informazioni importanti per verificare se ci sono segmenti del processo che necessitano di essere ottimizzati per evitare il generarsi di code.

Nel paragrafo seguente viene descritta l'applicazione della metodologia proposta ad un particolare caso di studio, ovvero la cura del criptorchidismo.

6.3 Case Study: il trattamento chirurgico del criptorchidismo

Il Reparto di Urologia dell'Ospedale Isala di Zwolle (The Netherlands) è specializzato nel trattamento di patologie dell'apparato urinario maschile e femminile, ed in particolare degli organi riproduttivi. Nel corso dello studio ci si è soffermati su una specifica patologia, ovvero sul processo di trattamento del criptorchidismo.

Come descritto nel paragrafo precedente, il primo step dello studio ha riguardato l'approfondimento del processo oggetto di indagine, in modo da realizzare, sulla base delle prescrizioni delle Linee

Guida e delle evidenze fornite dalla Letteratura Scientifica, un modello 'de jure' del processo con il Declare Designer. Sono state pertanto esaminate le Linee Guida dell'Associazione Europea di Urologia ¹¹ ed altri Studi trovati all'interno della banca dati Pubmed¹².

Come riportato nelle Linee Guida, il criptorchidismo è l'assenza di uno o di entrambi i testicoli dallo scroto, a causa di una loro scorretta, incompleta, o assente discesa; si tratta di una delle patologie più comuni tra gli infanti: essa colpisce infatti circa lo 0,8 – 1,8% di bambini fino dell'età di 1 anno, il 3% dei neonati ed il 21% di nati prematuri [99]. E' stato inoltre constatato che, se non curata adeguatamente, tale patologia può diventare causa di infertilità e di cancro [100]: pertanto, la diagnosi precoce ed il trattamento adeguato del criptorchidismo appaiono fondamentali per prevenire la degenerazione della patologia. Come spiegato nelle Linee Guida, nonostante nel 15-20% dei casi i testicoli ritenuti discendono grazie a cure ormonali, l'assunzione di tali sostanze può rivelarsi pericolosa e compromettere la spermatogenesi; è stato inoltre dimostrato che i testicoli discesi grazie alle cure farmacologiche, in molti casi tendono a risalire nel tempo: il trattamento ormonale, pertanto, viene solitamente sconsigliato. D'altra parte, la percentuale di successo del trattamento chirurgico del criptorchidismo, denominato orchidopessi, è del 70-90% [101]. Per trattare chirurgicamente il criptorchidismo, possono essere applicati l'approccio laparoscopico e quello tradizionale di chirurgia open, anche se risulta ancora aperto il dibattito in merito a quale delle due tecniche sia preferibile [102]. In accordo a quanto affermato da Gapany et Al [103], attualmente nessuna tecnica di diagnostica per immagini è superiore alla laparoscopia per il trattamento dei testicoli non palpabili, in quanto essa consente sia la diagnosi che il trattamento chirurgico se la presenza della patologia viene confermata. Qualora l'intervento, per via di complicanze, non possa essere completato per via laparoscopica, uno switch verso l'approccio open è comunque applicabile: tale decisione viene presa dai chirurghi prima o durante l'intervento nei casi in cui sia ritenuto opportuno per la sicurezza del paziente, ad esempio nei casi in cui si ha difficoltà a visualizzare la struttura anatomica con l'approccio laparoscopico o per insorgenza di complicanze [104] [105]. Quando viene eseguito un intervento di orchidopessi, può essere effettuato un esame istologico per indagare la natura del tessuto e rimuovere a scopo preventivo anomalie sospette: una associazione tra il criptorchidismo ed il cancro ai testicoli è infatti stata osservata da più di un secolo [106]. Come riportato in [107], quando viene diagnosticato criptorchidismo unilaterale con presenza di cellule maligne, l'orchietomia – una procedura

¹¹ http://www.uroweb.org/fileadmin/guidelines/EAU_GO_Manual_November_28th_2012.pdf

¹² <http://www.ncbi.nlm.nih.gov/pubmed>

chirurgica per rimuovere il testicolo attraverso una incisione nella parte bassa dell'addome laterale – è da consigliarsi rispetto all'orchidopessi.

Nel presente lavoro, il processo di trattamento chirurgico del criptorchidismo è stato modellato mediante l'impiego del Declare Designer. Nella realizzazione del modello sono state considerate le attività presenti all'interno del log del processo estratto dal Sistema Informativo del Reparto di Urologia dell'Ospedale di Zwolle (The Netherlands). I constraint tra tali attività sono stati posti considerando le prescrizioni delle Linee Guida e della Letteratura Scientifica. Il modello realizzato, mostrato in figura (in inglese per rispettare la lingua originaria del log di eventi), è composto da 10 attività e 13 constraint derivanti da 4 diversi template.

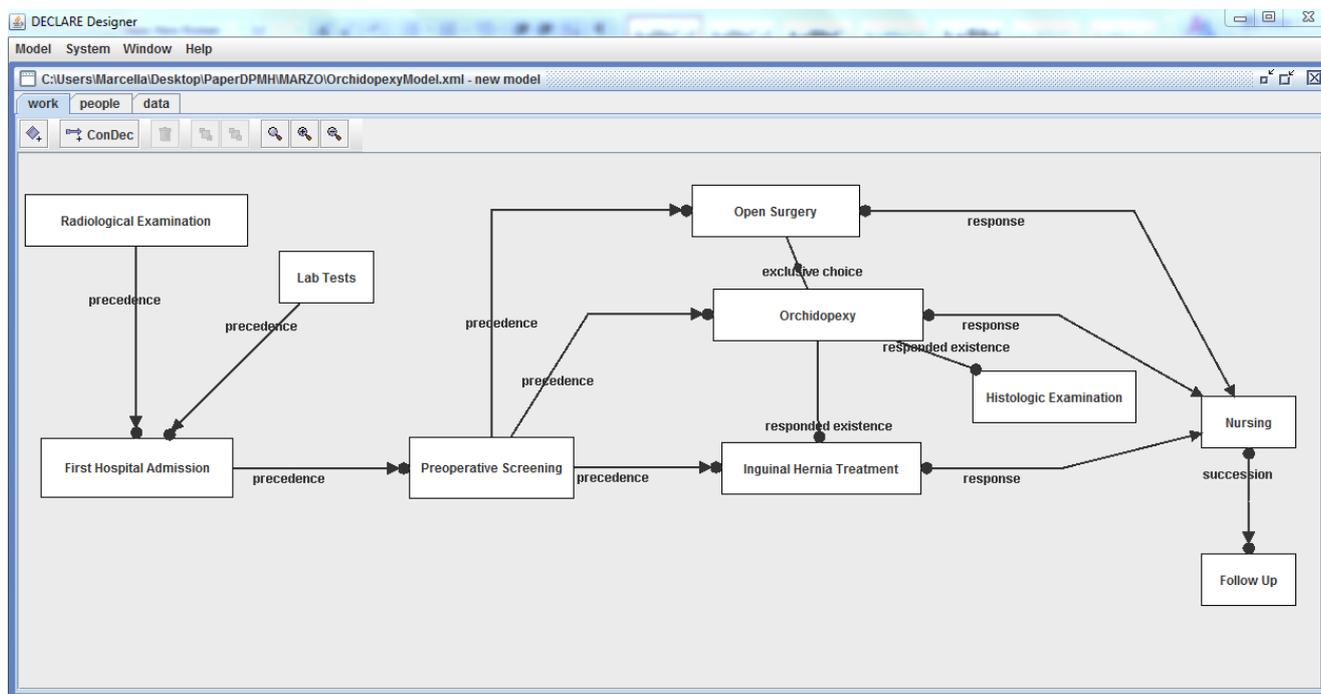


Fig.6.3.2: Il trattamento del criptorchidismo secondo le Linee Guida: rappresentazione con Declare

Di seguito viene illustrato il significato di ciascun constraint alla luce del caso di studio esaminato:

- Precedence tra Radiological Examination ed First Hospital Admission e tra Lab Test e First Hospital Admission: il ricovero in ospedale è solitamente preceduto dagli esami richiesti per stabilire la presenza del criptorchidismo ed effettuare la diagnosi; tra questi esami, i più comuni sono quelli radiologici ed esami di laboratorio.
- Precedence tra First Hospital Admission e Preoperative Screening: per essere sottoposto allo screening preoperatorio, il paziente deve essere prima ricoverato.

- Precedence tra Screening Preoperatorio ed intervento in chirurgia open, tra screening preoperatorio ed orchidopessi, tra screening preoperatorio e trattamento dell'ernia inguinale: in accordo alle Linee Guida per la Chirurgia dell'Organizzazione Mondiale della Sanità¹³, tutti i trattamenti chirurgici non possono essere eseguiti senza che esami preliminari per prevenire rischi per il paziente siano stati effettuati. Per quanto riguarda la denominazione delle attività, è importante sottolineare che, in accordo alla denominazione delle attività da parte dell'Isala Hospital, con il termine orchidopessi si intende il trattamento del criptorchidismo effettuato mediante la chirurgia laparoscopica, mentre l'espressione open surgery è riferito specificamente al trattamento in chirurgia open.
- Response tra orchidopexy e nursing, chirurgia open e nursing, trattamento dell'ernia inguinale e nursing: in accordo alle linee guida citate, un periodo di monitoraggio post-operatorio deve essere previsto dopo ogni intervento chirurgico per tenere sotto controllo le condizioni del paziente dopo l'intervento.
- Responded Existence tra esame istologico ed orchidopessi: in accordo alle European Guidelines of Urology, una biopsia può essere eseguita contestualmente all'orchidopessi quando si ritiene raccomandabile analizzare la natura del tessuto.
- Succession tra Nursing e Follow Up: dopo il monitoraggio post-operatorio, i pazienti si sottopongono ad un controllo per verificare se il trattamento è stato efficace ed il paziente può essere dimesso, oppure sono necessari ulteriori esami e/o interventi. D'altra parte, il follow up è comunque generalmente preceduto dal nursing.
- Responded Existence tra trattamento dell'ernia inguinale ed orchidopessi: in accordo alle European Guidelines of Urology, l'ernia inguinale può essere operata contestualmente all'intervento di orchidopessi.
- Exclusive Choice tra orchidopessi e chirurgia open: per trattare il criptorchidismo, o l'uno o l'altro approccio vengono solitamente applicati [108].

Una volta realizzato il modello de jure del processo con Declare, è stata applicata la conformance checking al fine di investigare la corrispondenza di tale modello alla realtà, rappresentata dall'event log del processo estratto dal Sistema Informativo del Reparto di Urologia dell'Ospedale di Zwolle. Il modello è stato infatti 'matchato' con l'event log mediante i plugin Declare Checker e Declare Analyzer.

¹³ http://www.who.int/patientsafety/safesurgery/knowledge_base/WHO_Guidelines_Safe_Surgery_finalJun08.pdf

I risultati ottenuti con i componenti Declare Replayer e Declare Diagnoser del Declare Checker plugin sono mostrati nelle immagini 6.3.3 e 6.3.4.

Come mostrato in figura 6.3.3, la 'fitness', ovvero la corrispondenza tra il modello de jure e l'event log è di 0.831 (in un range, come spiegato precedentemente, che varia da 0 per fitness assente, a 1 per fitness completa). Dalla fig. 6.3.4, è invece possibile dedurre che la conformance peggiore si ottiene in riferimento alle attività diagnostiche (esami radiologici e di laboratorio) eseguite per la diagnosi del criptorchidismo. Tale risultato, consultando i medici dell'Isala Hospital, è stato interpretato alla luce del fatto che, nella prassi quotidiana, i pazienti svolgono gli esami privatamente al di fuori dell'ospedale, e vengono direttamente ricoverati per subire il trattamento. Un basso grado di conformance è anche associato alla chirurgia open, che risulta essere eseguita solo in 4 tracce (ovvero per 4 pazienti) nell'event log.

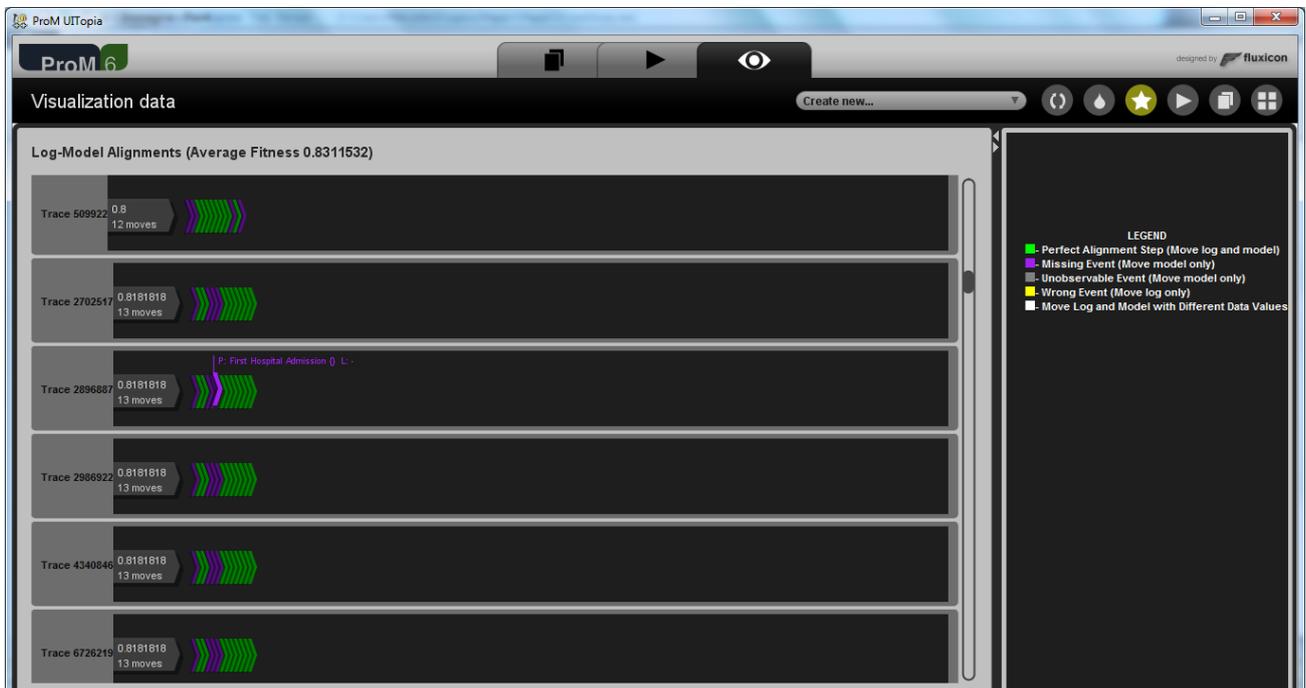


Fig.6.3.3: Conformance Checking mediante il Declare Replayer

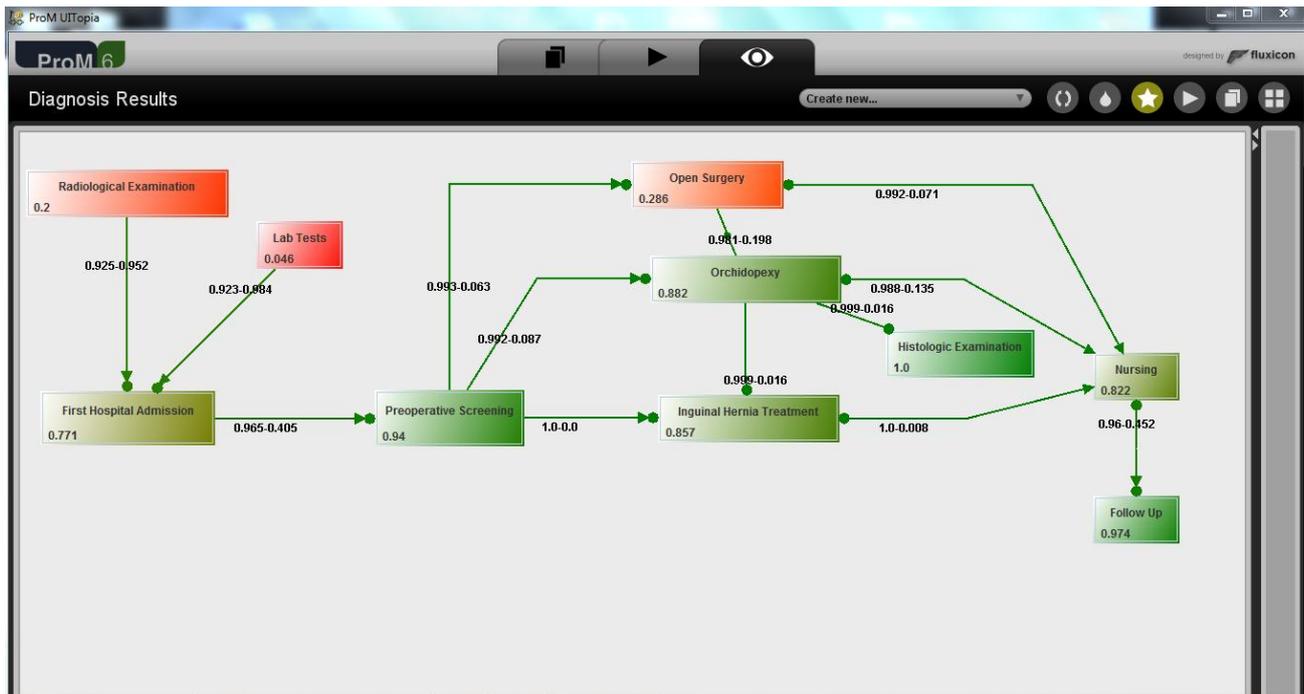


Fig.6.3.4: Conformance Checking mediante il Declare Visualizer

Attraverso il Declare Analyzer è stato inoltre possibile ottenere informazioni dettagliate in merito al soddisfacimento ed alla violazione dei constraint del modello nell'event log. I risultati ottenuti, in riferimento a ciascun constraint, sono elencati nella seguente tabella (in cui con FHA si indica la First Hospital Admission, con PS lo screening preoperatorio, con N il nursing, con OR l'orchidopessi, con IHT il trattamento dell'ernia inguinale, con RE gli esami radiologici, con LT gli esami di laboratorio e con FU il follow up) :

Constraint	Attivazioni	Violazioni	Fulfillments	Conflitti
Precedence [PS-OR]	197	7	190	0
Precedence [PS-OS]	4	0	4	0
Precedence [PS-IHT]	7	0	7	0
Precedence [RE-FHA]	172	162	10	0
Precedence [FHA-PS]	264	86	178	0
Precedence [LT-FHA]	172	168	4	0

Responded Existence [HE-OR]	3	2	1	0
Responded Existence [IHT-OR]	7	6	1	0
Exclusive Choice [OR-OS]	201	0	201	0
Succession[N-FU]	667	150	517	0
Response [OS-N]	4	0	4	0
Response [OR-N]	197	9	188	0
Response [IHT-N]	7	1	6	0

Tab.6.3.1: Conformance Checking mediante il Declare Analyzer

I risultati del Declare Analyzer hanno mostrato che in numerosi casi, il ricovero non è preceduto dagli esami radiologici e di laboratorio. Uno screenshot del Declare Analyzer che evidenzia tale tipo di deviazione è mostrato in figura 6.3.5.

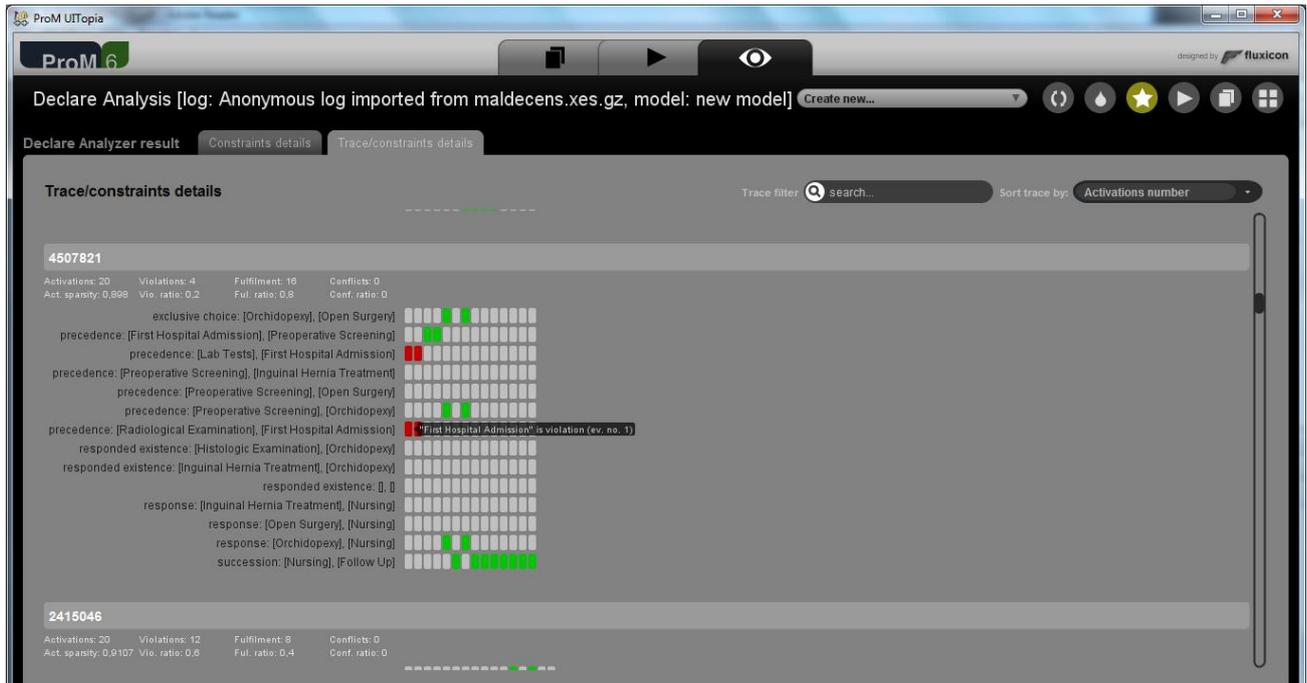


Fig.6.3.5: Conformance Checking mediante il Declare Analyzer

E' inoltre interessante osservare che in alcuni casi la precedenza tra screening preoperatorio e first hospital admission non è rispettata; tale anomalia è stata approfondita consultando i medici dell'ospedale, ai quali è stato fornito un elenco degli ID delle tracce (associate univocamente ai pazienti) in cui era stata registrata la deviazione. Secondo quanto riferito dai medici, tale anomalia si spiega considerando che a volte i pazienti vengono ricoverati in reparti differenti da quello di urologia perché affetti da patologie di altra natura, e successivamente trasferiti se sono sospette anche patologie dell'apparato urinario. La stessa considerazione può essere fatta per quei casi in cui non viene rispettata la precedenza tra screening preoperatorio e orchidopessi, in quanto i pazienti possono essere stati già sottoposti a trattamenti chirurgici in reparti differenti. Lo stesso può essere affermato in assenza del nursing dopo l'orchidopessi.

Altro interessante risultato è la violazione della risponde existence tra orchidopessi e trattamento dell'ernia inguinale: questo risultato è un tipico esempio in cui la libera scelta del medico e le condizioni del paziente trovano spazio, in quanto, a seconda di questi fattori, l'ernia può essere trattata da sola o in combinazione con l'orchidopessi.

Una volta analizzate le deviazioni, il processo è stato osservato dal punto di vista temporale. Per applicare l'analisi dei tempi è stato usato il plugin in ProM 'Extend a Declare model with time information', che riceve in input un modello dichiarativo del processo ed un log di eventi, e fornisce come output i tempi di attivazione dei constraint del modello nell'event log. Per semplicità computazionale, il modello iniziale è stato ridotto ai soli segmenti interessanti dal punto di vista temporale.

E' stato dunque realizzato il modello in figura 6.3.6, in cui viene incluso solo il constraint response tra le seguenti attività:

- First hospital Admission e Preoperative Screening
- Preoperative Screening e Orchidopessi
- Orchidopessi e Nursing
- Nursing e Follow Up,

in quanto rappresentative dei segmenti in cui si generano di consueto ritardi e code per il paziente.

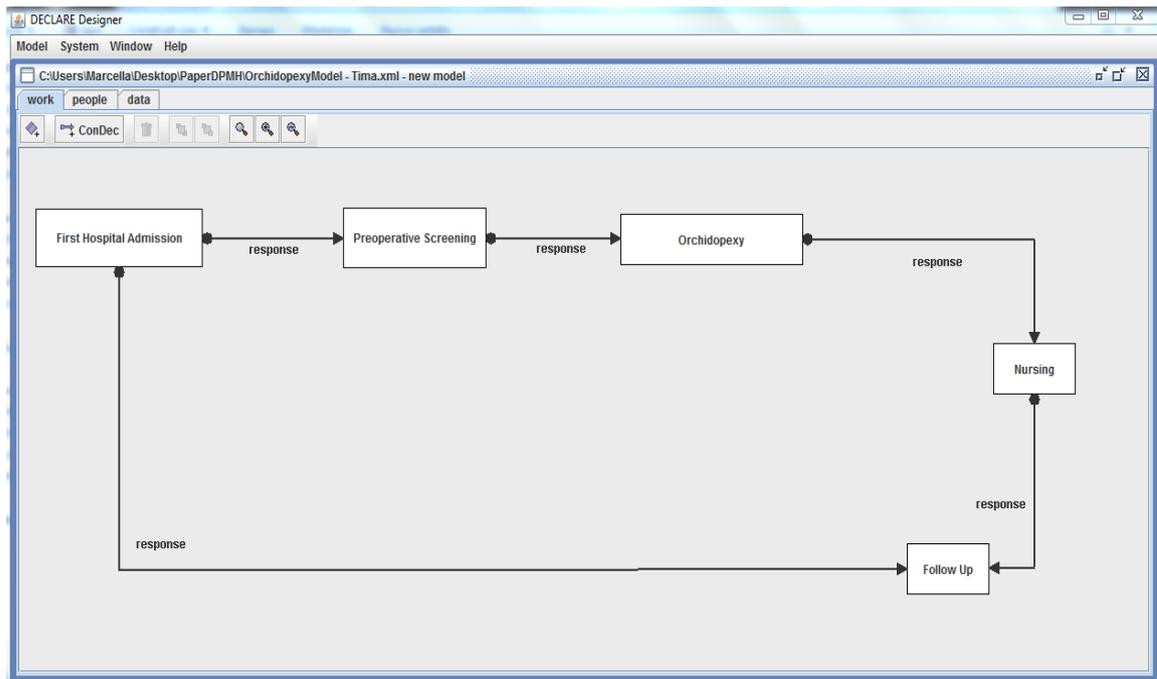


Fig.6.3.6: Time Model

Il match del modello con l'event log ha consentito di stimare il tempo di attivazione medio, minimo e massimo di ciascun constraint, rappresentativo della distanza temporale tra i due eventi legati dal constraint stesso, come mostrato in figura 6.3.7. In particolare, la stima del tempo intercorrente tra la First Hospital Admission ed il Follow Up – mostrata in figura 3.6.7 - , ha fornito una stima globale della durata dell'intero processo, che rappresenta il tempo di attesa del paziente per completare la cura ed è pari a 138 giorni.

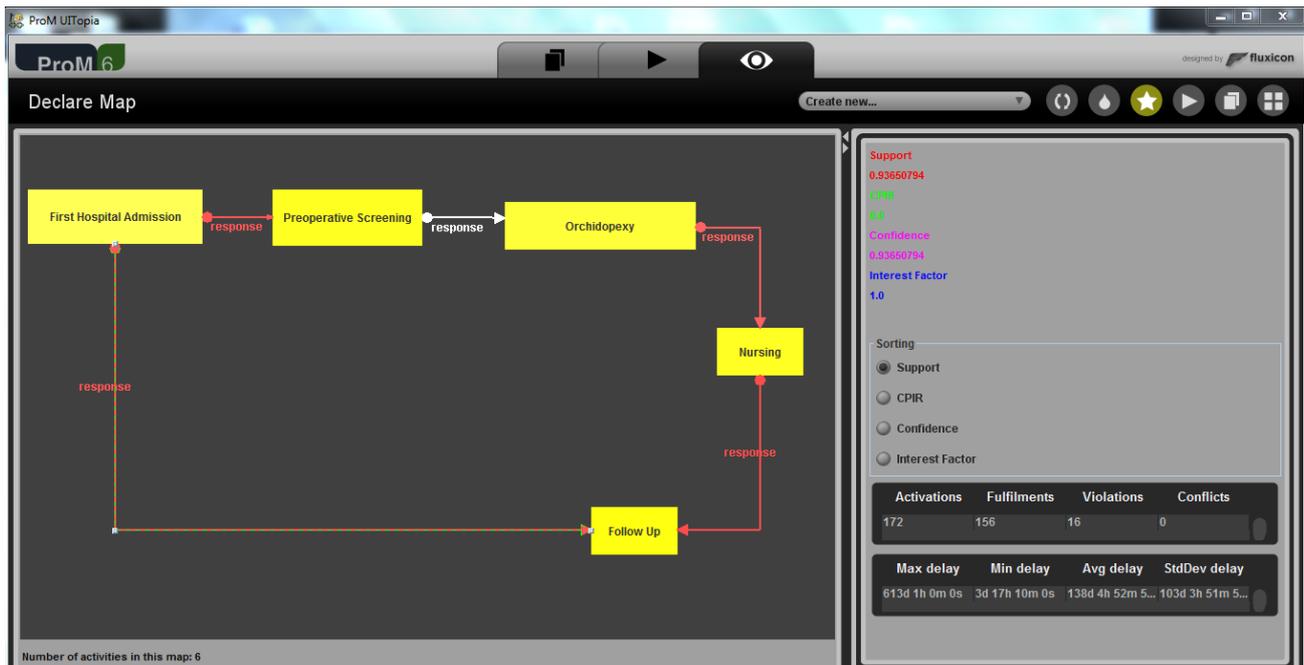


Fig.6.3.7: Analisi dei tempi attraverso il plugin 'Extend a Declare model with time information'

La tabella 6.3.2 riassume i risultati ottenuti, che sono stati confermati dai medici di riferimento.

Constraint	Max Delay	Min Delay	Average
Response [FHA-PS]	531g 20h 40m 0s	0g 0h 0m 0s	53g 21h 13m 2s
Response [PS-OR]	640d 15h 40m 0s	6g 9h 40m 0s	36g 12h 4m 38s
Response [OR-N]	29d 8h 19m 0s	0g 0h 1m 0s	0g 12h 20m 33s
Response [N-FU]	360g 5h 16m 0s	4g 21h 40m 0s	58g 1h 32m 39s
Response [FHA-FU]	613g 1h 0m 0s	3g 17h 10m 0s	138g 4h 52m 55s

Tab.6.3.2: Analisi dei Tempi

Come ci si aspettava, il minimo ritardo riguarda l'intervallo temporale tra intervento e nursing, dimostrando che il paziente viene correttamente monitorato nel periodo postoperatorio.

7. CONCLUSIONI

In questo lavoro di ricerca sono state analizzate diverse tecniche di analisi dei processi al fine di sperimentarne l'applicazione nel settore sanitario.

Partendo dalla modellazione dei processi sulla base dell'osservazione diretta e secondo linguaggi imperativo/procedurali (come quello caratterizzante le reti di Petri [51], è stato in seguito sperimentato un approccio basato sul linguaggio dichiarativo per la modellazione, e sul Process Mining per la verifica dei modelli realizzati [77] [15].

La metodologia proposta nel cap. 6 di questo lavoro si basa sull'impiego di Declare e del tool per Process Mining ProM per l'analisi dei processi sanitari. Da un lato, l'impiego di Declare per il design dei processi clinici si rivela una possibile soluzione per rappresentarne la flessibilità, consentendo di evidenziare solo le regole intrinseche del processo, ma non tutte le sue possibili modalità di esecuzione. D'altra parte, le tecniche di process mining consentono di investigare le deviazioni tra i modelli disegnati sulla base delle linee guida ed i processi reali, così come essi vengono eseguiti nella quotidiana pratica clinica. L'identificazione di tali deviazioni apre nuove prospettive di ricerca, verso errori commessi nella "pratica" clinica, e verso l'individuazione di interventi positivi per la gestione dei pazienti critici e per le cd. eccezioni terapeutiche.

Altro aspetto riguarda la possibilità offerta dalla metodologia proposta di tenere sotto controllo quei casi - di cui un esempio è il trattamento dell'ernia inguinale congiuntamente all'orchidopessi, trattato nel cap. 6 - in cui le condizioni del paziente, il grado della patologia, e la scelta del medico, determinano un'evoluzione del processo in una direzione piuttosto che in un'altra.

Bisogna anche sottolineare che l'elevato livello di monitoraggio dei processi raggiunto con l'impiego del process mining, consente la rilevazione puntuale degli errori, che costituisce a sua volta il presupposto per una politica di riduzione di rischi ed eventi avversi.

A dispetto dei lati positivi della metodologia proposta, si riscontrano tuttavia delle criticità da tenere in considerazione. La prima riguarda la difficoltà dei medici e del personale sanitario a 'familiarizzare' con un nuovo linguaggio, quale quello dichiarativo, per modellare i processi: nonostante il Declare sia dotato di una grafica piuttosto semplice, bisogna considerare che all'interno degli Ospedali si è abituati a ragionare secondo logiche sequenziali di workflow; pertanto, l'accettazione di un nuovo modello, richiede del tempo.

Inoltre, per poter applicare le tecniche di process mining, bisogna disporre di event log in cui per ogni processo ci sia un set definito di istanze, requisito che non è sempre soddisfatto, dal momento che i sistemi informativi process oriented sono di recente introduzione nel settore sanitario.

Dall'analisi del caso di studio, ed in particolare delle deviazioni riscontrate, è risultato inoltre chiaro che, per una completa analisi del processo di cura del paziente secondo l'approccio descritto, bisognerebbe realizzare una completa integrazione dei sistemi informativi di tutti i reparti di uno stesso ospedale: in molti casi, infatti, i pazienti possono essere trasferiti e ciò può condurre ad identificare deviazioni (come quella descritta relativa alla precedenza tra screening preoperatorio e intervento) che non sono dovute ad anomalie del processo di cura, ma piuttosto ad una carenza del sistema informativo.

Prospettive di ricerca future dovrebbero dunque non soltanto focalizzarsi sull'applicazione delle tecniche di analisi di processo a diversi ambiti del settore sanitario, ma anche interessarsi agli aspetti inerenti la struttura dei sistemi informativi ospedalieri, spingendo verso l'adozione di architetture altamente integrate.

Bibliografia

- [1] R. Giorgetti, «Legislazione e organizzazione del Servizio Sanitario Nazionale,» *Maggioli Editore*, 2010.
- [2] E. E. Pessina, «Rapporto Oasi 2003. L'aziendalizzazione della Sanità in Italia,» 2003.
- [3] J. Healy, «Hospitals in a Changing Europe,» *European Observatory on Health Care Systems*, 2002.
- [4] R. Mans, «Workflow Support in the Healthcare Domain,» Eindhoven University of Technology Library, 2011.
- [5] G. Turchetti, «Health Technology Assessment,» *Engineering in Medicine and Biology Magazine*, 2010.
- [6] A.J Sutton, *Methods for Meta-Analysis in Medical Research*, Wiley, 2000.
- [7] D. Moher, «Improving the quality of reports of meta-analyses of randomised controlled trials: the QUORUM statement,» *The Lancet*, 1999.
- [8] M. Reichert, «IT Support for Healthcare Processes: premises, challenges, perspectives.,» *Data and Knowledge Engineering*, p. 58, 2007.
- [9] M. Helfert, «Preview of frameworks and an approach for healthcare process development,» *Information Management in Modern Enterprise: Issues and Solutions*, pp. 371-378, 2005.
- [10] W.T. Adams, «Computer Imaging and Workflow Systems in the Business Office,» *Healthcare Financial Management*, 1999.
- [11] W. van der Aalst e K. Hees, *Workflow Management: Models, Methods and Systems*, 1997.
- [12] T. Murata, «Petri nets: Properties, analysis and applications,» in *Proceedings of the IEEE*

volume 77, 1989.

- [13] A. Pepino, M. Rovani, A. Torri e M. Sansone, «Supply Chain for Regional Centralized Purchasing: An Application Model,» *Intelligent Information Management*, 2012.
- [14] A. Torri, M. Rovani e A. Pepino, «Healthcare Systems Analysis through the Workflow Management approach. A case study: Hypertension Center,» in *Data and Knowledge for Medical Decision Support*, Praga, 2013.
- [15] W. van der Aalst, *Process Mining: Discovery, Conformance and Enhancement of Business Process*, Springer, 2011.
- [16] M. Weske, *Business Process Management: Concepts, Languages, Architectures*, Springer, 1998.
- [17] D. Elzinga, «Business process management: survey and methodology,» *Engineering Management, IEEE Transactions*, pp. 119-128, 1995.
- [18] M. Weske, W. van der Aalst e H. Verbeek, «Advances in business process management,» *Data & Knowledge Engineering*, vol. 50, n. 1, pp. 1-8, 2004.
- [19] J. Hill, «Gartner's position on business process management,» *Gartner Research*, 2006.
- [20] W. van der Aalst, *Modern Business Process Automation*, Springer, 2010.
- [21] A. Ferri, W. Jost e M. Kirchmer, «Business Process Automation,» *August-Wilhelm Scheer-Springer*, 2004.
- [22] v. d. Aalst, «Process mining: discovering and improving Spaghetti and Lasagna processes,» *Computational Intelligence and Data Mining (CIDM) IEEE*, 2011.
- [23] M. Barnett, «Modeling & simulation in business process management,» in *BP Trends Newsletter, White Papers & Technical Briefs*, 2003.
- [24] V. Grover, S. Jeong, W. Kettinger e J. Teng, «The implementation of business process

- reengineering,» *Journal of Management Information Systems*, vol. 12, n. 1, pp. 109-144, 1995.
- [25] I. Tsiouras, *La progettazione del sistema di gestione per la qualità nelle organizzazioni ad alta intensità informativa. Dalla ISO 9000 alla modellazione del business*, FrancoAngeli, 2005.
- [26] D. Hollingsworth e U. Hampshire, «Workflow management coalition the workflow reference model,» *Workflow Management Coalition*, vol. 68, 1993.
- [27] J. Becker, M. Muehlen e M. Gille, «Workflow application architectures: classification and characteristics of workflow-based information systems,» in *Workflow handbook*, 2002.
- [28] F. Leymann, D. Roller e M. Schmidt, «Web services and business process management,» *IBM systems Journal*, vol. 41, n. 2, pp. 198-211, 2002.
- [29] O. Signore, «La Web Services Activity del W3C,» 2002.
- [30] P. Subramanian, A. Songer e J. Diekmann, «Visualizing process performance,» Stanford, 2002.
- [31] P. Hung e K. Kamalakar, «A secure workflow model,» *Proceedings of the Australasian information security workshop conference on ACSW frontiers. Australian Computer Society.*, 2003.
- [32] M. Alavi e D. Leidner, «Review: Knowledge management and knowledge management systems: Conceptual foundations and research issues,» *MIS quarterly*, pp. 107-136, 2001.
- [33] H. Plünnecke e W. Reisig, *Bibliography of Petri nets 1990*, Springer Berlin Heidelberg, 1991.
- [34] R. Bruni, J. Meseguer e U. Montanari, «Functorial models for petri nets,» *Information and Computation*, pp. 207-236, 2001.
- [35] W. Reisig e G. Rozenberg, *Lectures on Petri nets I- Basic models: advances in Petri nets*, Springer, 1998.
- [36] S. Kounev, «Performance modeling and evaluation of distributed component-based systems

- using queueing petri nets,» *Software Engineering, IEEE Transactions*, pp. 486-502, 2006.
- [37] G. Chiola, M. Marsan e G. Balbo, «Generalized stochastic Petri nets: A definition at the net level and its implications,» *Software Engineering, IEEE Transactions*, pp. 89-107, 1993.
- [38] D. Kartson, G. Balbo, S. Donatelli e G. Franceschinis, *Modelling with generalized stochastic Petri nets*, John Wiley & Sons, Inc, 2004.
- [39] V. Sassone, «On the algebraic structure of petri nets,» *Bulletin of the European Association for Theoretical Computer Science*, pp. 133-148, 2000.
- [40] R. Bruni e U. Montanari, «Zero-safe nets: Comparing the collective and individual token approaches,» *Information and computation*, pp. 46-89, 2000.
- [41] R. David e H. Alla, «Petri nets for modeling of dynamic systems: A survey,» *Automatica*, pp. 175-202, 1994.
- [42] G. Denaro e M. Pezze, «Petri nets and software engineering,» *Lectures on Concurrency and Petri Nets - Springer Berlin Heidelberg*, pp. 439-466, 2004.
- [43] K. Salimifard e M. Wright, «Petri net-based modelling of workflow systems: An overview,» *European journal of operational research*, pp. 664-676, 2001.
- [44] W. van der Aalst e A. Barros, «Advanced workflow patterns,» *Cooperative Information Systems - Springer Berlin Heidelberg*, 2000.
- [45] P. Merlin e D. Faber, «Recoverability of communication protocols,» *IEEE Transactions on Communication*, pp. 1036-1043, 1976.
- [46] C. Ramchandani, «Analysis of asynchronous concurrent systems by timed Petri nets,» 1973.
- [47] M. Molloy, «Performance analysis using stochastic Petri nets,» *Computers - IEEE Transactions*, pp. 913-917, 1982.
- [48] M. Ajmone Marsan, G. Conte e G. Balbo, «A class of generalized stochastic Petri nets for the

- performance evaluation of multiprocessor systems,» *ACM Transactions on Computer Systems*, pp. 93-122, 1984.
- [49] W. van der Aalst, «Interval timed coloured Petri nets and their analysis,» *Application and Theory of Petri Nets - Springer Berlin Heidelberg*, pp. 453-472, 1993.
- [50] C. Lindemann, «Performance modelling with deterministic and stochastic Petri nets,» *ACM SIGMETRICS Performance Evaluation Review*, 1998.
- [51] W. van der Aalst, «The application of Petri nets to workflow management,» *Journal of circuits, systems, and computers*, pp. 21-66, 1998.
- [52] J. Billington, S. Christensen e e. Al., «The Petri Net Markup Language: Concepts, Technology, and Tools,» *Applications and Theory of Petri Nets - Springer Berlin Heidelberg*, pp. 483-505, 2003.
- [53] T. Freitag, «WoPeD-workflow petri net designer,» *BA Karlsruhe*, 2005.
- [54] A. Eckleder e T. Freytag, «WoPeD 2.0 goes BPEL 2.0,» *AWPN*, pp. 75-80, 2008.
- [55] W. van der Aalst e K. Lassen, «Translating unstructured workflow processes to readable BPEL: Theory and implementation,» *Information and Software Technology*, pp. 131-159, 2008.
- [56] W. van der Aalst, «Woflan: a Petri-net-based workflow analyzer,» *Systems Analysis Modelling Simulation*, pp. 345-358, 1999.
- [57] W. van der Aalst, «Verification of workflow nets,» *Application and Theory of Petri Nets - Springer Berlin Heidelberg*, pp. 407-426, 1997.
- [58] F. Biagini e M. Campanino, *Elementi di probabilità e statistica*, Springer, 2006.
- [59] B. Abel-Smith e E. Mossialos, «Cost Containment and Health Care Reform: A Study of the European Union,» *Health Policy*, pp. 89-132, 1994.

- [60] M. Amatucci e S. Mele, «Centralized Purchasing in Healthcare: A Comparison between National and International Experiences,» in *15th Annual Conference AIES of the future Healthcare: Institutions, Efficiency and Equity*, 2010.
- [61] B. Tony, «Developing New Forms of Partnership With the 'Market' in the Procurement of Public Services,» *Public Administration*, pp. 81-102, 2006.
- [62] E. Bakker, H. Walker, F. Schotanus e C. Harland, «Choosing an Organizational Form: The Case of Collaborative Procurement Initiatives,» *International Journal of Procurement Management*, pp. 297-317, 2008.
- [63] C. Austin, J. Trimm e P. Spbczak, «Information Systems and Strategic Management,» *Health Care Management Review*, pp. 26-33, 1995.
- [64] B. Forouzan, *TCP/IP Protocol Suite*, New York: McGraw-Hill, 2002.
- [65] F. Curbera, M. Duftler, R. Khalaf, W. Nagy, N. Mukhi e S. Weerawarana, «Unraveling the Web Services Web: An Introduction to SOAP, WSDL, and UDDI,» *IEEE Internet Computing*, pp. 86-93, 2002.
- [66] M. De Marco, *I sistemi informativi aziendali. Temi di attualità*, Franco Angeli, 2000.
- [67] T. Davenport e J. Short, «The new industrial engineering: information technology and business process redesign,» *Sloan management review*, 2000.
- [68] M. Reichert e R. Lenz, «It support for healthcare processes - premises, challenges, perspectives,» *Data & Knowledge Engineering*, pp. 39-58, 2007.
- [69] W. Jiang, T. Au e K. Tsui, «A statistical process control approach to business activity monitoring,» *Iie Transactions*, pp. 235-249, 2007.
- [70] W. van der Aalst e J. Weijters, «Process mining: a research agenda,» *Computers in industry*, pp. 231-244, 2004.

- [71] D. Hilbert e D. Redmiles, «Extracting usability information from user interface events,» *ACM Computing Surveys (CSUR)*, pp. 384-421, 2000.
- [72] W. van der Aalst, T. Weijters e L. Maruster, «Workflow mining: Discovering process models from event logs,» *Knowledge and Data Engineering - IEEE Transactions*, pp. 1128-1142, 2004.
- [73] A. Rozinat e W. van der Aalst, «Conformance checking of processes based on monitoring real behavior,» *Information Systems*, pp. 64-95, 2008.
- [74] W. van der Aalst, H. Reijers e A. Weijters, «Business process mining: An industrial application,» *Information Systems*, pp. 713-732, 2007.
- [75] B. van Dongen, A. de Medeiros, H. Verbeek, A. Weikters e W. van der Aalst, «The ProM Framework: A New Era in Process Mining Tool Support,» *Applications and Theory of Petri Nets - Springer Berlin Heidelberg*, pp. 444-454, 2005.
- [76] A. Rebugue e D. Ferreira, «Business process analysis in healthcare environments: A methodology based on process mining,» *Information System*, 2012.
- [77] N. Mulyar, M. Pesic, W. van der Aalst e M. Peleg, «Declarative and procedural approaches for modelling clinical guidelines: addressing flexibility issues,» *Business Process Management Workshops*, pp. 335-346, 2008.
- [78] F. Maggi, A. J. Mooij e W. van der Aalst, «User-guided discovery of declarative process models,» *Computational Intelligence and Data Mining (CIDM) IEEE Symposium*, pp. 192-199, 2011.
- [79] M. Pesic e W. van der Aalst, «A declarative approach for flexible business process management,» in *Business Process Management Workshops*, Heidelberg, 2006.
- [80] M. Pesic e W. van der Aalst, «A declarative approach for business process management,» in

Business Process Management Workshop, 2006.

- [81] M. Vardi, «An automata-theoretic approach to linear temporal logic,» *Logics for concurrency - Springer Berlin Heidelberg*, pp. 238-266, 1996.
- [82] J. Lavis, E. Paulsen, A. Oxman e R. Moynihan, «Evidence-informed health policy 2: Survey of organizations that support the use of research evidence,» *Implementation Science* 3, 2008.
- [83] D. Eddy, «Evidence-based medicine: a unified approach,» *Health Affairs*, pp. 9-17, 2005.
- [84] M. Hay, T. Weisner, S. Subramanian, N. Duan, E. Niedzinski e R. Kravitz, «Harnessing experience: exploring the gap between evidence-based medicine and clinical practice,» *Journal of evaluation in clinical practice*, pp. 707-713, 2008.
- [85] L. Cochrane, C. Olson, S. Murray, M. Dupuis, T. Tooman e S. Hayes, «Gaps between knowing and doing: Understanding and assessing the barriers to optimal health care,» *Journal of Continuing Education in the Health Professions*, pp. 94-102, 2007.
- [86] N. E. Cahill e D. Heyland, «Bridging the guideline practice gap in critical care nutrition: A review of guideline implementation studies,» *Journal of Parenteral and Enteral Nutrition*, pp. 653-659, 2010.
- [87] F. Hajjaj, M. Salek, M. Basra e A. Finlay, «Nonclinical influences, beyond diagnosis and severity, on clinical decision making in dermatology: understanding the gap between guidelines and practice,» *British Journal of Dermatology*, pp. 789-799, 2010.
- [88] E. Newnham e A. Page, «Bridging the gap between best evidence and best practice in mental health,» *Clinical Psychology Review*, pp. 127-142, 2010.
- [89] J. Cornu, O. Cussenot, F. Haab e B. Lukas, *European Urology*, pp. 450-456, 2010.
- [90] A. Freeman e K. Sweeney, «Why general practitioners do not implement evidence: qualitative study,» *BMJ*, pp. 1-5, 2001.

- [91] S. Bernheim, J. Ross, H. Krumholz e E. Bradley, «Influence of patients' socioeconomic status on clinical management decisions: a qualitative study,» *Annals of Family Medicine*, pp. 53-59, 2008.
- [92] R. Grol, «Has guideline development gone astray? yes,» *BMJ*, pp. 394-395, 2010.
- [93] R. Mans, M. Schonenberg, W. van der Aalst, G. Leonardi, S. Panzarasa, A. Cavallini e S. Quaglini, «Process mining techniques: an application to stroke care,» 2008.
- [94] J. Weerdt, F. Caron, J. Vanthienen e B. Baesens, «Getting a grasp on clinical pathway data: An approach based on process mining,» *Emerging Trends in Knowledge Discovery and Data Mining*, pp. 22-35, 2013.
- [95] S. Shrivastava e S. Wheeler, «Architectural support for dynamic reconfiguration of distributed workflow applications,» *IEEE Proceedings Software Engineering*, pp. 155-162, 1998.
- [96] M. Beyer, K. Kuhn, C. Meiler, S. Jablonski e R. Lenz, «Towards a flexible, process-oriented it architecture for an integrated healthcare network,» in *Proceedings of the 2004 ACM symposium on Applied computing*, 2004.
- [97] M. A. Grando, D. Glasspool e J. Fox, «etri nets as a formalism for comparing expressiveness of workflow-based clinical guideline languages,» 2009.
- [98] A. Seyfang, B. Martinez-Salvador, R. Serban, J. Wittenberg, S. Miksch, M. Marcos, A. Ten Teije e K. Rosenbrand, «Maintaining formal models of living guidelines efficiently,» *Artificial Intelligence in Medicine*, pp. 441-445, 2007.
- [99] M. Smolko, G. Kaplan e W. Brock, «Location and fate of the nonpalpable testis in children,» vol. 129, n. 6, 1983.
- [100] F. Ferro, A. Lais e L. Gonzalez-Serva, «Benefits and afterthoughts of laparoscopy for the nonpalpable testis,» vol. 156, n. 6.

- [101] P. Jones, «Approaches to orchidopexy,» *British Journal of Urology*, vol. 75, n. 6, pp. 693-696, 1995.
- [102] J. Guo, Z. Liang, H. Zhang, C. Yang, J. Pu, H. Mei, L. Zheng, F. Zeng e Q. Tong, «Laparoscopic versus open orchidopexy for non-palpable undescended testes in children: a systemic review and meta-analysis,» *Pediatric Surgery International Journal*, vol. 27, n. 9, pp. 943-952, 2011.
- [103] C. Gapany, P. Frey, F. Cachat, F. Guidinchet, P. Jichlinski, B. Meyrat, P. Ramseyer, G. Theintz e B. Burnand, «Management of cryptorchidism in children: guidelines.,» *Swiss Med Wkly*, vol. 138, pp. 492-498, 2008.
- [104] C. Domiguez, «Expertise in laparoscopic surgery: anticipation and affordances,» 2001.
- [105] M. Mathers, H. Sperling, S. Ribben e S. Roth, «The undescended testis: Diagnosis, treatment and long-term consequences,» *Dtsch Arztebl Int*, vol. 106, n. 33, pp. 527-532, 2009.
- [106] M. Wood e J. Elder, «Cryptorchidism and testicular cancer: separating fact from finction,» *The Journal of Urology*, pp. 452-461, 2009.
- [107] E. Rogers, S. Teahan, H. Gallacher, M. Butler, R. Grainer, T. McDermott e J. Thornill, «The role of orchiectomy in the management of postpubertal cryptorchidism,» *The Journal of Urology*, pp. 851-854, 1998.
- [108] J. Guo, Z. Liang, H. Zhang, C. Yang, J. Pu, H. Mei, L. Zheng, F. Zeng e Q. Tong, «Laparoscopic versus open orchidopexy for non-palpable undescended testes in children: a systemic review and meta-analysis,» *Pediatric Surgery International*, pp. 943-952, 2011.
- [109] J. Banks, S. Carson, S. Nelson e D. Nicol, *Discrete Event Simulation*, 2000.
- [110] S. Alter, «Information Systems: A Management Perspective,» Addison-Wesley, 1999.
- [111] W. van der Aalst, *Modern Business Process Automation*, Springer, 2010.

[113] Z. Tang, L. Weavind, J. Mazabob, E. Thomas, M. Chu-Weininger e T. Johnson, «Workflow in intensive care unit remote monitoring: A time-and-motion study,» vol. 35, n. 9, 2007.

RINGRAZIAMENTI

In questo giorno così importante sento di dover rivolgere un pensiero a tutte le persone che hanno accompagnato e sostenuto fino alla fine il mio percorso di studi.

Prima di tutto rivolgo un pensiero ai miei nonni, che purtroppo non sono qui a sentirmi, ma che sono stati i primi ad inculcarmi tutti i valori in cui credo, tra cui l'importanza dello studio e della famiglia. Ringrazio i miei genitori, che hanno sempre avuto fiducia in me, dandomi il supporto necessario a raggiungere tutto ciò che ho conquistato oggi; insieme a mia sorella Alessandra, da sempre il mio più solido esempio. Un ringraziamento speciale a mio fratello Giuseppe, che è sempre riuscito a starmi vicino e a farmi sorridere nei momenti difficili.

Ringrazio sentitamente il Prof. Alessandro Pepino, non soltanto per gli insegnamenti che mi ha trasmesso durante questo percorso di ricerca, ma anche per l'entusiasmo e per la fermezza con cui è stato capace di guidarmi nelle esperienze di questi anni.

Con profonda stima ringrazio la Prof.ssa Maria Triassi, per le numerose opportunità di crescita che costantemente sostiene in favore di noi Dottorandi, e per costituire un solido punto di riferimento.

Un ringraziamento speciale va anche al mio amico e collega Adriano Torri, per la disponibilità, lealtà e simpatia che lo caratterizzano, riuscendo a farmi sorridere anche nelle giornate più pesanti.

Rivolgo anche un pensiero a tutti i membri del Dipartimento di Computer Science dell'Università di Eindhoven, con i quali ho vissuto un'esperienza unica ed indimenticabile.

Ringrazio inoltre mia Zia Loredana, che mi ha insegnato a scrivere, i miei zii Mimmo, Maria e Enza - sempre presenti in tutte le occasioni felici e tristi della mia vita - la mia madrina Mariolina - che da 29 anni veglia su di me con costanza e premura - e zia Elvira, che alla sua veneranda età non smette di preoccuparsi per me.

Rivolgo poi un pensiero affettuoso a tutti i miei cugini, presenti e non. In particolare Carmen per avermi sopportata negli ultimi mesi di stesura della tesi, e alla quale mi lega un profondo affetto; Ilaria e Giulia con i rispettivi mariti, da sempre come sorelle/fratelli, e Mario, costante punto di riferimento.

Ringrazio inoltre il mio fidanzato Stefano, che mi sopporta ormai da un anno, sostenendo le mie scelte con dedizione e pazienza.

Un pensiero sentito va poi a tutti i miei amici: in particolare Valentina - la migliore amica di sempre - il mio alter ego Pierluigi, la mia cara Rosaria, la mia amica storica Dionea, e Antonietta.

Con affetto ringrazio la mia collega Giusy, con cui ho avuto il piacere di condividere giornate di lavoro e di divertimento, e con la quale spero di restare in contatto quotidiano anche in futuro.