

POLITECNICO DI MILANO

Polo territoriale di Como

Scuola di Ingegneria dell'Informazione

Corso di Laurea Magistrale in Ingegneria Informatica



Access control for the information model of
business processes

Relatore: Prof. Giuseppe Pozzi

Tesi di Laurea di:

Giorgio Zinetti Matr. 769884

A.A. 2012-2013

POLITECNICO DI MILANO

Polo territoriale di Como

Scuola di Ingegneria dell'Informazione

Corso di Laurea Magistrale in Ingegneria Informatica



Access control for the information model of
business processes

Relatore: Prof. Giuseppe Pozzi

Tesi di Laurea di:

Giorgio Zinetti Matr. 769884

A.A. 2012-2013

Ringraziamenti

Desidero ringraziare il mio relatore, il professore Giuseppe Pozzi, per la grande disponibilità e cortesia dimostratami, nonché per il prezioso aiuto fornitomi nella stesura della tesi, consigli preziosi che mi hanno aiutato ad intraprendere ogni volta le scelte più appropriate. Ringrazio la mia famiglia per avermi permesso di raggiungere questo traguardo e per essermi stata vicina non solo durante i cinque anni universitari ma anche in tutta la mia carriera scolastica.

Giorgio.

Sommario

Ci sono numerosi aspetti da considerare quando si modella un processo aziendale (business process). Fino ad oggi la ricerca si è concentrata sullo studio dello stesso da una prospettiva di controllo di flusso, mentre la modellazione delle informazioni correlate al processo è rimasta affidata ai tradizionali metodi come i modelli entit relazione ER o il classico class diagram standard UML. Gli ultimi citati sono però strumenti che non forniscono metodologie per modellare e collocare le informazioni specifiche per/in un processo aziendale quindi questi due aspetti di modellazione fino ad oggi sono stati visti come due aspetti separati. In particolare i tradizionali metodi di rappresentazione non consentono di rappresentare esplicitamente i vincoli e i permessi sulle informazioni con l'evolversi dei processi. In quest'opera viene proposto un modello a soluzione di questo problema e la sua integrazione in un WfMS.

Abstract

There are a number of aspects to be considered when modeling a business process, and research has mainly focused on the control flow perspective. The conceptual modeling of process-related information has been mainly left to traditional data models, like the ER model or UML class and object diagrams; these, however, are generic data models, and they do not offer means to express information features specific for business processes; as a result, the process structure and information are modeled as separate aspects. In particular traditional models don't allow designer to represent constraints and rights on information according to the workflow. In this work a model to solve this problem is proposed with the integration in a WfMS.

Indice

Introduzione	1
1 Approccio	3
1.1 Obiettivi	3
1.2 Pianificazione	4
2 Stato dell'arte	5
2.1 Data modeling	5
2.2 Data models	6
2.2.1 ER	6
2.2.2 Costrutti ER	7
2.3 Business process modeling	17
2.3.1 BPMN	18
2.3.2 Costrutti BPD	18
2.3.3 Utilizzo della BPMN	23
2.3.4 Livelli di dettaglio	24
2.3.5 Cenni notazioni aggiuntive	26
2.3.6 XPDL	27
2.4 Modelli ibridi	34
2.4.1 Seamless and temoral conceptual modeling	34
3 Motivation scenario (Un caso di studio)	38
3.1 Processo	39
3.1.1 Gli attori del processo	39
3.1.2 I task	40
3.2 Base di dati	41
3.2.1 Entità	42

3.3	BPMN e XPDL	43
3.4	Altri scenari	43
4	Soluzione	45
4.1	Privilegi	48
4.1.1	Privilegi attori	49
4.1.2	Privilegi in base al task	49
4.2	Modello complessivo	50
4.3	XPDL	50
4.3.1	Da ER a XML	51
4.3.2	Notazione vincoli	52
5	Integrazione modello in un WfMS	63
5.1	WfMS	63
5.2	Integrazione	63
6	Conclusioni e sviluppi futuri	65
6.1	Sviluppi futuri	65
6.2	Conclusioni	66
A	BPMN Esempi completi	67
	Bibliografia	89

Elenco delle figure

2.1	Le entità sono rappresentate con dei rettangoli contenente il titolo nome dell'oggetto.	7
2.2	Le relazioni sono rappresentate da delle linee continue, a queste linee si può interporre un rombo per indicare il tipo di relazione. .	8
2.3	E' possibile avere relazioni ricorsive, queste sono indicate facendo tracciando una linea che parte dall'entità e finisce sull'entità stessa.	8
2.4	Esempio di relazione multipla.	9
2.5	Esempio di relazione con cardinalità 0,N a 1,N.	9
2.6	Esempio di relazione con cardinalità 0,N a 1,1.	9
2.7	Esempio di relazione con cardinalità 0,1 a 1,1.	10
2.8	Esempio di relazione ricorsiva con cardinalità.	10
2.9	Esempio di relazione multipla con cardinalità.	11
2.10	Esempio di schema ER con entità relazioni e attributi. Gli attributi sono indicati con degli ovali contenente il nome dell'attributo. . .	12
2.11	Esempio di schema ER con entità con chiave. Per indicare le chiavi il nome dell'attributo viene sottolineato.	14
2.12	Una entità debole viene rappresentata con un doppio rettangolo; la chiave parziale viene sottolineata tratteggiandola. Inoltre, la relazione che la lega all'entità proprietario viene disegnata con un doppio rombo.	15
2.13	L'evento start viene indicato con un cerchietto vuoto.	19
2.14	L'evento intermedio viene indicato con un doppio cerchietto. . . .	19
2.15	L'evento end viene indicato con un cerchietto vuoto con un tratto molto marcato.	20
2.16	L'attività elementare Task viene indicata con un rettangolo, al cui interno viene scritto il nome/titolo del task.	20

2.17	Le attività da intendersi come sottoprocessi, vengono indicate con un rettangolo con un + al loro interno.	20
2.18	Le diramazioni (gateway) sono indicate con un quadrato inclinato di 45 gradi.	21
2.19	Biforcazione con due percorsi alternativi.	22
2.20	Biforcazione o ricongiungimento in due attività parallele si può indicare con un + all'interno del rombo.	23
2.21	I connettori sequenziali vengono indicati con una freccia, che collega i vari task.	23
2.22	Lo scambio di messaggi all'interno dei processi viene indicato con una freccia a linea tratteggiata.	24
2.23	Le associazioni vengono indicate con una freccia avente una linea costituita da puntini.	24
2.24	Corsia di un'unità organizzativa, indicata con un rettangolo che andrà a circoscrivere tutte i task e le decisioni relative all'unità.	25
2.25	Sottocorsie all'interno di una corsia.	25
2.26	Il tipo oggetto dati viene indicato con un disegno stilizzato di un foglio.	26
2.27	I raggruppamenti vengono rappresentati come linee tratteggiate che circondano tutti i processi relativi.	26
2.28	Le note e i commenti vengono segnalate da delle parentesi quadre aperte ed una linea tratteggiata che la collega al punto del processo da commentare.	27
2.29	Tabella eventi bpmn	28
2.30	BPMN, tre task in serie.	29
2.31	BPMN esempio con sdoppiamento in due attività parallele.	31
2.32	BPMN sdoppiamento processo e ricongiungimento.	32
2.33	Lo schema workflow Radiological Exam process proposto come caso di studio.	34
2.34	Il diagramma TimeER modellato per il processo caso di studio. La classica notazione ER è estesa con le specifiche nella dimensione temporale (BT per bitetemporal, VT per valid time, LF per life span)	35
2.35	Rappresentazione grafica di un'entità core.	36
2.36	Esempio finale del modello proposto da Combi-Degani.	37

3.1	Schizzo diagramma processo di certificazione, gli attori coinvolti sono 3: Responsabile qualità, Auditor Consulente e Certificatore.	39
3.2	Use case del caso di studio mostra le funzioni relative al processo dei tre attori.	40
3.3	Esempio di schema ER su cui si basano le normative.	41
3.4	La base di dati completa. La base di dati delle normative vengono estese con la possibilità di tracciare gli audit e memorizzare i compiti assegnati per conformare i punti norma non in regola. . .	42
3.5	BPMN caso di studio disegnato con together workflow editor. . .	44
4.1	Business Process semplificato del caso di studio. Esso verrà posto nel modello in alto e fungerà da linea temporale.	46
4.2	Per identificare il processo attivo esso viene evidenziato colorando lo sfondo di un giallo acceso, questa soluzione per porta con sè limiti di accessibilità. è stato pensato di utilizzare un giallo in quanto è un colore che risalta molto e genera un forte contrasto con il nero dei testi e ne permette una facile lettura.	47
4.3	In questo caso il processo attivo viene semplicemente evidenziato da un tratto più marcato.	47
4.4	Per aumentare ulteriormente l'accessibilità i processi inattivi vengono disegnati con una tonalità di grigio.	48
4.5	Nella parte inferiore del modello è mostrato lo ER al momento in cui è attivo una fase del processo, in questa immagine è riportato lo schema ER completo.	55
4.6	Il privilegio di insert sulle entità è indicato con un più nell'angolo.	56
4.7	Il privilegio di delete sulle entità è indicato con un meno nell'angolo.	56
4.8	Il privilegio di insert e delete è indicato con entrambi i simboli, più e meno nell'angolo in alto a destra.	56
4.9	Per indicare gli attori si utilizza la metodologia UML, quindi una figura di persona stilizzata facile da riprodurre anche a mano libera con al di sotto della stetta il nome dell'attore.	57
4.10	Per indicare l'attore attivo si è pensato di scolorire con una tonalità di grigio gli attori inattivi, facendo risaltare in nero quello attivo.	58

4.11	Per migliorare ulteriormente l'accessibilità del modello gli attori inattivi vengono rimpiccioliti e l'attore attivo viene marcato con un tratto più spesso.	59
4.12	Quest'immagine raccoglie tutti gli elementi introdotti; nella parte superiore presente il modello BPMN semplificato con attivo il task Stesura report e invio a certificatore, sulla destra vengono riportati gli attori interessati a questa fase del processo, in questo caso solo il consulente, e nella parte inferiore viene riportato lo schema ER. Come possiamo vedere in questa fase del processo l'attore consulente ha solo privilegi di read sulle entità del database.	60
4.13	In questa seconda immagine viene riportato la situazione quando il task Esecuzione audit è attiva. Come possiamo vedere gli attori coinvolti sono Manager e Consulente, nell'immagine viene però riportata la situazione relativa al consulente essendo lo stesso evidenziato. Leggendo l'ER notiamo che ha la possibilità di inserire nuovi compiti, eliminarli e commentarli.	61
4.14	In quest'ultimo esempio viene mostrato il modello come sarebbe se comprendesse anche delle feature cromatiche. Il risultato è sicuramente di maggiore impatto.	62
5.1	Esempio di integrazione del modello in eclipse.	64

Elenco delle tabelle

4.1	Esempio di tabella dei permessi relativi ad un dato o a una entità a seconda dell'utente attivo e del task attivo.	50
-----	--	----

Introduzione

Ci sono numerosi aspetti da considerare quando si modella un business process; i più comuni sono il controllo del flusso, le risorse e la gestione delle informazioni. La ricerca si è focalizzata per lo più nel capo del controllo del flusso e un grande numero di linguaggi per definire processi sono stati sviluppati per meglio definire le funzionalità dei processi nei vari domini di applicazione. La modellazione delle risorse e dei dati ha avuto meno attenzione da parte dei ricercatori. In particolare la modellazione concettuale dei dati è stata messa in secondo piano e ci si è affidati ai classici e collaudati metodi tradizionali come il largamente diffuso modello ER (entità-relazione) o l'UML (UNIFIED MODELING LANGUAGE) class diagram. Questi modelli tuttavia possono essere adottati in diverse circostanze essendo molto generici. I classici linguaggi permettono di definire le entità e le loro relazioni ma non offrono la possibilità di dichiarare relazioni-funzioni-features specifiche con i business process, e a quali dati un task ha accesso. Il risultato è che la struttura dei processi e quella dei dati sono modellati come due differenti aspetti a livello concettuale e l'effettivo utilizzo dei dati in un processo di business è solo espresso al livello di implementazione tipicamente attraverso le sue variabili. Quando si modella un processo di business si dovrebbe dare una speciale importanza agli aspetti legati ai permessi e privilegi, quindi privilegi di lettura, scrittura, inserimento e cancellazione. I tradizionali strumenti ER e UML non offrono esplicitamente la possibilità di rappresentare questi vincoli quindi un modello esteso che quindi li modelli fornirebbe uno strumento completo attualmente non esistente. In quest'opera viene definita una connessione tra le prospettive di business process control flow e quella delle informazioni (dati) permettendo ai designer di catturare specifiche informazioni per i processi di business e concettualmente collegarle a informazioni legati alla struttura dei dati e i permessi sugli stessi. In particolare viene proposto un linguaggio che permetta graficamente di indicare il tipo di permesso sulle variabili a seconda della fase del processo e a

seconda dell'attore coinvolto. Oggigiorno i sistemi informativi sono presenti nella totalità delle società. Processi di sviluppo software devono ricalcare le esigenze del business e il trattamento dei dati in maniera appropriata è diventato essenziale. Pensiamo solamente ai problemi legati al trattamento dei dati personali e le complicazioni legate alla privacy, pensiamo alla manipolazione di dati bancari, un singolo errore nelle proprietà di accesso ai servizi potrebbe portare a perdite ingenti e danni gravissimi; per questo e per altri motivi la ricerca deve indagare soluzioni che si adeguino alle moderne esigenze legate al processo di sviluppo software. In questa opera viene proposto un modello e la sua implementazione in un sistema collaborativo WfMS.

Capitolo 1

Approccio

Questo capitolo descrive l'approccio utilizzato durante in il progetto di tesi descrivendone gli obiettivi iniziali e le prime decisioni prese.

1.1 Obiettivi

Per limitare il campo di studio è stato deciso di definire fin da subito gli obiettivi che il modello avrebbe dovuto perseguire. Come obiettivi principali del modello sono stati individuati i seguenti:

- Privilegi su entità e suoi attributi differenti secondo l'attore, cioè la possibilità di capire i diritti sulle entità e le proprietà della stessa da parte dei vari protagonisti del processo.
- Privilegi su entità e suoi attributi secondo la fase del processo. Cioè la possibilità di tracciare i privilegi sulle entità e le proprietà della stessa nello svolgersi del processo.
- Integrazione modello in un WfMS.
- Definizione di un linguaggio utile all'esportazione del modello.

Il modello avrebbe dovuto ricalcare il più possibile i modelli già esistenti per conferire allo stesso intuitività agli utenti dello stesso. L'adozione, laddove possibile, degli standard UML BPMN è sistematica per rendere il modello conforme al mondo sia della modellazione di process che della modellazione di dati.

1.2 Pianificazione

- Per prima cosa è stata fatta una ricerca concernente le principali tecnologie utilizzate nella definizione dei modelli di business process e delle basi di dati, quindi è stata eseguita un'analisi dei costrutti principali di questi modelli per meglio inquadrare gli elementi e le caratteristiche che possono essere riutilizzate nel modello desiderato. Nello specifico sono stati analizzati il modello BPMN per i business process e il classico schema entità relazione per il data modelling. Uno studio approfondito è stato effettuato anche per il classico class diagram UML molto interessante viste le sue proprietà.
- Successivamente sono stati analizzati i lavori che hanno cercato di conciliare i due mondi, le tecniche utilizzate e i risultati ottenuti.
- Per testare il lavoro è stato creato un caso di studio che potesse rispecchiare quelle che sono le caratteristiche che il modello deve amministrare, sulla base del quale poi pensare una soluzione ottimale e di facile lettura.
- Il caso di studio viene modellato secondo gli standard studiati nel capitolo Stato dell'arte
- Il business process è stato modellato con il Together work-flow editor per poterne analizzare l'XPDL e ragionare su come potessero essere integrate nel modello Le novità introdotte dal modello.
- Sulla base delle ricerche effettuate viene pensato un modello che risolve i problemi sollevati ma mantenga un livello di conformità agli standard accettabile.
- Le novità introdotte vengono modellate di modo da poter essere introdotte nella notazione XPDL
- Viene descritto nel dettaglio il modello cercando di renderlo più semplice e chiaro possibile.
- Viene fatta una proposta per una possibile integrazione del modello in un WfMS.
- Vengono tratte le conclusioni e vengono proposte delle possibili ricerche/-lavori futuri per dare continuità al lavoro.

Capitolo 2

Stato dell'arte

Come prima cosa è stata fatta una ricerca sugli strumenti esistenti utilizzati per modellare le strutture di dati e i business process di modo da acquisire una visione molto ampia e poter far tesoro delle tecniche utilizzate in questi due mondi e poterli conciliare.

2.1 Data modeling

Il data modeling è un processo usato per definire e analizzare i requisiti necessari a supportare i processi di business che accompagnano i sistemi informativi con basi di dati. Il processo di data modeling coinvolge professionisti nel design dei dati che lavorano a stretto contatto con gli utilizzatori del sistema informativo e i responsabili del business in analisi.

Ci sono tre diversi modelli prodotti nel processo di progettazione. I requisiti sono inizialmente schematizzati in un modello concettuale che è essenzialmente un set di requirements tecnologici riguardo al sistema di dati usato. Il modello è poi tradotto in un modello logico che può essere tradotto in un database vero e proprio. L'ultima trasformazione prevede il passaggio dal modello logico all'implementazione vera e propria del database organizzato quindi in tabelle, utenze, studi di performance etc.. Il data modeling definisce non solo gli elementi del modello ma anche le relazioni tra gli elementi della struttura.

Le tecniche di data modeling sono codificate e standardizzate in maniera consistente. l'utilizzo di queste tecniche è fortemente consigliato per tutti i processi che richiedono un elevato standard.

Il modello dei dati può essere utilizzato da vari progetti in vari momenti differenti. La modellazione è progressiva quindi non avremo mai uno stadio finale del modello, invece avremo un modello vivente che aderisce alle esigenze del business process e ne segue i cambiamenti. Il modello va quindi conservato in una posizione facilmente accessibile di modo che possa essere recuperato espanso e modificato col tempo.

2.2 Data models

I dati possono essere modellizzati secondo una moltitudine di maniere, secondo il caso in esame si può scegliere il metodo che più si adatta al caso in esame. Questi modelli definiscono la maniera in cui i dati possono essere immagazzinati organizzati e manipolati. Il modello più utilizzato e famoso che viene preso in esame è il modello ER (entità-Relazione).

2.2.1 ER

Il modello Entità-Relazione (modello ER) è un modello concettuale dei dati. Come tale descrive ad alto livello la realtà modellata indipendentemente da come i dati verranno logicamente e fisicamente rappresentati. Il modello ER definisce uno schema concettuale dei dati. Uno schema descrive la struttura o forma dei dati, ma non descrive i dati veri e propri, vale a dire le istanze o occorrenze dello schema. Ad esempio, lo schema ER può dire che l'entità teatro ha gli attributi nome e indirizzo ma non dice quali sono gli effettivi valori di questi attributi. I valori verranno inseriti alla fine della progettazione in fase di popolamento della base di dati. Facendo un paragone con la programmazione orientata agli oggetti, lo schema corrisponde alla nozione di classe e una istanza dello schema è un oggetto della classe. Lo schema ER è costituito da due componenti:

- Diagramma ER: è una rappresentazione diagrammatica della realtà che evidenzia le entità, le relazioni tra entità e gli attributi di entrambi;
- Documentazione: una spiegazione (in linguaggio naturale) dei concetti presenti nel diagramma ER;
- Regole aziendali: sono delle regole che aggiungono dei vincoli sui dati che non sono rappresentabili con il modello ER.

2.2.2 Costrutti ER

Entità

Rappresentano concetti complessi e di rilievo che descrivono classi di oggetti con esistenza autonoma. Una istanza di una entità è un oggetto della classe rappresentata. Ad esempio, possibili entità della rete teatrale sono: teatro, spazio teatrale, dipendente, spettacolo, attore, notizia. Una entità ha un nome univoco all'interno dello schema concettuale e viene rappresentata nel diagramma ER con un rettangolo con il nome dell'entità all'interno:

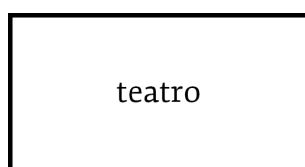


Figura 2.1: Le entità sono rappresentate con dei rettangoli contenente il titolo nome dell'oggetto.

Relazioni

Rappresentano legami logici, significativi per la realtà modellata, tra due o più entità. Una istanza di una relazione è una coppia (ennupla in generale) di istanze di entità partecipanti alla relazione. Formalmente, una relazione è un sottoinsieme del prodotto cartesiano delle classi che la compongono. Essendo un insieme, tutti le istanze di una relazione sono distinte.

Ad esempio, esiste una relazione, che possiamo chiamare luogo, tra le entità teatro e spazio teatrale. Questa relazione associa ad un teatro gli spazi teatrali che il teatro usa per gli spettacoli e viceversa essa associa ad uno spazio teatrale i teatri che usano quello spazio. Ogni relazione ha un nome univoco tra le relazioni e le entità dello schema. E' preferibile usare un sostantivo come nome per non indurre ad individuare un verso nella relazione. Una relazione viene rappresentata mediante un rombo con il suo nome all'interno e da linee che connettono la relazione con le entità componenti:

E' possibile avere relazioni ricorsive, cioè che legano la stessa entità. Ad esempio, la relazione collega definita sull'entità dipendente o la relazione genitore sull'entità persona. La relazione collega è simmetrica, la relazione genitore non lo

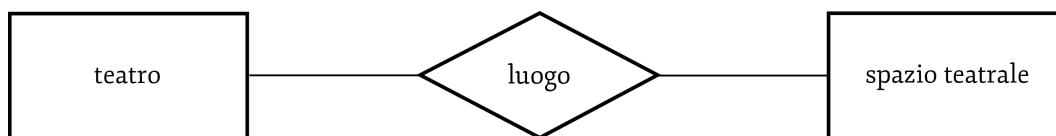


Figura 2.2: Le relazioni sono rappresentate da delle linee continue, a queste linee si può interporre un rombo per indicare il tipo di relazione.

è. Nel caso di relazioni non simmetriche, occorre definire i ruoli di partecipazione dell'entità alla relazione. Nel caso di genitore, i ruoli sono padre e figlio:

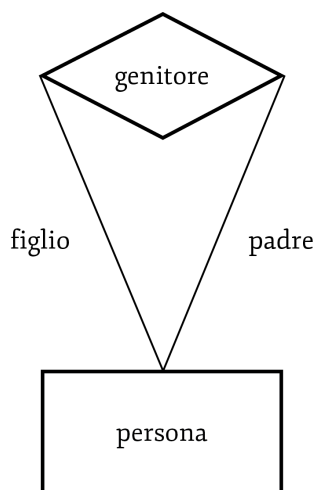


Figura 2.3: E' possibile avere relazioni ricorsive, queste sono indicate facendo tracciando una linea che parte dall'entità e finisce sull'entità stessa.

Inoltre è possibile avere relazioni che associano più di due entità. Ad esempio, la relazione fornitura associa un fornitore che rifornisce di un certo prodotto un qualche dipartimento:

Per ogni entità partecipante ad una relazione viene specificata una cardinalità di relazione. Essa è una coppia di numeri naturali che specifica il numero minimo e massimo di istanze di relazione a cui una istanza dell'entità può partecipare. E' possibile indicare con la costante N un numero generico maggiore di uno quando la cardinalità non è nota con precisione. Nel diagramma ER, la cardinalità di relazione di una entità etichetta l'arco che lega l'entità alla relazione. Seguono alcuni esempi:

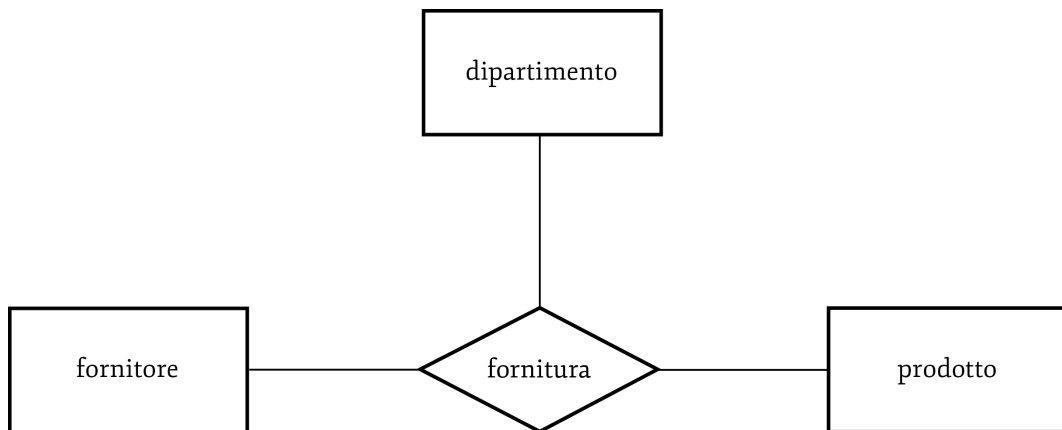


Figura 2.4: Esempio di relazione multipla.

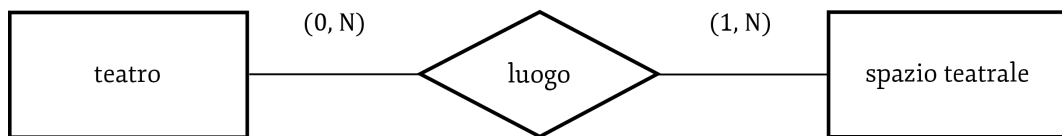


Figura 2.5: Esempio di relazione con cardinalità 0,N a 1,N.

In questo caso un teatro ha nessun, uno o molti spazi teatrali e uno spazio teatrale ha uno o più teatri associati. Questo tipo di relazione in cui entrambe le cardinalità massime sono maggiori di uno si chiama molti a molti. La partecipazione dell'entità teatro si dice opzionale (cardinalità minima uguale a zero) e la partecipazione dell'entità spazio teatrale si dice obbligatoria (cardinalità minima uguale a uno).

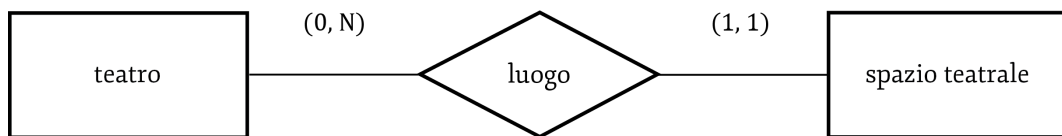


Figura 2.6: Esempio di relazione con cardinalità 0,N a 1,1.

In questo caso un teatro ha nessun, uno o molti spazi teatrali e uno spazio teatrale ha esattamente un teatro associato. Dunque non vi sono spazi condivisi

da più teatri. Questo tipo di relazione in cui una cardinalità massima è pari a uno e l'altra è maggiore di uno si chiama uno a molti.

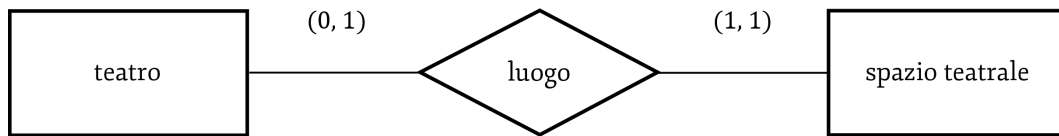


Figura 2.7: Esempio di relazione con cardinalità 0,1 a 1,1.

In questo caso un teatro ha nessun o uno spazio teatrale e uno spazio teatrale ha esattamente un teatro associato. Dunque non vi sono teatri con più spazi e neppure spazi condivisi da più teatri. Questo tipo di relazione in cui entrambe le cardinalità massime sono uguali a uno si chiama uno a uno.

Di seguito proponiamo un esempio di relazione ricorsiva con ruoli e cardinalità. Ogni persona ha esattamente un padre e zero o più figli:

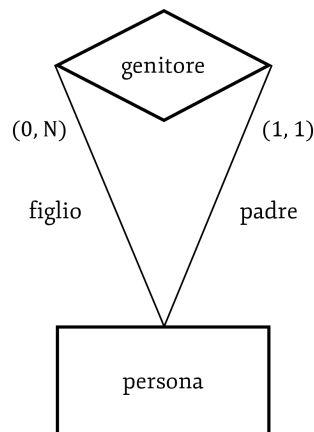


Figura 2.8: Esempio di relazione ricorsiva con cardinalità.

Infine mostriamo un esempio di relazione ternaria con cardinalità:

Si ricordi che la cardinalità di una entità si riferisce al numero di istanze della relazione a cui quella entità può partecipare. Ad esempio, nel caso di fornitore, la cardinalità si riferisce al numero di coppie prodotto e dipartimento che il fornitore rifornisce. Dunque una cardinalità massima pari a uno per l'entità fornitore si ha quando un fornitore rifornisce un solo dipartimento di un solo prodotto. Attributi
Un attributo è un concetto che ha una struttura semplice e non possiede proprietà

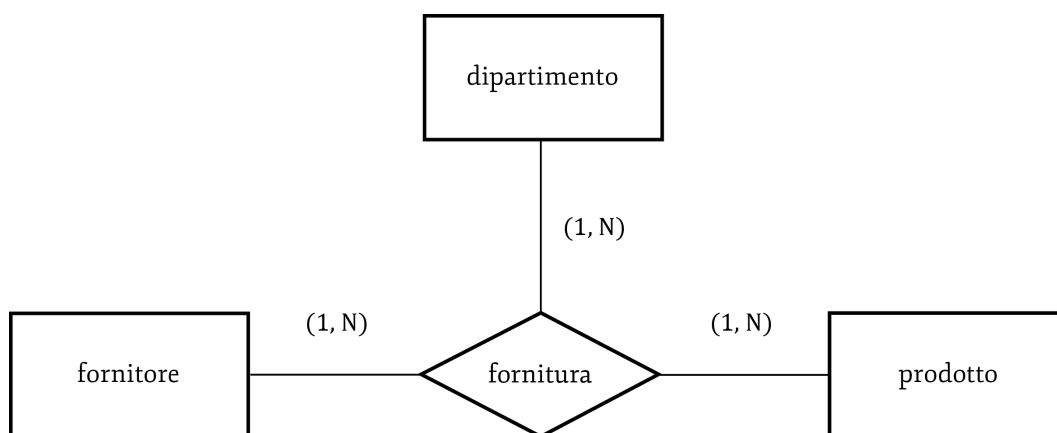


Figura 2.9: Esempio di relazione multipla con cardinalità.

rilevanti associate. Un attributo non ha esistenza autonoma ma è associato ad una entità o ad una relazione. Un attributo può essere semplice, multivalore, composto o calcolato:

- Un attributo semplice ha, per ogni istanza dell'entità associata, un unico valore atomico di un certo tipo di base. Ad esempio, attributi semplici possono essere nome, indirizzo di posta elettronica e pagina web;
- Un attributo multivalore può avere, per ogni istanza dell'entità associata, più valori. Ad esempio, un attributo multivalore potrebbe essere telefono per il quale possono essere noti più numeri;
- Un attributo composto è costituito da più attributi correlati. Ad esempio un attributo composto potrebbe essere indirizzo formato dagli attributi via, numero, e città;
- Un attributo calcolato è un attributo il cui valore è ottenuto calcolandolo da quello di altri attributi. Al esempio l'incasso totale si ottiene moltiplicando il numero di spettatori per il prezzo del biglietto (supponendo che il prezzo sia fisso).

Un attributo può essere inoltre obbligatorio o opzionale. Un attributo obbligatorio deve sempre avere un valore valido. Un attributo opzionale può non avere un valore a causa di uno dei seguenti motivi:

- Il valore dell'attributo non esiste. Ad esempio, nel caso di un attributo email per una persona, la persona non possiede un indirizzo di posta elettronica;
- Il valore dell'attributo esiste ma non è noto. Quindi, la persona ha un indirizzo di posta elettronica ma non ci è noto quale sia;
- Non è noto se il valore dell'attributo esista. Quindi, non sappiamo se la persona abbia un indirizzo di posta elettronica.

Un attributo può essere associato ad una entità o ad una relazione. Nel secondo caso esso assume un valore per ogni coppia (o ennupla) di istanze partecipanti alla relazione. Ad esempio, una relazione che lega i dipendenti ai teatri in cui essi lavorano può avere attributi data di assunzione e ruolo.

Un attributo ha un nome univoco all'interno dell'insieme degli attributi della stessa entità o relazione. Dunque attributi di entità o relazioni diverse possono essere omonimi. Un attributo si rappresenta graficamente con una ellisse (con una doppia ellisse se multivalore, con una ellisse tratteggiata se calcolato) con il nome all'interno collegata all'entità, alla relazione o all'attributo composto di appartenenza. Segue un esempio:

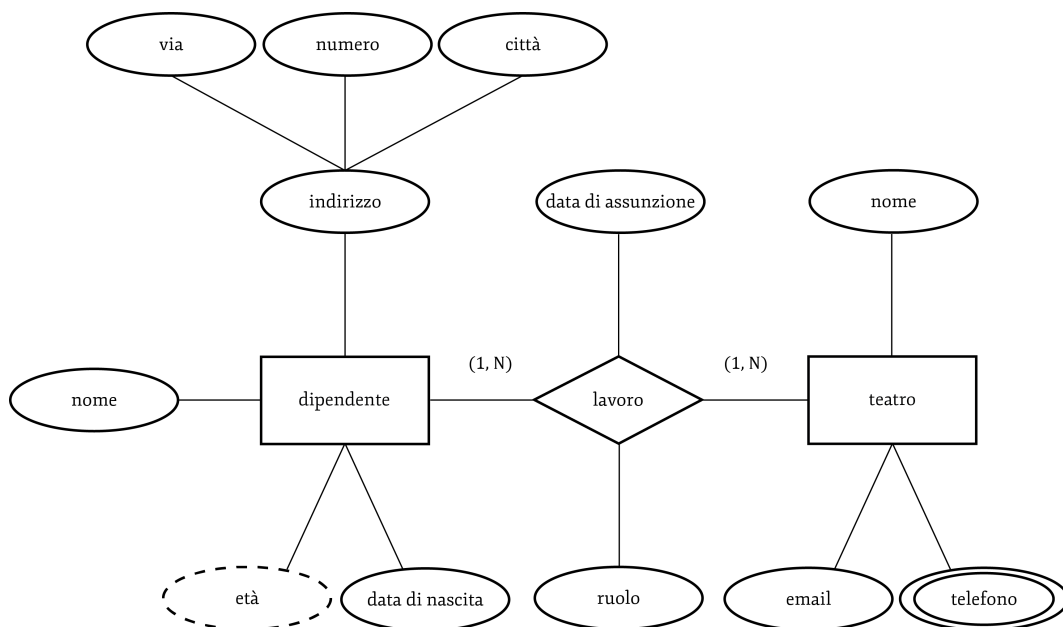


Figura 2.10: Esempio di schema ER con entità relazioni e attributi. Gli attributi sono indicati con degli ovali contenente il nome dell'attributo.

Chiavi

Per ciascuna entità occorre specificare una chiave primaria. Una chiave è un insieme di uno o più attributi con due caratteristiche:

- **Univocità:** i valori degli attributi identificano univocamente le istanze dell'entità nel micro-mondo modellato. Vale a dire che nell'universo del discorso dell'applicazione non possono esistere due istanze diverse con lo stesso valore per la chiave;
- **Minimalità:** rimuovendo un qualsiasi attributo dall'insieme perdiamo il requisito di univocità.

Ogni insieme di attributi che verifica la prima proprietà è detto superchiave. Una chiave è dunque una superchiave minimale. E' possibile che esistano più chiavi per una entità. Tali chiavi sono dette chiavi candidate. Tra queste occorre sceglierne una che è detta chiave primaria. Esistono dei criteri di decisione per la scelta della chiave primaria:

- Gli attributi opzionali non possono far parte della chiave primaria. Dunque chiavi che contengono attributi opzionali non possono diventare primarie;
- sono preferibili, per motivi di efficienza, chiavi piccole, cioè con pochi attributi;
- E' preferibile, per motivi di efficienza, scegliere chiavi con attributi che vengono utilizzate da molte operazioni o da operazioni molto frequenti;

Se non si riesce a trovare una chiave che soddisfa questi requisiti è possibile introdurre un ulteriore attributo sintetico che serve solo ad identificare le istanze dell'entità. Questi attributi sono solitamente detti codici.

Facciamo un esempio. In un opportuno micro-mondo, uno studente può essere identificato dalla chiave matricola e dalla chiave nome e cognome. Possiamo scegliere come chiave primaria la matricola che è più breve. Una superchiave di studente è matricola (o nome e cognome) unita con un qualsiasi insieme di attributi di studente. Si noti che il concetto di chiave è definito rispetto al nel micro-mondo modellato. Vedremo tra breve un esempio in cui matricola non è una chiave per studente. Nel diagramma ER, il nome gli attributi che formano la chiave primaria viene sottolineato:

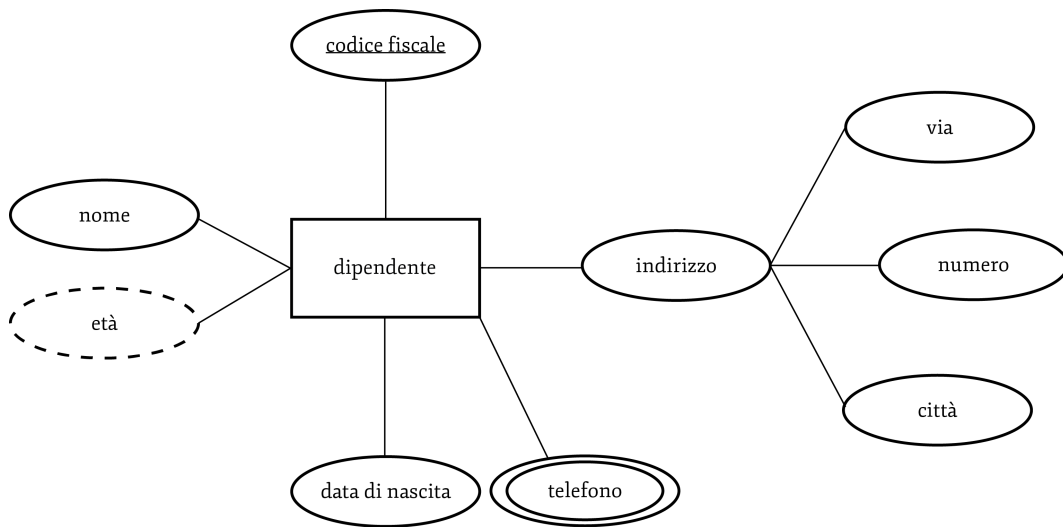


Figura 2.11: Esempio di schema ER con entità con chiave. Per indicare le chiavi il nome dell'attributo viene sottolineato.

Talvolta una entità non ha una chiave ma è identificata componendo una propria chiave parziale con la chiave di un'altra entità alla quale è associata mediante una relazione. L'entità priva di chiave propria si chiama entità debole, l'entità associata a questa si chiama entità proprietario e la relazione che le lega si dice identificante. Affinchè l'entità debole possa essere univocamente identificata, essa deve partecipare alla relazione identificante con vincolo di partecipazione (1,1). Vale a dire, per ogni istanza dell'entità debole, deve esistere esattamente una istanza dell'entità proprietario (che la identifica). Una identificazione esterna può coinvolgere più entità proprietarie. Inoltre, una entità proprietario può essere a sua volta debole purchè non si formino cicli di entità deboli.

Ad esempio, in una base di dati che modella una rete di università, vi sono le entità studente e università. Ogni studente è dotato di una matricola, che lo identifica solo all'interno dell'università in cui studia. Lo studente viene identificato univocamente componendo la matricola con la chiave dell'entità università (ad esempio nome). In tal caso, l'entità debole è studente, l'entità proprietaria è università, la chiave parziale di studente è matricola e la chiave di studente è matricola e nome dell'università.

Una entità debole viene rappresentata con un doppio rettangolo; la chiave parziale viene sottolineata tratteggiandola. Inoltre, la relazione che la lega all'entità

proprietario viene disegnata con un doppio rombo. Segue un esempio:

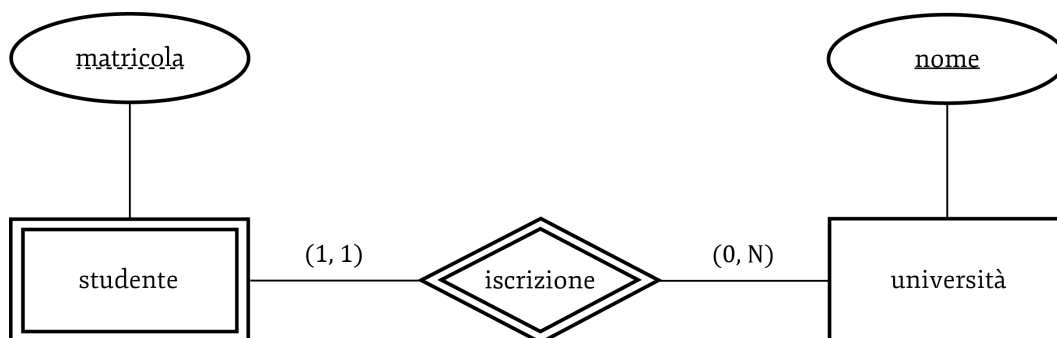


Figura 2.12: Una entità debole viene rappresentata con un doppio rettangolo; la chiave parziale viene sottolineata tratteggiandola. Inoltre, la relazione che la lega all'entità proprietario viene disegnata con un doppio rombo.

Specializzazioni

Talvolta una entità può essere specializzata in una o più sotto-entità. Le sotto-entità ereditano dall'entità genitore i suoi attributi e le sue relazioni. Inoltre, le sotto-entità possono definire nuovi attributi e nuove relazioni non presenti nell'entità genitore. L'insieme delle istanze di una sotto-entità è un sottoinsieme dell'insieme delle istanze dell'entità genitore. Il concetto di sotto-entità è affine al concetto di sotto-classe nei linguaggi di programmazione orientati agli oggetti. Talvolta, il modello ER esteso con la specializzazione prende il nome di modello Entità-Relazione esteso (Enhanced Entity-Relationship Model, EER)

Vediamo un esempio. Una persona è identificata da codice fiscale, nome e cognome e ha alcuni recapiti telefonici, di cui di vuole registrare il numero, l'operatore e il tipo (fisso o mobile). Uno studente universitario è una persona per la quale si vuole memorizzare la matricola e il corso di laurea a cui è iscritto. Uno corso di laurea è identificato da nome, facoltà e università. Possiamo rappresentare questi requisiti in questo diagramma ER esteso con la specializzazione.

L'entità studente eredita dall'entità persona gli attributi codice fiscale, nome e cognome e la relazione recapito. In particolare la chiave di studente è quella di persona, cioè codice fiscale. Inoltre, studente definisce un nuovo attributo matricola e una nuova relazione iscritto. Si noti il simbolo di sottoinsieme sull'arco che connette l'entità persona alla sotto-entità studente.

Una entità può specializzarsi in più sotto-entità. La specializzazione è totale se ogni istanza dell'entità genitore corrisponde ad una istanza di qualche sotto-entità, oppure parziale altrimenti. Una specializzazione è disgiunta se gli insiemi di istanze delle sotto-entità sono disgiunti, altrimenti è sovrapposta. Ad esempio, supponiamo di estendere il precedente diagramma con l'entità lavoratore, sotto-entità di persona. Un lavoratore è caratterizzato da una professione e da un impiego presso un'azienda. Il diagramma ER risultante è il seguente.

La lettera O all'interno del cerchio da cui partono gli archi verso le sotto-entità studente e lavoratore sta ad indicare che la specializzazione è di tipo sovrapposto (overlapping); infatti esistono studenti lavoratori. Per specializzazioni di tipo disgiunto si usa la lettera D (disjoint). Le specializzazioni totali vengono disegnate con una doppia linea che connette l'entità genitore al cerchio di diramazione dei collegamenti alle sotto-entità, quelle parziali da una linea singola. Nel nostro caso, la specializzazione è parziale in quanto esistono persone che non sono né studenti e né lavoratori.

Infine, una sotto-entità può avere più entità genitore. Essa eredita gli attributi e le relazioni da tutti i genitori. In tal caso occorre fare attenzione a non ereditare due volte lo stesso attributo. Questo accade se le entità genitore sono a loro volta sotto-entità di una stessa entità nonno, e quindi ereditano le stesse proprietà da quest'ultima. Inoltre occorre rinominare eventuali attributi diversi ma omonimi. Questo accade se l'insieme unione di tutti gli attributi delle entità genitore contiene attributi differenti ma omonimi. L'ereditarietà multipla generalmente non viene ammessa per le classi dei linguaggi orientati agli oggetti.

Estendiamo il precedente diagramma con l'entità studente-lavoratore, sotto-entità di studente e di lavoratore. Per uno studente lavoratore viene specificata una percentuale di rivalutazione del voto di laurea, in funzione dell'intensità dell'impegno lavorativo. Otteniamo il seguente diagramma ER.

Uno studente lavoratore eredita gli attributi e le relazioni di persona (una sola volta), quelli di studente e quelli di lavoratore. Inoltre definisce un nuovo attributo rivalutazione. Il modello ER può essere descritto in termini di sé stesso mediante il seguente diagramma ER. Si noti che l'informazione sulla chiave è stata modellata mediante un attributo chiave della relazione appartenenza-E che associa gli attributi alle entità corrispondenti. Possiamo immaginare che, per ogni coppia entità-attributo della relazione tale attributo abbia valore vero se e soltanto se l'attributo fa parte della chiave dell'entità

2.3 Business process modeling

Manager ed analisti tendono a migliorare l'efficienza e l'efficacia dei processi, ovvero a ridurre i costi e ad accrescere la qualità intesa come soddisfazione del cliente. In questo campo, il Business Process Modeling (BPM) è l'attività di rappresentazione dei processi aziendali nelle due ottiche:

- La situazione attuale, detta as-is
- La situazione futura desiderata, detta to-be

La mappatura dei processi reali (as-is) e di quelli a tendere (to-be) sono due attività di analisi nettamente distinte, che portano a definire i miglioramenti necessari per passare dai processi rilevati nell' as-is a quelli formalizzati nel to-be. Gli interventi possono essere di tipo incrementale ed essere inclusi nell'ambito del BPM, oppure di tipo radicale, aprendo così la tematica della reingegnerizzazione dei processi aziendali (Business Process Reengineering o BPR). Gli interventi possono riguardare sia la tecnologia che l'organizzazione, e comportano normalmente anche una attività di formazione sui nuovi processi. Esistono software proprietari di modellazione dei processi, quali ADONIS, Aris, Pro2Work, IBPM e SYSTEM ARCHITECT (Telelogic-IBM), che talora garantiscono un'interoperabilità fra standard chiusi e proprietari, e con gli standard aperti di modellazione, in modo da evitare una costosa perdita di informazione nella migrazione dei dati da un linguaggio all'altro. Il software implementa una metodologia proprietaria organizzativa e di modellazione, fatta di particolari oggetti e regole, che è embedded (incapsulata) nel prodotto. L'utilizzo della metodologia è legato all'acquisto di licenze, con profilo di utente e come amministratore, del relativo prodotto. I linguaggi possono essere uno strumento di rappresentazione dei processi e supporto decisionale ai manager, ed un potente tool di programmazione. In questo caso, mentre il processo viene pensato e disegnato per via grafica, il tool genera parti del codice necessario all'automazione di processi esistenti (nell'ambito del Workflow e del Work Force Automation) o all'esecuzione del nuovo processo. Fra questi linguaggi, ricordiamo il Business Process Modeling Notation (BPMN), Business Process for Execution Language (BPEL), Unified Modeling Language (UML).

2.3.1 BPMN

La notazione BPMN (<http://www.bpmn.org>) è sviluppata dalla Business Process Management Initiative e dallo Object Management Group (<http://www.omg.org>), associazioni no-profit che raccolgono operatori nel campo dell'informatica e dell'analisi organizzativa. La BPMN ha come obiettivo fornire uno standard di rappresentazione efficace facile da utilizzare e da comprendere da parte degli utenti business interessati al problema della modellazione, progettazione ed eventuale informatizzazione dei processi aziendali: analisti di processo che costruiscono le bozze iniziali dei processi organizzativi in esame o da progettare, programmatori e sviluppatori delle applicazioni informatiche per la gestione di tali processi, e infine manager e dirigenti responsabili della gestione e del monitoraggio dei processi stessi. La BPMN è sostanzialmente una derivazione del formalismo dei flow chart ma con alcune aggiunte e modificazioni che permettono di superarne alcuni limiti nella modellazione dei processi aziendali. Permette di costruire dei diagrammi di processo (BPD Business Process Diagram) che rappresentano in pratica dei grafi o reti costituiti da oggetti rappresentati dalle attività di processo, collegati da flussi di controllo che definiscono la relazione logica, le dipendenze e l'ordine di esecuzione delle attività stesse. La versione 1.0 dello standard è stata rilasciata nel 2004, mentre la versione 2.0 è oggi (fine 2010) disponibile in una versione beta. La BPMN consente spesso una stretta integrazione con i sistemi di sviluppo software. Sono infatti disponibili applicazioni che consentono al modellista di rappresentare i dettagli di un processo tramite BPMN e traducono poi tale modello in programmi software per la gestione del processo stesso.

2.3.2 Costrutti BPD

Un BPD si costruisce utilizzando alcuni elementi grafici la cui caratteristica principale è la semplicità interpretativa per gli analisti di business e dei processi organizzativi. La notazione richiama anche, come detto, simboli classici del flow-chart, con attività rappresentate tipicamente da rettangoli e punti di decisione da rombi. Lo standard BPMN definisce alcuni elementi grafici di base, generalmente sufficienti per modellizzare una vasta casistica di processi, e ai quali comunque si possono aggiungere integrazioni ed elementi addizionali per dare maggiore efficacia rappresentativa nei casi di processi molto complessi ma senza modificare

l'impostazione base della notazione usata. Le quattro categorie fondamentali di elementi grafici sono le seguenti:

- Elementi di flusso (flow object);
- Connettori (connecting object) ;
- Corsie (swimlane)
- Artefatti (artifact)

Elementi di flusso

Evento: è rappresentato da un cerchio e rappresenta qualcosa che avviene nel corso di un processo. Gli eventi possono avere una causa che li determina (trigger) e un possibile esito (result). Ci sono tre tipi di eventi a seconda della loro collocazione all'interno del flusso di un processo: inizio (start), evento intermedio (intermediate) e termine (end)

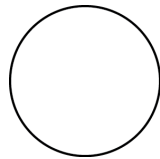


Figura 2.13: L'evento start viene indicato con un cerchietto vuoto.

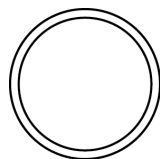


Figura 2.14: L'evento intermedio viene indicato con un doppio cerchietto.

Attività (activity): rappresentata con un rettangolo smussato, indica genericamente un compito o operazione svolti all'interno del processo considerato. Un'attività può rappresentare un compito elementare e atomico (task) ossia non ulteriormente scomposto nell'analisi del processo in questione, oppure un sottoprocesso che potrà venire ulteriormente scomposto nei suoi task elementari. In

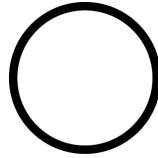


Figura 2.15: L'evento end viene indicato con un cerchietto vuoto con un tratto molto marcato.

questo secondo caso si usa apporre il simbolo + nella parte bassa centrale del rettangolo.

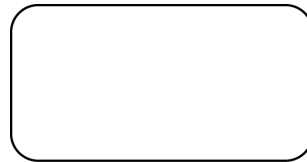


Figura 2.16: L'attività elementare Task viene indicata con un rettangolo, al cui interno viene scritto il nome/titolo del task.

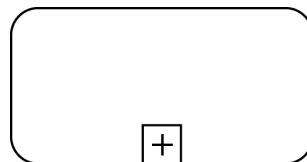


Figura 2.17: Le attività da intendersi come sottoprocessi, vengono indicate con un rettangolo con un + al loro interno.

Diramazione (gateway): simboleggiata con un rombo, definisce i punti del processo in cui i flussi delle attività divergono oppure convergono. è utilizzata per rappresentare i tradizionali punti di decisione come nei classici flow chart, ma anche semplici biforcazioni del flusso delle attività in attività parallele o viceversa il ricongiungimento di attività parallele in un flusso unico. Opportuni simboli di chiarimento potranno specificare il tipo di meccanismo di controllo dei flussi di attività che si biforcano o convergono. Ad esempio, una biforcazione o icongiungimento in due attività parallele si può indicare con un + all'interno del rombo (NB: versione BPMN 2.0).

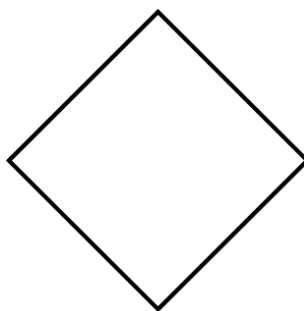


Figura 2.18: Le diramazioni (gateway) sono indicate con un quadrato inclinato di 45 gradi.

Connettori

Se in un processo gli elementi di flusso (eventi, attività o diramazioni) sono ciò che avviene effettivamente, essi devono essere logicamente collegati tra loro. A ciò servono i connettori. Connettore sequenziale (sequence flow): si disegna con una freccia piena ed è usato per indicare l'ordine logico-sequenziale tra attività o eventi di un processo.

Flusso di messaggio (message flow): rappresentato da una linea tratteggiata con una freccia vuota, simboleggia il fatto che un messaggio viene scambiato tra due diverse attività o entità partecipanti al processo, una che trasmette e l'altra che riceve il messaggio.

Associazione (association): indicata con una linea a puntini e una freccia a punta aperta, è usata per indicare un semplice legame tra dati, testi e altri oggetti. Sono usate anche per indicare gli input e gli output delle attività.

Corsie

Servono a specificare l'organizzazione delle attività e dei relativi flussi in gruppi diversi in relazione alle funzioni organizzative coinvolte, alle responsabilità operative e di gestione, ecc. La BPMN ha due tipi di corsie. Corsia di un'unità organizzativa (pool): rappresenta un'entità organizzativa ben definita che svolge un proprio processo eventualmente interagendo con altre unità organizzative. La notazione grafica è un rettangolo e serve a distinguere ciò che viene svolto da quella unità organizzativa rispetto a ciò che è responsabilità di altri e sarà contenuto in altre corsie.

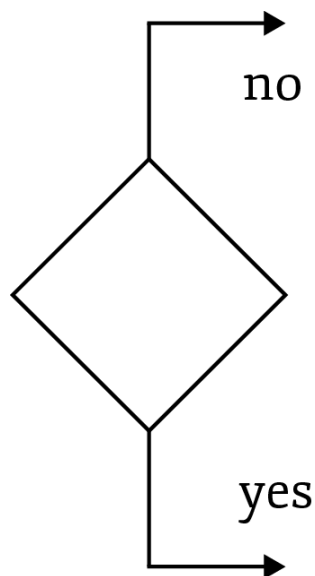


Figura 2.19: Biforcazione con due percorsi alternativi.

Le corsie sono spesso utilizzate nel caso di macroprocessi che coinvolgono due o più entità di business distinte e vengono fisicamente separate nel BPD. Le attività contenute in ogni corsia sono considerate come facenti parte di singoli processi separati che però interagiscono. Cos può accadere pertanto che il flusso di processo di ogni attività rimanga interamente dentro la propria corsia, ma che ci siano connessioni come ad es. messaggi. Sottocorsie (lane): sono una suddivisione della corsia di un'unità in partizioni che coprono l'intera lunghezza della corsia, verticalmente o orizzontalmente. Sono usate per specificare ulteriormente l'organizzazione delle attività nel processo rispetto alle componenti organizzative.

Si noti la differenza essenziale nell'uso di corsie e di sottocorsie. Mentre le corsie sono destinate a rappresentare processi distinti effettuati da entità diverse ma che interagiscono, le sottocorsie sono usate più efficacemente per rappresentare lo stesso processo il cui flusso viene spezzato in sottoprocessi o attività di cui sono responsabili specifiche unità organizzative o individui.

Artefatti

Gli artefatti costituiscono ulteriori simboli che possono essere aggiunti a un BPD per specificare ulteriori oggetti o entità che hanno rilevanza per un processo, chiarire raggruppamenti logici o operativi, aggiungere annotazioni e spiegazioni,

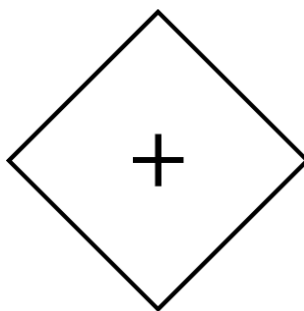


Figura 2.20: Biforcazione o ricongiungimento in due attività parallele si può indicare con un + all'interno del rombo.



Figura 2.21: I connettori sequenziali vengono indicati con una freccia, che collega i vari task.

ecc. Dati (data object): servono a descrivere i tipi di dati che sono necessari o prodotti da un'attività. Sono collegati alle attività attraverso il connettore associazione.

Raggruppamento (group): rappresentato da un rettangolo arrotondato con tratto punto-linea, serve a scopo di specificazione, analisi e documentazione. In altri termini serve a chiarire ulteriormente a chi legge il BPD l'organizzazione del processo, ma non fa parte di per sé dei flussi o del contenuto del processo stesso.

Nota (annotation): permette al modellista di aggiungere un testo di chiarimento o commento al fine di fornire ulteriori informazioni a chi deve interpretare il BPD.

In alcuni casi i modellisti creano e aggiungono propri simboli per fornire ulteriori dettagli al BPD. Tali simboli, comunque, non modificano la struttura base del processo, che è rappresentata dalle attività, dalle diramazioni, dai connettori.

2.3.3 Utilizzo della BPMN

In generale, esistono due tipi di diverse situazioni in cui la BPMN viene utilizzata. La prima si riferisce al caso dei processi che si svolgono interamente in azienda; possibili esempi sono i diagrammi di fig. 10 e di fig. 14. I flussi delle attività

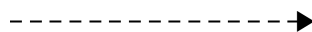


Figura 2.22: Lo scambio di messaggi all'interno dei processi viene indicato con una freccia a linea tratteggiata.

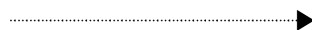


Figura 2.23: Le associazioni vengono indicate con una freccia avente una linea costituita da puntini.

rimangono essenzialmente entro la corsia del processo non attraversando mai i confini dell'unità organizzativa coinvolta. Flussi di messaggio possono peraltro attraversare i confini della corsia ovvero dell'unità organizzativa. Un classico caso di processi interni così modellizzati è quello dei processi che sono da considerarsi privati ossia esclusivamente interni a un'azienda o unità organizzativa: le attività interne non sono direttamente visibili dall'esterno (ad es. da un cliente). Il secondo tipo di situazioni si riferisce a processi collaborativi tra due o più entità di business (aziende, organizzazioni, unità, ecc.). I diagrammi assumono il punto di vista generale di una terza parte come osservatore esterno. Ciascuna entità sviluppa un proprio processo ma ci sono momenti di interazione (ad es. scambi di messaggi) tra attività dei distinti processi: tali momenti di interazione sono quelli che sono visibili dall'esterno ossia anche senza che siano noti i dettagli interni delle attività di ciascun processo. Nel caso dei processi collaborativi in genere ciascuno di essi è modellato a un livello di dettaglio non troppo spinto, ossia si rimane agli elementi generali del processo (naturalmente sono possibili eccezioni). Un esempio di BPD di questo tipo è quello presentato prima in fig. 13.

2.3.4 Livelli di dettaglio

Come già in altri tipi di notazione (flow chart, IDEF, ecc.), anche usando la BPMN si può scegliere di modellare un processo spingendosi a livelli di dettaglio molto spinti (ad es. la singola operazione o elaborazione elementare) o rimanere a livelli più elevati (macroattività, ecc.). Andare a livelli di dettaglio più profondi significa in sostanza scomporre le attività nelle singole sotto-attività, e così via.

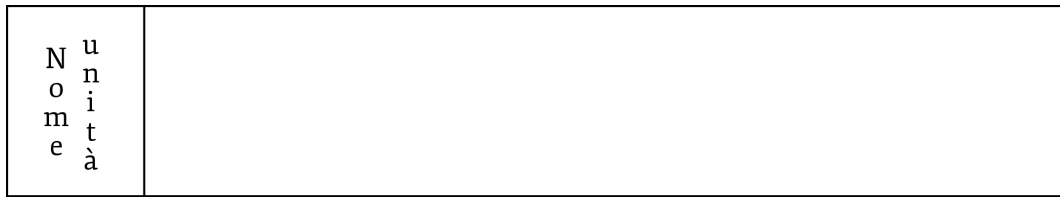


Figura 2.24: Corsia di un'unità organizzativa, indicata con un rettangolo che andrà a circoscrivere tutte i task e le decisioni relative all'unità.

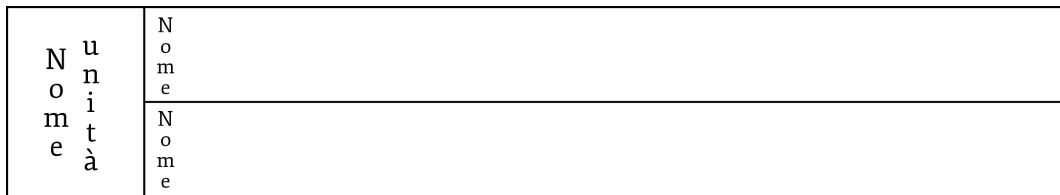


Figura 2.25: Sottocorsie all'interno di una corsia.

La scelta del livello di dettaglio dipende da molti fattori, e non è sempre facile. Innanzitutto dipende dall'obiettivo dell'analista: in generale, se si deve fornire un BPD che sia direttamente di utilizzo per la programmazione potrebbe essere necessario spingersi a un livello molto profondo, mentre se ci si limita a una descrizione del processo che si focalizza su altri aspetti (ad es. solamente i punti di interazione con altri processi) in tal caso il dettaglio di alcune sotto-attività potrebbe essere trascurato. Vi sono poi situazioni in cui una scomposizione dettagliata delle sotto-attività o operazioni che compongono una certa attività è inutile o addirittura impossibile: ad es. quando non è possibile scomporre in modo chiaro un'attività in una sequenza logica di sotto- operazioni che si ripetono sempre nello stesso modo. Ad es. vi sono attività che comportano decisioni o valutazioni umane non formalizzabili a priori, o incorporano un determinato grado di incertezza o variabilità che l'operatore risolve al momento sulla base della propria esperienza; in tali casi l'analista può dover fermare la scomposizione del processo a livello di una descrizione generale, non potendo spingersi oltre. La scomposizione a livelli di dettaglio successivi è un'operazione che spesso si svolge per gradi: si disegna innanzitutto un diagramma a livello di macroattività, e poi lo si scompone successivamente. Nelle figg. 18 e 19, riprese da White 2004,

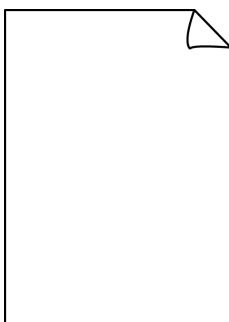


Figura 2.26: Il tipo oggetto dati viene indicato con un disegno stilizzato di un foglio.



Figura 2.27: I raggruppamenti vengono rappresentati come linee tratteggiate che circondano tutti i processi relativi.

si evidenzia il BPD di un processo a modesto livello di dettaglio che poi viene scomposto.

2.3.5 Cenni notazioni aggiuntive

Nei diagrammi più completi possono comparire altri elementi che specificano alcuni dettagli del processo, così come indicato nelle varie release della notazione BPMN. I simboli che specificano i diversi tipi di evento che possono innescare un qualche tipo di attività o decisione (eventi catching con il simbolo dentro il cerchio di colore chiaro) oppure che sono da esse generati (eventi throwing - con il simbolo dentro il cerchio di colore scuro). Gli eventi di inizio (start) possono essere solo catching, e quelli di fine (end) solo throwing, mentre quelli intermedi possono essere entrambi: in corrispondenza a questi il cerchio esterno sarà a linea singola sottile, a linea singola spessa, oppure a doppia linea. Vediamo nel dettaglio il significato dei possibili eventi aggiuntivi: - messaggio (message):

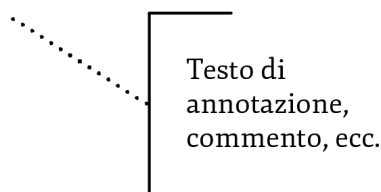


Figura 2.28: Le note e i commenti vengono segnalate da delle parentesi quadre aperte ed una linea tratteggiata che la collega al punto del processo da commentare.

l'evento consiste nella ricezione o invio di un messaggio - timer (non per eventi end): viene specificato che deve passare un certo periodo di tempo oppure si deve attendere un certo momento (data/ora) perché il processo continui - errore (error non per eventi start): significa una situazione anomala che comporta un evento di gestione del processo (che può essere genericamente descritto a margine), l'interruzione prematura del processo stesso, o un'attività non specificata - cancellazione (cancel non per eventi start): tipico in processi che riguardano transazioni; questo evento significa la cancellazione della transazione stessa, con il possibile invio di messaggi alle parti coinvolte. - compensazione (compensation): il verificarsi di questo evento significa la necessità che una certa attività complementi e compensi lo svolgimento di un'altra attività - condizione (conditional non per eventi end): è usato per determinare l'avvio di una certa attività nel momento in cui si verifichi una condizione esterna al processo (ad es. il cambiamento nello stato dell'ambiente) - legame (link non per eventi start): servono per collegare due sezioni diverse di un processo o denotare cicli - segnalazione (signal): significa l'emissione di un segnale che determina l'inizio di un'attività o ne è l'esito - terminazione (terminate solo per eventi end): tutte le attività del processo terminano istantaneamente al verificarsi di questo evento - multiple: significa che l'avvio o termine di un'attività sono legati al verificarsi di molteplici eventi insieme, che possono essere specificati a margine

2.3.6 XPDL

L'XPDL (XML Process Definition Language) è un linguaggio di markup ideato per la definizione dei processi di lavoro, standardizzato dalla WFMC (WorkFlow Management Coalition). Per processo di lavoro si intende l'insieme di operazioni


























	start	intermediate	end
basic			
message			
timer			
rule			
exception			
cancellation			
compensation			
link			
multiple			
termination			

Figura 2.29: Tabella eventi bpmn

che servono per implementare un'idea e portarla alla sua realizzazione utilizzando le risorse (umane e fisiche) all'interno di un'azienda. L'XPDL è tratto dall'XML e definisce un schema standard per la rappresentazione di un processo di business attraverso un workflow. L'obiettivo di tale standardizzazione è quello di fornire un punto di riferimento unico per la rappresentazione dei processi di lavoro e per garantire l'interoperabilità tra le applicazioni software che gestiscono tali processi.

XPDL è usato da più di 80 differenti prodotti oggi giorno per la comunicazione di modelli di processi. Più le società andranno ad utilizzare linguaggi di modellazione di business process la necessità di avere un linguaggio comune di comunicazione sarà sempre più utile.

Di seguito sono riportati alcuni esempi di XPDL, il primo mostra una semplice sequenza tra dall'attività A all'attività B.

Il tag `xpdl` indica l'inizio di un file XPDL, all'interno dello stesso si definiscono gli attributi di accesso, l'id e il nome del processo.

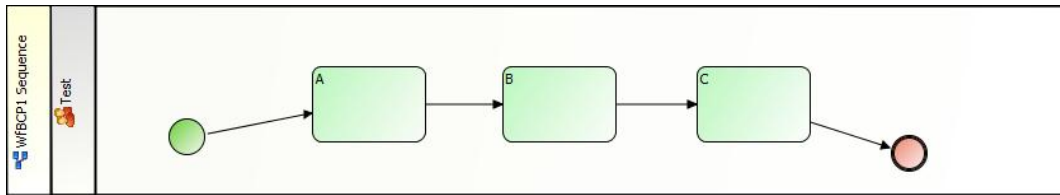


Figura 2.30: BPMN, tre task in serie.

```

1 <xpdl:WorkflowProcess AccessLevel="PUBLIC" Id="WfBCP1_Sequence"
  Name="WfBCP1_Sequence">
2   ...
  
```

Dopo di che si apre un tag header in cui si definiscono data di creazione e descrizione.

```

1
2 ..
3   <xpdl:ProcessHeader DurationUnit="D">
4     <xpdl:Created>2004-08-31 22:28:17</xpdl:Created>
5     <xpdl:Description>execute activities A, B and C in
      sequence</xpdl:Description>
6   </xpdl:ProcessHeader>
7   <xpdl:RedefinableHeader PublicationStatus="UNDER.TEST">
8     <xpdl:Author>Giorgio Zinetti</xpdl:Author>
9   </xpdl:RedefinableHeader>
10
11 ...
  
```

A questo punto inizia la vera e propria definizione del processo, si inizia de-

finendo le varie attività all'interno di un tag `Activities`, delle stesse si definisce un nome, una descrizione, un ID e l'attore che copie l'attività.

```
1
2  ....
3
4  <xpdl:Activities>
5      <xpdl:Activity Id="A3" Name="A">
6          <xpdl:Description>Just press 'Complete' button</
          xpdl:Description>
7          <xpdl:Implementation>
8              <xpdl:No />
9          </xpdl:Implementation>
10         <xpdl:Performers>
11             <xpdl:Performer>test</xpdl:Performer>
12         </xpdl:Performers>
13         <xpdl:NodeGraphicsInfos>
14             <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
              FillColor="187,247,190" Height="60" IsVisible=
              "true" LaneId="workflow_patterns_pool1_lane1"
              ToolId="JaWE" Width="90">
15                 <xpdl:Coordinates XCoordinate="206"
                  YCoordinate="48" />
16             </xpdl:NodeGraphicsInfo>
17         </xpdl:NodeGraphicsInfos>
18     </xpdl:Activity>
19
20  .....
```

Dopo aver definito tutte le attività si passa alle transizioni, all'interno di un tag `transitions` si definisce il tipo di transizione e le attività coinvolte.

```
1
2 ...
3
4 <xpdl:Transitions>
5   <xpdl:Transition From="A3" Id="WfBCP1.Sequence.Tra1" Name=
6     " Transition" To="B5">
7     <xpdl:ConnectorGraphicsInfos>
8       <xpdl:ConnectorGraphicsInfo FillColor="0,0,0"
9         IsVisible="true" Style="NO_ROUTING_ORTHOGONAL"
10        ToolId="JaWE" />
11     </xpdl:ConnectorGraphicsInfos>
12   </xpdl:Transition>
13 </xpdl:Transitions>
```

Il secondo esempio mostra una sequenza che si divide in due attività parallele B e C.

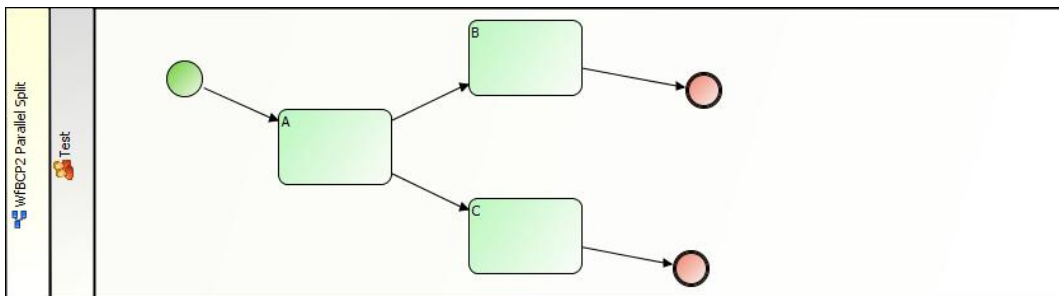


Figura 2.31: BPMN esempio con sdoppiamento in due attività parallele.

Le differenze nell'XPDL sono a livello di transazioni, i task vengono definiti in modo standard, ovviamente si modificheranno solo le destinazioni delle transazioni.

```
1    ...
2    <xpdl:Transitions>
3        <xpdl:Transition From="A4" Id=" WfBCP2_Parallel_Split_Tra1"
4            Name=" Transition" To="B6">
5            <xpdl:ConnectorGraphicsInfos>
6                <xpdl:ConnectorGraphicsInfo FillColor="0,0,0"
7                    IsVisible=" true" Style="NO.ROUTING.ORTHOAGONAL"
8                    ToolId="JaWE" />
9            </xpdl:ConnectorGraphicsInfos>
10        </xpdl:Transition>
11        <xpdl:Transition From="A4" Id=" WfBCP2_Parallel_Split_Tra2"
12            Name=" Transition" To="C5">
13            <xpdl:ConnectorGraphicsInfos>
14                <xpdl:ConnectorGraphicsInfo FillColor="0,0,0"
15                    IsVisible=" true" Style="NO.ROUTING.ORTHOAGONAL"
16                    ToolId="JaWE" />
17            </xpdl:ConnectorGraphicsInfos>
18        </xpdl:Transition>
19    </xpdl:Transitions>
20    ...
```

Il terzo esempio mostra una scelta esclusiva (XOR) sempre tra le attività B e C. In questo esempio viene introdotto il task per identificare un merging di due percorsi paralleli.

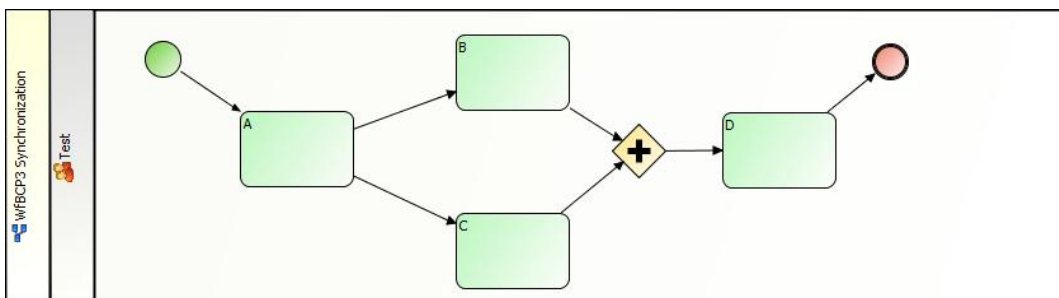


Figura 2.32: BPMN sdoppiamento processo e ricongiungimento.

```

1
2
3 ...
4     <xpdl:Activity Id="A5" Name="A">
5         <xpdl:Description>Just press 'Complete' button</
           xpdl:Description>
6         <xpdl:Implementation>
7             <xpdl:No/>
8         </xpdl:Implementation>
9         <xpdl:Performers>
10            <xpdl:Performer>test</xpdl:Performer>
11        </xpdl:Performers>
12        <xpdl:TransitionRestrictions>
13            <xpdl:TransitionRestriction>
14                <xpdl:Split Type="Parallel">
15                    <xpdl:TransitionRefs>
16                        <xpdl:TransitionRef Id="
                           WfBCP3_Synchronization_Tra1"/>
17                        <xpdl:TransitionRef Id="
                           WfBCP3_Synchronization_Tra2"/>
18                    </xpdl:TransitionRefs>
19                </xpdl:Split>
20            </xpdl:TransitionRestriction>
21        </xpdl:TransitionRestrictions>
22        <xpdl:NodeGraphicsInfos>
23
24 ...
25
26        <xpdl:Activity Id="D1">
27            <xpdl:Route GatewayType="Parallel"/>
28            <xpdl:TransitionRestrictions>
29                <xpdl:TransitionRestriction>
30                    <xpdl:Join Type="Parallel"/>
31                </xpdl:TransitionRestriction>
32            </xpdl:TransitionRestrictions>
33 ...
34
35        </xpdl:Activity>
36 ...

```

2.4 Modelli ibridi

In questa sezione sono raccolti le esperienze e i modelli che racchiudono e utilizzano parzialmente nozioni e informazioni di questi due mondi sopra descritti.

2.4.1 Seamless and temporal conceptual modeling

Carlo Combi e Sara Degani affrontano il problema dell'assenza di un linguaggio che unisca la modellazione dei dati e la modellazione dei processi, quelli visti nelle sezioni precedenti; Nel loro lavoro (referenza) affrontano questo problema e raggiungono ad una soluzione molto interessante; Per prima cosa introducono un motivation scenario che sulla base del quale dibattito e le esigenze che un linguaggio ibrido dovrebbe risolvere.

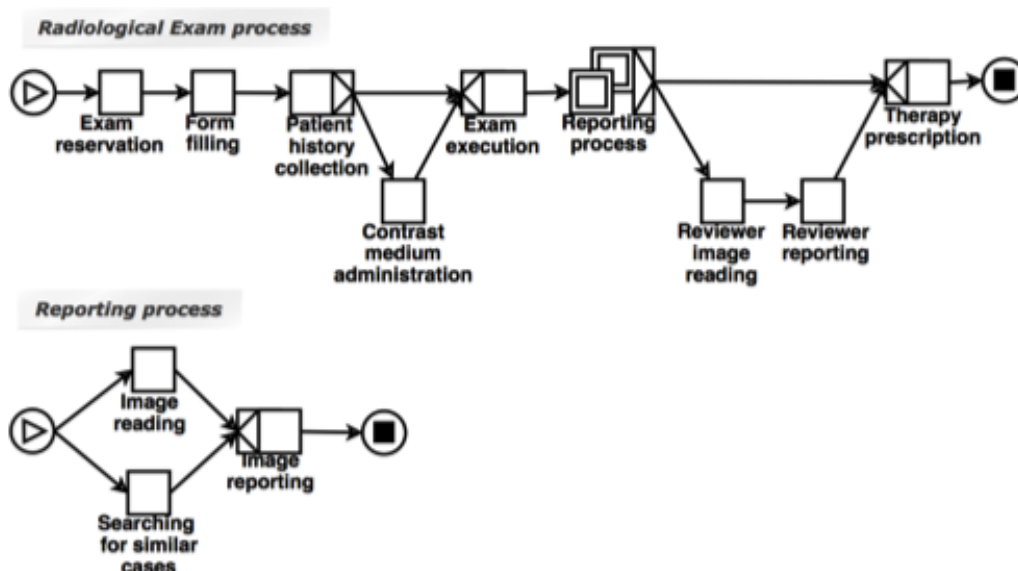


Figura 2.33: Lo schema workflow Radiological Exam process proposto come caso di studio.

Nello specifico analizzano un processo medico, oltre a modellarne il business process ne disegnano anche lo schema della base di dati utilizzando un Temporal ER che aggiunge alla normale notazione una dimensione temporale ai dati includendo nelle definizioni le notazioni VT per valid time, BT per bitemporal e LS per lifespan. Gli aspetti temporali di questo modello pur essendo molto interessanti verranno messi in disparte e invece sarà analizzato in dettaglio come database e BPMN sono connessi tra loro.

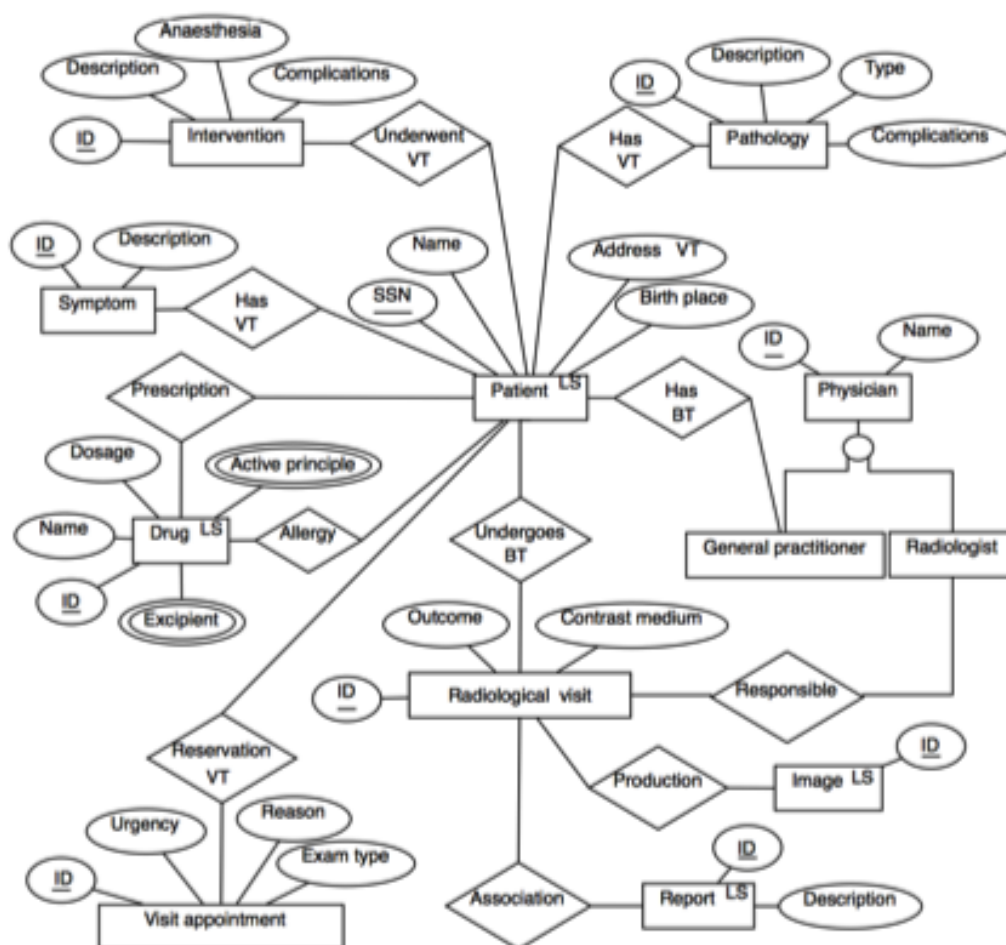


Figura 2.34: Il diagramma TimeER modellato per il processo caso di studio. La classica notazione ER è estesa con le specifiche nella dimensione temporale (BT per bitetemporal, VT per valid time, LF per life span)

Successivamente analizzano il processo e distinguono i processi che hanno influenza sulla base di dati. Questi task sono chiamate entità core. Quindi vengono

studiate le associazioni delle entità alle fasi del processo, per esempio si osserva nel caso di studio che ogni caso del processo analizza un singolo esame radiologico e ogni task maneggia informazioni in relazione solamente a quello specifico esame radiologico. Per esempio il exam reservation task che raccoglie i dati personali del paziente che si compreso dalla visita; il task therapy prescription immagazzina le prescrizioni che sono a loro volta incluse nella visita, possiamo così affermare che le informazioni chiave di questo processo sono quelle che descrivono l'esame radiologico.

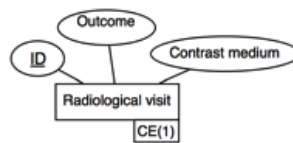


Figura 2.35: Rappresentazione grafica di un'entità core.

In ultimo collegano graficamente queste core entities con lo schema TimeEr associato.

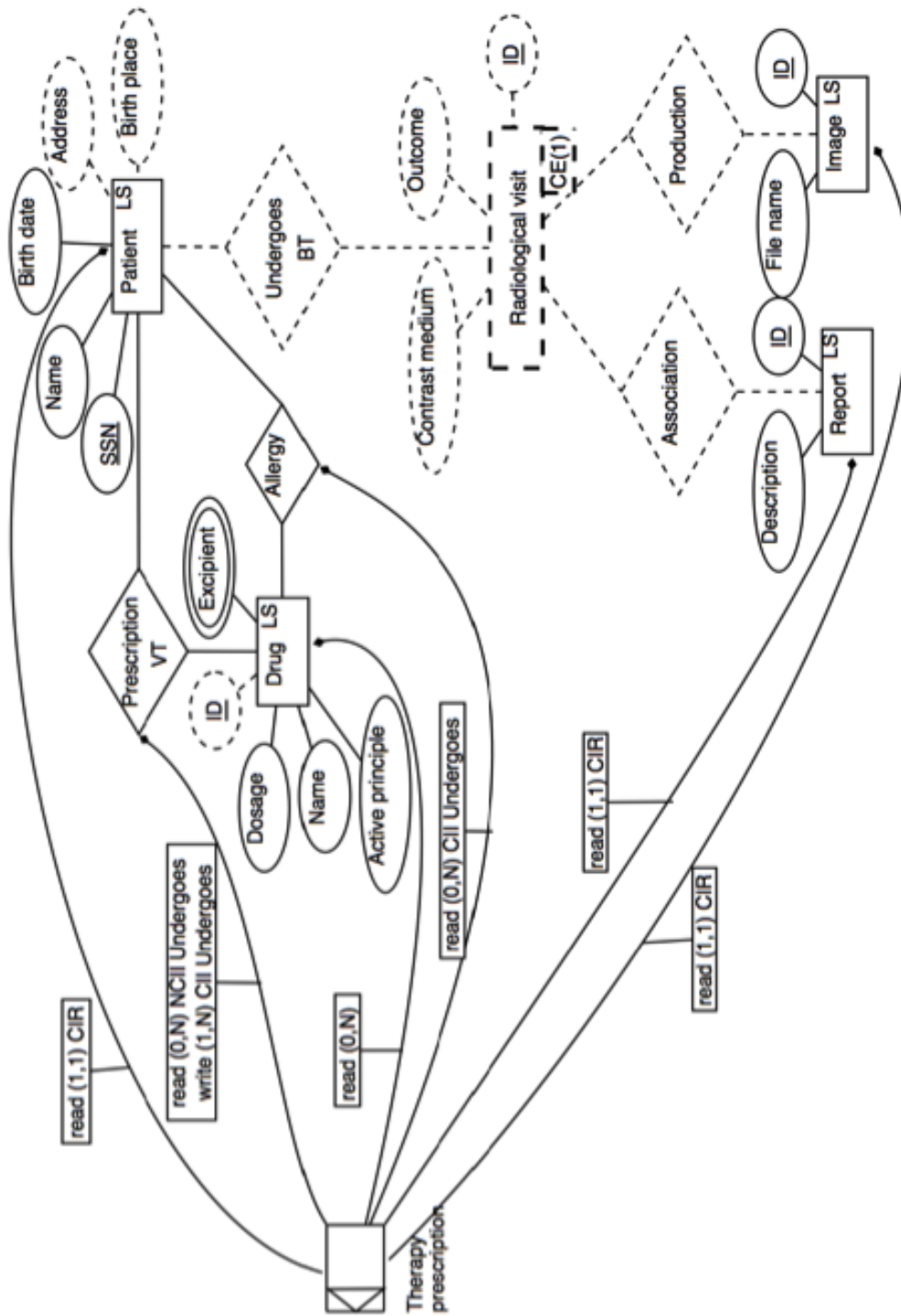


Figura 2.36: Esempio finale del modello proposto da Combi-Degani.

Capitolo 3

Motivation scenario (Un caso di studio)

In questo capitolo viene in trattato un caso di studio. Lo scenario è quello della certificazione in ambito aziendale, quindi audit per certificare l'attinenza a standard e ottenere attestati di qualità (norme ambientali, qualità sicurezza etc..). Il processo di certificazione è accompagnato dall'utilizzo di un applicazione software utilizzabile da pi attori che quindi possono accedere e modificare dati della sessa base di dati in momenti differenti. Per raggiungere una certificazione bisogna aderire ad una serie di regole/vincoli contenuti nella norma di riferimento. Secondo il settore industriale e delle dimensioni dell'azienda queste norme possono essere applicabili o non applicabili, se ad esempio una norma impone il controllo sui fumi di scarico per la certificazione ambientale, una società senza scarichi avrà questa norma non applicabile. Una volta identificate i punti norma ne si valuta la conformità (si o no). Il consulente notificherà al cliente le norme non conformi e assegnerà dei compiti chiamati task per allinearsi alla normativa. Una volta allineatosi il consulente, rianalizzerà la situazione e manderà un report al certificatore autorizzato che secondo la regolarità dell'audit assegnerà o no la certificazione.

L'applicazione fornisce una piattaforma contenente la norma con i suoi punti norma, permette ai consulenti di valutare punto per punto, assegnare task e rivedere i propri report. Permette ai clienti di visionare i report e i task assegnati. Infine permette al revisore di consultare il report.

3.1 Processo

Per prima cosa analizziamo il processo di certificazione.

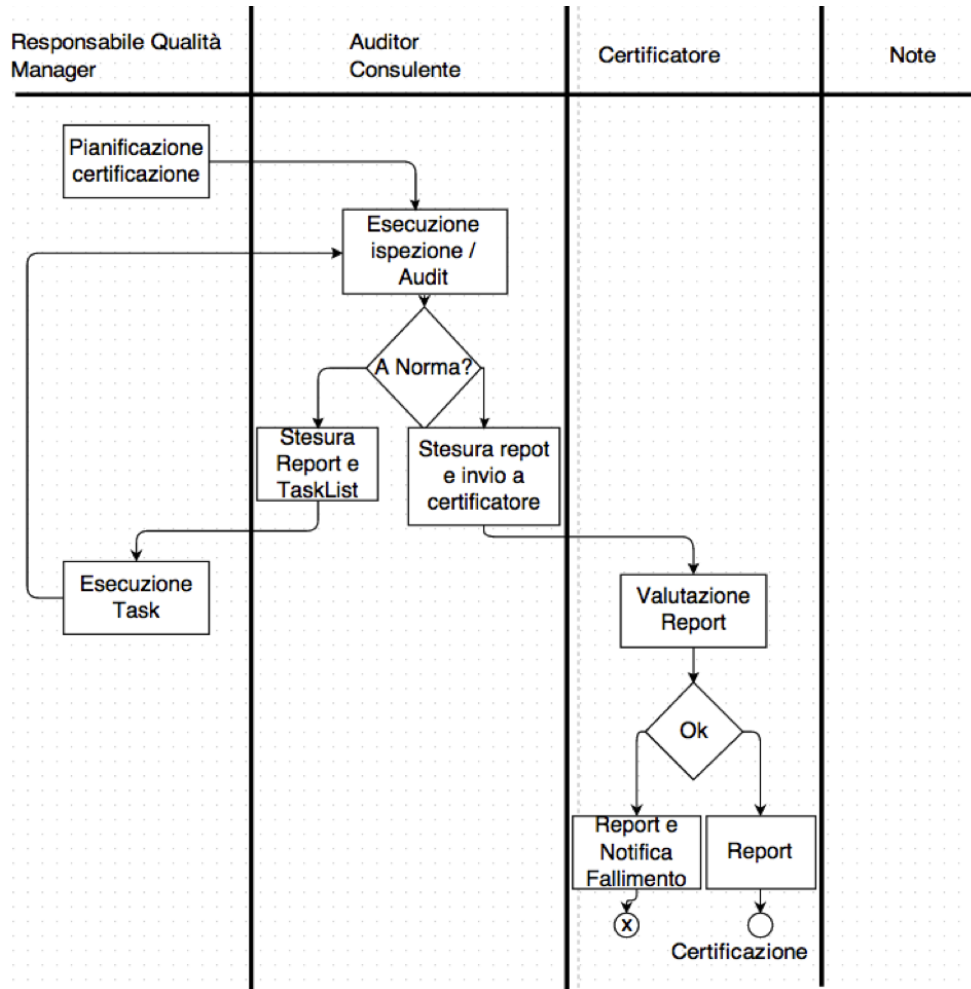


Figura 3.1: Schizzo diagramma processo di certificazione, gli attori coinvolti sono 3: Responsabile qualità, Auditor Consulente e Certificatore.

3.1.1 Gli attori del processo

Gli attori coinvolti nel processo di certificazione sono tre.

- Il manager: Il manager è chi attiva il processo nel momento in cui la sua azienda ha bisogno di una qualche certificazione. Sarà lui l'incaricato allo svolgimento dei task assegnati;

- Il consulente: Il consulente è un esperto in materia di certificazione capace di valutare le normative e assegnare compiti alle aziende di modo che raggiungano la conformità;
- Il certificatore: Il certificatore è l'autorità col potere di validare o invalidare l'intero processo.

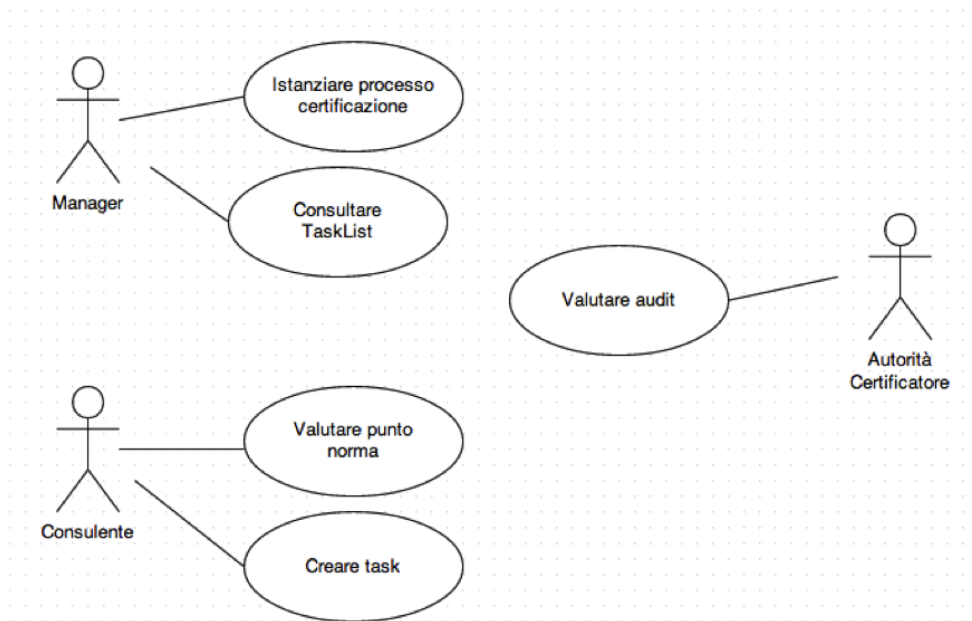


Figura 3.2: Use case del caso di studio mostra le funzioni relative al processo dei tre attori.

3.1.2 I task

- Pianificazione certificazione:

Una volta che un'azienda ha il bisogno di ottenere una certificazione attraverso il suo responsabile della qualità contatta un consulente certificatore di modo che esegua i test necessari

- Esecuzione ispezione audit:

Il consulente valuta la norma punto per punto ne valuta la conformità, la non conformità o la non applicabilità. Ai punti norma non conformi assegna dei compiti da eseguire per allinearsi alla normativa.

- Stesura report e task list:
è creato un documento con i risultati dell'audit e un elenco di task da eseguire da parte dell'azienda per allinearsi alla normativa.
- Esecuzione compiti:
La società cliente esegue i compiti assegnati per raggiungere la conformità. Una volta terminato l'allineamento viene notificato l'avvenimento al consulente.
- Stesura report e invio a certificatore:
è steso un report per l'autorità certificatrice.
- Valutazione report:
Il certificatore valuta la conformità del report e decide se approvarlo o meno.
- Notifica fallimento:
In caso la certificazione abbia esito negativo, è notificato il fallimento indicando le cause del rifiuto.
- Report certificazione avvenuta:
In caso tutto sia corretto, è notificata l'avvenuta certificazione.

3.2 Base di dati

La base di dati standard sui cui si basano le normative è la seguente:

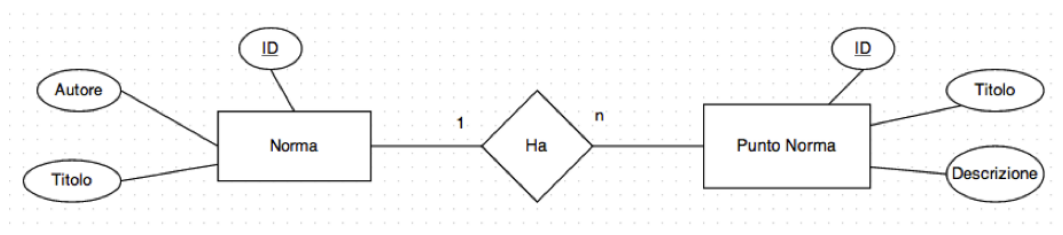


Figura 3.3: Esempio di schema ER su cui si basano le normative.

La base di dati a cui si appoggerà invece l'applicazione vede aggiungersi la possibilità di creare sessioni e di assegnare compiti(task) ai clienti.

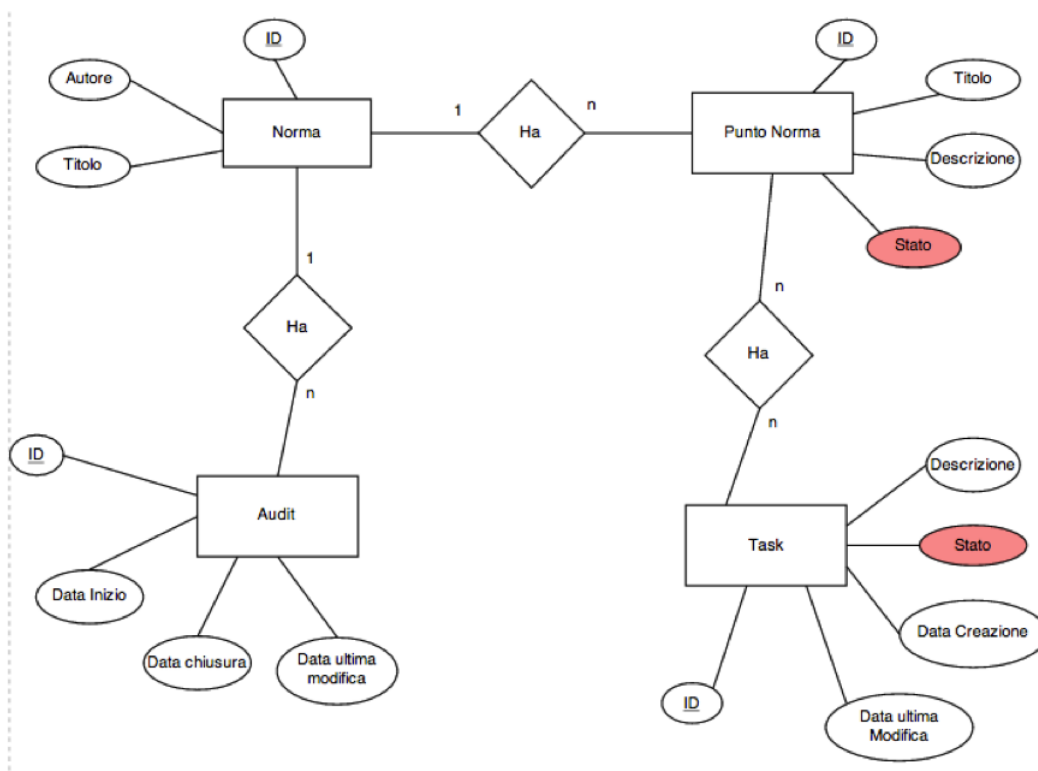


Figura 3.4: La base di dati completa. La base di dati delle normative vengono estese con la possibilità di tracciare gli audit e memorizzare i compiti assegnati per conformare i punti norma non in regola.

3.2.1 Entità

- Audit:

L'audit è l'entità che indentifica il processo di certificazione. è contraddistinto da un ID, una data di inizio, una data di fine e una data riportante l'ultima modifica.

- Norma:

La norma è un insieme di punti norma. è contraddistinta da un ID, un autore, e un titolo.

- Punto norma:

Il punto norma è un parametro valutabile dal consulente. La valutazione può essere di tre tipi, positiva, negativa e non applicabile nel caso in cui

non sia rilevante nel caso analizzato. Il punto norma p caratterizzato da un ID, da un titolo, da una descrizione (il testo effettivo) e da uno stato (positivo, negativo, N.A.).

- Task:

Il task è un compito assegnato dal consulente all'azienda per far sì che un punto normi cambi dallo stato non conforme a quello conforme. è caratterizzato da un ID da una data di modifica, da una data di creazione, da una descrizione e da uno stato.

3.3 BPMN e XPDL

Una volta pensato concettualmente il modello lo stesso è stato disegnato secondo norma con l'applicazione Together work-flow editor che permette di modellare il business process ed esportarlo in XPDL.

3.4 Altri scenari

Lo scenario qui proposto è solamente uno dei tanti casi in cui gli accessi variano a seconda della fase del processo e dell'attore che si interfaccia all'applicazione. Modellare questi aspetti è di vitale importanza per permettere agli sviluppatori del sistema di non commettere questo tipo di errore. Pensiamo all'ambito bancario o a quello medico. Transazioni non autorizzate potrebbero essere eseguite in maniera fraudolenta per un errore di concessione di privilegi, peggio ancora un errore di somministrazione di farmaci potrebbe essere commesso dando autorità di farlo a chi non ne dispone. Oggigiorno questo tipo di modellazione non è possibile con alcuno strumento.

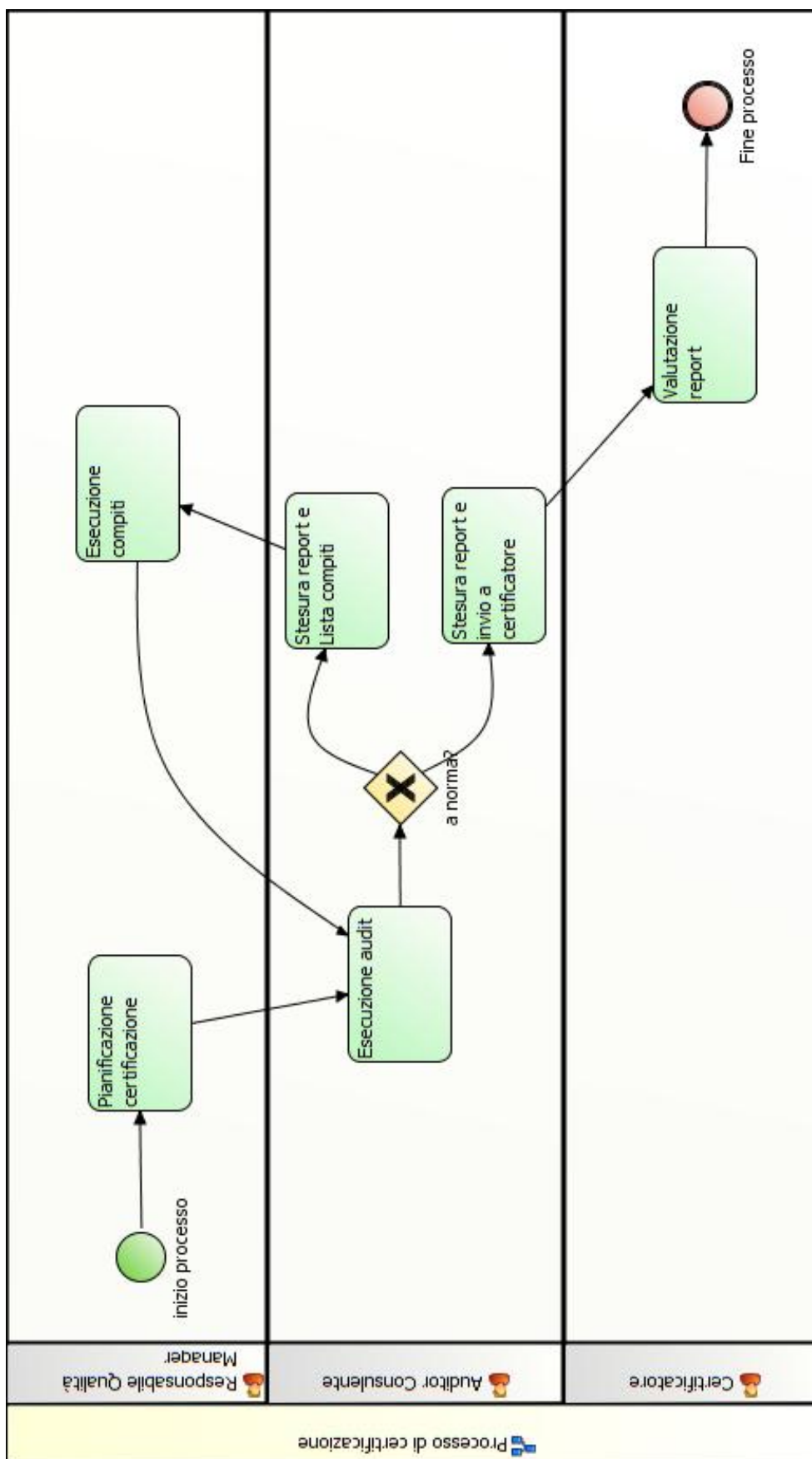


Figura 3.5: BPMN caso di studio disegnato con together workflow editor.

Capitolo 4

Soluzione

Partendo dalla lista dei requirements si è cercato di risolverli tutti pensando ad una soluzione ottimale per ciascuno di essi per poi studiare il modo migliore per integrarli in un solo modello. Questo approccio permette di avere inizialmente una visione a comparti stagni del problema per poi via via innalzarsi fino ad avere tutto il quadro del problema in vista.

Lista requirements linguaggio modellazione:

- Privilegi su varie entità e attributi
- Privilegi a degli attori
- Privilegi in base al Task
- Privilegi in base allo stato del processo

Vista l'ambizione del progetto non è stato possibile utilizzare un modello su un solo livello come ad esempio i classici BPMN o i più comuni diagrammi UML, ma si è dovuto evolvere su più livelli, quindi sono stati pensati meccanismi di iterazione per permettere all'utente di esplorare il modello analizzando porzioni di informazioni alla volta. Le informazioni sono mostrate il modello su 2 piani, una linea temporale del processo è costituita dal business process in alto nel modello.

Rispetto al classico BPMN sono state apportate delle modifiche di modo da alleggerire lo stesso e renderlo di più immediata lettura; è stato pensato di eliminare il concetto di corsie per permettere al disegno di svilupparsi in maniera più lineare

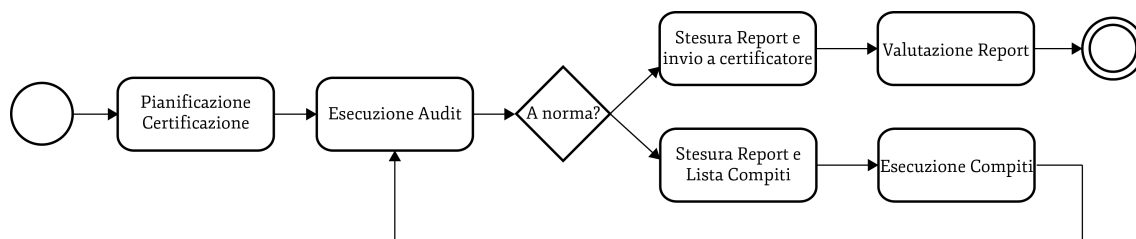


Figura 4.1: Business Process semplificato del caso di studio. Esso verrà posto nel modello in alto e fungerà da linea temporale.

lungo una direttrice orizzontale. L'utente per visualizzare i dettagli relativi al singolo task può, se si trova ad interagire con uno strumento informatizzato, cliccare sul task interessato e visualizzarle, in caso di documento cartaceo viene stampato una pagina per ogni task per ogni task se richiesto una pagine per definire i privilegi dei singoli attori. In questo modo la leggibilità del modello è molto alta, potendo facilmente l'utente navigare tra le varie fasi del processo.

- Per indicare l'inizio di un processo viene utilizzato il classico cercholino utilizzato in tutti gli standard.
- Per indicare i task si utilizzano i classici rettangoli con angoli smussati, contenenti il titolo del task.
- Le transazioni sono indicate con la classica freccia.
- Le biforcazioni e i merging sono identificati da un rombo.
- La conclusione del processo viene indicata da un doppio cerchio.

Per indicare il task attivo si sono pensate varie soluzioni che mostrassero un task alla volta. A seconda del task attivo, quello preso in esame, nella parte inferiore del modello apparirà il corrispondente schema entità relazione.

La prima soluzione pensata è per rappresentare i task attivi è stata la più creativa e di immediata comprensione, cioè colorare il task in esame di un colore vivace, come ad esempio un giallo acceso. Questa soluzione però ha dei limiti per quanto riguarda l'accessibilità in caso di utenti con problemi di vista o in caso di stampe monocromatiche. è stato pensato di utilizzare un giallo in quanto è

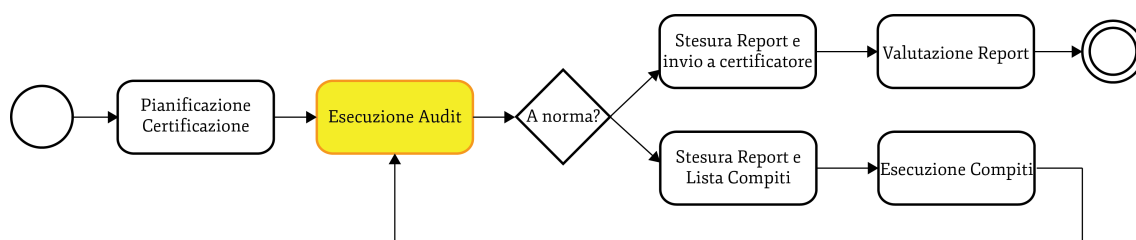


Figura 4.2: Per identificare il processo attivo esso viene evidenziato colorando lo sfondo di un giallo acceso, questa soluzione per porta con sè limiti di accessibilità. è stato pensato di utilizzare un giallo in quanto è un colore che risalta molto e genera un forte contrasto con il nero dei testi e ne permette una facile lettura.

un colore che risalta molto e genera un forte contrasto con il nero dei testi e ne permette una facile lettura.

La seconda soluzione pensata prevede l'utilizzo di un tratto più marcato, tipo grassetto, per disegnare il task attivo. Questa soluzione supera i limiti del modello precedente, in aggiunta per marcare ulteriormente il task utilizza un colore meno acceso per indicare i task meno accesi.

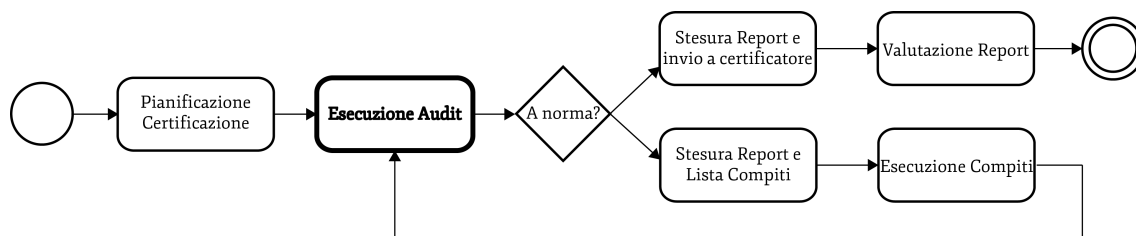


Figura 4.3: In questo caso il processo attivo viene semplicemente evidenziato da un tratto più marcato.

Tante altre soluzioni potrebbero essere adottate, come ad esempio l'utilizzo di sottolineature o caratteri maiuscoli, la seconda soluzione pensata ha soddisfatto in maniera ragionevole.

Nella parte inferiore del modello viene riportato invece il modello dei dati secondo la fase del business process attiva, l'utente può selezionare i vari task del business process e osservare l'evoluzione dei dati e dei permessi sugli stessi.

Per modellare la base di dati è stato pensato di seguire i canonici standard di modellazione entità-relazione descritti anche nel secondo capitolo.

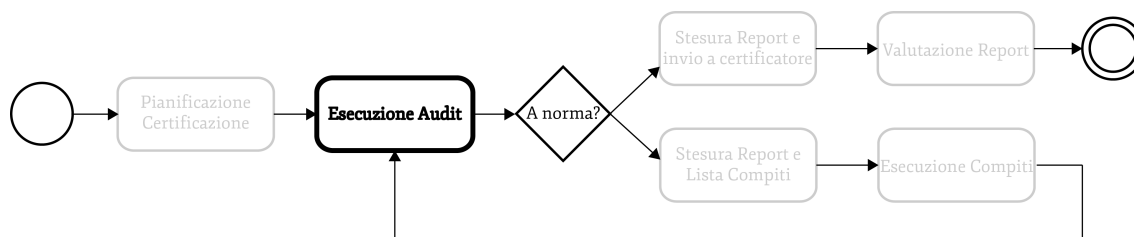


Figura 4.4: Per aumentare ulteriormente l'accessibilità i processi inattivi vengono disegnati con una tonalità di grigio.

4.1 Privilegi

I privilegi sulle entità o sugli attributi della stessa sono i classici attributi di read-select , write-update, delete e insert-create utilizzati da una moltitudine di modelli;

- Read-Select: Permette all'utente di leggere/vedere le entità e i suoi attributi in maniera mirata nel database.
- Write-Update: Permette all'utente con questo permesso di modificare l'entità e o i suoi attributi.
- Delete: Permette all'utente di eliminare un'istanza dell'oggetto.
- Insert: Permette all'utente di inserire un'istanza dell'oggetto e i suoi attributi.

Per segnalare se un'entità o un attributo è accessibile all'utente, è definita una convenzione che prevede l'utilizzo di linee tratteggiate per definire il contorno delle entità visibile, linee continue per definire le entità modificabili. Le entità e gli attributi non visibili non saranno inseriti nello schema. Il fatto di non inserire alcune entità nel modello potrebbe ingannare gli utenti del modello per ovviare a questo problema viene allegato al documento uno schema completo dei dati. In dettaglio quindi:

- Read: questo permesso viene indicato mostrando nello schema entità relazione l'entità

- Write: questo permesso viene indicato mostrando nello schema l'entità che avrà le linee di demarcazione continue, in caso l'entità o l'attributo abbia solo il diritto di read lo stesso avrà linee di demarcazione tratteggiate
- Delete e Insert: questi permessi sono mostrati inserendo nell'entità dei simboli + e -. Rispettivamente + per l'insert e - per il delete.

Questa metodologia molto intuitiva permette ad un veloce colpo d'occhio di identificare le entità e gli attributi importanti per ciascun attore in tutte le fasi del processo.

4.1.1 Privilegi attori

Il secondo problema affrontato è quello di distinguere i privilegi secondo l'attore. Per prima cosa bisogna pensare ad una convenzione per modellare gli attori, in questo caso è stato pensato di adottare il metodo standard UML di use case, quindi un umano stilizzato con il nome dell'attore al di sotto. Il classico attore use case è stato però modificato disegnandolo a mezzobusto per rendere la figura più piacevole alla vista.

Ogni attore coinvolto nel task attivo viene riportato nella scena. L'utente del modello potrà selezionare uno alla volta gli attori e a seconda di quello selezionato il modello di dati muterà. L'attore attivo sarà indicato con un'evidenziazione come può essere una dimensione maggiore, un nero più vivo o una sottolineatura. Questi metodi descritti grossomodo si equivalgono, è stato deciso di adottare un insieme di tutti i metodi per permettere una distinzione tra attore attivo e inattivo molto rapida e inequivoca.

In questo modo possiamo dire che lo use case verrà implicitamente integrato in questo modello, di fatto le connessioni/linee dello use case potranno essere viste come la presenza o meno dell'attore nel processo attivo o meno. Quindi volendo analizzare tutti i task in cui un attore è coinvolto basta correre tutti i task e annotare in quali è presente.

4.1.2 Privilegi in base al task

Come spiegato nell'incipit l'utente può analizzare i permessi e i dati task per task sfogliando le pagine se il documento è cartaceo o cliccando se il modello è in formato multimediale. Per scorrere da un task all'altro a seconda del mezzo

utilizzato, cartaceo o multimediale, l'utente può nel primo caso sfogliare fino a trovare la pagina in cui il task di interesse è evidenziato, nel secondo caso basta cliccare con il mouse o qualsiasi puntatore utilizzato sul task di interesse.

4.2 Modello complessivo

Il risultato complessivo è un insieme di 3 grafici, il primo in alto che funge da linea temporale è il modello semplificato di BPMN, il secondo che indica lo stato e i privilegi sulle entità e la struttura dei dati stessi è un classico schema ER, il terzo schema sulla destra funge da use case per distingere i privilegi dei singoli attori. Il risultato finale pur introducendo molte informazioni non risulta caotico anzi, un'occhio già allenato a modelli informatici probabilmente sarà subito in grado di capire il significato di ogni aspetto di questo modello. Il fatto di essere molto intuitivo e di facile lettura per tutti gli addetti ai lavori rende il modello molto interessante non presentando di fatto grosse difficoltà nell'apprendimento dello stesso.

4.3 XPDL

In questa sezione viene descritto come le novità introdotte vengono fuse con le convenzioni XPDL. L'elenco dei permessi può essere visto come una tabella associata ad ogni dato del processo, sia esso un'entità o un attributo. La tabella muterà però a seconda del task attivo e dell'utente preso in esame dal visualizzatore.

<i>Utente</i>	<i>Read</i>	<i>Write</i>	<i>Delete</i>	<i>Insert</i>
<i>Consulente</i>	X	X		
<i>Auditor</i>	X			
<i>Certificatore</i>	X	X	X	X

Tabella 4.1: Esempio di tabella dei permessi relativi ad un dato o a una entità a seconda dell'utente attivo e del task attivo.

Nel linguaggio XPDL ogni elemento viene contraddistinto da un ID, questo semplifica molto la risoluzione del nostro problema. A questo punto non rimane che trasformare in formato XML la base di dati. Dopo di che nella definizione

XML XPDL di ogni task avendo i riferimenti sia degli utenti, contraddistinti da un ID e della base di dati, possiamo definire per ogni utente nel task in esame i permessi specifici.

4.3.1 Da ER a XML

La ricerca di metodi esistenti per la creazione di XML partendo da schemi ER ha portato molti risultati. Per agevolare ulteriormente la risoluzione del problema è stato pensato di concepire un algoritmo di traduzione ad hoc per il problema in esame di modo da ricalcare, laddove possibile, le convenzioni e i canoni già adottate dall'XPDL.

Al nostro scopo servono 3 nuovi tag; un tag per le entità, uno per i suoi attributi, e uno per le connessioni. Per prima cosa sono definite tutte le entità:

```
1
2 <entities>
3   <entity id="1" name="Audit">
4     ..
5   </entity>
6   <entity id="2" name="punto_norma">
7     ..
8   </entity>
9 </entities>
```

dopo aver definito tutte le entità definiamo tutti gli attributi assegnando un id e definendo il tipo di variabile. Per semplificare le cose gli attributi saranno compresi tra i tag della relativo oggetto:

```
1
2 <entities>
```

```
3 <entity id="1" name="audit">
4   <attributes>
5     <attribute id="1" name="id" type="id">
6     </attribute>
7     <attribute id="2" name="Data_cchiusura" type="date">
8     </attribute>
9   </attributes>
10 </entity>
11 </entities>
```

per ultimo vengono definite le relazioni fra le entità ricalcando quello che viene fatto il XPDL per le transazioni:

```
1
2 <Relations>
3   <Relation From="1" To="2" Type="NN">
4   </Relation>
5   <Relation From="2" To="3" Type="1N">
6   ..
7   </Relation>
8
9 </Relations>
```

4.3.2 Notazione vincoli

Ora che abbiamo tutti gli elementi principali nel nostro XML dobbiamo decidere come inserire i vincoli.

Una proposta è quella di creare un tag constraint per ogni vincolo, quindi avremo N tag dove N è uguale al numero di attributi per il numero dei task per il numero di attori.

```
1
2 <constraints>
3   <constraint actor="1" entity="1" attribute="1" permit="RW">
4   </constraint>
5   <constraint actor="2" entity="1" attribute="1" permit="R">
6   </constraint>
7 </constraints>
```

Questa soluzione porta ad avere un esponenziale numero di tag, per limitarne la vastità si potrebbe definire un tipo di vincolo default che si applica a tutte le combinazioni non specificate:

```
1
2 <constraints defaultValue="R">
3   <constraint actor="1" entity="1" attribute="1" permit="RW">
4   </constraint>
5 </constraints>
```

In questo modo ridurremo sensibilmente il numero totale di tag necessari. Un'altra metodologia potrebbe essere quella di creare n tag per ogni tipo di vincolo e inserire all'interno degli stessi le combinazioni di ID che appartengono a questo tipo di vincolo;

```
1
2 <constraints>
3   <constraint permit="RW">
```

```
4 <Combination actor="1" entity="1">
5 </Combination>
6 <Combination actor="2" entity="3">
7 </Combination>
8 <Combination actor="2" entity="2">
9 </Combination>
10 </constraint>
11 <constraint permit="R">
12 <Combination actor="3" entity="1">
13 </Combination>
14 </constraint>
15 ..
16 ..
17 </constraints>
```

Questa soluzione ci soddisfa al pari della precedente forse la precedente presenta leggero un vantaggio in termini di leggibilità quindi adotteremo la prima.

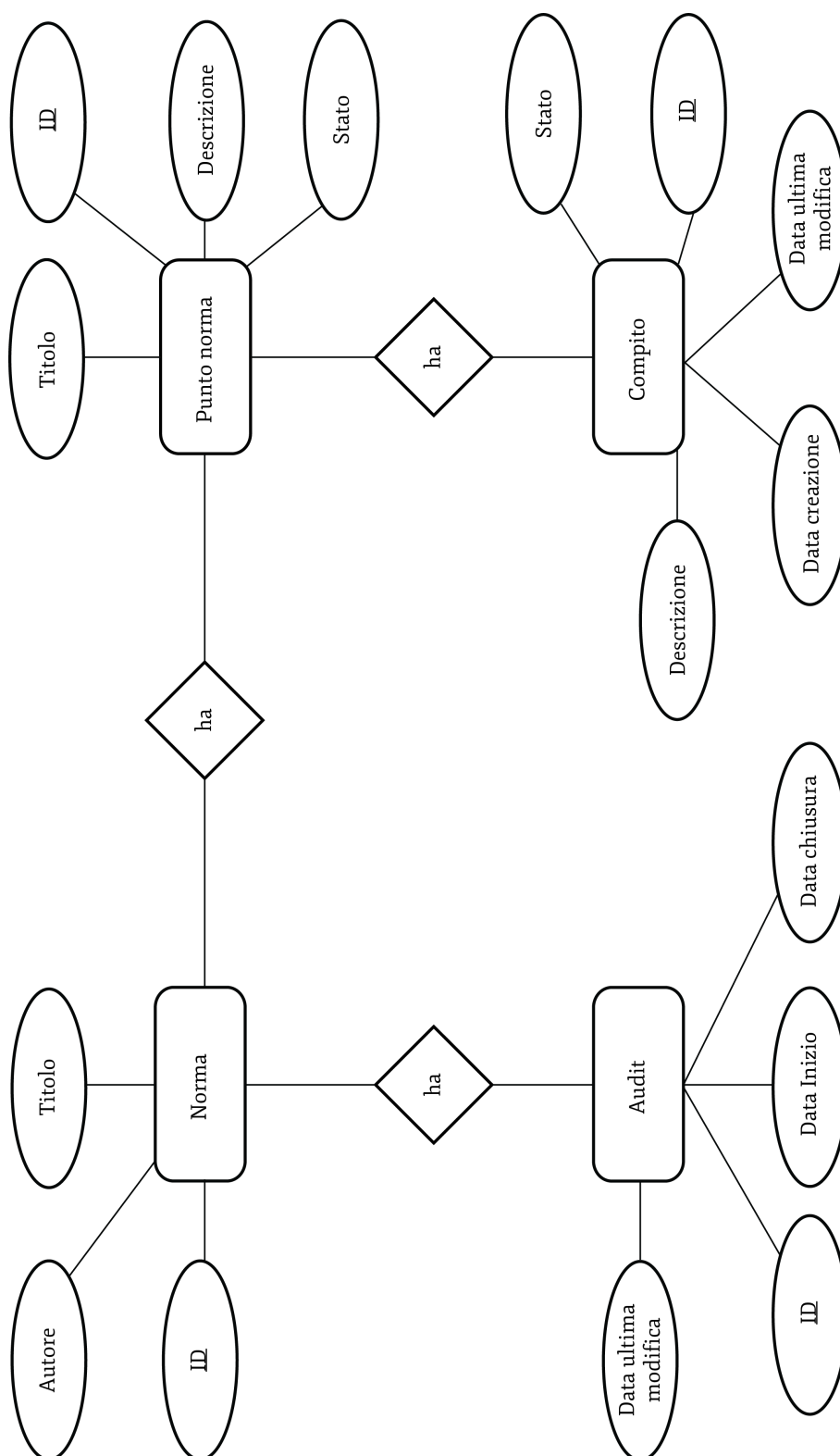


Figura 4.5: Nella parte inferiore del modello è mostrato lo ER al momento in cui è attivo una fase del processo, in questa immagine è riportato lo schema ER completo.

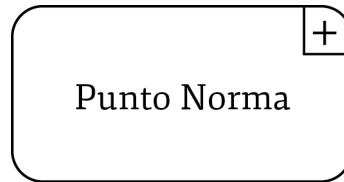


Figura 4.6: Il privilegio di insert sulle entità è indicato con un più nell'angolo.

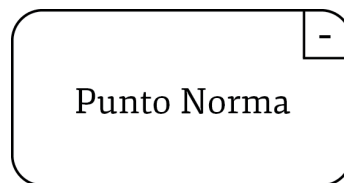


Figura 4.7: Il privilegio di delete sulle entità è indicato con un meno nell'angolo.

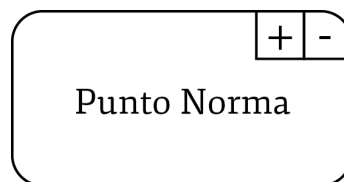


Figura 4.8: Il privilegio di insert e delete è indicato con entrambi i simboli, più e meno nell'angolo in alto a destra.

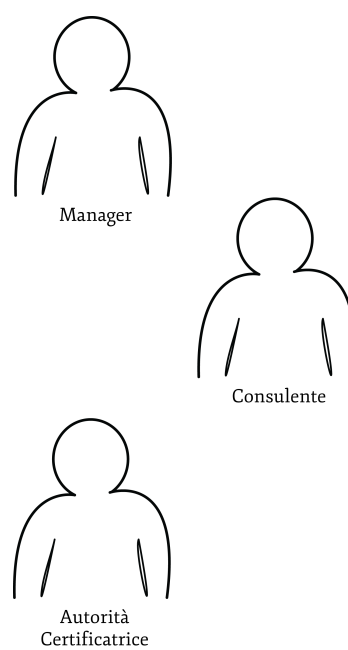


Figura 4.9: Per indicare gli attori si utilizza la metodologia UML, quindi una figura di persona stilizzata facile da riprodurre anche a mano libera con al di sotto della stetta il nome dell'attore.

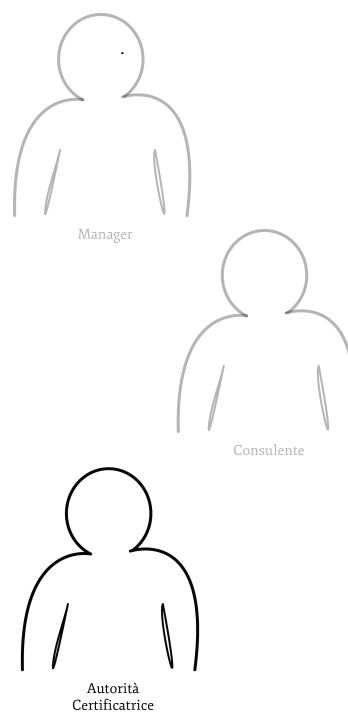


Figura 4.10: Per indicare l'attore attivo si è pensato di scolorire con una tonalità di grigio gli attori inattivi, facendo risaltare in nero quello attivo.

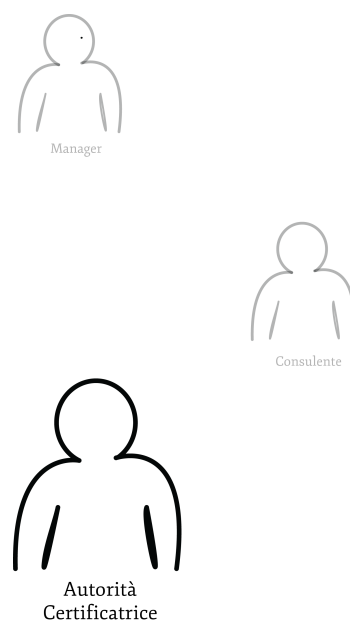


Figura 4.11: Per migliorare ulteriormente l'accessibilità del modello gli attori inattivi vengono rimpiccioliti e l'attore attivo viene marcato con un tratto più spesso.

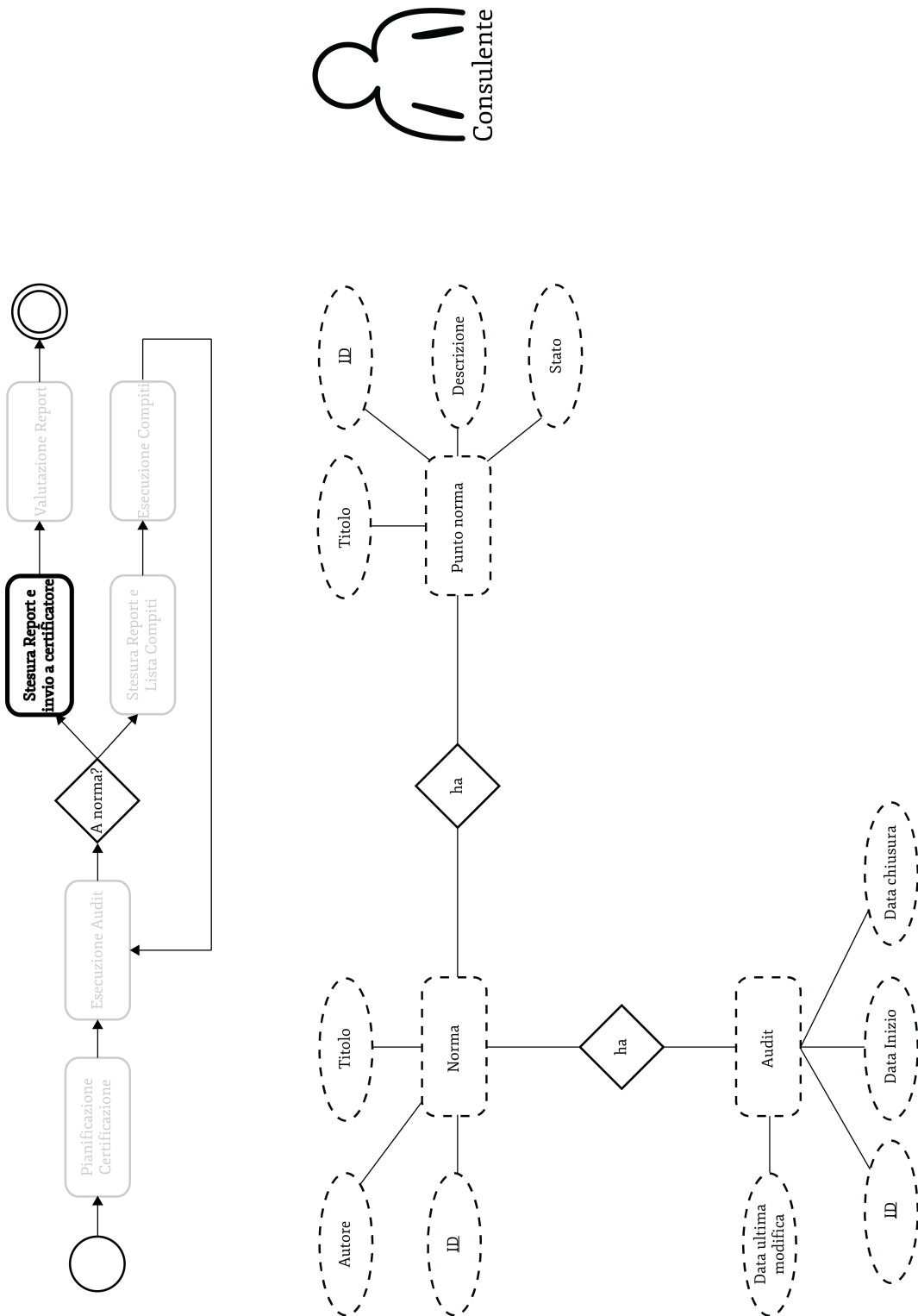


Figura 4.12: Quest'immagine raccoglie tutti gli elementi introdotti; nella parte superiore presente il modello BPMN semplificato con attivo il task Stesura report e invio a certificatore, sulla destra vengono riportati gli attori interessati a questa fase del processo, in questo caso solo il consulente, e nella parte inferiore viene riportato lo schema ER. Come possiamo vedere in questa fase del processo l'attore consulente ha solo privilegi di read sulle entità del database.

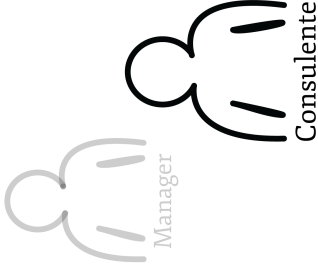
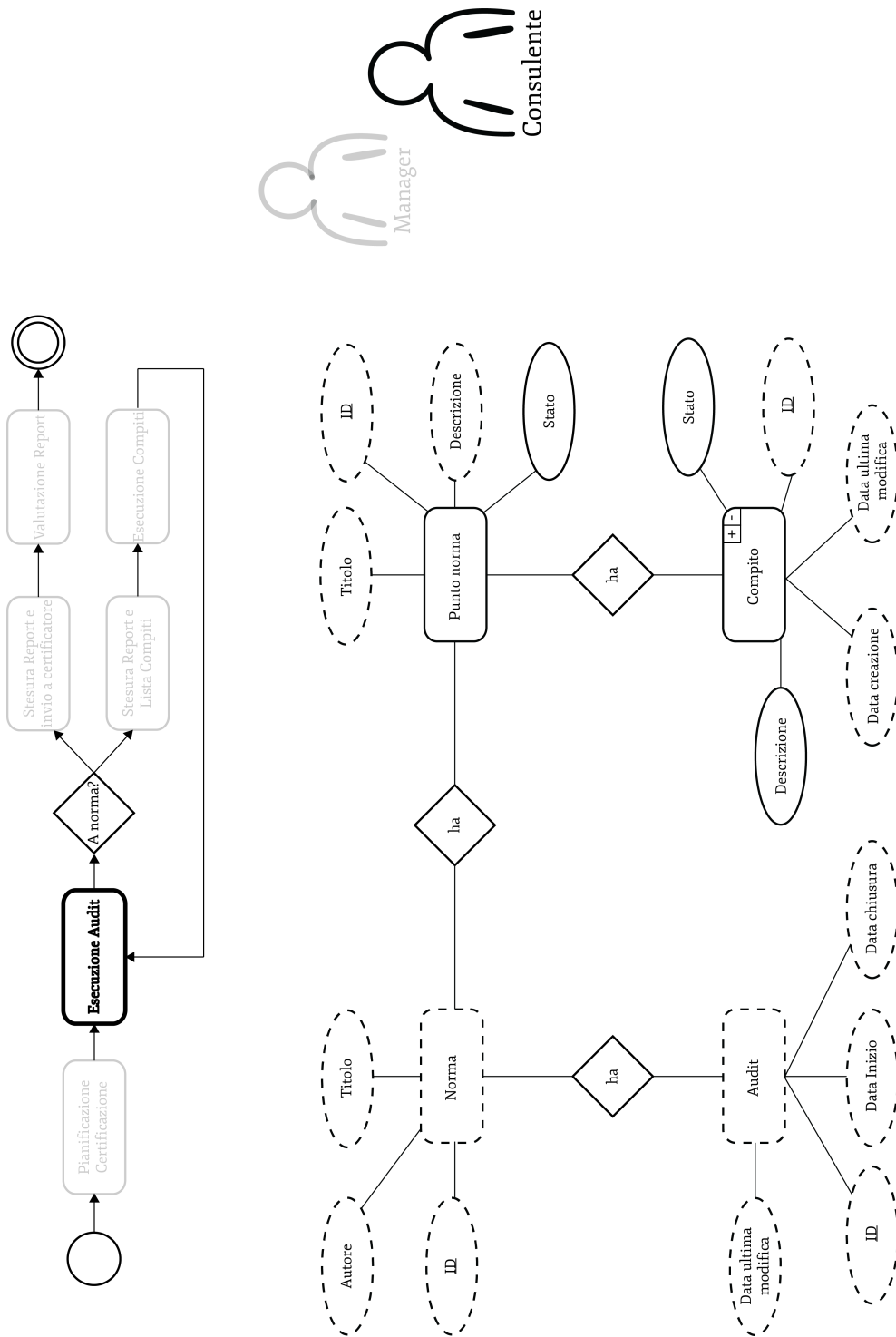


Figura 4.13: In questa seconda immagine viene riportato la situazione quando il task Esecuzione audit è attiva. Come possiamo vedere gli attori coinvolti sono Manager e Consulente, nell'immagine viene però riportata la situazione relativa al consulente essendo lo stesso evidenziato. Leggendo l'ER notiamo che ha la possibilità di inserire nuovi compiti, eliminarli e commentarli.

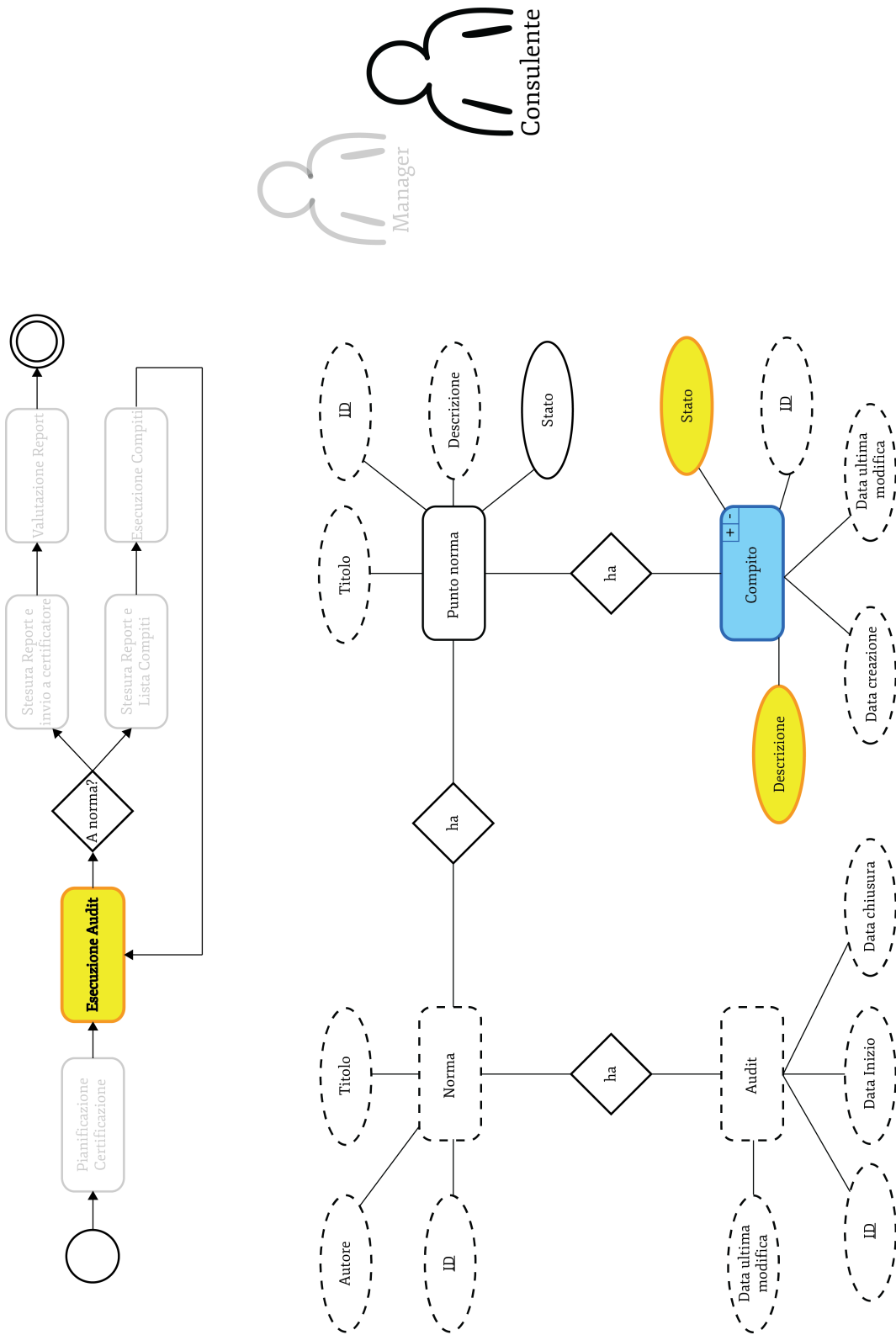


Figura 4.14: In quest'ultimo esempio viene mostrato il modello come sarebbe se comprendesse anche delle feature cromatiche. Il risultato è sicuramente di maggiore impatto.

Capitolo 5

Integrazione modello in un WfMS

5.1 WfMS

In organizzazioni complesse, la pianificazione e la gestione di diversi business process diventa difficile, inoltre il controllo dei flussi di lavoro pu richiedere tanto tempo quanto lesecuzione del lavoro stesso. È possibile risolvere questi problemi mediante l'utilizzo di un Workflow management system che gestisca attivamente il coordinamento di attivit tra i partecipanti del processo. Un WFMS quindi orchestra(coordina) le istanze di processo di un'azienda tenendo conto dei dati e dei partecipanti associati ad ogni istanza. Nelle organizzazioni moderne orchestrare il lavoro di tutte le risorse è diventato sempre più complesso e dispendioso, per tamponare questo probleme WfMS vengono implementati e integrati nei sistemi informativi dell'organizzazione. Essi permettono una ottimale condivisione delle infomrazioni durante tutte le fasi del progetto permettendo allo stesso di procedere più velocemente e ricalcare quelle che sono le esigenze imposte dai progettisti/architetti.

5.2 Integrazione

Il modello proposto si pone l'obbiettivo principale di agevolare gli sviluppatori durante la fase di implementazione del software, potendo creare classi di utenze con diversi permessi di accesso a alle variabili a seconda dello stato attuale del

processe. L'integrazione quindi deve essere accessibile agli sviluppatori e deve poter essergli fornita attraverso degli strumenti a loro congeniali. La soluzione ottimale a questo punto è quella di integrare nell'IDE (Integrated development environment) dell'organizzazione un plugin per la realizzazione dei modelli e la rapida consultazione degli stessi. Il modello viene salvato e aggiornato di volta in volta sulla repository dell'organizzazione per poter permettere a tutte le utenze di consultare la versione aggiornata e lo storico del versioning. I moderni IDE forniscono oltre all'ambiente di sviluppo anche un potente set di strumenti utilizzati da business analyst, software architect e altre figure IT. L'integrazione del modello in uno di questi IDE sembra la soluzione più pratica e trasportabile. Le altre utenze interessate dal modello sono gli utilizzatori dello stesso, essi avranno accesso al modello tramite dei file esportati del software di modellazione.

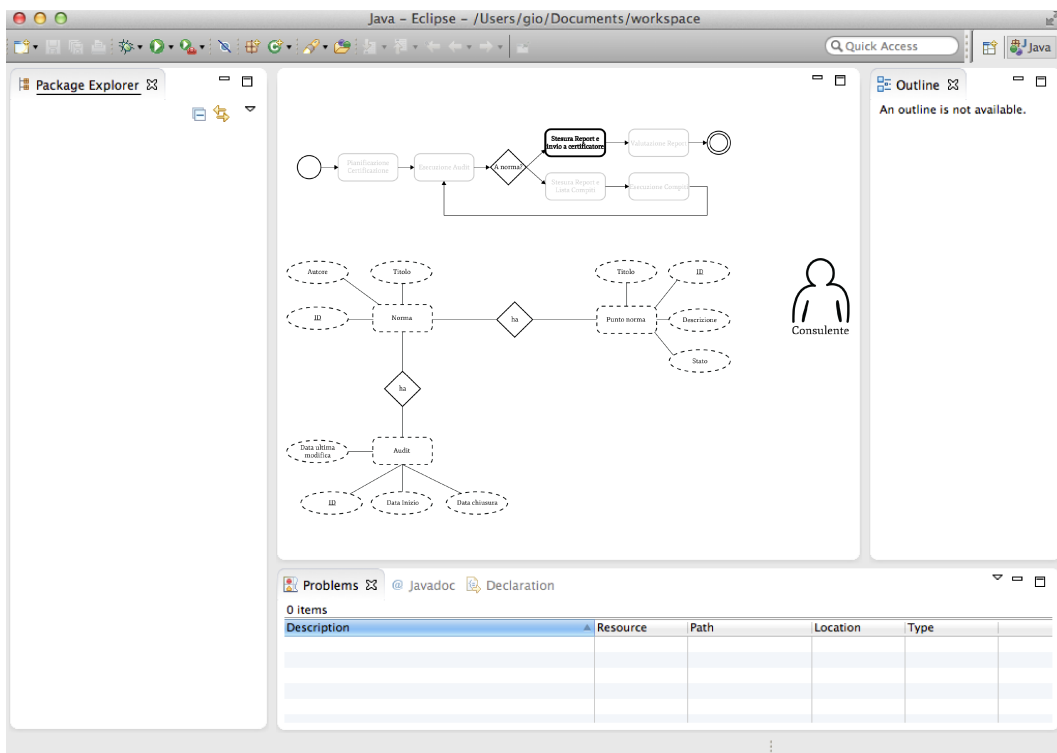


Figura 5.1: Esempio di integrazione del modello in eclipse.

Capitolo 6

Conclusioni e sviluppi futuri

6.1 Sviluppi futuri

Il modello si adatta molto bene a casi molto generici, le prime estensioni che si possono fare al modello riguarda sicuramente la possibilità di descrivere differenti modelli per raffigurare le informazioni. Il modello proposto utilizza il classico ER, ma si potrebbe facilmente modificare sostituendo l'ER con un class diagram, in questo caso il modello assumerebbe una potenza maggiore potendo definire, non solo la visibilità e i permessi sulle variabili, ma anche sui metodi delle varie classi. Con il class diagram entrano in gioco altri tipi di problematiche legati ai tipi di variabili e alla struttura delle classi, il discorso andrebbe approfondito. In secondo luogo si potrebbero utilizzare un diverso linguaggio di definizione dei processi, nel nostro caso è stato usato una versione ad hoc del BPMN ma si sarebbe potuto usare un qualsiasi altro modello. Sperimentazioni in questo senso potrebbero essere condotte. L'integrazione in un WfMS è stata solamente abbozzata se si dovesse ritenere il modello valido uno sviluppo chiaro sarebbe quindi l'implementazione dello stesso. Altre ricerche si potrebbero effettuare su altri tipi di vincoli; ad esempio vincoli temporali, vincoli scatenati da eventi o da stati delle informazioni stesse. Questo ibrido process-information crea un set di possibilità molto ampio, in ambiti come quello bancario, medico e legale dove gli aspetti legati alle tempistiche e permessi possono fornire una notevole quantità di spunti.

6.2 Conclusioni

In questa tesi è stato portato a galla il problema di come creare una connessione tra il controllo dei business process e il controllo delle informazioni in particolare il controllo di accesso alle stesse. È stata introdotta una notazione formale per la modellazione di situazioni ibride e un sistema per rendere il modello interoperabile (XPDL). Viene proposta una possibile integrazione in un WfMS. L'esigenza di poter modellare vincoli nei sistemi informativi moderni

Appendice A

BPMN Esempi completi

Esempio 1

```
1 <xpdl:WorkflowProcess AccessLevel="PUBLIC" Id="WfBCP1_Sequence"
  Name="WfBCP1_Sequence">
2   <xpdl:ProcessHeader DurationUnit="D">
3     <xpdl:Created>2004-08-31 22:28:17</xpdl:Created>
4     <xpdl:Description>execute activities A, B and C in
      sequence</xpdl:Description>
5   </xpdl:ProcessHeader>
6   <xpdl:RedefinableHeader PublicationStatus="UNDER_TEST">
7     <xpdl:Author>Sasa Bojanic</xpdl:Author>
8   </xpdl:RedefinableHeader>
9   <xpdl:Activities>
10    <xpdl:Activity Id="A3" Name="A">
11      <xpdl:Description>Just press 'Complete' button</
        xpdl:Description>
12      <xpdl:Implementation>
13        <xpdl:No/>
14      </xpdl:Implementation>
15      <xpdl:Performers>
16        <xpdl:Performer>test</xpdl:Performer>
17      </xpdl:Performers>
18      <xpdl:NodeGraphicsInfos>
19        <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
          FillColor="187,247,190" Height="60" IsVisible=
```

```

    "true" LaneId=" workflow_patterns_pool1_lan1 "
    ToolId="JaWE" Width="90">
20     <xpdl:Coordinates XCoordinate="206"
        YCoordinate="48" />
21     </xpdl:NodeGraphicsInfo>
22 </xpdl:NodeGraphicsInfos>
23 </xpdl:Activity>
24 <xpdl:Activity Id="B5" Name="B">
25     <xpdl:Description>Just press 'Complete' button</
        xpdl:Description>
26     <xpdl:Implementation>
27         <xpdl:No />
28     </xpdl:Implementation>
29     <xpdl:Performers>
30         <xpdl:Performer>test</xpdl:Performer>
31     </xpdl:Performers>
32     <xpdl:NodeGraphicsInfos>
33         <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
            FillColor="187,247,190" Height="60" IsVisible=
            "true" LaneId=" workflow_patterns_pool1_lan1 "
            ToolId="JaWE" Width="90">
34             <xpdl:Coordinates XCoordinate="356"
                YCoordinate="48" />
35             </xpdl:NodeGraphicsInfo>
36         </xpdl:NodeGraphicsInfos>
37 </xpdl:Activity>
38 <xpdl:Activity Id="C4" Name="C">
39     <xpdl:Description>Just press 'Complete' button</
        xpdl:Description>
40     <xpdl:Implementation>
41         <xpdl:No />
42     </xpdl:Implementation>
43     <xpdl:Performers>
44         <xpdl:Performer>test</xpdl:Performer>
45     </xpdl:Performers>
46     <xpdl:NodeGraphicsInfos>
47         <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
            FillColor="187,247,190" Height="60" IsVisible=
            "true" LaneId=" workflow_patterns_pool1_lan1 "
            ToolId="JaWE" Width="90">
48             <xpdl:Coordinates XCoordinate="509"
                YCoordinate="48" />
```

```
49         </xpd:NodeGraphicsInfo>
50     </xpd:NodeGraphicsInfos>
51 </xpd:Activity>
52 <xpd:Activity Id="WfBCP1_Sequence_act1">
53     <xpd:Event>
54         <xpd:StartEvent Trigger="None" />
55     </xpd:Event>
56     <xpd:NodeGraphicsInfos>
57         <xpd:NodeGraphicsInfo BorderColor="0,0,0"
58             FillColor="102,204,51" Height="31" IsVisible="
59             true" LaneId="workflow_patterns_pool1_lane1"
60             ToolId="JaWE" Width="31">
61             <xpd:Coordinates XCoordinate="92" YCoordinate
62                 ="88" />
63         </xpd:NodeGraphicsInfo>
64     </xpd:NodeGraphicsInfos>
65 </xpd:Activity>
66 <xpd:Activity Id="WfBCP1_Sequence_act2">
67     <xpd:Event>
68         <xpd:EndEvent Result="None" />
69     </xpd:Event>
70     <xpd:NodeGraphicsInfos>
71         <xpd:NodeGraphicsInfo BorderColor="0,0,0"
72             FillColor="236,120,98" Height="31" IsVisible="
73             true" LaneId="workflow_patterns_pool1_lane1"
74             ToolId="JaWE" Width="31">
75             <xpd:Coordinates XCoordinate="661"
76                 YCoordinate="101" />
77         </xpd:NodeGraphicsInfo>
78     </xpd:NodeGraphicsInfos>
79 </xpd:Activity>
80 </xpd:Activities>
81 <xpd:Transitions>
82     <xpd:Transition From="A3" Id="WfBCP1_Sequence_Tra1" Name="
83         Transition" To="B5">
84         <xpd:ConnectorGraphicsInfos>
85             <xpd:ConnectorGraphicsInfo FillColor="0,0,0"
86                 IsVisible="true" Style="NO.ROUTING.ORTHOGONAL"
87                 ToolId="JaWE" />
88         </xpd:ConnectorGraphicsInfos>
89     </xpd:Transition>
90     <xpd:Transition From="B5" Id="WfBCP1_Sequence_Tra2" Name="
```

```

      " Transition" To="C4">
80     <xpdl:ConnectorGraphicsInfos>
81       <xpdl:ConnectorGraphicsInfo FillColor="0,0,0"
          IsVisible="true" Style="NO.ROUTING.ORTHOGONAL"
          ToolId="JaWE" />
82     </xpdl:ConnectorGraphicsInfos>
83   </xpdl:Transition>
84   <xpdl:Transition From="WfBCP1_Sequence_act1" Id="
      WfBCP1_Sequence_tra1" To="A3">
85     <xpdl:ConnectorGraphicsInfos>
86       <xpdl:ConnectorGraphicsInfo FillColor="0,0,0"
          IsVisible="true" Style="NO.ROUTING.ORTHOGONAL"
          ToolId="JaWE" />
87     </xpdl:ConnectorGraphicsInfos>
88   </xpdl:Transition>
89   <xpdl:Transition From="C4" Id="WfBCP1_Sequence_tra2" To="
      WfBCP1_Sequence_act2">
90     <xpdl:ConnectorGraphicsInfos>
91       <xpdl:ConnectorGraphicsInfo FillColor="0,0,0"
          IsVisible="true" Style="NO.ROUTING.ORTHOGONAL"
          ToolId="JaWE" />
92     </xpdl:ConnectorGraphicsInfos>
93   </xpdl:Transition>
94 </xpdl:Transitions>
95 </xpdl:WorkflowProcess>

```

Secondo esempio

```

1 <xpdl:WorkflowProcess AccessLevel="PUBLIC" Id="
  WfBCP2_Parallel_Split" Name="WfBCP2_Parallel_Split">
2   <xpdl:ProcessHeader DurationUnit="D">
3     <xpdl:Created>2004-08-31 22:34:21</xpdl:Created>
4     <xpdl:Description>executes activity A, and after that
      activities B and C in parallel</xpdl:Description>
5   </xpdl:ProcessHeader>
6   <xpdl:RedefinableHeader PublicationStatus="UNDER_TEST">

```

```
7     <xpdl:Author>Sasa Bojanic</xpdl:Author>
8 </xpdl:RedefinableHeader>
9 <xpdl:Activities>
10   <xpdl:Activity Id="A4" Name="A">
11     <xpdl:Description>Just press 'Complete' button</
12       xpdl:Description>
13     <xpdl:Implementation>
14       <xpdl:No/>
15     </xpdl:Implementation>
16     <xpdl:Performers>
17       <xpdl:Performer>test</xpdl:Performer>
18     </xpdl:Performers>
19     <xpdl:TransitionRestrictions>
20       <xpdl:TransitionRestriction>
21         <xpdl:Split Type="Parallel">
22           <xpdl:TransitionRefs>
23             <xpdl:TransitionRef Id="
24               WfBCP2.Parallel.Split.Tra2" />
25             <xpdl:TransitionRef Id="
26               WfBCP2.Parallel.Split.Tra1" />
27           </xpdl:TransitionRefs>
28         </xpdl:Split>
29       </xpdl:TransitionRestriction>
30     </xpdl:TransitionRestrictions>
31     <xpdl:NodeGraphicsInfos>
32       <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
33         FillColor="187,247,190" Height="60" IsVisible=
34         "true" LaneId="workflow_patterns_pool2_lan1"
35         ToolId="JaWE" Width="90">
36       <xpdl:Coordinates XCoordinate="180"
37         YCoordinate="80" />
38     </xpdl:NodeGraphicsInfo>
39   </xpdl:NodeGraphicsInfos>
40 </xpdl:Activity>
41 <xpdl:Activity Id="B6" Name="B">
42   <xpdl:Description>Just press 'Complete' button</
43     xpdl:Description>
44   <xpdl:Implementation>
45     <xpdl:No/>
46   </xpdl:Implementation>
47   <xpdl:Performers>
48     <xpdl:Performer>test</xpdl:Performer>
```

```
41     </xpd:Performers>
42     <xpd:NodeGraphicsInfos>
43         <xpd:NodeGraphicsInfo BorderColor="0,0,0"
44             FillColor="187,247,190" Height="60" IsVisible=
45             "true" LaneId="workflow_patterns_pool2.lan1"
46             ToolId="JaWE" Width="90">
47             <xpd:Coordinates XCoordinate="330"
48                 YCoordinate="10" />
49         </xpd:NodeGraphicsInfo>
50     </xpd:NodeGraphicsInfos>
51 </xpd:Activity>
52 <xpd:Activity Id="C5" Name="C">
53     <xpd:Description>Just press 'Complete' button</
54     xpd:Description>
55     <xpd:Implementation>
56     <xpd:No />
57 </xpd:Implementation>
58 <xpd:Performers>
59     <xpd:Performer>test</xpd:Performer>
60 </xpd:Performers>
61 <xpd:NodeGraphicsInfos>
62     <xpd:NodeGraphicsInfo BorderColor="0,0,0"
63         FillColor="187,247,190" Height="60" IsVisible=
64         "true" LaneId="workflow_patterns_pool2.lan1"
65         ToolId="JaWE" Width="90">
66     <xpd:Coordinates XCoordinate="330"
67         YCoordinate="150" />
68     </xpd:NodeGraphicsInfo>
69 </xpd:NodeGraphicsInfos>
70 </xpd:Activity>
71 <xpd:Activity Id="WfBCP2_Parallel_Split_act1">
72     <xpd:Event>
73         <xpd:StartEvent Trigger="None" />
74     </xpd:Event>
75     <xpd:NodeGraphicsInfos>
76     <xpd:NodeGraphicsInfo BorderColor="0,0,0"
77         FillColor="102,204,51" Height="31" IsVisible="
78         true" LaneId="workflow_patterns_pool2.lan1"
79         ToolId="JaWE" Width="31">
80     <xpd:Coordinates XCoordinate="91" YCoordinate
81         ="41" />
82     </xpd:NodeGraphicsInfo>
```



```
70         </xpd:NodeGraphicsInfos>
71     </xpd:Activity>
72     <xpd:Activity Id="WfBCP2_Parallel_Split_act2">
73         <xpd:Event>
74             <xpd:EndEvent Result="None" />
75         </xpd:Event>
76         <xpd:NodeGraphicsInfos>
77             <xpd:NodeGraphicsInfo BorderColor="0,0,0"
78                 FillColor="236,120,98" Height="31" IsVisible="
79                 true" LaneId="workflow_patterns_pool2_lan1"
80                 ToolId="JaWE" Width="31">
81                 <xpd:Coordinates XCoordinate="490"
82                     YCoordinate="190" />
83             </xpd:NodeGraphicsInfo>
84         </xpd:NodeGraphicsInfos>
85     </xpd:Activity>
86     <xpd:Activity Id="WfBCP2_Parallel_Split_act3">
87         <xpd:Event>
88             <xpd:EndEvent Result="None" />
89         </xpd:Event>
90         <xpd:NodeGraphicsInfos>
91             <xpd:NodeGraphicsInfo BorderColor="0,0,0"
92                 FillColor="236,120,98" Height="31" IsVisible="
93                 true" LaneId="workflow_patterns_pool2_lan1"
94                 ToolId="JaWE" Width="31">
95                 <xpd:Coordinates XCoordinate="500"
96                     YCoordinate="50" />
97             </xpd:NodeGraphicsInfo>
98         </xpd:NodeGraphicsInfos>
99     </xpd:Activity>
100 </xpd:Activities>
101 <xpd:Transitions>
102     <xpd:Transition From="A4" Id="WfBCP2_Parallel_Split_Tra1"
103         Name="Transition" To="B6">
104         <xpd:ConnectorGraphicsInfos>
105             <xpd:ConnectorGraphicsInfo FillColor="0,0,0"
106                 IsVisible="true" Style="NO_ROUTING_ORTHOGONAL"
107                 ToolId="JaWE" />
108         </xpd:ConnectorGraphicsInfos>
109     </xpd:Transition>
110     <xpd:Transition From="A4" Id="WfBCP2_Parallel_Split_Tra2"
111         Name="Transition" To="C5">
```

```
100         <xpdl:ConnectorGraphicsInfos>
101             <xpdl:ConnectorGraphicsInfo FillColor="0,0,0"
                IsVisible="true" Style="NO.ROUTING.ORTHOGONAL"
                ToolId="JaWE" />
102         </xpdl:ConnectorGraphicsInfos>
103     </xpdl:Transition>
104     <xpdl:Transition From="WfBCP2_Parallel_Split_act1" Id="
                WfBCP2_Parallel_Split_tra1" To="A4">
105         <xpdl:ConnectorGraphicsInfos>
106             <xpdl:ConnectorGraphicsInfo FillColor="0,0,0"
                IsVisible="true" Style="NO.ROUTING.ORTHOGONAL"
                ToolId="JaWE" />
107         </xpdl:ConnectorGraphicsInfos>
108     </xpdl:Transition>
109     <xpdl:Transition From="C5" Id="WfBCP2_Parallel_Split_tra2"
                To="WfBCP2_Parallel_Split_act2">
110         <xpdl:ConnectorGraphicsInfos>
111             <xpdl:ConnectorGraphicsInfo FillColor="0,0,0"
                IsVisible="true" Style="NO.ROUTING.ORTHOGONAL"
                ToolId="JaWE" />
112         </xpdl:ConnectorGraphicsInfos>
113     </xpdl:Transition>
114     <xpdl:Transition From="B6" Id="WfBCP2_Parallel_Split_tra3"
                To="WfBCP2_Parallel_Split_act3">
115         <xpdl:ConnectorGraphicsInfos>
116             <xpdl:ConnectorGraphicsInfo FillColor="0,0,0"
                IsVisible="true" Style="NO.ROUTING.ORTHOGONAL"
                ToolId="JaWE" />
117         </xpdl:ConnectorGraphicsInfos>
118     </xpdl:Transition>
119 </xpdl:Transitions>
120 </xpdl:WorkflowProcess>
```

Terzo esempio

```

2
3 <xpdl:WorkflowProcess AccessLevel="PUBLIC" Id="
  WfBCP3_Synchronization" Name="WfBCP3_Synchronization">
4   <xpdl:ProcessHeader DurationUnit="D">
5     <xpdl:Created>2004-08-31 22:38:46</xpdl:Created>
6     <xpdl:Description>synchronize execution of two parallel
      activities</xpdl:Description>
7   </xpdl:ProcessHeader>
8   <xpdl:RedefinableHeader PublicationStatus="UNDER_TEST">
9     <xpdl:Author>Sasa Bojanic</xpdl:Author>
10  </xpdl:RedefinableHeader>
11  <xpdl:Activities>
12    <xpdl:Activity Id="A5" Name="A">
13      <xpdl:Description>Just press 'Complete' button</
        xpdl:Description>
14      <xpdl:Implementation>
15        <xpdl:No/>
16      </xpdl:Implementation>
17      <xpdl:Performers>
18        <xpdl:Performer>test</xpdl:Performer>
19      </xpdl:Performers>
20      <xpdl:TransitionRestrictions>
21        <xpdl:TransitionRestriction>
22          <xpdl:Split Type="Parallel">
23            <xpdl:TransitionRefs>
24              <xpdl:TransitionRef Id="
                WfBCP3_Synchronization_Tra1"/>
25              <xpdl:TransitionRef Id="
                WfBCP3_Synchronization_Tra2"/>
26            </xpdl:TransitionRefs>
27          </xpdl:Split>
28        </xpdl:TransitionRestriction>
29      </xpdl:TransitionRestrictions>
30      <xpdl:NodeGraphicsInfos>
31        <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
          FillColor="187,247,190" Height="60" IsVisible=
            "true" LaneId="workflow_patterns_pool3_lan1"
            ToolId="JaWE" Width="90">
32          <xpdl:Coordinates XCoordinate="150"
            YCoordinate="80"/>
33        </xpdl:NodeGraphicsInfo>
34      </xpdl:NodeGraphicsInfos>

```

```
35     </xpdl:Activity>
36     <xpdl:Activity Id="B7" Name="B">
37         <xpdl:Description>Just press 'Complete' button</
           xpdl:Description>
38         <xpdl:Implementation>
39             <xpdl:No/>
40         </xpdl:Implementation>
41         <xpdl:Performers>
42             <xpdl:Performer>test</xpdl:Performer>
43         </xpdl:Performers>
44         <xpdl:NodeGraphicsInfos>
45             <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
           FillColor="187,247,190" Height="60" IsVisible=
           "true" LaneId="workflow_patterns_pool3_lan1"
           ToolId="JaWE" Width="90">
46                 <xpdl:Coordinates XCoordinate="320"
           YCoordinate="20" />
47             </xpdl:NodeGraphicsInfo>
48         </xpdl:NodeGraphicsInfos>
49     </xpdl:Activity>
50     <xpdl:Activity Id="C6" Name="C">
51         <xpdl:Description>Just press 'Complete' button</
           xpdl:Description>
52         <xpdl:Implementation>
53             <xpdl:No/>
54         </xpdl:Implementation>
55         <xpdl:Performers>
56             <xpdl:Performer>test</xpdl:Performer>
57         </xpdl:Performers>
58         <xpdl:NodeGraphicsInfos>
59             <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
           FillColor="187,247,190" Height="60" IsVisible=
           "true" LaneId="workflow_patterns_pool3_lan1"
           ToolId="JaWE" Width="90">
60                 <xpdl:Coordinates XCoordinate="320"
           YCoordinate="160" />
61             </xpdl:NodeGraphicsInfo>
62         </xpdl:NodeGraphicsInfos>
63     </xpdl:Activity>
64     <xpdl:Activity Id="D6" Name="D">
65         <xpdl:Description>Just press 'Complete' button</
           xpdl:Description>
```

```
66         <xpdl:Implementation>
67             <xpdl:No/>
68         </xpdl:Implementation>
69         <xpdl:Performers>
70             <xpdl:Performer>test</xpdl:Performer>
71         </xpdl:Performers>
72         <xpdl:NodeGraphicsInfos>
73             <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
74                 FillColor="187,247,190" Height="60" IsVisible=
75                 "true" LaneId="workflow_patterns_pool3_lan1"
76                 ToolId="JaWE" Width="90">
77                 <xpdl:Coordinates XCoordinate="530"
78                     YCoordinate="81"/>
79             </xpdl:NodeGraphicsInfo>
80         </xpdl:NodeGraphicsInfos>
81     </xpdl:Activity>
82     <xpdl:Activity Id="D1">
83         <xpdl:Route GatewayType="Parallel"/>
84         <xpdl:TransitionRestrictions>
85             <xpdl:TransitionRestriction>
86                 <xpdl:Join Type="Parallel"/>
87             </xpdl:TransitionRestriction>
88         </xpdl:TransitionRestrictions>
89         <xpdl:NodeGraphicsInfos>
90             <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
91                 FillColor="255,229,124" Height="43" IsVisible=
92                 "true" LaneId="workflow_patterns_pool3_lan1"
93                 ToolId="JaWE" Width="43">
94                 <xpdl:Coordinates XCoordinate="443"
95                     YCoordinate="90"/>
96             </xpdl:NodeGraphicsInfo>
97         </xpdl:NodeGraphicsInfos>
98     </xpdl:Activity>
99     <xpdl:Activity Id="WfBCP3_Synchronization_act1">
100         <xpdl:Event>
101             <xpdl:StartEvent Trigger="None"/>
102         </xpdl:Event>
103         <xpdl:NodeGraphicsInfos>
104             <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
105                 FillColor="102,204,51" Height="31" IsVisible="
106                 true" LaneId="workflow_patterns_pool3_lan1"
107                 ToolId="JaWE" Width="31">
```

```
97         <xpdl:Coordinates XCoordinate="74" YCoordinate
98             ="24" />
99     </xpdl:NodeGraphicsInfo>
100 </xpdl:NodeGraphicsInfos>
101 </xpdl:Activity>
102 <xpdl:Activity Id="WfBCP3_Synchronization_act2">
103     <xpdl:Event>
104         <xpdl:EndEvent Result="None" />
105     </xpdl:Event>
106     <xpdl:NodeGraphicsInfos>
107         <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
108             FillColor="236,120,98" Height="31" IsVisible="
109             true" LaneId="workflow_patterns_pool3_lane1"
110             ToolId="JaWE" Width="31">
111         <xpdl:Coordinates XCoordinate="647"
112             YCoordinate="26" />
113     </xpdl:NodeGraphicsInfo>
114 </xpdl:NodeGraphicsInfos>
115 </xpdl:Activity>
116 </xpdl:Activities>
117 <xpdl:Transitions>
118     <xpdl:Transition From="A5" Id="WfBCP3_Synchronization_Tra1
119         " Name="Transition" To="B7">
120     <xpdl:ConnectorGraphicsInfos>
121         <xpdl:ConnectorGraphicsInfo FillColor="0,0,0"
122             IsVisible="true" Style="NO-ROUTING-ORTHOGONAL"
123             ToolId="JaWE" />
124     </xpdl:ConnectorGraphicsInfos>
125 </xpdl:Transition>
126 <xpdl:Transition From="A5" Id="WfBCP3_Synchronization_Tra2
127     " Name="Transition" To="C6">
128     <xpdl:ConnectorGraphicsInfos>
129         <xpdl:ConnectorGraphicsInfo FillColor="0,0,0"
130             IsVisible="true" Style="NO-ROUTING-ORTHOGONAL"
131             ToolId="JaWE" />
132     </xpdl:ConnectorGraphicsInfos>
133 </xpdl:Transition>
134 <xpdl:Transition From="B7" Id="WfBCP3_Synchronization_Tra3
135     " Name="Transition" To="D1">
136     <xpdl:ConnectorGraphicsInfos>
137         <xpdl:ConnectorGraphicsInfo FillColor="0,0,0"
138             IsVisible="true" Style="NO-ROUTING-ORTHOGONAL"
```

```

    ToolId="JaWE" />
126     </xpd:ConnectorGraphicsInfos>
127 </xpd:Transition>
128 <xpd:Transition From="C6" Id="WfBCP3_Synchronization_Tra4
    " Name="Transition" To="D1">
129     <xpd:ConnectorGraphicsInfos>
130         <xpd:ConnectorGraphicsInfo FillColor="0,0,0"
            IsVisible="true" Style="NO.ROUTING.ORTHOGONAL"
            ToolId="JaWE" />
131     </xpd:ConnectorGraphicsInfos>
132 </xpd:Transition>
133 <xpd:Transition From="D1" Id="WfBCP3_Synchronization_tra1
    " To="D6">
134     <xpd:ConnectorGraphicsInfos>
135         <xpd:ConnectorGraphicsInfo FillColor="0,0,0"
            IsVisible="true" ToolId="JaWE" />
136     </xpd:ConnectorGraphicsInfos>
137 </xpd:Transition>
138 <xpd:Transition From="WfBCP3_Synchronization_act1" Id="
    WfBCP3_Synchronization_tra2" To="A5">
139     <xpd:ConnectorGraphicsInfos>
140         <xpd:ConnectorGraphicsInfo FillColor="0,0,0"
            IsVisible="true" Style="NO.ROUTING.ORTHOGONAL"
            ToolId="JaWE" />
141     </xpd:ConnectorGraphicsInfos>
142 </xpd:Transition>
143 <xpd:Transition From="D6" Id="WfBCP3_Synchronization_tra3
    " To="WfBCP3_Synchronization_act2">
144     <xpd:ConnectorGraphicsInfos>
145         <xpd:ConnectorGraphicsInfo FillColor="0,0,0"
            IsVisible="true" Style="NO.ROUTING.ORTHOGONAL"
            ToolId="JaWE" />
146     </xpd:ConnectorGraphicsInfos>
147 </xpd:Transition>
148 </xpd:Transitions>
149 </xpd:WorkflowProcess>
```

Listing completo caso di studio.

```
1 <?xml version="1.0" encoding="UTF-8" standalone="no" ?>
2 <xpdl:Package xmlns:xpdl="http://www.wfmc.org/2008/XPDL2.1" xmlns=
  "http://www.wfmc.org/2008/XPDL2.1" xmlns:xsi="http://www.w3.
  org/2001/XMLSchema-instance" Id="newpkg1" Name="newpkg1"
  xsi:schemaLocation="http://www.wfmc.org/2008/XPDL2.1 http://
  www.wfmc.org/standards/docs/bpmnxpdl_31.xsd">
3   <xpdl:PackageHeader>
4     <xpdl:XPDLVersion>2.1</xpdl:XPDLVersion>
5     <xpdl:Vendor>(c) Together Teamsolutions Co., Ltd.</
      xpdl:Vendor>
6     <xpdl:Created>2013-06-08 10:59:46</xpdl:Created>
7   </xpdl:PackageHeader>
8   <xpdl:Participants>
9     <xpdl:Participant Id="newpkg1_par1" Name="Auditor">
10      <xpdl:ParticipantType Type="HUMAN" />
11    </xpdl:Participant>
12    <xpdl:Participant Id="corsia1" Name="Responsabile_Qualit\
      a_Manager">
13      <xpdl:ParticipantType Type="HUMAN" />
14      <xpdl:ExtendedAttributes>
15        <xpdl:ExtendedAttribute Name="JaWE_TYPE" Value="
          LANE.DEFAULT" />
16      </xpdl:ExtendedAttributes>
17    </xpdl:Participant>
18    <xpdl:Participant Id="corsia2" Name="Auditor_Consulente">
19      <xpdl:ParticipantType Type="HUMAN" />
20      <xpdl:ExtendedAttributes>
21        <xpdl:ExtendedAttribute Name="JaWE_TYPE" Value="
          LANE.DEFAULT" />
22      </xpdl:ExtendedAttributes>
23    </xpdl:Participant>
24    <xpdl:Participant Id="corsia3" Name="Certificatore">
25      <xpdl:ParticipantType Type="HUMAN" />
26      <xpdl:ExtendedAttributes>
27        <xpdl:ExtendedAttribute Name="JaWE_TYPE" Value="
          LANE.DEFAULT" />
28      </xpdl:ExtendedAttributes>
29    </xpdl:Participant>
30  </xpdl:Participants>
31  <xpdl:Pools>
```



```
32     <xpdl:Pool BoundaryVisible="true" Id="newpkg1_pool1"
33         MainPool="true" Name="Processo_di_certificazione"
34         Orientation="HORIZONTAL" Process="processo1">
35         <xpdl:Lanes>
36             <xpdl:Lane Id="newpkg1_pool1_lan1" Name="
37                 Responsabile_Qualit`a_Manager">
38                 <xpdl:NodeGraphicsInfos>
39                     <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
40                         FillColor="220,220,220" IsVisible="
41                         true" ToolId="JaWE" />
42                     <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
43                         FillColor="220,220,220" IsVisible="
44                         true" ToolId="JaWE" />
45                 </xpdl:NodeGraphicsInfos>
46                 <xpdl:Performers>
47                     <xpdl:Performer>corsia1</xpdl:Performer>
48                 </xpdl:Performers>
49             </xpdl:Lane>
50             <xpdl:Lane Id="newpkg1_pool1_lan2" Name="Auditor_
51                 Consulente">
52                 <xpdl:NodeGraphicsInfos>
53                     <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
54                         FillColor="220,220,220" IsVisible="
55                         true" ToolId="JaWE" />
56                     <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
57                         FillColor="220,220,220" IsVisible="
58                         true" ToolId="JaWE" />
59                 </xpdl:NodeGraphicsInfos>
60                 <xpdl:Performers>
61                     <xpdl:Performer>corsia2</xpdl:Performer>
62                 </xpdl:Performers>
63             </xpdl:Lane>
64             <xpdl:Lane Id="newpkg1_pool1_lan3" Name="
65                 Certificatore">
66                 <xpdl:NodeGraphicsInfos>
67                     <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
68                         FillColor="220,220,220" IsVisible="
69                         true" ToolId="JaWE" />
70                     <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
71                         FillColor="220,220,220" IsVisible="
72                         true" ToolId="JaWE" />
73                 </xpdl:NodeGraphicsInfos>
```

```

57             <xpdl:Performers>
58                 <xpdl:Performer>corsia3</xpdl:Performer>
59             </xpdl:Performers>
60         </xpdl:Lane>
61     </xpdl:Lanes>
62     <xpdl:NodeGraphicsInfos>
63         <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
64             FillColor="255,255,215" IsVisible="true"
65             ToolId="JaWE" />
66     </xpdl:NodeGraphicsInfos>
67 </xpdl:Pool>
68 </xpdl:Pools>
69 <xpdl:WorkflowProcesses>
70     <xpdl:WorkflowProcess Id="processo1" Name="Processo_di_
71         certificazione">
72         <xpdl:ProcessHeader>
73             <xpdl:Created>2013-06-08 11:00:39</xpdl:Created>
74         </xpdl:ProcessHeader>
75         <xpdl:Activities>
76             <xpdl:Activity Id="start1" Name="inizio_processo">
77                 <xpdl:Event>
78                     <xpdl:StartEvent Trigger="None" />
79                 </xpdl:Event>
80                 <xpdl:NodeGraphicsInfos>
81                     <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
82                         FillColor="102,204,51" Height="31"
83                         IsVisible="true" LaneId="
84                         newpkg1_pool1_lan1" ToolId="JaWE"
85                         Width="31">
86                     <xpdl:Coordinates XCoordinate="69"
87                         YCoordinate="62" />
88                     </xpdl:NodeGraphicsInfo>
89                 </xpdl:NodeGraphicsInfos>
90             </xpdl:Activity>
91             <xpdl:Activity Id="attivit 1" Name="
92                 Pianificazione_certificazione">
93                 <xpdl:Implementation>
94                     <xpdl:No />
95                 </xpdl:Implementation>
96             <xpdl:Performers>
97                 <xpdl:Performer>corsia1</xpdl:Performer>
98             </xpdl:Performers>

```

```

90         <xpdl:NodeGraphicsInfos>
91             <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
                FillColor="187,247,190" Height="60"
                IsVisible="true" LaneId="
                newpkg1_pool1_lan1" ToolId="JaWE"
                Width="90">
92                 <xpdl:Coordinates XCoordinate="168"
                    YCoordinate="47" />
93             </xpdl:NodeGraphicsInfo>
94         </xpdl:NodeGraphicsInfos>
95     </xpdl:Activity>
96 <xpdl:Activity Id="attivit 2" Name="Esecuzione_
    audit">
97     <xpdl:Implementation>
98         <xpdl:No />
99     </xpdl:Implementation>
100    <xpdl:Performers>
101        <xpdl:Performer>corsia2</xpdl:Performer>
102    </xpdl:Performers>
103    <xpdl:TransitionRestrictions>
104        <xpdl:TransitionRestriction>
105            <xpdl:Join Type="Exclusive" />
106        </xpdl:TransitionRestriction>
107    </xpdl:TransitionRestrictions>
108    <xpdl:NodeGraphicsInfos>
109        <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
                FillColor="187,247,190" Height="60"
                IsVisible="true" LaneId="
                newpkg1_pool1_lan2" ToolId="JaWE"
                Width="90">
110            <xpdl:Coordinates XCoordinate="196"
                YCoordinate="46" />
111        </xpdl:NodeGraphicsInfo>
112    </xpdl:NodeGraphicsInfos>
113 </xpdl:Activity>
114 <xpdl:Activity Id="split1" Name="a_norma?">
115     <xpdl:Route GatewayType="Exclusive" />
116     <xpdl:TransitionRestrictions>
117         <xpdl:TransitionRestriction>
118             <xpdl:Split Type="Exclusive">
119                 <xpdl:TransitionRefs>

```

```
120         <xpdl:TransitionRef Id="
121             transizione4" />
122         <xpdl:TransitionRef Id="
123             transizione5" />
124     </xpdl:TransitionRefs>
125 </xpdl:Split>
126 </xpdl:TransitionRestriction>
127 </xpdl:TransitionRestrictions>
128 <xpdl:NodeGraphicsInfos>
129     <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
130         FillColor="255,229,124" Height="43"
131         IsVisible="true" LaneId="
132         newpkg1_pool1_lan2" ToolId="JaWE"
133         Width="43">
134         <xpdl:Coordinates XCoordinate="332"
135             YCoordinate="55" />
136     </xpdl:NodeGraphicsInfo>
137 </xpdl:NodeGraphicsInfos>
138 </xpdl:Activity>
139 <xpdl:Activity Id="attivit 3" Name="Stesura _
140     report _e _Lista _compiti">
141     <xpdl:Implementation>
142         <xpdl:No />
143     </xpdl:Implementation>
144     <xpdl:Performers>
145         <xpdl:Performer>corsia2</xpdl:Performer>
146     </xpdl:Performers>
147     <xpdl:NodeGraphicsInfos>
148         <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
149             FillColor="187,247,190" Height="60"
150             IsVisible="true" LaneId="
151             newpkg1_pool1_lan2" ToolId="JaWE"
152             Width="90">
153             <xpdl:Coordinates XCoordinate="433"
154                 YCoordinate="10" />
155         </xpdl:NodeGraphicsInfo>
156     </xpdl:NodeGraphicsInfos>
157 </xpdl:Activity>
158 <xpdl:Activity Id="attivit 4" Name="Stesura _
159     report _e _invio _a _certificatore">
160     <xpdl:Implementation>
161         <xpdl:No />
```

```
148         </xpd:Implementation>
149     <xpd:Performers>
150         <xpd:Performer>corsia2</xpd:Performer>
151     </xpd:Performers>
152     <xpd:NodeGraphicsInfos>
153         <xpd:NodeGraphicsInfo BorderColor="0,0,0"
154             FillColor="187,247,190" Height="60"
155             IsVisible="true" LaneId="
156             newpkg1_pool1_lan2" ToolId="JaWE"
157             Width="90">
158             <xpd:Coordinates XCoordinate="436"
159                 YCoordinate="100" />
160         </xpd:NodeGraphicsInfo>
161     </xpd:NodeGraphicsInfos>
162 </xpd:Activity>
163 <xpd:Activity Id="attivit " Name="Esecuzione _
164     compiti">
165     <xpd:Implementation>
166         <xpd:No />
167     </xpd:Implementation>
168     <xpd:Performers>
169         <xpd:Performer>corsia1</xpd:Performer>
170     </xpd:Performers>
171     <xpd:NodeGraphicsInfos>
172         <xpd:NodeGraphicsInfo BorderColor="0,0,0"
173             FillColor="187,247,190" Height="60"
174             IsVisible="true" LaneId="
175             newpkg1_pool1_lan1" ToolId="JaWE"
176             Width="90">
177             <xpd:Coordinates XCoordinate="483"
178                 YCoordinate="40" />
179         </xpd:NodeGraphicsInfo>
180     </xpd:NodeGraphicsInfos>
181 </xpd:Activity>
182 <xpd:Activity Id="attivit 6" Name="Valutazione _
183     report">
184     <xpd:Implementation>
185         <xpd:No />
186     </xpd:Implementation>
187     <xpd:Performers>
188         <xpd:Performer>corsia3</xpd:Performer>
189     </xpd:Performers>
```

```
178         <xpdl:NodeGraphicsInfos>
179             <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
                FillColor="187,247,190" Height="60"
                IsVisible="true" LaneId="
                newpkg1_pool1_lan3" ToolId="JaWE"
                Width="90">
180                 <xpdl:Coordinates XCoordinate="574"
                    YCoordinate="51" />
181             </xpdl:NodeGraphicsInfo>
182         </xpdl:NodeGraphicsInfos>
183     </xpdl:Activity>
184     <xpdl:Activity Id="fineprocesso" Name="Fine_
        processo">
185         <xpdl:Event>
186             <xpdl:EndEvent Result="None" />
187         </xpdl:Event>
188         <xpdl:NodeGraphicsInfos>
189             <xpdl:NodeGraphicsInfo BorderColor="0,0,0"
                FillColor="236,120,98" Height="31"
                IsVisible="true" LaneId="
                newpkg1_pool1_lan3" ToolId="JaWE"
                Width="31">
190                 <xpdl:Coordinates XCoordinate="729"
                    YCoordinate="67" />
191             </xpdl:NodeGraphicsInfo>
192         </xpdl:NodeGraphicsInfos>
193     </xpdl:Activity>
194 </xpdl:Activities>
195 <xpdl:Transitions>
196     <xpdl:Transition From="start1" Id="transizione1"
        To="attivit 1">
197         <xpdl:ConnectorGraphicsInfos>
198             <xpdl:ConnectorGraphicsInfo FillColor="
                0,0,0" IsVisible="true" Style="
                NO_ROUTING_SPLINE" ToolId="JaWE" />
199         </xpdl:ConnectorGraphicsInfos>
200     </xpdl:Transition>
201     <xpdl:Transition From="attivit 1" Id="
        transizione2" To="attivit 2">
202         <xpdl:ConnectorGraphicsInfos>
203             <xpdl:ConnectorGraphicsInfo FillColor="
                0,0,0" IsVisible="true" Style="
```

```
NO_ROUTING_SPLINE" ToolId="JaWE" />
204     </xpd:ConnectorGraphicsInfos>
205 </xpd:Transition>
206 <xpd:Transition From=" attivit 2 " Id="
      transizione3" To=" split1 ">
207     <xpd:ConnectorGraphicsInfos>
208         <xpd:ConnectorGraphicsInfo FillColor="
              0,0,0" IsVisible="true" Style="
              NO_ROUTING_SPLINE" ToolId="JaWE" />
209     </xpd:ConnectorGraphicsInfos>
210 </xpd:Transition>
211 <xpd:Transition From=" attivit 3 " Id="
      transizione6" To=" attivit 5 ">
212     <xpd:ConnectorGraphicsInfos>
213         <xpd:ConnectorGraphicsInfo FillColor="
              0,0,0" IsVisible="true" Style="
              NO_ROUTING_SPLINE" ToolId="JaWE" />
214     </xpd:ConnectorGraphicsInfos>
215 </xpd:Transition>
216 <xpd:Transition From=" attivit 4 " Id="
      transizione8" To=" attivit 6 ">
217     <xpd:ConnectorGraphicsInfos>
218         <xpd:ConnectorGraphicsInfo FillColor="
              0,0,0" IsVisible="true" Style="
              NO_ROUTING_SPLINE" ToolId="JaWE" />
219     </xpd:ConnectorGraphicsInfos>
220 </xpd:Transition>
221 <xpd:Transition From=" attivit 6 " Id="
      transizione9" To=" fineprocesso">
222     <xpd:ConnectorGraphicsInfos>
223         <xpd:ConnectorGraphicsInfo FillColor="
              0,0,0" IsVisible="true" Style="
              NO_ROUTING_SPLINE" ToolId="JaWE" />
224     </xpd:ConnectorGraphicsInfos>
225 </xpd:Transition>
226 <xpd:Transition From=" attivit 5 " Id="
      transizione7" To=" attivit 2 ">
227     <xpd:ConnectorGraphicsInfos>
228         <xpd:ConnectorGraphicsInfo FillColor="
              0,0,0" IsVisible="true" Style="
              NO_ROUTING_SPLINE" ToolId="JaWE">
```

```
229         <xpdl:Coordinates XCoordinate="406"
230             YCoordinate="89" />
231     </xpdl:ConnectorGraphicsInfo>
232 </xpdl:ConnectorGraphicsInfos>
233 </xpdl:Transition>
234 <xpdl:Transition From="split1" Id="transizione4"
235     Name="No" To="attivit 3">
236     <xpdl:ConnectorGraphicsInfos>
237         <xpdl:ConnectorGraphicsInfo FillColor="
238             0,0,0" IsVisible="true" Style="
239             NO_ROUTING_SPLINE" ToolId="JaWE">
240             <xpdl:Coordinates XCoordinate="418"
241                 YCoordinate="176" />
242             </xpdl:ConnectorGraphicsInfo>
243         </xpdl:ConnectorGraphicsInfos>
244     </xpdl:Transition>
245     <xpdl:Transition From="split1" Id="transizione5"
246     Name="Si" To="attivit 4">
247     <xpdl:ConnectorGraphicsInfos>
248         <xpdl:ConnectorGraphicsInfo FillColor="
249             0,0,0" IsVisible="true" Style="
250             NO_ROUTING_SPLINE" ToolId="JaWE">
251             <xpdl:Coordinates XCoordinate="423"
252                 YCoordinate="274" />
253             </xpdl:ConnectorGraphicsInfo>
254         </xpdl:ConnectorGraphicsInfos>
255     </xpdl:Transition>
256 </xpdl:Transitions>
257 </xpdl:WorkflowProcess>
258 </xpdl:WorkflowProcesses>
259 <xpdl:ExtendedAttributes>
260     <xpdl:ExtendedAttribute Name="EDITING_TOOL" Value="
261         Together_Workflow_Editor" />
262     <xpdl:ExtendedAttribute Name="EDITING_TOOL_VERSION" Value="
263         "4.8-1-20130204-2300-TAB-2.4-1" />
264     <xpdl:ExtendedAttribute Name="JaWE_CONFIGURATION" Value="
265         default" />
266 </xpdl:ExtendedAttributes>
267 </xpdl:Package>
```

Bibliografia

- [1] Carlo Combi and Sara Degani: *Seamless (and Temporal) Conceptual Modeling of Business Process Information*, *ADBIS (2)*, pp. 23-32 , 01/2011
- [2] Johann Eder and Mária Bieliková and A Min Tjoa: *ADBIS 2011, Research Communications, Proceedings II of the 15th East-European Conference on Advances in Databases and Information Systems*, vol 789, pp. 20-23, September 2011, Vienna, Austria
- [3] Elisa Bertino, Elena Ferrari e Vijay Atluri: *The Specification and Enforcement of Authorization Constraints in Workflow Management Systems*, *ACM Trans. Inf. Syst. Secur.*, vol. 2, no. 1, pp. 65-104, 1999
- [4] James B.D. Joshi, Member, IEEE, Elisa Bertino, Fellow, IEEE, Usman Latif, and Arif Ghafoor, Fellow, IEEE: *A Generalized Temporal Role-Based Access Control Model*, *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 1, pp. 4-23
- [5] Elisa Bertino and Piero A. Bonatti and Elena Ferrari: *A temporal role-based access control model*, *ACM Trans. Inf. Syst. Secur.*, vol. 4, no. 3, pp. 191-233 , 2001
- [6] Carlo Combi, Sara Degani and Christian S. Jensen: *Capturing Temporal Constraints in Temporal ER Models*
- [7] W.M.P. van der Aalst, M. Adams, A.H.M. ter Hofstede, M. Pesic and H. Schonenberg: *Flexibility as a Service*, *BPM Center Report BPM-08-09*, *BPMcenter.org*, 2008.
- [8] Workflow Management Coalition: *Workflow Standard*
- [9] Stephen A. White: *XPDL and BPMN* , e *Workflow Handbook 2003*, 2003

- [10] W.M.P. van der Aalst and A.H.M. ter Hofstede: *YAWL: Yet Another Workflow Language (Revised version)*, *information Systems*, vol.30, no. 4, pp. 245-275, 2005