

## 1. La gestione dell'azienda basata sui processi

I problemi relativi all'analisi delle aziende e alla diagnosi e ristrutturazione dei loro processi hanno acquistato negli anni recenti una sempre maggior rilevanza, essendo legati a concetti come:

- il passaggio da una visione funzionale dell'azienda ad una visione per processi,
- la gestione del cambiamento come un'esigenza permanente per l'azienda,
- la ristrutturazione dei processi al fine di migliorare le prestazioni dell'azienda.

A causa del cambiamento continuo l'azienda si trova oggi a valutare costantemente la sua posizione sul mercato e a cercare nuove strategie e nuove tecniche per innovarsi ed acquisire un vantaggio competitivo rispetto alle aziende concorrenti.

Il cambiamento interessa innanzitutto il contesto **esterno** dell'azienda: clienti, fornitori, concorrenti, dinamiche di mercato, leggi, regolamenti comunitari, e così via. Ad esempio cresce costantemente l'esigenza dei clienti di ottenere prodotti e servizi sempre più personalizzati, e questo provoca un mercato sempre più frammentato che richiede un'azienda sempre più flessibile e "snella", in grado di modificarsi rapidamente in termini di nuove linee di prodotto, qualità dei servizi, cura dei clienti.

Alla crescente complessità del contesto esterno corrisponde una crescente complessità dell'organizzazione **interna** dell'azienda, al fine di incrementare l'efficienza delle sue attività e l'efficacia del controllo e del coordinamento delle attività stesse per raggiungere gli obiettivi che l'azienda si prefigge.

In questo testo l'azienda sarà considerata come un **sistema** "aperto" all'interazione col mondo esterno, che ad esso cerca di adattarsi e nello stesso tempo cerca di influenzarlo; questa continua interazione fa sì che un'azienda moderna sia un sistema:

- **complesso** e caratterizzato da un gran numero di componenti (strutture, attività, prodotti, attori, risorse,...),
- **distribuito**, basato su sistemi di comunicazione eterogenei e geograficamente dispersi,
- **aleatorio**, condizionato da frequenti e imprevedibili eventi perturbatori, sia esterni (mercati, borse, regolamenti, leggi,...) che interni (guasti, incidenti, dimissioni,...).

### 1.1 L'Approccio Sistemico

La complessità dell'azienda vista come un sistema rende poco praticabile l'analisi e lo studio dell'azienda stessa con le tecniche tradizionali di tipo analitico.

Nell'approccio analitico l'analisi globale di un sistema, visto come un insieme di attività che trasformano oggetti in ingresso in oggetti in uscita, prende in considerazione delle "macro" variabili caratteristiche del sistema in analisi che possono classificarsi in **livelli** e **flussi**: i livelli sono variabili di stato del sistema e indicano il valore attuale di "oggetti" che si "accumulano" nel sistema come risultato dei **flussi** di oggetti in ingresso e uscita dal sistema stesso (prodotti in un magazzino, clienti in una banca, impiegati di un'azienda, e così via). Hanno particolare interesse i sistemi in stato stazionario cioè sistemi in cui il tasso  $\lambda$  di arrivo ( $n^\circ$  di oggetti in ingresso nell'unità di tempo) è in media uguale a quello di uscita.

Nella pratica, molti sistemi possono rappresentarsi come un sistema di code del tipo illustrato in Figura 1.1.

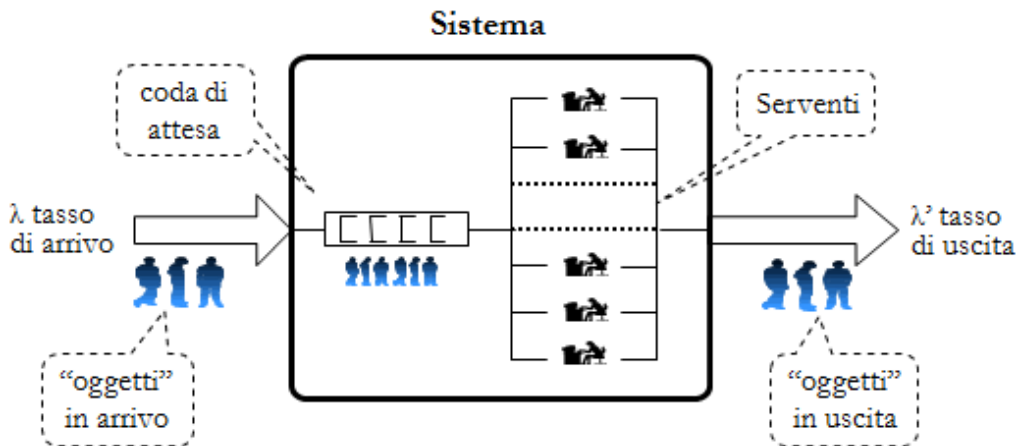


Figura 1.1: L'azienda come sistema

Per caratterizzare il sistema possiamo introdurre un certo numero di variabili di cui prenderemo in considerazione i valori medi (riferiti ad una certa unità di tempo):

S : numero di serventi	L : numero di oggetti in coda
$N = L + S$ : numero di oggetti nel sistema	
$\mu$ : tasso di servizio	$T = 1/\mu$ : tempo di servizio
$\Lambda_c$ : tempo di attesa (trascorso in coda)	
$\Lambda = \Lambda_c + 1/\mu$ : tempo di ciclo (trascorso complessivamente nel sistema)	

In condizioni stazionarie, cioè con un numero di "oggetti" che entrano all'incirca uguale a quello di "oggetti" che escono ( $\lambda \approx \lambda'$ ) valgono le relazioni:

$N = \lambda * \Lambda$	$L = \lambda * \Lambda_c$	leggi di Little
-------------------------	---------------------------	-----------------

Le leggi di Little valgono sotto condizioni abbastanza generali e forniscono direttamente delle indicazioni molto utili. Ad esempio se un'agenzia di assicurazioni tratta 120000 pratiche (pr) in un anno e in media, in un momento qualsiasi, ci sono 600 pratiche in agenzia (in attesa o in "lavorazione") possiamo chiederci qual è il tempo di ciclo; supposto che in un anno ci siano circa 50 settimane lavorative avremo:

se 120000 pr/anno  $\rightarrow \lambda = 120000/50 = 240$  pr/sett. (tasso di arrivo)  
 da  $N = \lambda * \Lambda = 600$  pr  $\rightarrow \Lambda = 600/240 = 2.5$  sett. (tempo di ciclo)

e possiamo concludere che una pratica richiede circa due settimane e mezza per essere evasa. Come altro semplice esempio consideriamo un checkpoint di sicurezza in un aeroporto e supponiamo che: 1) la lunghezza media (L) della coda sia di 16.5 passeggeri e 2) il checkpoint smaltisca 600 passeggeri all'ora ( $\lambda = 10$  passeggeri al minuto), avremo:

da  $L = \lambda * \Lambda_c = 16.5$  pass  $\rightarrow \Lambda_c = 16.5/10 = 1.65$  min

per cui un passeggero aspetta in media 1.65 minuti.

Questi esempi mostrano due semplici (ma significative) implicazioni delle leggi di Little:

1. dei tre indicatori di efficienza del sistema ( $\lambda$ ,  $N$  o  $L$ ,  $\Lambda$  o  $\Lambda_c$ ) il gestore del processo può focalizzarsi solo su due di loro (ad esempio quelli che sono più facilmente misurabili!), il terzo è fornito dalle leggi;
2. per un certo livello di "carico" del sistema ( $\lambda \approx \lambda'$  può essere interpretato come il carico di lavoro che il sistema può sopportare in un certo tempo) l'unico modo per ridurre le code è ridurre il tempo di ciclo (e viceversa).

Le leggi di Little possono usarsi anche in fase di progetto del sistema (e del relativo processo); in tal caso si può operare sui parametri:

$S$  = numero di serventi in parallelo (ad esempio dottori in pronto soccorso, cassieri al supermarket,...);

$L$  = lunghezza massima ammessa nella coda (ad esempio numero di chiamate messe in attesa al call-center, numero di pazienti in attesa di essere visitati al pronto soccorso,...);  $N = L+S$  sarà il numero massimo di oggetti (pazienti, clienti,...) ammessi nel sistema.

E' chiaro che se aumentiamo  $L$  diminuisce la soddisfazione del cliente, se aumentiamo  $S$  aumenta il carico di lavoro che il sistema può sopportare e la coda diminuisce ma aumentano anche i costi: il bilanciamento fra questi due fattori potrebbe fornire una buona stima del numero  $S$  di serventi da prevedere in fase di progetto; resta comunque la difficoltà di valutare i costi, ad esempio il "costo" relativo all'attesa al pronto soccorso è differente da quello al supermarket!

Osserviamo infine che finora abbiamo considerato il sistema come una "scatola chiusa" approssimata mediante una coda e dei serventi, e quindi è stato costruito un **modello** del sistema piuttosto semplice (e di cui la teoria delle code fornisce delle soluzioni analitiche): non è pensabile di estendere l'analisi al processo interno al sistema modellando le attività in modo analogo a quello con cui abbiamo modellato il sistema perché la rete di code risultante è raramente trattabile in modo così semplice.

L'approccio sistemico non è quindi proponibile come strumento generale con cui modellare e studiare i processi aziendali.

Ci si può chiedere a questo punto se esistono approcci diversi alla costruzione di modelli in grado di fornire risultati significativi e fortunatamente la risposta è positiva. Osserviamo anzitutto che per i processi (o sistemi) che ci interessano l'evoluzione nel tempo non è continua ma si passa da uno stato ad un altro stato solo quando si verificano degli **eventi**, ad esempio arriva un ordine, un paziente è dimesso, un macchinario utensile termina una lavorazione e così via. Questo fa rientrare i processi aziendali in una classe di sistemi, quella dei **sistemi a eventi discreti**, che allo stato attuale permette di utilizzare degli strumenti estremamente potenti come la simulazione, questo approccio sarà trattato diffusamente nel seguito.

## 1.2 L'Organizzazione Funzionale

L'organizzazione tradizionale con cui un'azienda si struttura per raggiungere i propri obiettivi è chiamata **funzionale** in quanto raggruppa tutte le attività simili all'interno dell'azienda; ad esempio tutte le risorse di una casa editrice specializzate nella vendita, indipendentemente dal fatto che si occupino di linee diverse di prodotto come l'editoria scolastica, quella universitaria o le enciclopedie, sono assegnate alla stessa funzione Marketing&Vendite. Le **funzioni** sono quindi aggregazioni di uomini e mezzi necessari per lo svolgimento di una determinata tipologia di attività; tali attività richiedono le stesse competenze e utilizzano lo stesso tipo di risorse e di tecnologie, e sono raggruppate in una **unità organizzativa** (ad esempio Acquisti, Vendite, Produzione e così via) sotto un'unica responsabilità.

Una tipica organizzazione funzionale comprende ad esempio le seguenti funzioni:

1. **Produzione** : comprende le attività necessarie a produrre i beni o i servizi per il cliente, a partire dai beni e servizi acquistati dai fornitori e usando le risorse aziendali;

2. **Gestione della Produzione** : comprende le attività decisionali riguardanti la redditività delle attività di produzione; la gestione implica il controllo continuo delle entrate e dei costi di produzione per coordinare le attività al fine di raggiungere gli obiettivi desiderati;
3. **Gestione della Qualità** : comprende le attività atte a garantire la qualità non solo dei prodotti ma anche del processo di produzione (visto come il presupposto per garantire la qualità dei prodotti finali); le attività sono inserite in un sistema di gestione della qualità che controlla la conformità dell'organizzazione a norme come la ISO 9000 (e seguenti);
4. **Marketing e Vendite** : le attività di marketing hanno l'obiettivo di identificare, anticipare e soddisfare le esigenze dei clienti, e sono quindi collegate alle attività di vendita che gestiscono il processo di scambio di prodotti e servizi dell'azienda con i clienti a fronte dei loro pagamenti;
5. **Gestione Generale** : è l'insieme delle attività su cui ricade la responsabilità dei profitti e delle perdite dell'azienda, per cui questa funzione coordina tutte le altre per sostenere l'integrità e la crescita dell'azienda stessa;
6. **Contabilità e Finanza** : il compito delle attività finanziarie è quello di assicurare i mezzi finanziari necessari al funzionamento dell'impresa e di creare valore a partire dal bilancio societario; la contabilità gestisce essenzialmente i flussi di cassa dell'azienda;
7. **Ricerca e Sviluppo** : comprende e organizza in modo sistematico le attività creative dell'azienda che riguardano la ricerca di nuovi prodotti e nuovi sistemi di produzione, l'uso efficiente delle risorse e le iniziative di innovazione e diversificazione;
8. **Gestione delle Risorse Umane** : comprende tutte le attività relative alla gestione della forza lavoro dell'azienda inclusa la concezione di un ambiente di lavoro gradevole e la ricerca delle competenze necessarie per raggiungere gli obiettivi aziendali.

In questo tipo di organizzazione ogni funzione gestisce le sue risorse aggregandole per competenze, ha un suo budget e dei suoi obiettivi, ed è responsabile della gestione delle attività a cui assegna le risorse necessarie.

La necessità di fare delle valutazioni sulle attività è legata al concetto che il fine ultimo di ogni attività svolta dall'azienda dovrebbe essere quello di creare dei prodotti che vengano incontro ai desideri dei clienti o, detto in altro modo, *creino del valore per il cliente*; ovviamente non tutte le attività di un'azienda possono creare del valore che poi il cliente percepisce (ad esempio la manutenzione di un'apparecchiatura non riguarda direttamente il cliente) ma quelle che creano valore meritano una particolare attenzione!

Con l'introduzione del concetto di valore l'azienda può essere "letta" secondo una prospettiva detta *catena del valore di Porter*: l'azienda è vista come un sistema di attività generatrici di valore, inteso come il prezzo che il consumatore è disposto a pagare per il prodotto che soddisfa pienamente i propri bisogni; nel prossimo paragrafo questo concetto sarà discusso più in dettaglio.

### 1.3 La Catena del Valore e i Processi

Il concetto del valore induce una classificazione delle attività aziendali in attività che aggiungono valore (VA) per il cliente, attività che non aggiungono valore (NVA) per il cliente, e attività che aggiungono valore per l'azienda (VAA). Nella prospettiva della catena del valore le attività che genericamente aggiungono valore (VA e VAA) possono appartenere fondamentalmente a due tipi, Attività Primarie e Attività di Supporto, come mostrato in Figura 1.2.

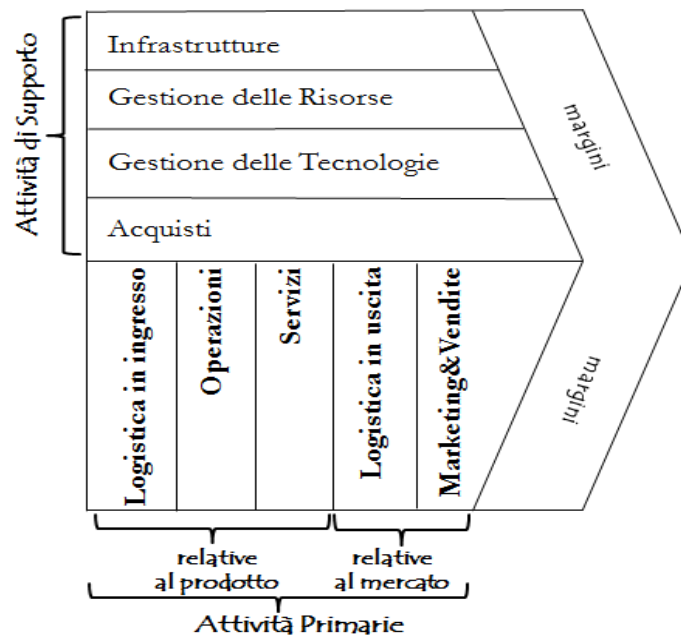


Figura 1.2: La Catena del Valore

Le attività primarie possono essere relative al prodotto (attività che aggiungono valore al prodotto o servizio erogato al cliente) o relative al mercato. Le attività relative al prodotto sono classificabili come:

- **Logistica in ingresso:** sono tutte quelle attività connesse al ricezione di materie prime, componenti, semi-lavorati che l'azienda acquista dai suoi fornitori e comprendono i controlli in arrivo, lo smistamento, il deposito nei magazzini e la preparazione per le lavorazioni successive;
- **Operazioni:** includono le attività direttamente rivolte alla produzione dei prodotti o all'erogazione dei servizi ai clienti, i controlli, la confezione dei prodotti, la gestione degli impianti, e così via;
- **Servizi:** comprendono tutte le attività legate alla cura del cliente e al "dopo-vendita", come quelle relative alla gestione delle manutenzioni e delle riparazioni, gli aggiornamenti, l'addestramento dei clienti e così via.

Le attività primarie relative al mercato comprendono le attività che si occupano della gestione dei prodotti finiti dalle unità produttive verso i clienti e quindi sono classificate come:

- **Logistica in uscita:** comprendono lo smistamento dei prodotti e la gestione del magazzino in uscita, l'elaborazione degli ordini, la programmazione delle spedizioni, il trasporto e la distribuzione verso i clienti;
- **Marketing e vendite:** queste attività includono lo studio del comportamento dei clienti, la scelta dei prodotti, la determinazione dei prezzi, la scelta e la gestione dei canali di vendita (uso di grossisti e dettaglianti, punti di vendita propri, grande distribuzione, via Internet,...), la comunicazione pubblicitaria, le promozioni, e così via.

Le attività di supporto comprendono tutte quelle attività che l'azienda deve svolgere per fare in modo che le attività primarie possano essere svolte nel modo migliore possibile, esse possono riguardare:

- **Acquisti:** comprendono tutte le attività connesse alla ricerca e all'acquisto dei beni (materie prime, componenti, semi-lavorati, servizi) che saranno trasformati in prodotti finiti, alle migliori condizioni per l'azienda (in termini di costo, tempi di consegna, qualità e così via);
- **Gestione delle Tecnologie:** riguardano le iniziative di ricerca e sviluppo di nuovi prodotti, nuovi materiali, nuovi processi di produzione, l'automazione della produzione, l'ottimizzazione e così via;

- **Gestione delle Risorse:** queste attività riguardano anzitutto le risorse umane come ricerca di competenze, assunzione, pianificazione delle carriere, politiche di incentivazione, addestramento, riqualificazione e così via;
- **Infrastrutture:** le attività relative includono tutto quello che riguarda la direzione generale, l'organizzazione, la gestione dell'informatica, l'amministrazione, la finanza, gli affari legali, i rapporti con gli enti pubblici e la gestione della qualità.

Le attività generatrici di valore producono i **ricavi** dell'azienda e richiedono delle risorse per la loro esecuzione, per cui determinano anche i **costi**; come mostrato in Figura 1.2 il **margin**e per l'azienda è la differenza tra i ricavi e i costi.

I concetti di catena del valore da una parte e di architettura funzionale dall'altra sono ovviamente complementari e quindi è certamente utile analizzare il valore delle attività gestite dalle funzioni aziendali. L'organizzazione funzionale presenta però degli inconvenienti:

- favorisce una struttura statica, organizzata gerarchicamente con linee di comando e di controllo rigide;
- non permette di individuare le responsabilità verso il cliente finale;
- ha una visione ristretta, focalizzata al proprio interno, per cui le comunicazioni con le altre funzioni spesso non sono pienamente soddisfacenti e il coordinamento necessario fra le attività aziendali risulta carente;
- il miglioramento riguarda le attività che la funzione gestisce, per cui è limitato e può avere effetti negativi a livello globale.

Per poter impostare in modo più efficace il problema dell'analisi delle attività e del miglioramento dell'efficienza ci si è resi conto che è necessaria una visione dell'azienda (e dei meccanismi che al suo interno regolano l'avanzamento del lavoro) più ampia di quella fornita dall'architettura funzionale.

Questa visione è fornita dal concetto di **processo** ed è illustrata in Figura 1.3. Le **funzioni** aziendali, associate ad **unità organizzative** dell'azienda, sono rappresentate da rettangoli a bordi arrotondati, contenenti le attività gestite dalla funzione stessa (e rappresentate con rettangoli). Il **processo** "ricuce" le attività a partire da un evento esterno generato da un cliente (che ad esempio invia un ordine) e può "attraversare" diverse funzioni aziendali per ritornare al cliente (che riceve una spedizione con i prodotti ordinati).

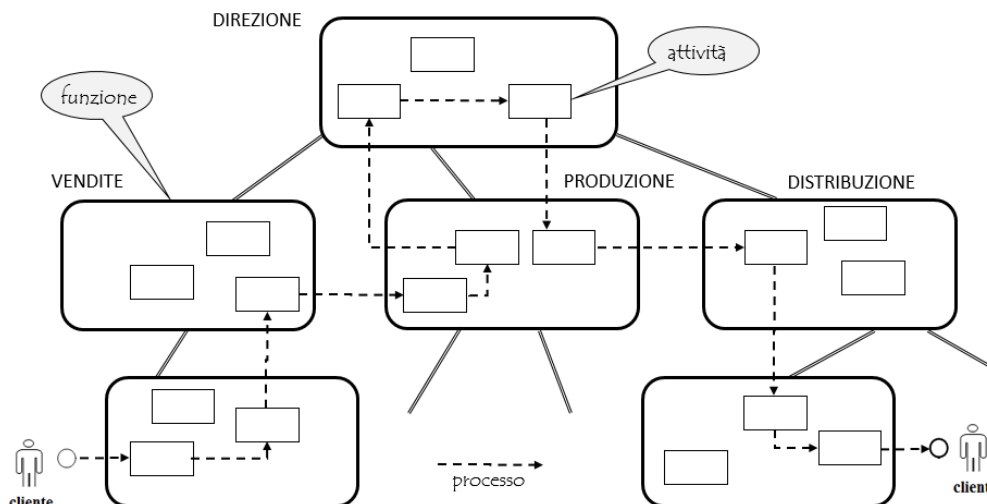


Figura 1.3: Funzioni aziendali e Processi

La figura rappresenta "graficamente" la situazione attuale, in cui continua ad esistere la struttura dell'azienda basata su unità organizzative e funzioni ma su questa struttura si tende a far "emergere" la rete dei processi e di conse-

guenza a introdurre metodologie di gestione che della visione per processi fanno il loro punto di forza; si può dire quindi che la tendenza attuale è il passaggio da una azienda organizzata per funzioni ad una azienda orientata ai processi.

Possiamo definire un **processo** aziendale come un insieme di **attività** (controlli e azioni) tra loro interrelate per realizzare un **risultato** definito e misurabile, il prodotto o servizio che trasferisce **valore** al fruitore (al **cliente**) del prodotto o servizio stesso.

Il processo deve avere un **nome** ed è importante assegnare dei nomi significativi; generalmente questo si ottiene adottando una struttura del tipo:

<verbo di azione all'infinito> <nome o frase nominale>

Il nome dovrebbe possibilmente riferirsi al risultato del processo, ad esempio Preparare bilancio oppure Rilasciare certificato.

Un processo in azienda è caratterizzato da un insieme di proprietà che hanno come riscontro diverse esigenze che i processi stessi dovrebbero soddisfare, come illustrato nella Tabella 1.1.

	Proprietà	Esigenze
1.	risponde ad un evento esterno eseguendo delle attività	valutare l'efficienza del processo in termini di tempi
2.	deve raggiungere certi obiettivi	valutare lo scostamento fra la strategia aziendale e i risultati del processo
3.	deve fornire un risultato	misurare i risultati del processo
4.	consuma delle risorse aziendali	valutare l'efficienza del processo in termini di costi
5.	deve soddisfare i requisiti dei clienti	migliorare la qualità
6.	è vincolato da regole interne ed esterne	assicurare la conformità alle regole
7.	ha generalmente un responsabile	gestire e monitorare il processo

Tabella 1.1: Proprietà di un Processo

Le proprietà di un processo sono molto importanti e meritano un'attenta riflessione; possiamo dire che il processo:

1. risponde ad un evento esterno eseguendo delle attività: possiamo considerare il processo come la risposta che l'azienda mette in atto quando accade qualcosa nel mondo esterno. E' quindi fondamentale il concetto di **evento**, inteso come l'accadere, in un certo istante di tempo, di un fatto rilevante per l'azienda e a cui l'azienda deve reagire; il modo in cui reagisce è appunto un insieme di **attività** eseguite in un certo ordine. Ad un evento è in genere associata un'informazione, ad esempio al tempo t arriva un ordine del cliente Rossi: l'azienda reagisce con la prima attività del processo **Gestire Ordini** (l'attività Protocollo ordine) che termina all'istante t' con la conclusione dell'ultima attività (Consegnare pacco). Poiché il cliente si aspetta di ricevere gli articoli ordinati "prima possibile" il tempo intercorso fra gli istante t e t' è un parametro importante (chiamato in genere tempo di ciclo), di qui l'esigenza di misurarlo e ridurlo per quanto è possibile. Un processo di questo tipo è chiamato "end-to-end" perché inizia e termina sul cliente;
2. deve raggiungere certi obiettivi: il processo è lo strumento con cui l'azienda persegue i propri obiettivi. Ad esempio per un'azienda di noleggio auto che ha come obiettivo raggiungere un certo budget alla fine dell'anno il processo **Noleggiare auto** ad un cliente è chiaramente importante, da qui l'esigenza di misurare lo scostamento fra

l'obiettivo finale e la situazione attuale che potrebbe consigliare, ad esempio, una campagna promozionale nell'ultimo trimestre;

3. deve fornire un risultato: il risultato per il cliente del processo deve essere chiaramente *identificabile* (questo implica anzitutto identificare i clienti del processo!) e *misurabile* (ad esempio in termini di oggetti prodotti o servizi erogati in un certo tempo); possiamo avere diversi "clienti" in un processo (esterni e/o interni) ma ciascuno deve ricevere un risultato. Ad esempio nel processo **Gestire ordini** il cliente riceve un certo prodotto e l'azienda riceve un pagamento;
4. consuma delle risorse aziendali: il processo durante la sua esecuzione consuma delle risorse aziendali come il tempo dei dipendenti (che hanno un costo orario), materiali, energia, servizi di manutenzione e così via; di qui l'esigenza di valutare i costi in modo da poter determinare il prezzo dei beni prodotti;
5. deve soddisfare i requisiti dei clienti: i clienti scelgono un prodotto in base a certi requisiti di qualità che hanno in mente, per cui il processo deve adeguarsi per quanto possibile a questi requisiti (che possono variare nel tempo!) cercando di migliorare continuamente i prodotti attuali e (possibilmente) prevedere il tipo e la qualità dei beni o dei servizi che il cliente potrebbe richiedere in futuro;
6. è vincolato da regole interne ed esterne: ogni azienda ha delle regole interne che possono determinare l'esecuzione di alcuni processi; ad esempio in una banca "il cassiere non può consegnare ad un cliente una somma superiore a quella coperta dal denaro depositato sul conto e dal fido che il cliente ha a disposizione"; analogamente possono esserci regole esterne (leggi, regolamenti comunitari, contratti nazionali di lavoro e così via) che devono essere rispettate (ad esempio "per noleggiare un'auto il cliente deve avere una patente valida"), da qui l'esigenza che il processo tenga conto di tutte le possibili situazioni in cui queste regole potrebbero essere violate;
7. ha generalmente un responsabile (il cosiddetto "Process Owner"): è colui che ha la responsabilità globale del processo; opera in maniera trasversale rispetto alle funzioni, deve gestire e monitorare il processo verificando le tempistiche di realizzazione e gli output intermedi e finali.

L'importanza di considerare processi "end-to-end" frequentemente non è compresa a fondo; spesso sono chiamati processi degli "insiemi di azioni" svolte in azienda e che sarebbe corretto chiamare in altro modo, ad esempio:

- procedura o metodo: se l'insieme rappresenta il modo in cui viene svolta un'attività o un insieme limitato di attività all'interno di una funzione organizzativa;
- area di processo: se ci si riferisce ad un insieme di processi correlati fra loro; ad esempio l'area chiamata Customer Relationship Management (CRM) *non* è un processo;
- frammenti di processo: sono insiemi di attività correlate fra loro ma interne ad una singola funzione;
- sotto-processi: sono processi *interni* ad un processo "end-to-end".

La distinzione diventa rilevante se si pensa di intraprendere delle azioni di ottimizzazione di questi "pseudo-processi": è infatti noto che delle ottimizzazioni locali possono avere delle conseguenze negative sui processi globali!

Queste osservazioni permettono di comprendere il ruolo fondamentale che i processi assumono in azienda: anche se l'azienda continua per tradizione ad articolarsi per funzioni tutte le considerazioni riguardanti il miglioramento delle prestazioni aziendali dovrebbero essere basate su come effettivamente si svolge il lavoro all'interno dell'azienda e quindi sui suoi processi.



## 1.4 L'Organizzazione e la Gestione basata sui Processi

Le considerazioni espresse nei paragrafi precedenti indicano chiaramente l'importanza del passaggio da un'organizzazione dell'azienda di tipo funzionale ad un'organizzazione orientata ai processi.

In quest'ultimo decennio la disciplina che tratta l'utilizzo dei processi per la gestione dell'azienda è chiamata generalmente **Business Process Management** o **BPM**.

Il BPM è un insieme di metodi, tecniche e strumenti per gestire nel modo migliore l'azienda e può considerarsi come il risultato di un'evoluzione partita con la fine della II° guerra mondiale che ha raccolto risultati (metodi, tecniche, strumenti) da vari filoni di ricerca, classificabili orientativamente in tre ampie categorie e nel cui contesto il BPM assume diversi significati:

1. **gestione aziendale:** il BPM è una filosofia (e una metodologia) per gestire l'azienda organizzandola "intorno" ai suoi processi operativi, coinvolgendo con l'azienda (le sue funzioni e le sue risorse), i clienti, i fornitori e i partner;
2. **controllo della qualità:** il BPM è un insieme di tecniche (statistiche) finalizzate al miglioramento della qualità di prodotti (o servizi) attraverso l'analisi della loro variabilità e di quella dei relativi processi di produzione (o erogazione);
3. **tecnologie ICT:** il BPM è un insieme di tecnologie e strumenti per progettare, analizzare, controllare, eseguire e ristrutturare i processi operativi dell'azienda.

La Tabella 1.2 riporta metodologie, tecniche e strumenti che hanno contribuito a questa evoluzione; i limiti di questo testo non permettono di descrivere, sia pure brevemente, i punti toccati nella tabella, comunque diversi di questi saranno ripresi nella metodologia che sarà illustrata nel Capitolo 4.

Gestione Aziendale	Controllo della Qualità	Tecnologie ICT
<i>Catena del Valore</i> (Porter) <i>Balanced Score Card</i> <i>Activity based mangement</i> <i>Framework architetturali</i> (SCOR, COBIT,...)	<i>Lean production</i> <i>Total Quality Management</i> <i>SixSigma</i> <i>Lean SixSigma</i> <i>Continuous improvement</i> <i>Capability Maturity Model</i>	<i>IT architectures</i> (Zachman) <i>Enterprise architectures</i> (TOGAF) <i>Metodologie software</i> <i>Linguaggi</i> (UML, BPMN) <i>Simulatori di processo</i> <i>Motori di workflow</i> <i>Sistemi informativi d'azienda</i> (ERP,CRM, ...) <i>Business Intelligence</i> <i>Business rules</i>

Tabella 1.2: Evoluzione del BPM

Il risultato è un insieme, ancora a contorni piuttosto vaghi, di concetti, definizioni, tecniche, strumenti, metodologie a volte contrastanti ma che comunque si richiamano alla sigla BPM immettendo sul mercato centinaia di prodotti commerciali fra cui è difficile districarsi. Per chiarire la situazione che si è creata osserviamo che le tre categorie sopra accennate rappresentano tre **punti di vista** che si riflettono in azienda in **ruoli** distinti come specificato in Tabella 1.3.

Concetti	Gestione Aziendale	Controllo della Qualità	Tecnologie ICT
----------	--------------------	-------------------------	----------------

Punti di vista	<u>Gestionale</u>	<u>della Qualità</u>	<u>Tecnologico</u>
Ruoli	<u>Responsabile gestionale</u>	<u>Responsabile della Qualità</u>	<u>Progettista BPM</u>

Tabella 1.3: Ruoli nel BPM

Possiamo allora dire che:

1. il Responsabile gestionale ha bisogno di strumenti per:
  - a) controllare l'esecuzione dei processi, gestendo in modo diretto l'attivazione di applicazioni e l'accesso a dati preesistenti (per esempio dati ereditati da applicazioni obsolete),
  - b) ridurre i rischi e garantire la conformità a leggi e regolamenti,
  - c) curare la distribuzione di documenti, dati e compiti per le risorse aziendali fornendo le giuste informazioni in tempo utile e nel posto richiesto;
2. il Responsabile della Qualità è interessato a strumenti per controllare la qualità dei prodotti/servizi forniti al cliente a partire dalla qualità dei relativi processi (valutabile in termini di difettosità, rilavorazioni, ritardi,...) e dalla qualità delle componenti e delle materie prime acquistate dai fornitori;
3. il Progettista BPM utilizza modelli, linguaggi e metodologie per descrivere, progettare e simulare i processi aziendali al fine di supportare la strategia dell'azienda nel perseguire i propri obiettivi nel modo più efficiente ed efficace possibile; i modelli di processo devono quindi permettere l'analisi di tempi, costi e qualità dei processi di produzione (o erogazione) dei prodotti (o servizi) forniti al cliente.

Risulta evidente a questo punto che è fondamentale costruire una specifica dei processi aziendali che possa essere facilmente compresa da tutti i ruoli aziendali interessati ai processi e quindi funzionare da "linguaggio comune". Il linguaggio che è stato adottato in questo testo è attualmente lo standard internazionale per la specifica dei processi, chiamato BPMN (Business Process Management and Notation - versione 2) ed è stato emesso dal gruppo di standardizzazione internazionale OMG (Object Management Group) insieme ad altri linguaggi come il Business Motivation Model (BMM) e lo UML (Unified Modeling Language) che saranno anch'essi considerati nel seguito.

Il BPM si pone degli obiettivi certamente importanti:

- a) ottimizzare i processi "interni" dell'azienda: il BPM deve creare una cultura aziendale ed un contesto in cui coordinare l'applicazione di metodologie e strumenti in grado di gestire attività di miglioramento continuo dei processi alternati a passi di miglioramento radicale che possono rendersi indispensabili in momenti critici in cui è necessario introdurre delle iniziative di innovazione. La Figura 1.4 illustra un andamento di questo tipo.

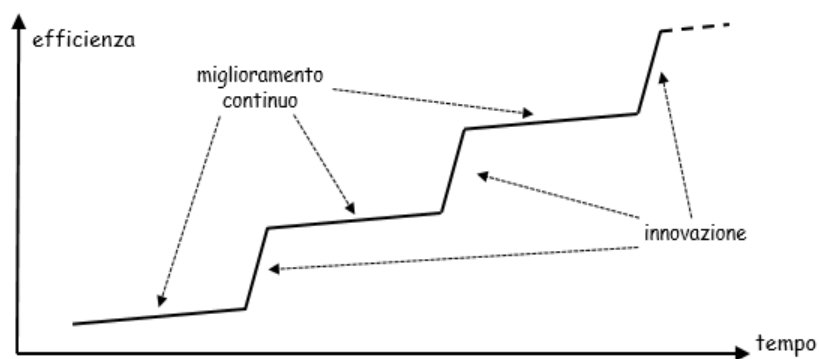
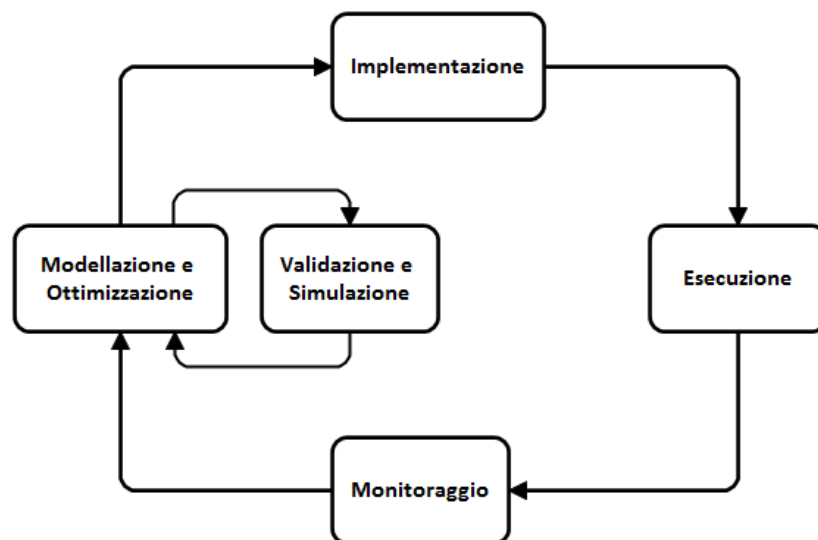


Figura 1.4: Andamento nel tempo dell'efficienza dei processi

In particolare il BPM deve rispondere a quesiti di base come:

- quali passi sono veramente necessari? chi deve eseguirli?
  - come devono essere svolti e quali competenze richiedono?
  - quali sono i risultati attesi e come valutarli?
  - come eliminare gli sprechi lungo il processo?
- b) facilitare l'allineamento dei processi gestionali e produttivi agli obiettivi strategici dell'azienda (migliorando produttività, gestione delle risorse e qualità);
- c) migliorare la flessibilità dell'organizzazione aziendale in modo da rispondere "rapidamente" ai cambiamenti di scenario;
- d) facilitare il flusso del processo fra i vari partecipanti cercando di eliminare le attese, in particolare dei clienti;
- e) automatizzare il flusso di controllo e dei documenti al proprio interno (tramite un motore di workflow) e, possibilmente, fra i vari partecipanti al processo (compresi clienti e fornitori);
- f) analizzare, modellare e misurare il processo introducendo degli indicatori di efficienza (KPI - Key Performance Indicators) per individuare punti critici;
- g) utilizzare il modello per fare delle previsioni (analisi "what if");
- h) migliorare l'ambiente di lavoro eliminando i passi ripetitivi, noiosi o poco utili mediante un'applicazione "intelligente" delle tecnologie
- i) ripensare i processi avendo come obiettivo il miglioramento della qualità e dell'efficienza.

La costruzione del modello del processo riveste un ruolo importante nel **ciclo di vita** del processo stesso. Come già discusso un processo non è una struttura statica ma si evolve nel tempo passando attraverso le fasi specificate in Figura 1.5.



"Figura 1.5: Ciclo di vita dei processi

La Tabella 1.4 descrive brevemente i compiti delle varie fasi e gli attori relativi, cioè i ruoli aziendali incaricati di

condurre le fasi stesse.

<u>Fasi</u>	<u>Compiti</u>	<u>Attori</u>
<u>Modellazione e Ottimizzazione</u>	- costruire il modello - definire gli indicatori - ottimizzare il modello attraverso simulazioni e analisi "what if"	analista
<u>Validazione e Simulazione</u>	- validare il modello attraverso l'animazione e la simulazione - verificare il modello sugli indicatori attuali	analista
<u>Implementazione</u>	- tradurre il modello sul motore di workflow	sviluppatore
<u>Esecuzione</u>	- eseguire il workflow	gestore
<u>Monitoraggio</u>	- controllare gli indicatori - identificare le possibilità di miglioramento - decidere la ristrutturazione	analista e gestore

Tabella 1.4: Fasi del ciclo di vita dei processi

Il ciclo di vita inizia con la fase di **modellazione** del processo; il modello viene specificato con il linguaggio BPMN e quindi **validato** attraverso dei passi di animazione e **simulazione** in cui i risultati sul modello sono confrontati con i dati attuali degli indicatori per assicurare una buona "aderenza" del modello rispetto alla realtà aziendale. Una volta validato il modello può essere **ottimizzato** attraverso azioni di ristrutturazione che possono essere verificate con la simulazione e l'analisi "what if" di scenari differenti; il ciclo "interno" termina quando il modello prodotto mostra dei risultati soddisfacenti sul simulatore.

Nella fase di **implementazione** il modello viene "tradotto" nelle specifiche di input del **motore di workflow** per essere quindi automatizzato durante l'**esecuzione** del processo. Il motore è un sistema software che, ricevuta in ingresso la versione eseguibile del modello, provvede a gestire il processo durante l'esecuzione, si occupa di interagire con i vari attori partecipanti coordinando le attività. Poiché il motore tiene traccia di tutte le operazioni che sono effettuate il **monitoraggio** utilizza questi dati per capire se le prestazioni del processo rientrano nei livelli previsti o si rende necessario un nuovo ciclo di ottimizzazione del processo stesso.

Ovviamente non sempre si ha a disposizione in azienda un sistema di esecuzione di questo tipo per cui l'esecuzione del processo avviene sovente in modo "informale" attraverso l'opera degli esecutori; ad esempio è un certo impiegato che quando ha finito di svolgere un determinato compito sulla pratica provvede a portare la pratica stessa nell'ufficio in cui deve essere svolta l'attività successiva.

Anche gli oggetti (dati, documenti, archivi,...) trattati dal processo devono essere gestiti dal motore di workflow; la tendenza attuale per dati e documenti è di passare a versioni digitali e di caricare gli archivi nel sistema informativo aziendale, di conseguenza i motori di workflow dovrebbero integrare sistemi di gestione documentale e di accesso al sistema informativo.

La gestione del ciclo di vita, almeno per i processi rilevanti dell'azienda, rappresenta un impegno non indifferente. Per capire a che punto si trova l'azienda è stato introdotto un **modello di maturità dei processi** che prevede cinque **livelli di maturità** come illustrato nella Tabella 1.5.

liv.	tipo	Descrizione
1	operativo	il processo produce i beni ma sono definiti solo gli input, gli output, le risorse e gli

		oggetti che deve trattare; i risultati sono poco prevedibili perché dipendono da come le persone eseguono il processo;
2	<b>regolato</b>	il processo è operativo ed esistono regole per eseguirlo e risorse specifiche, è definito per macro attività fino a un livello che permette di pianificare la sua esecuzione e valutare alcuni indicatori; l'esecuzione è manuale o con l'ausilio di alcuni strumenti;
3	<b>definito</b>	il processo è regolato e definito a livello di attività che sono specificate completamente in termini di risorse, risultati e oggetti trattati; poiché la definizione completa delle attività le rende controllabili i risultati del processo sono prevedibili; il processo può essere automatizzato su un motore di workflow;
4	<b>gestito</b>	Il processo è definito e controllato usando tecniche quantitative, generalmente di tipo statistico; il processo può essere controllato in tempo reale a livello di istanza e quindi si possono prendere delle decisioni per ottimizzare l'esecuzione;
5	<b>ottimizzato</b>	Il processo è gestito quantitativamente e sono definite delle tecniche per ottimizzarlo in modo continuo basandosi sui risultati di molte istanze e di esigenze generali dell'azienda (ad esempio la necessità di accelerare la produzione in certi periodi di tempo).

Tabella 1.5: Livelli di maturità dei processi

La tabella mostra come per raggiungere i livelli di maturità più alti (a partire dal livello 3) sia necessario per l'azienda dotarsi di strumenti per la modellazione e l'esecuzione dei processi, e quindi sarebbe certamente utile che i modelli (anche quelli più "astratti") fossero a loro volta (in un qualche modo) eseguibili.

## 1.5 La Gestione dei Progetti

Il passaggio dell'orientamento di un'azienda da funzioni a processi è un **progetto** aziendale di particolare rilevanza per l'impatto che avrà su tutta l'azienda. A questo punto è necessario capire cosa è e cosa implica la gestione di un progetto e che relazione esiste fra il concetto di progetto e il concetto di processo.

Un **progetto** in letteratura è definito come una iniziativa dell'azienda di durata predefinita che si propone di realizzare un certo risultato (un prodotto o un servizio). Un progetto è quindi **unico** e **temporaneo** e si svolge attraverso un percorso che prevede dei **traguardi intermedi** per arrivare infine al conseguimento del suo **obiettivo**.

La prima caratteristica dei progetti è l'**unicità**: ogni progetto produce un unico prodotto o un unico servizio. Spesso all'interno di un'organizzazione aziendale un progetto che nasce ne ricalca, per grandi linee, uno già esistente e questa similitudine si traduce in due ordini di vantaggi: innanzitutto la pianificazione risulta facilitata se il nuovo progetto aderisce ad un modello noto e già collaudato e poi alcuni risultati tangibili, i **deliverables**, del nuovo progetto possono essere recuperati da progetti precedenti.

Occorre comunque tener presente che anche se dei progetti sono simili e si possono riciclare alcune componenti di base un progetto è e resta unico nella sua peculiarità e il gestore del progetto (o **Project Manager**) deve essere molto attento ad evidenziare gli elementi di differenza nella gestione delle similitudini tra progetti.

Seconda caratteristica importante è la **temporaneità**: ogni progetto ha una durata finita con un inizio ed una fine ben stabiliti. Sia inizio che fine sono fasi delicate che possono portare, se mal impostate, a criticità nella conduzione dell'intero ciclo progettuale.

Esistono numerose categorie di progetti che accanto alle caratteristiche generali che abbiamo visto presentano caratteristiche particolari; possiamo avere ad esempio:

1. progetti aerospaziali o militari (costruzione e lancio di un satellite),

2. progetti di cambiamento strategico (fusioni di aziende),
3. progetti di telecomunicazione (messa in opera di una rete satellitare),
4. organizzazione di grandi eventi (olimpiadi, campionati mondiali, esposizioni),
5. progetti impiantistici (costruzione di dighe e di autostrade)
6. progetti informatici (sviluppo di sistemi informativi e di applicazioni),
7. progetti di aiuto allo sviluppo (progetti ONU per i paesi del Terzo Mondo),
8. progetti culturali e multimediali (produzioni teatrali e di film, mostre),
9. progetti di sviluppo di nuovi prodotti (nuovi modelli di auto),
10. progetti di ricerca e sviluppo (nuovi medicinali e nuove cure).

### 1.5.1 Gestire il progetto (Project Management)

Per assicurare che tutto il lavoro necessario a realizzare quanto richiesto in un progetto sia effettivamente fatto nei tempi previsti, rispettando il budget dei costi e con la qualità attesa è necessario un controllo ed un coordinamento continuo ed accurato di tutti i partecipanti. Il Project Manager assume questi compiti applicando conoscenze, metodologie, mezzi e tecniche sviluppate in questi ultimi decenni per raggiungere una serie di obiettivi fondamentali:

- il raggiungimento degli obiettivi specifici di progetto;
- il rispetto dei tempi;
- il controllo ed il contenimento dei costi;
- il rispetto del livello di qualità fissato;
- l'assegnazione dei compiti e delle responsabilità;
- l'utilizzo ottimale delle risorse;
- l'individuazione preventiva e la mitigazione dei rischi;
- l'evidenziazione delle criticità e dei cambiamenti;
- l'assunzione tempestiva di misure correttive;
- il controllo dell'avanzamento;
- la comunicazione delle informazioni agli attori interessati (stakeholder).

Tali obiettivi necessitano di adeguate azioni di coordinamento come:

- organizzare e sistematizzare le attività attraverso la definizione di regole comuni;
- identificare gli attori (diretti ed indiretti) e definire ruoli e responsabilità ;
- fornire agli attori del sistema tecniche e strumenti a supporto del pieno raggiungimento degli obiettivi specifici ed in conformità agli standard definiti;
- facilitare la comunicazione grazie anche all'utilizzo di un linguaggio comune;
- attuare meccanismi che consentano di capitalizzare le esperienze ed il riuso delle attività svolte;
- favorire la crescita professionale attraverso azioni di formazione e tutoring.

Un progetto viene organizzato in fasi successive nel tempo. Queste fasi, a loro volta trattate come veri e propri progetti autonomi, sono caratterizzate dalla produzione di uno o più risultati tangibili (deliverables) la cui verifica ed

approvazione permette di accedere alla fase successiva.

L'insieme delle fasi attraverso cui si sviluppa un progetto (cioè nasce e si evolve fino al suo compimento) viene chiamato ciclo di vita. Generalmente è composto dalle seguenti fasi:

- Apertura
- Pianificazione
- Esecuzione
- Monitoraggio e Controllo
- Chiusura

per cui il ciclo di vita si presenta come in Figura 1.6.

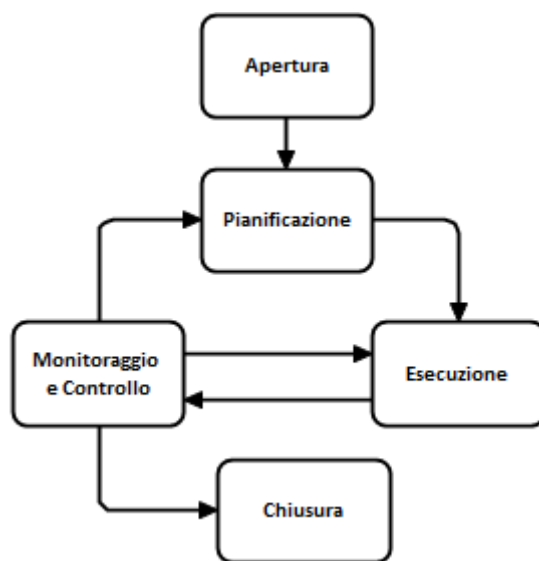


Figura 1.6: Ciclo di vita dei Progetti

Le fasi del progetto sono costituite da attività la cui specifica fa parte della metodologia adottata per la conduzione del progetto stesso; in questo testo faremo solo un breve cenno al contenuto delle varie fasi senza far riferimento a delle metodologie specifiche.

La fase di **Apertura** comprende l'inizializzazione del progetto (in cui al progetto viene assegnato un nome, un Project Manager e una posizione nel quadro complessivo dei progetti dell'azienda) e l'identificazione dell'ambiente in cui verrà svolto, degli attori e dei ruoli interessati (stakeholder) e dei vincoli a cui sarà soggetto (in termini di obiettivi, tempi, costi, risorse, budget, standard e così via).

La fase di **Pianificazione** del progetto a partire dagli obiettivi tracciati nella fase precedente si propone di specificare, con un Piano Esecutivo, la soluzione da realizzare e comprende:

- la descrizione di cosa dovrà essere fatto nel corso del progetto in termini di attività da svolgere finalizzate a produrre specifici deliverables,
- la stima delle risorse necessarie e dei tempi (producendo la schedulazione delle attività espressa sovente in forma grafica mediante un diagramma di Gantt),
- l'identificazione delle modalità di gestione dei rischi e della qualità,

- la costituzione del team di progetto (il Project Management Office),
- l'acquisizione delle risorse necessarie.

Nella fase di **Esecuzione** la soluzione viene sviluppata ed ultimata secondo quanto definito nel piano esecutivo. Le attività che la compongono comprendono l'azione del Project Manager nel guidare e coordinare il team di progetto e nel gestire la comunicazione puntuale a tutti i partecipanti al progetto dei progressi ottenuti.

Nella fase di **Monitoraggio** e **Controllo** si monitorizza e periodicamente si revisiona il programma in base agli avanzamenti riscontrati. Sono comprese nella fase azioni di controllo di valori non congruenti, la messa a punto di operazioni correttive ed eventuali modifiche al piano approvato, il controllo costante della qualità e la risoluzione di problemi possibilmente prima che si manifestino.

La fase di **Chiusura** sancisce il completamento dei lavori e la possibilità da parte degli utenti finali di iniziare ad utilizzare la soluzione adottata. In questa fase sono comprese le azioni di rilascio e accettazione formale del prodotto del progetto e la definizione degli aspetti amministrativi sia nei confronti del cliente che dei committenti del progetto.

### 1.5.2 Il rapporto fra i progetti e i processi

Il fatto che sia i processi che i progetti siano costituiti da attività, abbiano degli obiettivi, allochino delle risorse in base alle competenze, abbiano un inizio e una fine fa capire perché spesso nascono delle confusioni fra questi due concetti, dovute anche alla presenza frequente nelle aziende di entrambi i modi di operare. La Tabella 1.6 mette a confronto le differenze usando come esempio un'industria automobilistica.

Progetti	Processi
<b>unico:</b> un progetto è unico e non si ripeterà più (progetto della nuova auto BerlinaABC)	<b>ripetitivo:</b> il processo si ripete nel tempo (ha più istanze anche contemporanee), su lunghi periodi temporali (assemblaggio nel corso degli anni di milioni di auto del modello XYZ)
<b>ha una durata limitata:</b> il progetto della BerlinaABC ha la durata di 18 mesi dal 1/1/2015 al 30/6/2016	<b>è continuo:</b> la produzione avviene lungo una catena di assemblaggio che "sforna" continuamente nuovi esemplari del modello
<b>è schedato:</b> il progetto ha una data precisa di apertura e chiusura	<b>è guidato dagli eventi:</b> l'azienda produce in base all'arrivo degli ordini dai clienti
<b>ha un'apertura e una chiusura:</b> sono <u>fasi</u> importanti e relativamente complesse nella vita del progetto	<b>ha un inizio e una fine:</b> sono <u>eventi</u> che corrispondono semplicemente al termine della produzione di un'auto e all'inizio di lavorazione della successiva

Tabella 1.6: Confronto Progetti - Processi

Le differenze fra i due concetti non implicano, in ogni caso, che non possano crearsi delle sinergie significative. Consideriamo ad esempio il processo di gestione dei reclami nella casa automobilistica; nella maggioranza dei casi il processo si conclude in tempi medi "standard" con accordo delle parti, ma in alcuni casi i tempi si dilatano perché il disaccordo impone la via giudiziaria. In questi casi l'azienda può decidere di trattare la parte restante del processo come un progetto da chiudere entro tempi definiti.



Il concetto di progetto può quindi essere utilizzato per parti di processo (in casi specifici, ricordiamo che il progetto è unico) o sotto-processi ma può anche essere applicato su singole attività in cui la necessità di rispettare delle scadenze strette può suggerire di trattare l'attività come un progetto.

Viceversa il concetto di processo può essere utilmente applicato per modellare le fasi del progetto (che come si è detto comprendono in genere diverse attività da svolgere secondo certe logiche) e il ciclo di vita del progetto stesso; un motore di workflow può quindi essere un supporto importante per il Project Manager nella sua attività di coordinamento e controllo fornendogli anche tutta una serie di indicatori per valutare lo stato di avanzamento del progetto.

Lo sviluppo di modelli integrati per la progettazione ibrida progetto-processo sarebbero un importante argomento di ricerca applicata ma esula dai limiti del nostro testo.




## 1.6 L'Architettura d'Azienda

L'**Architettura d'Azienda** è un concetto che sta assumendo una rilevanza sempre maggiore a causa della crescente complessità delle aziende. Tale complessità deriva spesso da acquisizioni, fusioni e modifiche profonde delle aziende per cui diventa sempre più importante mettere a disposizione dell'organizzazione aziendale una rappresentazione, la più completa possibile, di tutto quello che esiste in azienda in termini di strategie, strutture, regole, risorse, processi, applicazioni, dati, attrezzature tecnologiche, infrastrutture informatiche e di comunicazione (ICT – Information and Communication Technologies), e così via.

L'architettura diventa quindi per le aziende una banca dati ad alto livello in grado di rispondere a domande riguardanti, ad esempio, la ridondanza di strutture a seguito di fusioni, l'esistenza di diverse applicazioni di gestione finanziaria (a volte incompatibili fra di loro), sistemi di contabilità, produzioni che parzialmente si sovrappongono, processi di gestione delle risorse umane da unificare, e così via. L'architettura è quindi uno strumento decisionale importante, utile ad esempio per evitare di creare delle ridondanze, basti pensare, ad esempio, nell'ambito delle Pubbliche Amministrazioni a dipartimenti diversi che richiedono fondi per creare sistemi simili.

Un ulteriore ambito in cui l'architettura riveste un ruolo cruciale riguarda la comunicazione delle direttive e delle informazioni all'interno dell'azienda e verso il mondo esterno, svolgendo in tal modo una funzione di base per il lavoro cooperativo.

Una componente fondamentale dell'architettura è ovviamente costituita dai processi e oltre ai processi stessi deve contenere la specifica di tutte informazioni necessarie a coordinarli. Osserviamo che esistono diverse forme di *coordinamento*, la Figura 1.7 illustra quelle più comuni, specificate mediante un meccanismo di invio e ricezione di **messaggi** basato su **eventi** (iniziali, intermedi o finali). Il significato dei simboli utilizzati (che saranno ripresi in dettaglio nel seguito) è descritto in Tabella 1.7.

Simbolo	Significato
	Attività
	Processo
	Evento iniziale
	Evento iniziale con ricezione di un messaggio






	Evento finale
	Evento finale con invio di un messaggio
	Evento intermedio con ricezione di un messaggio
	Evento intermedio con invio di un messaggio
	Messaggio

Tabella 1.7: Simboli grafici per i processi

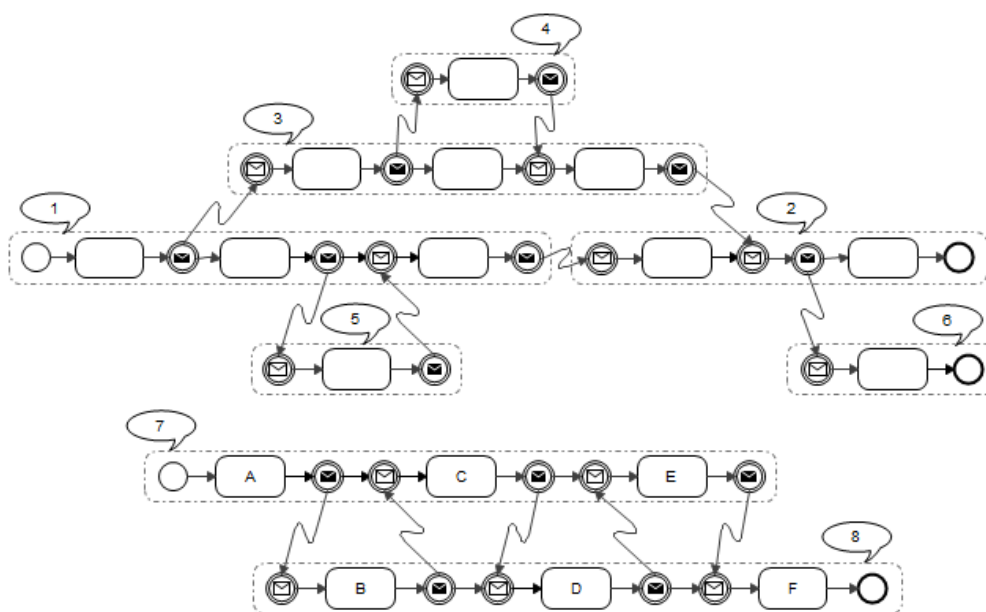


Figura 1.7: Coordinamento fra processi

La Figura 1.7 mostra i seguenti tipi di coordinamento fra gli otto processi:

- il processo (1) termina e attiva il processo (2) (*sequenzializzazione di processi*);
- il processo (3) richiama il processo (4) che si attiva, opera in modo indipendente e poi segnala al processo chiamante che è terminato (*chiamata asincrona*);
- il processo (1) richiama il processo (5) e aspetta che termini per poter andare avanti (*chiamata sincrona*);
- il processo (2) attiva il processo (6) che prosegue in modo indipendente;
- il processo (3) è attivato dal processo (1) che si attiva, opera in modo indipendente e poi segnala al processo chiamante che è terminato (*chiamata asincrona*);
- il processo (7) e il processo (8) si richiamano alternativamente (*processi cooperanti*); nel caso in figura l'ordine secondo il quale saranno eseguite le attività (o sotto-processi) è: A – B – C – D – E – F .

Ovviamente i casi mostrati in Figura 1.7 non esauriscono tutte le possibilità di coordinamento.

L'importanza delle architetture d'azienda è dimostrata dallo sviluppo negli ultimi anni di un insieme di **framework**

(schemi di riferimento) che hanno proposto delle strutture architettoniche in diversi contesti, come, ad esempio, TOGAF, SCOR e eTOM.

Occorre tener presente che l'architettura è distinta dai **modelli** (come quelli che esamineremo nel seguito di questo testo). Un'architettura è ad un livello più generico dei modelli in quanto opera a livello dell'intera azienda, mentre i modelli si riferiscono a **viste** particolari della realtà aziendale (ad esempio, la vista funzionale o quella strategica); i modelli sono molto più dettagliati, ad un livello tale per cui diventa possibile evidenziare gli aspetti critici delle realtà che rappresentano, quali, ad esempio, le risorse consumate, il prodotto/servizio erogato, le regole e i vincoli di trasformazione, i tempi e i costi, e così via. Possiamo quindi concludere che l'architettura comprende i vari modelli e le relazioni fra loro esistenti.

L'argomento architettura d'azienda non sarà trattato ulteriormente per motivi di spazio, nel testo ci occuperemo sostanzialmente di modelli per la specifica dei processi e dei sistemi informativi aziendali e, accanto a questi, degli altri modelli rilevanti per ottenere una descrizione accurata dell'azienda.

### **Bibliografia**

- M. Porter, Il vantaggio competitivo, Edizioni di Comunità, 1987.
- P. Kotler, Marketing Management, ISEDI, 1991.
- P.F. Camussone, Informatica organizzazione e strategie, McGraw-Hill, 2000
- M. Cantamessa, E. Cobos, C. Rafele, Il Project Management – Un approccio sistemico alla gestione dei progetti, ISEDI, 2007.
- S. Setti, Project & Process Management – La gestione integrata di progetti e processi: una sfida organizzativa, Franco Angeli, 2008.
- G. Bracchi, C. Francalanci, G. Motta, Sistemi Informativi d'Impresa, McGraw-Hill, 2010.

## 2. I modelli

Esistono nella pratica numerosi modelli per rappresentare diverse viste della realtà aziendale. Tra questi si annoverano sia modelli tradizionali, mutuati dall'analisi organizzativa e dall'analisi dei sistemi informativi, quali le carte di processo, i diagrammi di flusso (data flow diagram, DFD) e funzionali (IDEFO), sia modelli più innovativi quali ad esempio l'Action Workflow Diagram, in cui si rappresenta un processo attraverso l'interrelazione dei vari cicli cliente-fornitore da cui è composto.

Quello che è importante puntualizzare è che la scelta del modello dovrà essere guidata principalmente dalla individuazione degli aspetti che si intende evidenziare in funzione degli obiettivi dell'intervento, ricercando di conseguenza il modello più adatto a metterli in luce.

Un ragionamento analogo dovrà valere per l'eventuale utilizzo di strumenti automatizzati per rappresentare, modellare e archiviare i processi e per effettuare simulazioni o analisi varie, strumenti per i quali è presente una variegata offerta di mercato. In particolare ci interesseremo della specifica di modelli che siano utilizzabili per dare una descrizione formale dei processi; la specifica formale permette la generazione di un programma eseguibile tramite strumenti di gestione del flusso delle attività (motori di workflow).

Per comprendere a pieno la potenzialità fornita dal processo di modellazione analizziamo anzitutto il concetto di **modello**. Se vogliamo costruire delle analogie con le varie discipline dell'ingegneria è da notare che, indipendentemente dal settore nel quale si intende operare (realizzare una casa, un ponte, un meccanismo, un nuovo reparto dell'azienda,...) e dalla dimensione di quanto si intende realizzare, ogni qualvolta si voglia realizzare un'opera si procede programmando, come prima attività, uno studio di fattibilità basato su un modello.

Nel nostro caso siamo particolarmente interessati a modelli che specifichino la struttura e il comportamento di un'azienda e che (possibilmente) permettano di valutare sul modello l'effetto provocato da modifiche che potrebbero essere applicate all'azienda stessa.

Questi modelli, in tempi brevi e con costi contenuti, dovrebbero mettere progettisti e clienti in grado di poter:

- definire il modello con il giusto livello di astrazione in modo da potersi concentrare solo sugli aspetti salienti per l'analisi, isolando quanto è di interesse da tutti gli altri aspetti presenti nel sistema reale;
- valutare il comportamento del sistema in particolari situazioni di stress o di funzionamento determinando limiti e criticità che caratterizzano il sistema attuale;
- attuare delle simulazioni sul modello atte a comprendere cosa potrebbe accadere se si applicano delle variazioni al sistema.

Il ricorso all'uso dei modelli trova tanto più facilmente giustificazione quanto è più complesso il sistema che si intende studiare e/o realizzare. D'altronde più il sistema è complesso tanto più è necessario realizzare dei modelli che, con diversi livelli di astrazione, descrivano gli elementi salienti del sistema reale. Le semplificazioni comunque non devono invalidare la capacità del modello di rappresentare ciò che interessa nel sistema.

In generale si può affermare che si introducono dei modelli aziendali per raggiungere i risultati indicati nella Tabella 2.1.

1	Aumentare la comprensione del sistema oggetto di analisi e del quale si realizza il modello.
2	Predisporre uno strumento che, se pur non è completamente equivalente al sistema reale, consente, con un adeguato margine d'errore, di poter effettuare studi e simula-

	zioni che non sarebbe possibile realizzare sul sistema stesso.
3	Comunicare a tutti gli interlocutori quanto è importante conoscere sul sistema oggetto di studio e sulle modifiche ad esso apportate.
4	Simulare il modello confrontando i risultati ottenuti dalla simulazione con degli indicatori di efficienza rilevati nella realtà: questo permette di validare il modello e fornire la base per le future ristrutturazioni.
5	Descrivere in modo preciso i requisiti per lo sviluppo del software utile per il sistema.
6	Costituire una specifica del comportamento del sistema utilizzabile da un motore di workflow per la gestione del flusso delle attività descritto nel modello.

Tabella 2.1 : Proprietà dei modelli

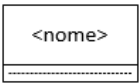
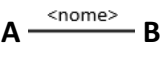
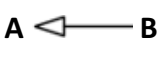
Per poter facilitare la gestione e l'evoluzione del modello dovrebbero essere assicurate alcune proprietà di base:

- **Accuratezza:** il modello deve descrivere il sistema che rappresenta correttamente, completamente e senza ambiguità.
- **Consistenza:** le diverse viste costituenti il modello devono completarsi vicendevolmente per formare un insieme coerente di conoscenze sul sistema.
- **Semplicità:** il modello deve poter essere compreso, senza troppi problemi, da persone estranee al processo di modellazione.
- **Manutenibilità:** il modello deve essere gestibile e le variazioni che si vogliono apportare al modello stesso devono essere facilmente applicabili.

Vediamo ora come procedere per specificare in modo formale i modelli che svilupperemo. L'approccio che seguiremo è basato fondamentalmente su tre livelli di specifica:

1. il **Meta-modello**
2. il **Modello**
3. le **Istanze**

A livello di Meta-modello devono essere specificati i **concetti** da utilizzare per descrivere la realtà aziendale (come vedremo ce ne saranno diversi!). Il linguaggio usato a questo livello è il **diagramma delle classi** del linguaggio **UML** (Unified Modeling Language). Questo linguaggio è attualmente lo standard per la specifica dei sistemi software e dei sistemi informativi ed è stato emesso dall'autorità per la standardizzazione **OMG** (Object Management Group). La Tabella 2.2 illustra gli elementi grafici fondamentali con cui costruire il diagramma, cioè un grafo diretto in cui le **classi** costituiscono i **nodi** e gli archi rappresentano le **relazioni** fra le classi stesse.

Simbolo	Significato
	<b>classe:</b> descrive un insieme di oggetti aventi le stesse <b>proprietà</b> e lo stesso <b>comportamento</b> ;
	relazione di <b>associazione:</b> specifica un insieme di <b>legami</b> fra gli oggetti di due classi A e B (es: <b>Studiante segue Corso</b> );
	relazione di <b>generalizzazione:</b> specifica un rapporto classe $\subseteq$ sotto-classe (es: <b>PERSONA <math>\leftarrow</math> OPERAIO</b> ) per cui gli oggetti della sotto-classe ereditano tutte le proprietà dei corrispondenti oggetti della classe (e possono aggiungerne delle altre);

<p>A ◊ — B</p>	<p>relazione di <b>aggregazione</b> specifica un rapporto <b>tutto</b> ◊ — <b>parti</b> (es: CAMPIONATO ◊ — SQUADRE) per cui oggetti <b>parti sono aggregati in un oggetto tutto</b> ;</p>
<p>A ◆ — B</p>	<p>relazione di <b>composizione</b>: specifica un rapporto: <b>oggetto-composto</b> ◆ — <b>oggetti-componenti</b> (es: LIBRO ◆ — CAPITOLI) per cui un <b>oggetto-composto è composto da oggetti-componenti</b>.</p>

Tabella 2.2: Simboli grafici per i diagrammi delle classi (UML)

Per comprendere meglio questi concetti consideriamo il semplice esempio di Figura 2.1.

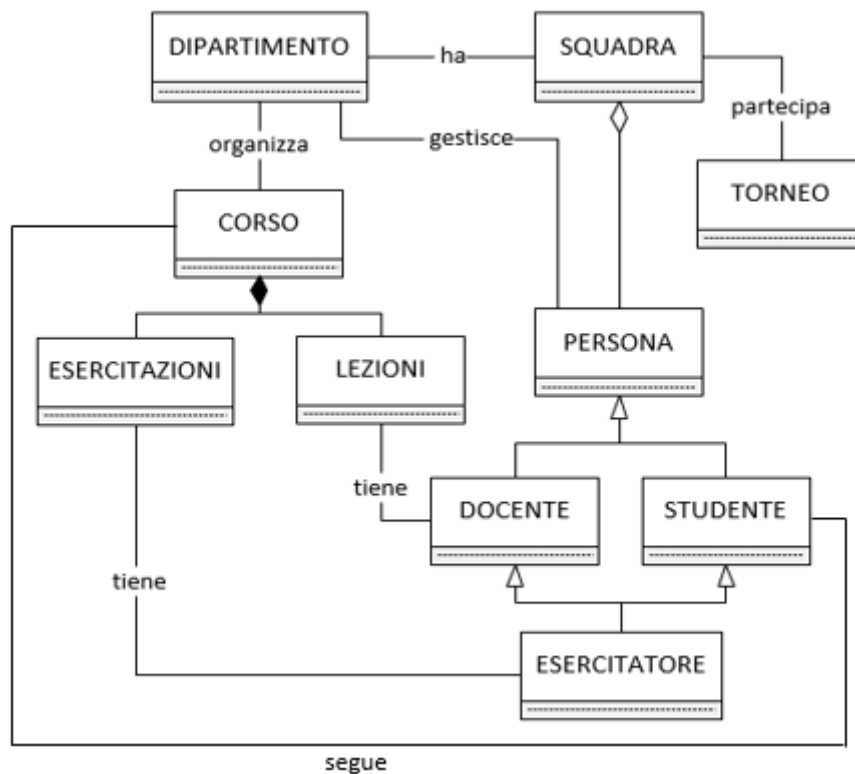


Figura 2.1: Esempio di diagramma delle classi (UML)

Nell'esempio è descritto un insieme di dipartimenti (classe **DIPARTIMENTO**), ogni dipartimento *organizza* diversi corsi (classe **CORSO**) e ogni corso è *composto\_da* (relazione di **composizione**) lezioni ed esercitazioni (classi **LEZIONI** e **ESERCITAZIONI**). Il dipartimento *gestisce* (relazione di **associazione**) delle persone (classe **PERSONA**). Un docente o uno studente è *una* persona (relazione di **generalizzazione**) ed un esercitatore *può\_essere\_sia\_un* docente *che\_uno* studente: **DOCENTE** e **STUDENTE** sono sotto- classi di **PERSONA** ed **ESERCITATORE** è *una* sotto-classe sia di **DOCENTE** che di **STUDENTE**. Uno studente *segue* uno o più corsi. Un docente *tiene* le lezioni mentre un esercitatore *tiene* le esercitazioni del corso. Un dipartimento *ha* delle squadre (classe **SQUADRA**) che *partecipano* a tornei universitari (classe **TORNEO**); una squadra *aggrega* (relazione di **aggregazione**) dei giocatori che sono persone (docenti o studenti). Osserviamo che la differenza fra le relazioni di composizione e di aggregazione riguarda il fatto che in una aggregazione se una squadra viene cancellata i suoi componenti non vengono cancellati, mentre se un dipartimento scompare

anche i i suoi corsi devono essere cancellati.

Nei nostri meta-modelli, i concetti saranno rappresentati mediante delle classi, e le relazioni (associazioni, generalizzazioni, aggregazioni, composizioni) saranno usate per specificare i rapporti fra i concetti stessi.

Poiché nei casi reali pensare ad un unico modello dell'azienda condurrebbe a rappresentazioni assolutamente incomprensibili prenderemo in considerazione cinque modelli distinti ma coordinati: **strategico**, **organizzativo**, **funzionale**, di **processo**, e di **implementazione**, per cui avremo cinque meta-modelli. Questi meta-modelli corrispondono a cinque punti di vista fondamentali sotto cui è possibile osservare l'azienda e verranno discussi estensivamente nel seguito.

A livello di **modello** si utilizzano i concetti descritti nei meta-modelli relativi per descrivere, mediante opportuni linguaggi di specifica, come una certa azienda è strutturata o si comporta in una determinata situazione. Ad esempio come il Pronto Soccorso di un certo ospedale tratta i pazienti che si presentano per essere curati. Quindi un modello è specifico per una certa azienda, benché aziende simili presentino dei modelli vicini tra loro.

A livello di **istanza** si prendono infine in considerazione i comportamenti effettivi dell'azienda messi in atto quando si verificano i casi reali (ad esempio cosa si fa nel Pronto Soccorso quando si presenta il paziente Rossi). Osserviamo che a livello di istanze possono sussistere contemporaneamente diverse istanze (nel Pronto Soccorso ci possono essere decine di pazienti) mentre abbiamo solo un modello.

Per comprendere meglio la definizione dei tre livelli faremo riferimento al processo con cui è trattato un paziente ogni qualvolta si reca nel Pronto Soccorso di un ospedale.

### Descrizione Caso: Pronto Soccorso

I pazienti arrivano al Pronto Soccorso (PS) dell'ospedale e vengono registrati. Un infermiere esperto effettua quindi il cosiddetto "triage" in cui è valutato il quadro clinico, e la gravità della situazione è espressa assegnando al paziente un codice colore:

Bianco (non urgente - paga il ticket),	Verde (urgenza differibile),
Giallo (urgenza),	Rosso (emergenza).

Inoltre, se è il caso, si fanno i prelievi al paziente e si effettuano gli esami necessari, i campioni sono inviati in laboratorio che, dopo un certo tempo, fornisce i risultati. La visita dal medico può concludersi con:

- 1) dimissioni,
- 2) ricovero (con trasferimento in un reparto di degenza dell'ospedale),
- 3) decesso (con trasferimento nell'obitorio).

Osserviamo che alcuni pazienti possono abbandonare il PS dopo il triage (ad esempio non intendono pagare il ticket).

Evidentemente la descrizione riguarda la vista di processo dell'azienda e il modello relativo è costruito seguendo la metodologia che illustreremo nel seguito. Il linguaggio con cui il modello è descritto è chiamato BPMN (Business Process Model and Notation, linguaggio per la descrizione dei modelli di processi aziendali) ed è attualmente, nella sua versione 2.0 rilasciata nel marzo del 2011 dal gruppo di standardizzazione OMG (Object Management Group), lo standard *de facto* per la descrizione dei processi aziendali. BPMN è un linguaggio grafico e il diagramma del processo PS è illustrato in Figura 2.2.

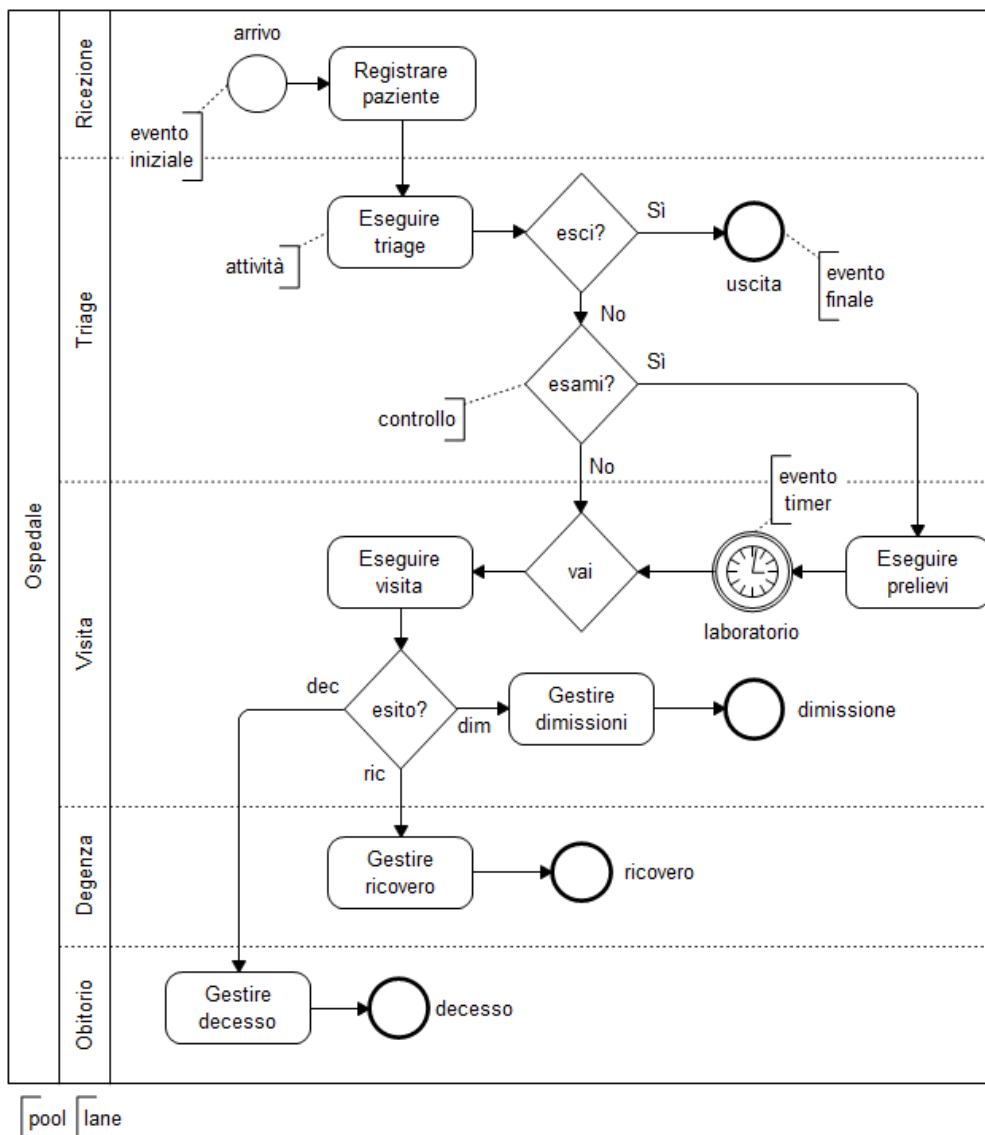


Figura 2.2: Modello del processo Gestire il Pronto Soccorso

I simboli che compaiono nel diagramma hanno il significato specificato nella Tabella 2.3.

Simbolo	Significato
	<b>rettangoli arrotondati</b> : rappresentano le attività eseguite durante l'erogazione del servizio offerto dal PS;
	<b>rombi</b> : rappresentano dei controlli su come proseguire nel flusso delle attività;
	<b>cerchi</b> : rappresentano gli <b>eventi</b> rilevanti nell'erogazione del servizio; gli eventi possono essere di tipo <b>iniziale</b> (cerchio sottile), di tipo





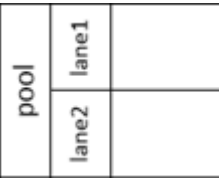
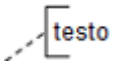
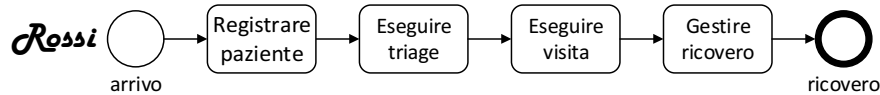
	<b>finale</b> (cerchio spesso), o di tipo <b>timer</b> (ritardo);
	<b>archi</b> : determinano il flusso di controllo, definendo delle sequenze di attività;
	corsie o <b>pool</b> orizzontali e sotto-corsie o <b>lane</b> : i pool rappresentano entità organizzative indipendenti (dette in genere partecipanti) che collaborano nel processo; le lane rappresentano componenti organizzative del partecipante o ruoli rivestiti da risorse del partecipante stesso; se un'attività è inserita in un pool o una lane vuol dire che utilizza risorse aziendali assegnate a quella componente;
	<b>testi</b> : sono commenti associabili a un elementi qualsiasi del modello per precisarne il significato.

Tabella 2.3: Simboli grafici per i diagrammi di processo (BPMN)

Osserviamo che questa descrizione è a livello di modello in quanto rappresenta il modo di comportamento generico dell'azienda PS, valido qualunque sia il paziente che si presenta (in condizioni più o meno gravi), mentre a livello di istanza il percorso effettivo del paziente *Rossi* che ha una gamba fratturata potrebbe essere:



A livello di meta-modello avremo invece la descrizione dei concetti (e delle loro relazioni) da usare per costruire il modello, per cui potremmo avere il meta-modello di processo (semplificato) riportato in Figura 2.3:

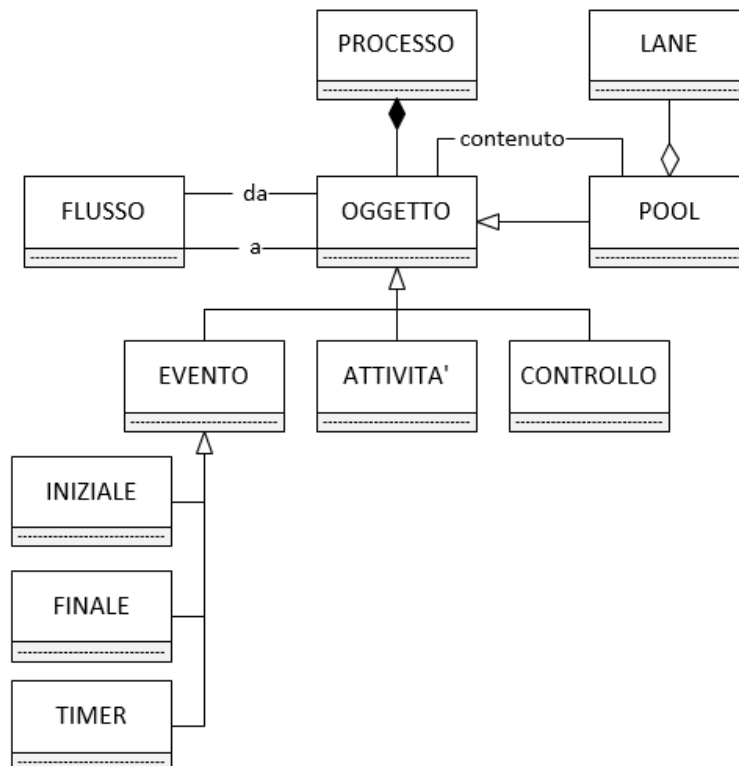


Figura 2.2: Meta-modello di processo

Un **processo** è *costituito da oggetti*; gli oggetti sono i nodi di un grafo diretto e possono essere **attività**, **controlli** o **eventi** (di tipo **iniziale**, **finale** o **timer**), e i **flussi** (archi) collegano un nodo di origine (*da*) a un nodo finale (*a*). Anche i **pool** (che possono *aggregare* delle *lane*) sono oggetti e ogni oggetto deve essere *contenuto* in un pool.

La struttura a livelli dei modelli può essere riassunta come mostrato in Figura 2.4:

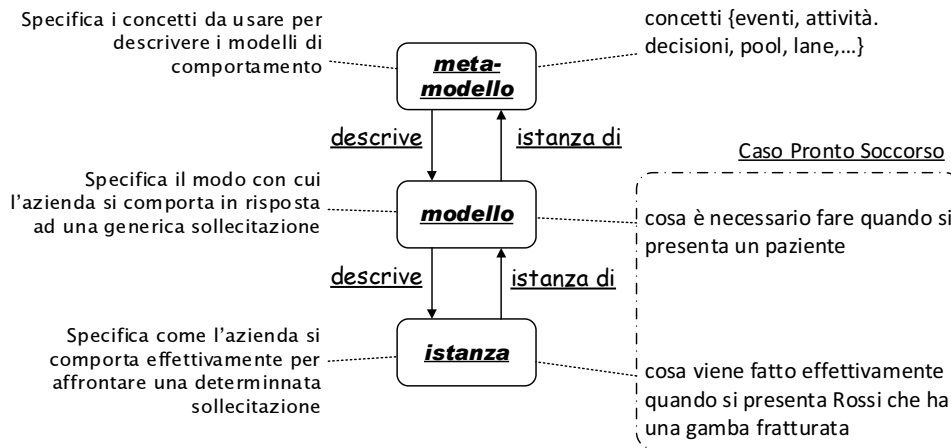


Figura 2.4: La struttura dei modelli

La struttura a tre livelli deriva da quella a quattro livelli utilizzata, ad esempio, per la definizione di UML. Al di sopra del meta-modello è possibile infatti definire un meta-meta-modello contenente le specifiche degli oggetti utilizzati a livello di meta-modello (ad esempio, meta-classi, meta-associazioni, ecc...), ma ai fini di questo testo i tre livelli introdotti sono certamente sufficienti.

## 2.1 Il Modello Strategico

La conoscenza (e la rappresentazione) della **strategia aziendale** costituisce ovviamente un punto di partenza per ogni iniziativa di miglioramento, non ha senso infatti parlare di miglioramento se non in relazione alla visione strategica dell'azienda, definita dalla sua direzione.

Alla base della costruzione di modelli per descrivere, in modo per quanto possibile completo e non ambiguo, la **strategia aziendale** ci sono oltre ai motivi generali indicati in Tabella 2.1 anche i seguenti:

- confrontare fra loro, per quanto possibile oggettivamente, diverse strategie aziendali al fine di compiere una scelta razionale,
- porre in atto delle azioni per allineare le funzionalità dell'azienda alla strategia dell'azienda stessa.

Poiché la strategia aziendale riflette le **motivazioni** di base dell'azienda prenderemo in considerazione il modello standard per specificare questi aspetti, il Business Motivation Model (BMM) emesso anch'esso dall'OMG.

Il BMM fornisce, a livello di meta-modello, un quadro molto generale dei concetti da utilizzare per descrivere le motivazioni (e di conseguenza le strategie) di un'azienda; tali motivazioni possono servire nella fase iniziale per stilare

il cosiddetto **business plan**, cioè la pianificazione iniziale delle attività aziendali, e in seguito per gestire le iniziative di miglioramento per far sì che i processi ristrutturati siano allineati alla strategia.

Nel BMM sono definiti quattro concetti fondamentali:

1. **FINI** : un fine descrive “cosa un’azienda vuol essere”,
2. **MEZZI** : un mezzo specifica “come l’azienda intende raggiungere un suo fine”,
3. **VALUTAZIONI** : cosa o chi valuta i mezzi usati rispetto ai fini che si intendono raggiungere,
4. **FATTORI** : un qualsiasi fatto che possa *influenzare* l’azienda nell’uso dei suoi mezzi e/o nel raggiungimento dei suoi fini.

La Figura 2.5 mostra il meta-modello BMM, semplificato rispetto alla definizione dello standard in quanto non sono state esplicitate tutte le possibili associazioni fra i concetti.

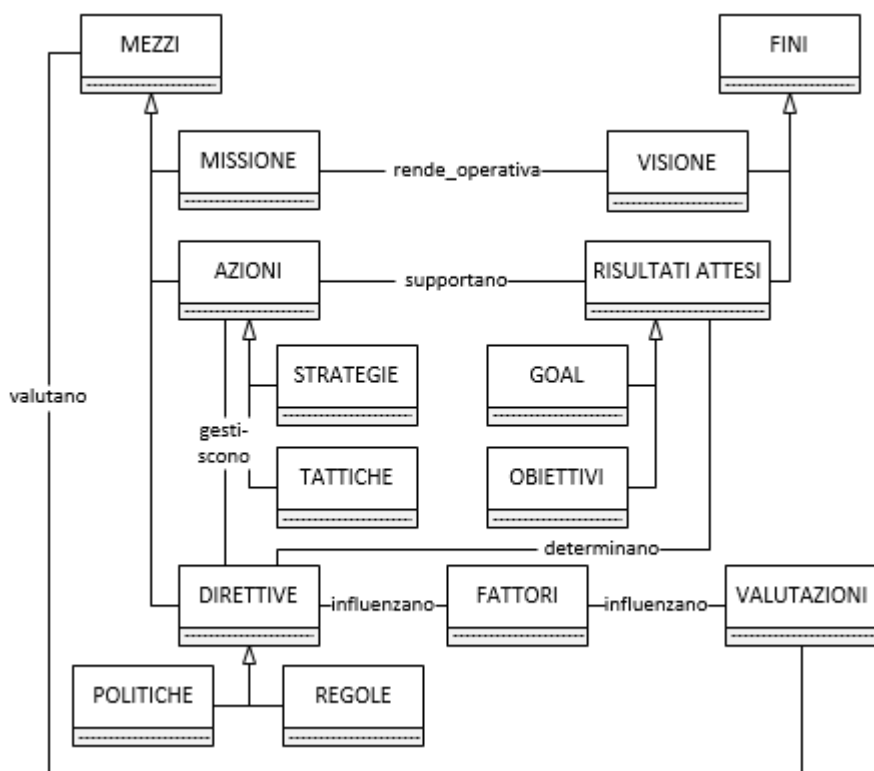


Figura 2.5: Il meta-modello BMM

Un **fine** comprende la **visione** dell’organizzazione e dei **risultati attesi**. La visione descrive il futuro stato dell’azienda e la **missione** dell’azienda ha lo scopo di *rendere\_operativa* la sua visione. Le **azioni**, espresse mediante **strategie** e **tattiche**, *supportano* i risultati attesi, che sono espressi come dei **goal** da raggiungere che vengono poi quantificati in **obiettivi**. I **mezzi** comprendono, oltre alla missione e alle azioni, le **direttive** (**politiche** e **regole**) mediante le quali si *gestiscono* le azioni e si *determinano* i risultati. I **fattori** sono dei fatti che possono *influenzare* le direttive e le **valutazioni** attraverso cui si *valutano* i mezzi messi in atto dall’azienda per raggiungere i propri fini.

Lo standard BMM non definisce una rappresentazione grafica particolare per specificare i modelli, per cui in gene-

re, quando bisogna specificare una istanza del modello relativa ad una certa azienda si utilizzano delle tabelle descrittive. Per comprendere meglio come si opera consideriamo un semplice caso di studio (ipotetico), una Azienda di Noleggio di Auto (ANA).

---

Come **Fine** principale l'azienda intende specificare la sua **Visione**:

V	<i>Diventare l'azienda di riferimento per i clienti business (in particolare le industrie) nelle regioni in cui l'azienda è attiva.</i>
---	---

Per raggiungere questa Visione l'azienda si è posta i seguenti **Goal**:

G1	<i>Posizionarsi a livello delle maggiori aziende nazionali di noleggio auto.</i>
G2	<i>Diventare il fornitore principale delle aziende leader nel settore industriale.</i>
G3	<i>Fornire dei veicoli sempre in condizioni ottimali.</i>
G4	<i>Avere sempre a disposizione dei veicoli quando e dove i clienti li richiedono.</i>

I Goal sono specificati (quantitativamente) in termini di **Obiettivi**:

O1	<i>Per la fine del 2014 essere classificati fra le prime sei maggiori compagnie di noleggio.</i>
O2	<i>Per la fine del 2014 essere classificati fra le prime tre maggiori compagnie di noleggio nel settore industriale.</i>
O3	<i>Durante il primo semestre del 2014 ridurre a meno del 3% la necessità di rimpiazzare auto presso i clienti (esclusi gli incidenti).</i>
O4	<i>Raggiungere il 90% di risposte positive nelle indagini di soddisfazione dei clienti per il primo semestre del 2015.</i>

Accanto ai Fini dovremmo tener conto dei Mezzi, cioè capacità, tecniche, strumenti o metodi da attivare per raggiungere i Fini; in particolare iniziamo a considerare la **Missione** dell'azienda:

M	<i>Fornire un eccellente servizio di noleggio in tutte le regioni italiane sia per i clienti singoli che per le aziende.</i>
---	--

Operativamente si perseguono i Fini attraverso delle **Azioni** che comprendono **Strategie** (generalmente si riferi-

scono ai Goal dell'azienda) e **Tattiche** (generalmente sono a più breve termine e fanno riferimento agli Obiettivi); per esempio, alcune Tattiche potrebbero essere:

T1	<i>Incoraggiare l'estensione del noleggio auto (sconti, promozioni...).</i>
T2	<i>Per le piccole filiali dare in outsourcing la manutenzione dei veicoli.</i>
T3	<i>Egalizzare il Kilometraggio per i veicoli della stessa età e categoria.</i>
T4	<i>Seguire rigorosamente i calendari di manutenzione delle case costruttrici.</i>

Fra le Azioni rivestono importanza le **Direttive** aziendali che comprendono **Politiche** e **Regole**. Per esempio, alcune Politiche potrebbero essere:

P1	<i>Il pagamento del noleggio deve essere garantito in anticipo.</i>
P2	<i>I veicoli noleggiati non possono varcare la frontiera.</i>
P3	<i>Le regole di noleggio devono essere conformi alle leggi e ai regolamenti nazionali e a quelli della regione in cui si trova la filiale.</i>

Mentre delle Regole potrebbero essere:

R1	<i>Il cliente deve rilasciare gli estremi della carta di credito, e questa deve esser controllata presso la banca emittitrice (regola associata alla politica P1).</i>
R2	<i>I guidatori delle auto a noleggio devono aver compiuto i 21 anni e devono esibire una patente di guida valida.</i>
R3	<i>Fra le auto di un certo gruppo bisogna noleggiare quella con il minor kilometraggio.</i>
R4	<i>Non assegnare un'auto se il Kilometraggio finale previsto è superiore a quello stabilito per la manutenzione.</i>

Alle regole è possibile associare un livello di applicazione (*obbligatoria, superabile con autorizzazione, giustificabile, consigliata*); ad esempio, la R2 è evidentemente obbligatoria.

I **Fattori** sono eventi o situazioni che quando si verificano influenzano il comportamento dell'azienda, in particolare le Direttive e le Valutazioni dei Mezzi che l'azienda pone in atto. I Fattori possono essere **Interni** (FI) o **Esterni** (FE), ad esempio:

FE1	<i>Due concorrenti più piccoli si sono fusi e ora in alcune regioni sono divenuti più grandi della nostra azienda.</i>
-----	--

FE2	Le prime due compagnie si sono impegnate in promozioni molto redditizie.
FE3	Alcune compagnie aeree low-cost sono diventate dei concorrenti pericolosi soprattutto nei centri dotati di piccoli aeroporti.
FI1	E' necessario proporre dei programmi di fidelizzazione per la clientela esistente e per attrarre nuovi clienti.
FI2	E' importante ristrutturare l'azienda organizzandola intorno alle città principali, con grosse filiali nei relativi aeroporti, medie filiali nelle periferie e agenzie negli hotel più prestigiosi.

Figura 2.6: Modello strategico per l'azienda ANA

Nel modello strategico le **Valutazioni** riguardano una stima di come i **Fattori** (interni o esterni all'azienda) influenzano i **Fini** che essa si pone e/o i **Mezzi** che mette in essere per raggiungere quei Fini. Osserviamo che queste stime sono in genere alquanto soggettive, per cui persone diverse (o anche le stesse persone in tempi diversi) possono fornire stime diverse: è importante quindi tenere traccia di chi e quando ha effettuato le stime in questione.

Lo standard BMM non prescrive tecniche particolari per le valutazioni, è comunque suggerita una tecnica di analisi strategica chiamata SWOT che sovente è usata in azienda per comprendere i propri punti di **Forza** (**Strenght**) e di **Debolezza** (**Weakness**) e per esaminare le **Opportunità** (**Opportunities**) e i **Rischi** (**Threats**) che si possono incontrare; opportunità e rischi riguardano fattori esterni, punti di forza e debolezza si riferiscono a fattori interni.

Nel caso dell'Azienda di Noleggio di Auto potremmo avere, ad esempio:

Azienda di Noleggio di Auto	
punti di Forza	punti di Debolezza
<ul style="list-style-type: none"> <li>- l'azienda ha una forte struttura di filiali e agenzie distribuite sul territorio,</li> <li>- gli ambienti e l'accoglienza sono molto amichevoli e curati</li> </ul>	<ul style="list-style-type: none"> <li>- il sito Internet è piuttosto carente</li> <li>- il turnover delle risorse umane è troppo elevato causando carenze nella gestione dei clienti</li> </ul>
Opportunità	Rischi
<ul style="list-style-type: none"> <li>- indagini di mercato mostrano che esistono forti opportunità nella fascia alta della clientela (anche industriale)</li> </ul>	<ul style="list-style-type: none"> <li>- compagnie aeree low-cost esercitano una forte concorrenza sulle tratte medie</li> <li>- politiche di pagamento nei centri urbani scoraggiano i clienti</li> </ul>

Figura 2.7: Analisi SWOT per l'azienda ANA

Il BMM ha ovviamente delle relazioni con le **unità organizzative** e i **processi** dell'azienda; illustreremo brevemente queste relazioni nei paragrafi riguardanti il modello organizzativo e quello dei processi.

## 2.2 Il Modello Organizzativo

Il modello organizzativo descrive la struttura “statica” dell’azienda e le risorse che essa utilizza per il processo (o i processi) in esame.

Esaminiamo anzitutto il meta-modello organizzativo mostrato in Figura 2.8:

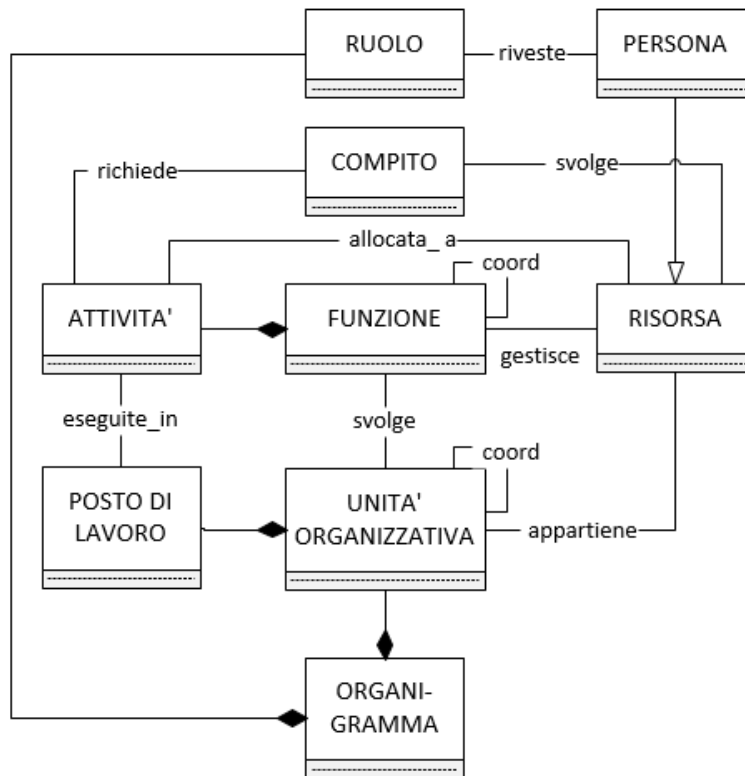


Figura 2.8: Meta-modello Organizzativo

Un **organigramma** aziendale è *composto* da **unità organizzative**. Una unità organizzativa è una componente strutturale dell’azienda (reparto, divisione, stabilimento, filiale,...), può *coordinare* altre unità ed è *composta* da **posti di lavoro** (ad esempio, l’ufficio protocollo è un posto di lavoro del reparto Amministrazione) in cui sono *eseguite* le **attività** aziendali.

Un organigramma è *composto* anche da **ruoli** (responsabile, esecutore,...) *rivestiti da persone* che sono **risorse** dell’azienda e *appartengono*, cioè sono gestite, da una unità organizzativa.

Ogni unità organizzativa *svolge* una **funzione** all’interno dell’azienda; una funzione può *coordinare* altre funzioni, è *composta* da attività *svolte* direttamente a livello della funzione e/o nei posti di lavoro che compongono l’unità, *gestisce* le risorse *appartenenti* alla unità a cui fa riferimento, e *alloca* (assegna) certe risorse alle attività basandosi sul fatto che una attività *richiede* lo svolgimento di **compiti** che possono *essere svolti* dalle risorse stesse.

A livello di Modello useremo diversi strumenti linguistici:

- **organigramma aziendale**: struttura generalmente ad albero in cui sono ordinate (in base a relazioni di responsa-

bilità) le **unità organizzative** e i **posti di lavoro** (nel seguito per brevità li chiameremo **unità** e **posti**) costituenti l'azienda. Osserviamo che la "radice" dell'albero è occupata dalla componente direzionale dell'azienda (amministratore delegato, direttore generale,...) mentre i posti di lavoro occupano le "foglie" dell'albero in quanto non devono coordinare componenti a livello più basso;

- **descrittori:** ad ogni nodo dell'organigramma è utile associare un **descrittore** contenente informazioni come:
  1. identificatore e nome dell'unità/posto,
  2. breve descrizione dell'unità/posto in cui specificare lo scopo che l'unità/posto ha all'interno dell'azienda,
  3. ruoli responsabili e capacità richieste: descrizione dei ruoli responsabili della unità/posto e delle caratteristiche delle persone che possono essere chiamate a rivestire quei ruoli.
- **mansionari:** per descrivere le risorse si utilizzano i mansionari, cioè delle matrici: [risorse / compiti che le risorse possono svolgere]

Nel mansionario potranno essere inserite altre informazioni come, ad esempio, orari di lavoro, regole per lo straordinario, e così via.

L'organigramma aziendale, per il caso del Pronto Soccorso, assume la forma grafica mostrata in Figura 2.9 (evidentemente siamo in questo caso a livello di istanza, stiamo parlando del pronto soccorso dell'ospedale X). Graficamente nei nodi dell'organigramma sono stati indicati il nome dell'unità/posto e il ruolo responsabile; sono possibili comunque altre convenzioni, ad esempio spesso è inserito direttamente il nome della persona che riveste il ruolo stesso.



Figura 2.9: Organigramma del PS

Riprendiamo ora il caso dell'Azienda di Noleggio di Auto considerata nel modello strategico. La Figura 2.10 illustra l'organigramma di una delle Agenzie dell'azienda (ricordiamo che tutte le agenzie sono organizzate allo stesso modo).

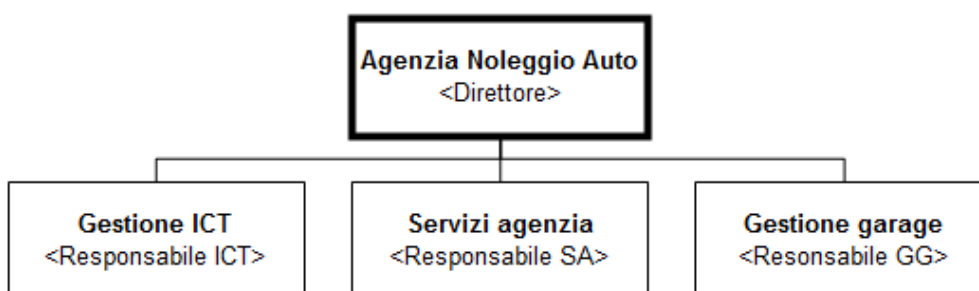




Figura 2.10: Organigramma dell'Agenzia

Il collegamento fra la struttura organizzativa e le attività dell'azienda è costituito dal concetto di **compito** (unità di lavoro). Durante l'esecuzione di un processo le attività sono abilitate quando vengono raggiunte dal flusso di controllo, ad esempio il sig. *Rossi* (gamba fratturata) ha eseguito il Triage e deve essere visitato. A quel punto l'organizzazione deve definire una regola per assegnare all'attività Eseguire visita le risorse in grado di svolgere i compiti che l'attività richiede: eseguire visita (medico) e assistere durante l'esecuzione (infermiere); fra tutti i medici e gli infermieri a disposizione e che in quel momento non sono impegnati su altre attività bisogna scegliere quelli da impegnare per il sig. *Rossi* seguendo una regola aziendale che è chiamata regola di allocazione. Se erano liberi e sono stati scelti il dr. Blu e l'inf. Gialli diremo che l'istanza dell'attività Eseguire visita relativa al paziente *Rossi* può *scattare* (cioè essere eseguita).

Contemporaneamente, altre coppie medico-infermiere potrebbero essere impegnate nella visita, ad esempio, dei pazienti *Verdi* e *Bianchi* : diremo allora che sono attive tre istanze dell'attività Eseguire visita che possono operare in quanto hanno ricevuto le risorse necessarie; potrebbe accadere che un ulteriore paziente si presenti per la visita e manchi ad esempio un medico: in tal caso il nuovo paziente entrerà in attesa che uno dei medici ritorni disponibile.

Esistono diversi tipi (e quindi regole) di allocazione, ad esempio:

1. **allocazione diretta**: una specifica persona viene assegnata a una determinata istanza di attività;
2. **allocazione per ruolo**: è la forma più comune, tutti i membri di un certo ruolo possono svolgere gli stessi compiti e quando un'istanza di attività è pronta e per essere eseguita richiede un compito associato al ruolo, un membro del ruolo è selezionato per svolgere il compito relativo e allocato all'attività. Poiché l'allocazione avviene a tempo di esecuzione solo le risorse (in quel ruolo) effettivamente disponibili possono essere selezionate; la selezione può avvenire in vari modi, ad esempio: la persona con più anzianità nel ruolo, quella impegnata di meno nella giornata, casualmente, e così via;
3. **allocazione differita**: l'allocazione avviene a tempo di esecuzione non sulla base di un ruolo definito in fase di progetto ma attraverso un passo esplicito di allocazione (si utilizza una particolare regola aziendale interna);
4. **separazione di doveri**: riguarda l'allocazione di persone diverse sull'attività di un processo (ad esempio, firma di due responsabili diversi su un ordine);
5. **gestione di casi**: riguarda l'allocazione di una stessa risorsa su tutte le attività di un processo (o sotto-processo), al fine di ridurre errori e tempi di esecuzione;
6. **allocazione organizzativa**: la persona è allocata in base alla posizione (ruolo) che essa riveste nell'organizzazione (ad esempio è necessario il manager dell'unità/posto che richiede i prodotti).

Le risorse aziendali rivestono ovviamente un ruolo fondamentale in azienda, in particolare le risorse umane, e devono quindi essere adeguatamente descritte; specificheremo la descrizione delle risorse nel capitolo riguardante la metodologia di analisi e modellazione dei processi.

Le unità organizzative rivestono ovviamente un ruolo fondamentale rispetto al modello strategico (BMM) dell'azienda. In particolare, ogni unità organizzativa:

- è definita dalle Strategie aziendali,
- partecipa nella definizione dei Fini aziendali e fornisce dei Mezzi per raggiungerli (nell'ambito delle sue responsabilità),
- è un Fattore che può assumere una rilevanza più o meno importante nell'azienda ed influenzare le altre unità,
- partecipa alle valutazioni aziendali e viene valutata.

## 2.3 Il Modello Funzionale

Il Modello Funzionale descrive il comportamento dell'azienda analizzando lo scambio di **oggetti** (informazioni, documenti, manufatti, prodotti,...) fra le sue **funzioni**, tenendo conto delle **risorse** impegnate e dei **vincoli** (o regole aziendali) da rispettare nello svolgimento delle **attività**.

Esaminiamo ora il meta-modello funzionale mostrato in Figura 2.11.

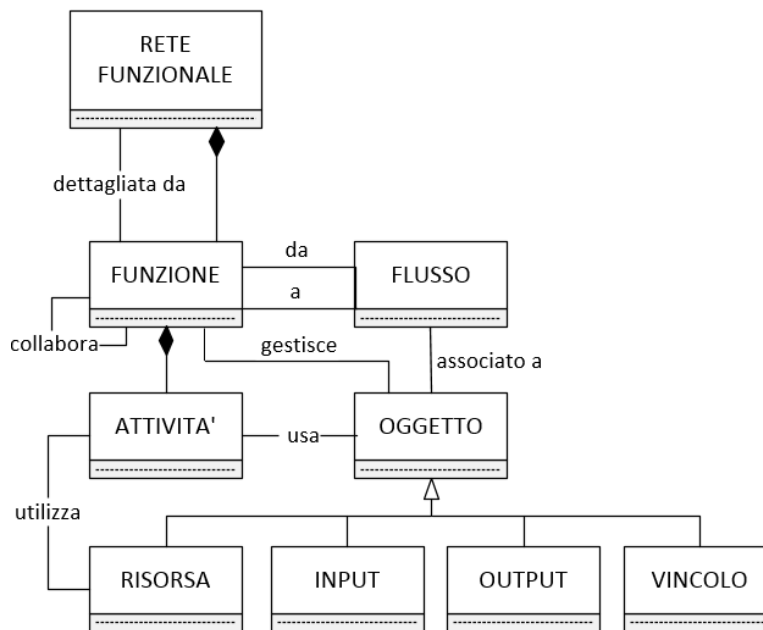


Figura 2.11: Meta-modello Funzionale

Una **funzione** è *composta da* una o più **attività** e può essere *dettagliata da* una **rete funzionale** composta a sua volta da funzioni a livello inferiore; la funzione a livello superiore *coordina* le funzioni a livello inferiore. Una funzione può *collaborare* con altre funzioni e *gestisce* degli **oggetti** (in particolare le **risorse** assegnate alla funzione). Un'attività *usa* degli oggetti (entità o informazioni presenti nel mondo reale, come ad esempio documenti e archivi) che *possono essere* di **input** o di **output**, e in questo caso sono modificati (cambiano di stato) in seguito all'esecuzione dell'attività stessa; gli oggetti di tipo **vincolo** rappresentano delle regole da rispettare nello svolgimento della funzione (e quindi delle sue attività) e non sono modificati. Le **risorse** (umane, tecnologiche, finanziarie,...) sono considerate come particolari oggetti dell'azienda, esse vengono *utilizzate* dalle attività *componenti* la funzione che *gestisce* le risorse stesse. Quando un oggetto in output da una funzione è anche in input a un'altra funzione (allo stesso livello) viene inserito nella rete un **flusso** cioè un arco diretto dalla funzione che produce a quella che riceve l'oggetto e a cui l'oggetto stesso è *associato*.

A livello di Modello useremo alcuni strumenti linguistici: il **linguaggio IDEF0**, per descrivere le reti funzionali, e la **matrice delle attività** per specificare le attività componenti la funzione in esame.

Il linguaggio IDEF0 permette di specificare l'articolazione funzionale di un'azienda mediante una **rete funzionale** (diagramma IDEF0) organizzata gerarchicamente a livelli.

La rete iniziale in un diagramma IDEF0 (chiamata **diagramma di contesto**) comprende soltanto la funzione che si vuole analizzare, usando la notazione indicata in Figura 2.12, e specifica il contesto in cui opera la funzione, cioè gli oggetti che essa scambia in qualche modo con il "resto del mondo" (che può comprendere il mondo esterno e/o altre funzioni dell'azienda).

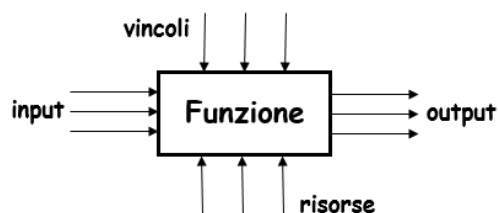


Figura 2.12: Notazione funzionale IDEF0: diagramma di contesto

Gli archi entranti o uscenti dalla funzione sono anche chiamati **flussi** e descrivono gli oggetti che in un qualche modo sono rilevanti per poter svolgere la funzione, le sotto-funzioni e le attività coordinate dalla funzione. In particolare, lungo il *lato sinistro* del rettangolo che rappresenta la funzione *entrano* gli oggetti in input (tali oggetti saranno assorbiti dalla funzione o saranno modificati), lungo il lato destro *escono* gli oggetti in output (oggetti creati dalla funzione o oggetti in input modificati), lungo il lato superiore *entrano* i vincoli, cioè i riferimenti a regole aziendali di cui bisogna tener conto durante l'esecuzione della funzione e lungo il lato inferiore *entrano* i riferimenti alle risorse che la funzione utilizza per eseguire le attività che la compongono.

A partire dalla rete di contesto, la funzione in esame può essere decomposta in reti di sotto-funzioni a livello 1 che a loro volta possono essere decomposte in sotto-funzioni a livello 2 e così via, fino a giungere a un livello di dettaglio in cui non ha senso decomporre ulteriormente: a questo livello abbiamo in generale una o più attività che vengono svolte nei posti di lavoro dell'azienda.

Poiché avevamo previsto nel meta-modello organizzativo che ad ogni unità corrispondesse una sola funzione (e viceversa), l'albero di decomposizione delle reti funzionali è isomorfo (cioè ha la stessa forma) all'organigramma organizzativo, per cui è utile farsi guidare nella decomposizione funzionale dall'organigramma stesso.

Per comprendere meglio il processo di decomposizione riprendiamo l'esempio dell'Azienda ANA; il diagramma di contesto per la funzione che gestisce una generica agenzia è mostrata in Figura 2.13.

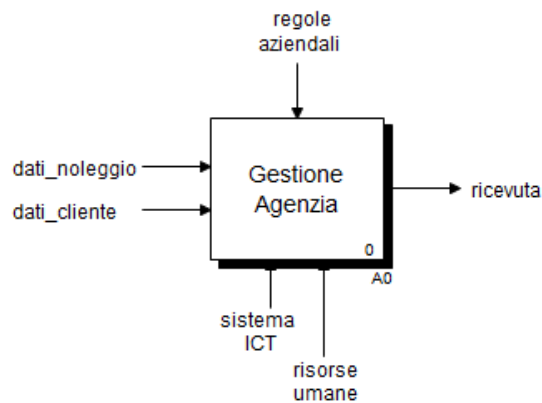


Figura 2.13: Diagramma di contesto Agenzia ANA (livello 0)

La Figura 2.14 mostra il primo livello di decomposizione della rete di contesto, a tale livello compaiono i posti di lavoro che costituiscono l'agenzia.

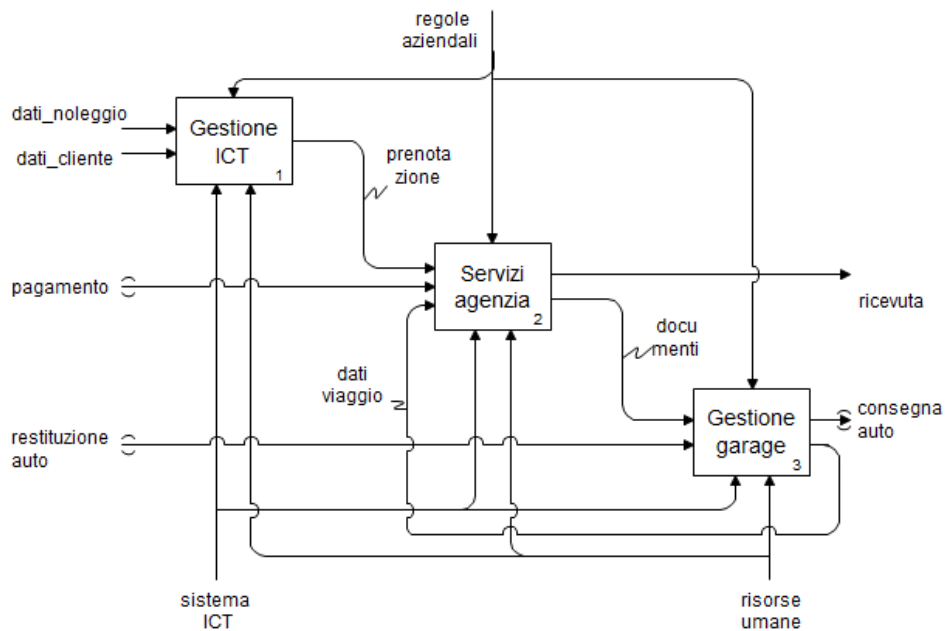


Figura 2.14: Rete ANA (livello 1)

Come possiamo osservare in questo esempio, gli input, i controlli e le risorse entranti nella funzione a livello 0 e gli output che ne escono devono essere "redistribuiti" tra le sotto-funzioni del livello 1; inoltre al livello di dettaglio 1 possono apparire oggetti che non sono rilevanti al livello precedente; tali oggetti sono quelli scambiati fra le sotto-funzioni (*prenotazione* e *documenti*) oppure sono considerati locali e in tal caso sono inseriti in un cosiddetto **tunnel** (), cioè una coppia di parentesi (*pagamento*, *consegna auto* e *restituzione auto*).

Consideriamo ora un caso più complesso, relativo all'acquisto di un'automobile; la Figura 2.15 mostra il diagramma di contesto per la funzione relativa.

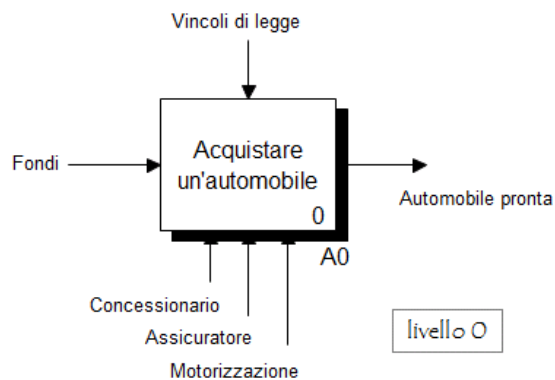


Figura 2.15: Diagramma di contesto Acquistare automobile (livello 0)

Al livello di dettaglio successivo (livello 1) abbiamo la rete mostrata in Figura 2.16 in cui compare la sotto-funzione *Ritirare automobile* insieme a tre attività (Registrare al PRA, Assicurare e Pagare bollo) che in quanto tali non sono ulteriormente decomposte.

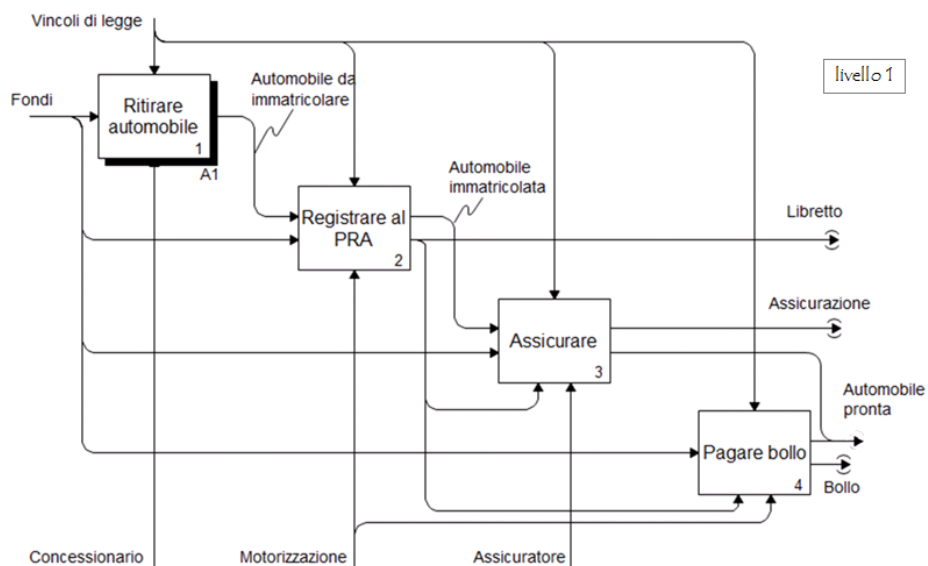


Figura 2.16: Rete Acquistare automobile (livello 1)

Decomponendo la sotto-funzione *Ritirare automobile* si ottiene la rete di Figura 2.17 a livello 2, in cui i nodi Stipulare contratto, Pagare e Consegnare automobile rappresentano delle attività eseguite nei posti di lavoro corrispondenti.

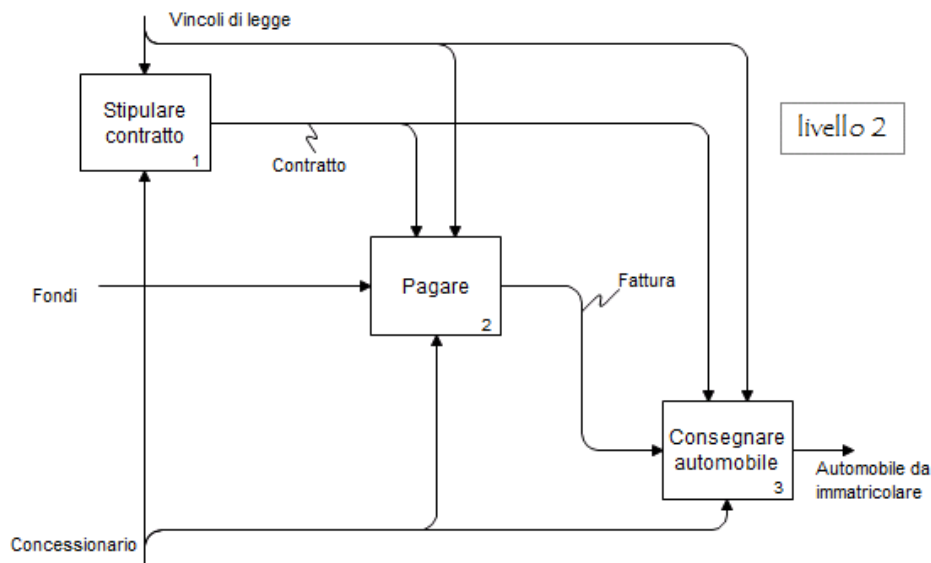


Figura 2.17: Rete Acquistare automobile (livello 2)

L'analisi funzionale, come mostrano gli esempi, è una procedura tipicamente "top down": si parte dalla funzione che si intende analizzare e man mano la si decompone per arrivare ad un livello in cui ci sono solo delle attività svolte in un posto di lavoro ben definito, per cui non ha senso decomporre ulteriormente.

Per completare la descrizione della rete funzionale ad ogni nodo della rete si può associare un descrittore che specifica alcune proprietà della funzione stessa come:

1. identificatore e nome della funzione,
2. breve descrizione della funzione,
3. elenco delle sotto-funzioni componenti,
4. elenco eventuali attività svolte da risorse (ad esempio, il responsabile) impegnate direttamente nella funzione (e non assegnate a sotto-funzioni o attività a livello inferiore).

Al termine della costruzione della rete funzionale le attività sono dettagliate tramite una **matrice delle attività** che comprende tipicamente delle informazioni del tipo:

1. identificatore e nome dell'attività,
2. obiettivo e breve descrizione dell'attività per precisare meglio l'obiettivo,
3. risorse necessarie per la sua esecuzione,
4. dati quantitativi sull'attività, come la sua durata (se è di tipo operativo) e gli eventuali costi fissi per la sua esecuzione (ad esempio, risorse che vengono consumate, come materiali monouso); se l'attività è seguita da controlli si possono inserire dati sulla probabilità di seguire un cammino piuttosto che un altro.

Specificheremo la descrizione delle attività nel capitolo riguardante la metodologia di analisi e modellazione dei processi.

## 2.4 Il Modello dei Processi

I modelli fino ad ora considerati, Strategico, Organizzativo e Funzionale, sono essenzialmente dei modelli statici che rappresentano la struttura dell'azienda, delle sue componenti e l'architettura delle sue funzioni, ma non è possibile risalire da questi modelli a come si comporta *effettivamente* l'azienda quando, ad esempio, al Pronto Soccorso si presenta il sig. *Rossi*.

Sorge quindi la necessità di specificare come avviene l'esecuzione dei processi e delle attività in azienda e inizieremo a prendere in considerazione un **Modello di Attività**; un tale modello deve descrivere l'insieme di esecuzioni possibili, o **istanze**, dell'attività in esame. Ad esempio, se noi consideriamo l'attività Eseguire triage del Pronto Soccorso in un certo istante potrebbero esistere (sono **attive**) tre istanze dell'attività relative ai pazienti *Rossi, Bianchi, Verdi*:

{Eseguire triage (*Rossi*), Eseguire triage (*Bianchi*), Eseguire triage (*Verdi*)}

Questo vuol dire che le risorse (umane e non) da utilizzarsi per eseguire i compiti dell'attività erano disponibili per i tre pazienti, e potrebbe accadere che un paziente successivo debba attendere perché alcune o tutte le risorse necessarie sono già impegnate: si crea in tal modo una **coda** di attesa per l'attività Eseguire triage.



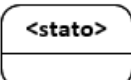
Un'istanza di attività si evolve nel tempo e può passare attraverso vari **stati**, ad esempio può essere attiva o essere in attesa di risorse, e così via; questa dinamica può essere descritta da un **diagramma stati/transizioni** in cui i nodi sono gli **stati** e gli archi le **transizioni** che permettono di passare da uno stato all'altro.

Le **transizioni** sono determinate da **eventi** (che identificano le transizioni stesse) e per ogni stato l'insieme delle transizioni uscenti determina i possibili stati che possono essere raggiunti a partire da quello in esame come conseguenza del verificarsi degli eventi corrispondenti.

Un **evento** è il verificarsi di un fatto in un certo istante di tempo che provoca una transizione cioè il *passaggio da uno stato a un altro* del sistema. Nel diagramma ci sono due nodi speciali, un evento iniziale - **start** - e un evento finale - **end** (quest'ultimo non è obbligatorio).

In un diagramma oltre a **stati semplici** possono esistere **stati composti**: una transizione che esce dal bordo dello stato composto disattiva tutti gli stati interni, mentre se entra sul bordo vengono attivati tutti gli stati interni iniziali (stati senza transizioni entranti). Gli stati hanno un nome e per denotare gli stati interni si usa una notazione puntata, ad esempio un nodo X interno ad un nodo composto Y verrà denotato Y.X.

I diagrammi stati/transizioni ammettono una semplice rappresentazione grafica i cui elementi sono illustrati nella Tabella 2.4.

Simbolo	Significato
	evento iniziale <b>start</b> ;
	evento finale <b>end</b> ;
	<b>stato</b> semplice del sistema;

	<p><b>stato composto</b> del sistema;</p>
	<p><b>transizione:</b> specifica il passaggio da uno stato all'altro del sistema al verificarsi dell'<b>evento e</b> .</p>

Tabella 2.4: Simboli grafici per i diagrammi stati/transizioni

La Figura 2.18 mostra un possibile modello stati/transizioni delle istanze di un'attività.

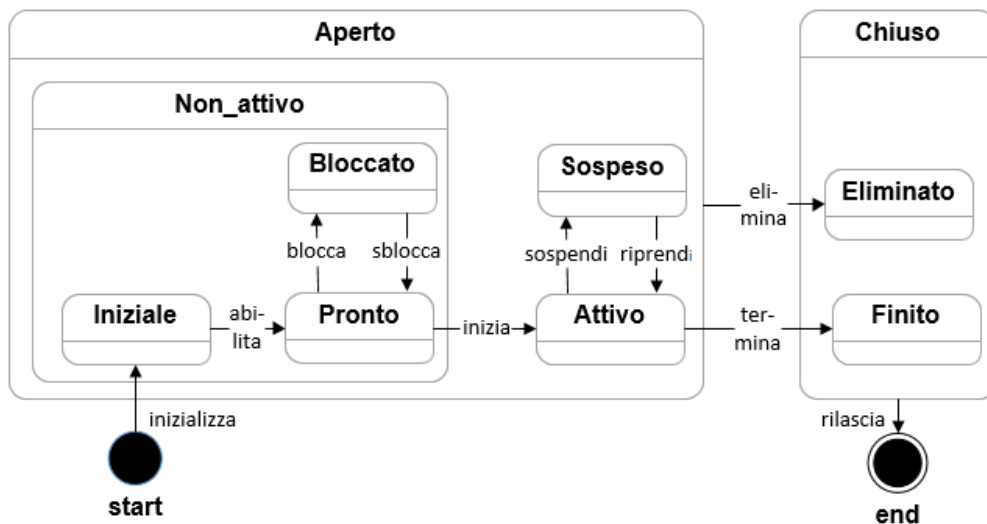


Figura 2.18: Modello stati/transizioni delle istanze di un'attività

A partire dall'evento **start** (l'attività può iniziare perché raggiunta dal flusso di controllo del processo) la transizione *inizializza* crea l'istanza di attività nello stato:

**Aperto.Non\_attivo.Iniziale**

e se le risorse necessarie alla sua esecuzione sono disponibili e vengono allocate all'istanza stessa, si passa (transizione *abilita*) allo stato:

**Aperto.Non\_attivo.Pronto**

Se esiste una condizione che impedisce di operare (transizione *blocca*) si passa nello stato: **Aperto.Non\_attivo.Bloccato**

ritornando poi allo stato Pronto (transizione *sblocca*), altrimenti l'istanza può *iniziare* e passa nello stato:

**Aperto.Attivo**

e quindi concludersi (transizione *termina*) passando nello stato:

**Chiuso.Finito**

Quando l'istanza è nello stato composto **Aperto** potrebbe verificarsi un evento che *elimina* l'attività e si passa nello stato:

**Chiuso.Eliminato**



quindi qualunque sia lo stato interno attivo l'istanza viene chiusa. Nello stato **Aperto.Attivo** può verificarsi un evento (transizione *sospendi*) per cui l'istanza deve essere sospesa e si entra nello stato:

**Aperto.Sospeso**

e poi può essere *ripresa*. Infine dallo stato **Chiuso** l'istanza *rilascia* le risorse che le erano state assegnate e si chiude la sua evoluzione (evento **end**).

Osserviamo che il diagramma di stato descrive “cosa potrebbe accadere”, e **non** “cosa accadrà”. Ricollegandoci all'esempio del Pronto Soccorso, l'arrivo del paziente *Rossi* in Triage genera un'istanza dell'attività Eeguire triage e supponiamo che l'istanza stessa passi attraverso gli stati:

**start** → Iniziale → Pronto → Attivo → Finito → **end**

Il cammino precedente è uno dei possibili cammini evolutivi nel diagramma degli stati che congiungono l'evento iniziale **start** all'evento finale **end**; l'insieme di questi cammini comprende tutte e sole le possibili evoluzioni di un'istanza dell'attività.

Ogni cammino ammette una rappresentazione duale in cui invece di mettere in evidenza gli stati sono messe in evidenza le transizioni, per cui l'evoluzione dell'attività Eeguire triage nel caso del paziente *Rossi* può essere rappresentata mediante il diagramma, chiamato **storia degli eventi**, di Figura 2.19 di semplice interpretazione. A partire dall'evento *start* ( $t_1$ ), l'attività passa nello stato **Iniziale** al tempo  $t_2$  (per effetto della transizione *inizializza*) e quindi negli stati **Pronto**, **Attivo** e **Finito** negli istanti  $t_3$ ,  $t_4$ ,  $t_5$  rispettivamente. Al tempo  $t_6$  le risorse che aveva acquisito sono rese nuovamente disponibili.

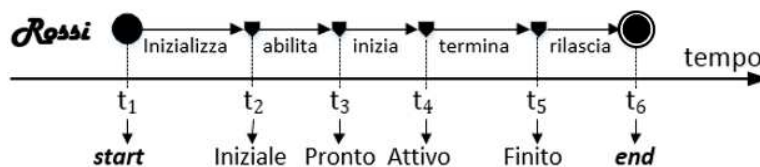


Figura 2.39: Storia degli eventi per il sig. *Rossi* nell'attività Eeguire triage

Osserviamo che svolgendosi lungo l'asse dei tempi gli eventi specificano la durata dei vari stati dell'attività (in una certa scala); ad esempio l'intervallo effettivo in cui il paziente *Rossi* viene trattato è quello compreso fra gli istanti  $t_4$  e  $t_5$ , gli altri intervalli sono di norma molto più brevi per cui generalmente è a questo intervallo che si fa riferimento quando si parla di durata di un'attività.

Quando ci sono risorse sufficienti per trattare istanze diverse che evolvono in parallelo, ad esempio sono attive le tre istanze {Eeguire triage (*Rossi*), Eeguire triage (*Bianchi*), Eeguire triage (*Verdi*)}, le storie si “intrecciano” generando una storia “complessiva” dell'attività mostrata in Figura 2.20.

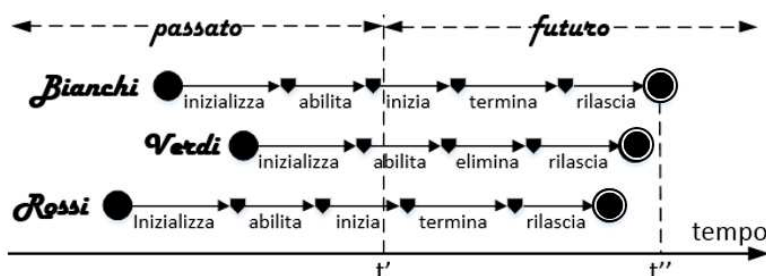


Figura 2.20: Storia degli eventi per l'attività Eseguire triage

Al tempo  $t'$ , le istanze Eseguire triage (*Rossi*) e Eseguire triage (*Bianchi*), sono pronte per iniziare, mentre Eseguire triage (*Verdi*) attende che vengano allocate le risorse: cosa accadrà non è ancora noto, e solo al tempo  $t''$  si saprà che le prime due sono terminate regolarmente mentre la seconda istanza è stata eliminata (ad esempio, il sig. *Verdi* è andato via dal Pronto Soccorso perché non poteva più aspettare).

In realtà il diagramma stati/transizioni può essere significativamente più complesso, si pensi, ad esempio, al caso in cui l'attività debba aggiornare una base dati; in tal caso è necessario introdurre delle transizioni che eseguano il ripristino dello stato della base di dati se l'attività non termina correttamente.

Un **Modello dei Processi** descrive l'insieme di esecuzioni possibili, o istanze, del processo in esame. Per esempio, se ritorniamo a considerare il modello che descrive il funzionamento del Pronto Soccorso (Figura 2.2) in un certo istante possono esistere tre istanze relative a tre pazienti, i sigg. *Rossi*, *Bianchi*, *Verdi* giunti in tempi diversi. A questi tre pazienti potrebbero corrispondere delle "storie" (sequenze di attività) diverse, ad esempio:

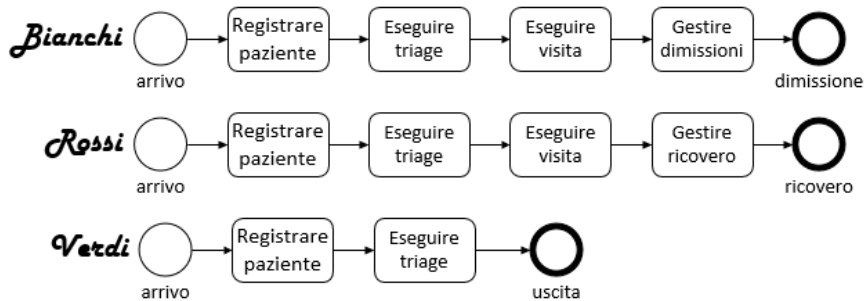


Figura 4: Istanze di processo

Poiché le tre istanze del processo evolvono in parallelo, introducendo per le istanze di un processo una storia degli eventi analoga a quella che abbiamo visto per le attività potremmo avere una situazione come quella illustrata in Figura 2.22: in un certo istante  $t'$  potremmo avere i pazienti *Bianchi* e *Rossi* in Eseguire visita e *Verdi* in Eseguire triage, ma solo al tempo  $t''$  sapremo che *Verdi* è uscito mentre *Bianchi* è stato dimesso e *Rossi* è stato ricoverato.

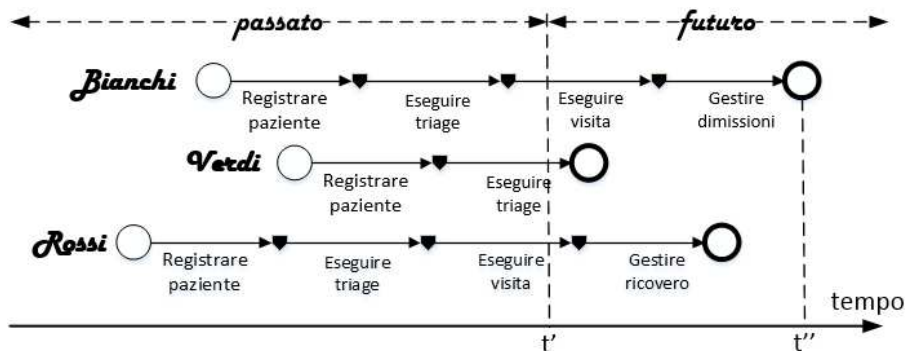


Figura 2.22: Storia degli eventi delle istanze del processo PS

Analogamente a quanto visto per le attività, la dinamica di un processo può essere modellata mediante un modello stati/transizioni che descrive le possibili evoluzioni delle istanze del processo, questo modello è illustrato in Figura

2.23.

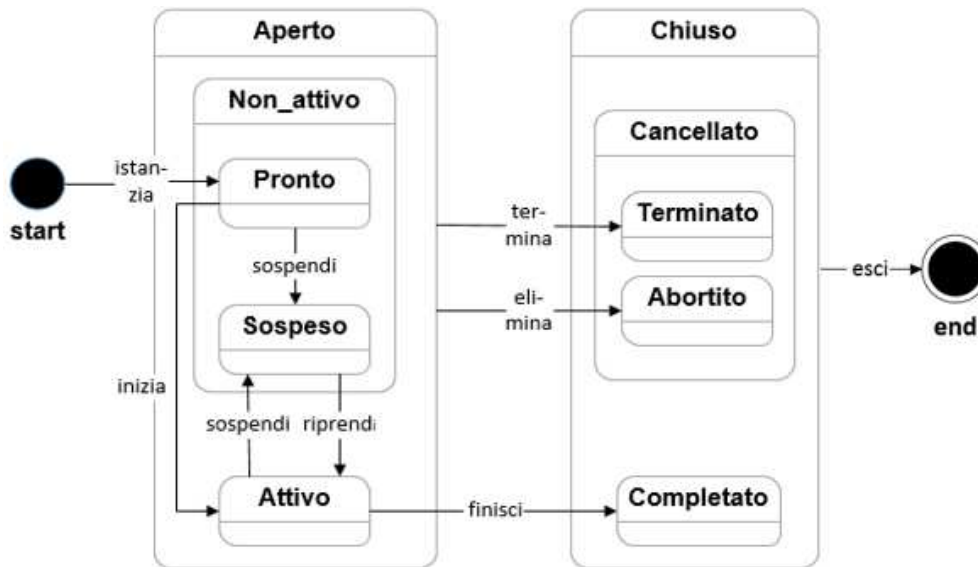


Figura 2.23: Modello stati/transizioni delle istanze di un Processo

A partire dall'evento **start** (il processo può iniziare perché accade l'evento) la transizione *istanzia* crea l'**istanza** del processo nello stato:

**Aperto.Non\_attivo.Pronto**

e se l'attività iniziale può partire si passa (transizione *inizia*) allo stato:

**Aperto.Attivo**

altrimenti si passa (transizione *sospendi*) nello stato:

**Aperto.Non\_attivo.Sospeso**

Nello stato **Aperto.Attivo** può verificarsi un evento (transizione *sospendi*) per cui l'istanza deve essere sospesa e passa nello stato:

**Aperto.Non\_attivo.Sospeso**

In questo stato nessuna attività è in grado di partire (ad esempio le risorse sono fuori orario di lavoro); l'istanza di processo può quindi ritornare attiva (transizione *riprendi*) e questo significa che delle attività possono riprendere; dallo stato attivo (transizione *finisci*) l'istanza di processo può passare nello stato:

**Chiuso.Completato**

che chiude correttamente l'istanza stessa. Mentre l'istanza è nello stato composto **Aperto** potrebbe verificarsi un evento che *elimina* l'istanza e quindi si passa nello stato:

**Chiuso.Cancellato.Abortito**

oppure la *termina* e si passa nello stato:

**Chiuso.Cancellato.Terminato**

Nel primo caso vengono terminate anche tutte le attività ancora attive, mentre nel secondo caso si permette alle attività ancora attive di terminare. La transizione *esci* completa l'esecuzione del processo (il log è memorizzato su una memoria sicura).

Se l'esecuzione del processo è controllata tramite un motore di workflow, la storia degli eventi del processo e delle singole attività è registrata durante l'esecuzione del processo stesso nel cosiddetto **giornale** (log) del sistema; il giornale conserva la traccia di tutti gli eventi che hanno scandito:

- l'evoluzione (in parallelo) di tutti i processi  $P_k$  ( $k=1, \dots, n$ ) gestiti dal motore di workflow;

- per ogni processo  $P_k$ , l'evoluzione (in parallelo) di tutte le sue istanze  $Q_i$  ( $i=1,\dots,m$ ) attive in un certo istante nel sistema;
- per ogni istanza  $Q_i$  di ogni processo  $P_k$ , l'evoluzione (in parallelo) di tutte le istanze di tutte le attività in esecuzione in un certo istante nel sistema.

Una generica registrazione nel giornale comprende quindi accanto al tempo in cui è accaduto l'evento un identificatore dell'istanza di processo, un identificatore dell'istanza di attività e il tipo di evento (ad esempio "fine attività").

A partire dal giornale è possibile ricostruire la storia di tutto quello che è accaduto nel sistema, per cui un interessante filone di ricerca, chiamato **process mining**, riguarda l'analisi del giornale per ricostruire i modelli e analizzare le code reali del sistema.

## 2.5 Il Modello di Implementazione

Nella fase di implementazione un processo aziendale, specificato mediante il linguaggio BPMN, deve essere "trattato" in una **specifica eseguibile** tramite un motore di workflow; una tale specifica dovrebbe comunque prevedere passi svolti manualmente, parzialmente automatizzati o automatizzati completamente dal motore con il supporto della infrastruttura informatica e tecnologica dell'azienda (comprendente la rete aziendale e il sistema informativo).

In questo testo non affronteremo il problema generale dell'implementazione, ci limiteremo a discutere il progetto del **sistema informativo** aziendale che deve essere costruito per fornire a tutti i processi aziendali i dati di cui hanno bisogno.

La strategia di costruzione suggerita in questo testo prevede che il sistema informativo sia progettato partendo dalla specifica dei processi aziendali e presenta quindi il vantaggio sostanziale di assicurare che i dati necessari siano a disposizione quando saranno richiesti dalle attività dei processi.

Per introdurre la strategia di passaggio dalla specifica dei processi alla loro implementazione osserviamo che questa strategia richiede un cambiamento di vista radicale: è necessario passare dalla modellazione del **flusso di lavoro** (coordinamento della attività costituenti il processo) alla specifica di **come** le attività devono operare in pratica, e quindi alla modellazione dei requisiti che gli **utenti** del processo richiedono al sistema informativo che deve supportarli nel loro lavoro.

Per chiarire meglio questo punto riprendiamo il processo PS di Figura 2.2 e consideriamo ad esempio le attività Registrazione paziente ed Esegui triage. Per simulare le attività è sufficiente conoscere i tempi e le risorse impegnate, ma per la loro implementazione è necessario specificare **come** devono essere svolte. Nel caso del Pronto Soccorso possiamo utilizzare una conoscenza esistente, fornita ad esempio da un esperto del settore, oppure andare in un ospedale e vedere come le attività sono effettivamente eseguite dalle risorse dedicate (l'analisi della situazione esistente sarà una strategia che useremo in seguito discutendo le metodologie di progetto); supponiamo che il risultato sia una descrizione delle attività del tipo indicato in Tabella 2.5.

<u>Registrazione paziente</u>	Il Paziente si presenta allo sportello Ricezione, l'Impiegato compila la parte Anagrafica del ModuloPaziente, lo consegna al Paziente e lo indirizza al settore Triage.
<u>Esegui triage</u>	L'Infermiere di triage interroga il Paziente, fa l'anamnesi e giudica le condizioni del paziente assegnandogli un codice di urgenza (colori Bianco, Verde, Giallo o Rosso), informa il paziente sul possibile pagamento di un ticket, eventualmente prescrive degli esami da effettuare e lo indirizza al settore Visita.

Tabella 2.5: Descrizione di attività del Pronto Soccorso

Come si può notare già da questo semplice esempio, passando dal diagramma (astratto) di processo di Figura 2.2 alla sua implementazione la "scatola nera" che modella un'attività (ad esempio Registrare paziente) deve essere "aperta" per descrivere come l'attività deve essere effettivamente svolta.

La descrizione fornita nella Tabella 2.5 è l'embrione di uno strumento linguistico chiamato **Caso d'Uso**; la specifica dei casi d'uso fa parte del linguaggio **UML** (Unified Modeling Language) che rappresenta attualmente lo standard per la specifica dei programmi e dei sistemi informativi in termini di oggetti. UML comprende ben 13 strumenti descrittivi ma noi ne prenderemo in considerazione solo due, il **diagramma dei casi d'uso** e il **diagramma delle classi**.

Discutere il meta-modello di UML (ovviamente molto complesso) non rientra negli obiettivi di questo testo, per cui descriveremo nel seguito i due diagrammi che verranno effettivamente usati.

### 2.5.1 Diagramma dei Casi d'Uso

Un **caso d'uso** rappresenta un insieme di **scenari di interazione** fra uno o più utenti e il sistema in esame; in particolare useremo i casi d'uso per specificare l'implementazione delle attività del processo nel senso che ad un'attività faremo corrispondere un caso d'uso che, ad un livello ancora indipendente dalla specifica piattaforma di implementazione, descrive come l'attività deve essere svolta.

Il linguaggio UML prevede una forma grafica per i casi d'uso: gli **attori** sono indicati come "*omini*" stilizzati e i **casi d'uso** con *ellissi*, gli *archi* associano gli attori ai casi d'uso, permettendo quindi la costruzione di **diagrammi dei casi d'uso** che specificano (ad un livello molto astratto) le interazioni degli attori col sistema. Nel caso del Pronto Soccorso otteniamo il diagramma indicato in Figura 2.24.

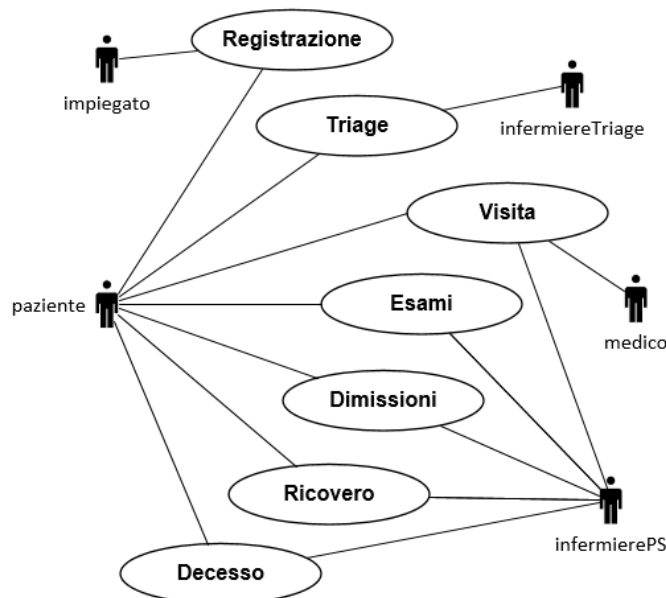


Figura 2.24: Diagramma dei casi d'uso Pronto Soccorso

La descrizione del caso d'uso è costituita da una serie di sezioni che descrivono:

- l'**obiettivo** del caso d'uso: in genere riprende l'obiettivo dell'attività;
- gli **attori**: comprendono attori **esterni** (ad esempio i pazienti) e attori **interni**, cioè le risorse aziendali allocate sull'attività; queste risorse possono essere sia attori umani (il o gli esecutori dell'attività) sia sistemi tecnologici (ad esempio il sistema con cui sono gestite le informazioni);
- le **pre-condizioni**: riguardano le situazioni che devono essersi verificate prima che l'attività possa essere eseguita;
- lo **scenario di esecuzione**: specifica l'insieme dei possibili cammini di esecuzione dell'attività costituiti da sequenze di **passi**; questi passi descrivono l'**interazione** fra gli attori esterni con quelli interni ed il sistema, e in tal caso sono detti **esterni** e avvengono attraverso l'**interfaccia** dell'attività (ad esempio l'impiegato riceve l'ordine dal cliente, l'infermiere consegna al paziente delle istruzioni per il medico curante, e così via); i passi **interni** riguardano le interazioni fra le risorse aziendali (ad esempio l'impiegato registra l'ordine nella base dati, l'infermiere inserisce i dati di anamnesi nel modulo del paziente, e così via). I passi interni possono riguardare:
  - invocazione (di altre applicazioni, ad esempio un editor di testi),
  - validazione (dei dati immessi, delle azioni compiute dagli attori, e così via),
  - verifica di regole di business (ad esempio, controllo della validità della carta di credito di un cliente),
  - gestione dei dati (lettura, cancellazione, modifica, scrittura, verifica dei vincoli di integrità).

Se nel processo l'attività in esame è seguita da controlli, nei passi dello scenario devono essere trattate tutte le condizioni che permettono di decidere successivamente i cammini di controllo da seguire;

- lo **schema dei dati** (o, più semplicemente, **schema**) del caso d'uso è il diagramma delle classi che descrive i dati di cui l'attività ha bisogno per poter essere eseguita;
- le **post-condizioni** riguardano le situazioni che diventano effettive dopo che l'attività è stata eseguita; anche in questo caso ci si riferisce in genere alla possibilità di eseguire le attività successive e/o all'invio di messaggi agli altri partecipanti.

Riprendendo l'esempio dell'attività Eseguire triage, gli utenti o, come sono chiamati in UML, gli **attori** del caso d'uso sono l'infermiere di triage T e il paziente P, e lo scenario di interazione può essere descritto, in modo più dettagliato di quanto illustrato nella Tabella 2.5, dalla sequenza di passi:

1. T interroga P e compila la parte *Anamnesi* del modulo,
2. T giudica le condizioni di P, gli assegna un codice di urgenza (colori Bianco, Verde, Giallo o Rosso) e gli comunica l'eventuale necessità di pagamento del ticket,
3. [eventualmente] T compila la parte *Esami da effettuare* del modulo,
4. T indirizza P al settore Visita.

La sequenza indicata è quella che viene generalmente seguita, potremmo avere delle varianti che descrivono scenari alternativi, ad esempio:

- 4'. P decide di lasciare il Pronto Soccorso

Questa interpretazione dei casi d'uso sottintende una struttura implementativa delle attività del tipo indicato in Figura 2.25.

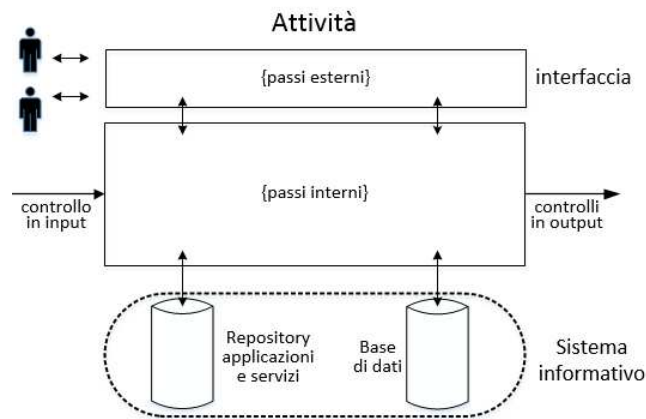


Figura 2.25: Modello di implementazione di un'attività

Per la specifica dei passi sono stati introdotti diversi formalismi ma in questo testo useremo in pratica il linguaggio naturale con dei semplici costrutti di controllo.

Supposto che nell'attività Registrare paziente si sia compilato il modulo paziente in forma elettronica lo scenario per il caso d'uso Eseguire triage può assumere la forma (*I* o *E* indicano passi interni o esterni) indicata nella Tabella 2.6.

1	<i>E</i>	T recupera sulla sua stazione di lavoro il modulo di P
2	<i>E</i>	T interroga P e compila la parte <i>Anamnesi</i> del modulo
3	<i>E</i>	T giudica le condizioni di P, gli assegna un codice di urgenza (colori Bianco, Verde, Giallo o Rosso), e gli comunica l'eventuale pagamento del ticket
4	<i>I</i>	T registra il modulo
5	<i>E</i>	P decide se continuare [esci? → si] P esce → controllo ① [esci? → no] P continua → controllo ②
6	<i>E</i>	T valuta la necessità di esami [esami? → si] P continua → controllo ③ [esami? → no] P continua → controllo ④
7	<i>I</i>	T aggiorna il modulo

Tabella 2.6: Descrizione dello scenario Eseguire triage

Gli archi di controllo ①, ②, ③, ④ sono indicati sul frammento del diagramma di processo PS che interessa l'attività Eseguire triage come indicato in Figura 2.26.

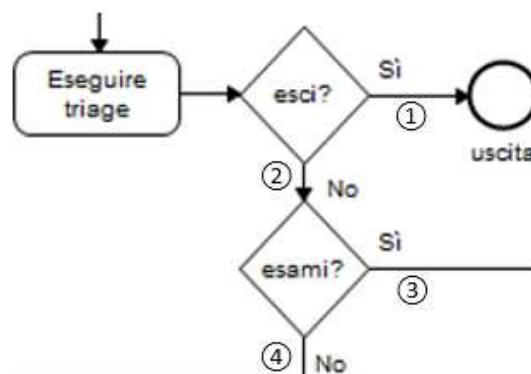
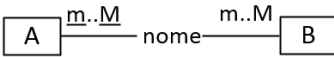
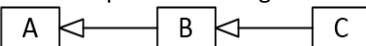


Figura 2.26: Attività Eseguire triage nel processo PS

Come mostra la tabella precedente, in fase di implementazione la specifica dell'attività "assorbe" la specifica dei controlli successivi, [esci?] ed [esami?] nel nostro esempio. Questa scelta deriva dall'ipotesi che durante l'esecuzione dell'attività vengano valutate le condizioni che determinano l'instradamento successivo del controllo.

### 2.5.2 Diagramma delle Classi

Il **diagramma delle classi** descrive le **classi** a cui gli **oggetti** che compongono un sistema, appartengono e le **relazioni** (**associazioni**, **aggregazioni** e **generalizzazioni**) esistenti tra loro. Abbiamo già preso in considerazione un esempio di diagramma delle classi con la Figura 2.1 e ora con la Tabella 2.7 preciseremo i concetti fondamentali già introdotti nella Tabella 2.2.

<b>oggetto</b>	- ha determinate proprietà (o <b>attributi</b> ) che possono assumere dei valori all'interno di opportuni <b>domini</b> di dati, - mostra determinati comportamenti, codificati in un insieme di <b>operazioni</b> , - i valori degli attributi e gli oggetti collegati definiscono lo <b>stato</b> dell'oggetto.
<b>classe</b>	- <b>oggetti</b> con la stessa struttura e gli stessi comportamenti costituiscono le <b>istanze</b> di una <b>classe</b> .
<b>attributi</b>	- sono definiti nella specifica della classe con una coppia: <b>nome : tipo</b> dove il <b>tipo</b> del dominio in cui l'attributo può assumere un valore può essere <i>primitivo</i> ( <i>integer, real, string,...</i> ) o <i>complesso</i> .
<i>relazione di associazione</i>  <u>&lt;nome&gt;</u>	- specifica un insieme di <b>legami</b> ( <b>istanze</b> dell'associazione) fra oggetti, - possiede delle <b>molteplicità</b> ; la notazione m..M nel diagramma:  specifica che ad un oggetto di A sono legati <b>almeno</b> m e al <b>massimo</b> M oggetti di B e, viceversa, ad un oggetto di B sono legati <b>almeno</b> <u>m</u> e al <b>massimo</b> <u>M</u> oggetti di A, con i vincoli: $0 \leq m \leq M, 0 \leq \underline{m} \leq \underline{M}, 1 \leq M, \underline{M} \leq *$ dove * sta per <u>numero intero</u> maggiore di 1 o, genericamente, per <u>molte</u> .
<i>relazione di generalizzazione</i>  ←	- specifica un rapporto: classe $\subseteq$ sotto-classe per cui la sotto-classe eredita tutte le proprietà della classe e può aggiungerne altre, - vale la proprietà <b>transitiva</b> per cui nel diagramma:  <b>se</b> $A \subseteq B$ e $B \subseteq C$ <b>è anche</b> $A \subseteq C$ ad esempio: PERSONA $\subseteq$ IMPIEGATO $\subseteq$ MANAGER
<i>relazione di aggregazione</i>  ◇	- specifica un rapporto: <b>tutto</b> ◇ <b>parti</b> per cui oggetti <b>parti</b> sono <b>aggregati</b> in un oggetto <b>tutto</b> ; ad esempio, con: CAMPIONATO ◇ SQUADRE ◇ GIOCATORI si descrive un Campionato che <i>aggrega</i> delle Squadre e ogni Squadra <i>aggrega</i> dei Giocatori - vale la proprietà <b>transitiva</b> per cui: <b>se</b> $A \diamond B$ e $B \diamond C$ <b>è anche</b> $A \diamond C$
	- specifica un rapporto: <b>oggetto</b> ◆ <b>componenti</b> per cui un




<p>relazione di composizione</p> 	<p><b>oggetto</b> composto è <i>composto da</i> oggetti <b>componenti</b>; ad esempio: LIBRO ◊— CAPITOLI ◊— PARAGRAFI (un Libro è <i>composto da</i> Capitoli e un Capitolo è <i>composto da</i> Paragrafi)</p> <p>- vale la proprietà <b>transitiva</b> per cui:          se A ◊— B e B ◊— C è <b>anche</b> A ◊— C</p> <p>- per la composizione vale una proprietà detta <b>dipendenza esistenziale</b> per cui i componenti non possono esistere senza che esista l'oggetto composto (se si cancella un oggetto composto devono essere cancellati anche i componenti); ricordiamo che questa proprietà non vale per la relazione di aggregazione.</p>
--	---

Tabella 2.7: Concetti fondamentali del Diagramma delle Classi

Nel nostro esempio del Pronto Soccorso il diagramma delle classi (iniziale) relativo alle due attività considerate potrebbe assumere la forma mostrata in Figura 2.27.

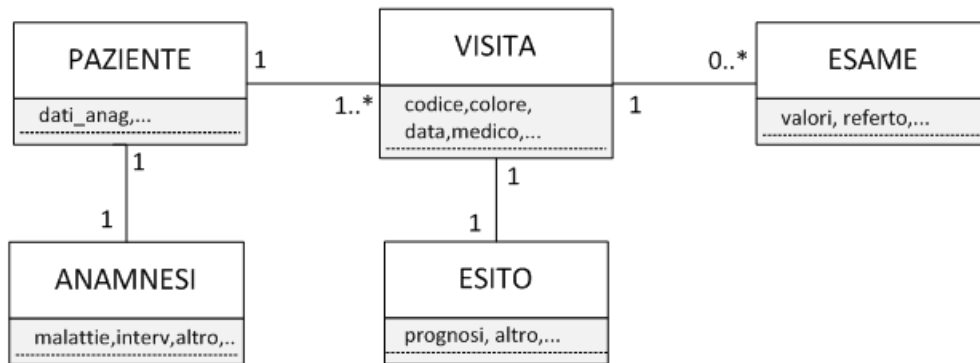


Figura 2.27: Diagramma iniziale delle classi per il Pronto Soccorso

Osserviamo che le classi descrivono sostanzialmente le informazioni che si trovano nel modulo che segue il paziente nel suo percorso nel Pronto Soccorso.

La specifica completa del caso d'uso Eseguire triage assume quindi la forma illustrata in Figura 2.28.

Nome	Eseguire triage
Obiettivo	Valutare le condizioni del paziente che si presenta al Pronto Soccorso
Attori	T: Infermiere di triage, P: paziente
Pre-condizioni	Il paziente è già stato registrato

Scenario di esecuzione	1	E	T recupera sulla sua stazione di lavoro il modulo di P
	2	E	T interroga P e compila la parte <i>Anamnesi</i> del modulo
	3	E	T giudica le condizioni di P, gli assegna un codice di urgenza (colori Bianco, Verde, Giallo o Rosso), e gli comunica l'eventuale pagamento del ticket
	4	I	T registra il modulo
	5	E	P decide se continuare [esci? → si] P esce → controllo ① [esci? → no] P continua → controllo ②
	6	E	T valuta la necessità di esami [esami? → si] P continua → controllo ③ [esami? → no] P continua → controllo ④
	7	I	T aggiorna il modulo
Schema	<pre> classDiagram     class PAZIENTE {         dati_anag,...     }     class VISITA {         codice,colore, data,medico,...     }     class ESAME {         valori,referto,...     }     class ANAMNESI {         malattie,interv,altro,..     }     class ESITO {         prognosi, altro,...     }     PAZIENTE "1" -- "1..*" VISITA     VISITA "1" -- "0..*" ESAME     PAZIENTE "1" -- "1" ANAMNESI     VISITA "1" -- "1" ESITO         </pre>		
Post-condizioni	(a) il Paziente esce (b) il Paziente si sposta su Eeguire prelievi (c) il Paziente si sposta su Eeguire visita		

Figura 2.28: Specifica del caso d'uso Eeguire triage

Nel seguito del testo non ci occuperemo dell'implementazione della dinamica dei processi, è comunque interessante approfondire gli aspetti del concetto di evento per la rilevanza che assumerà nella simulazione oltre che nell'implementazione dei processi.

### 2.5.3 Gli eventi

Abbiamo già detto che gli **eventi** descrivono il verificarsi in un certo istante di fatti di interesse dell'azienda. Questa definizione può essere approfondita introducendo una classificazione degli eventi secondo cui gli eventi possono appartenere a tre tipi: eventi di **azione**, eventi **temporali** ed eventi **condizionali** (Tabella 2.8).

<b>Eventi di azione</b>	- accadono quando un attore (può essere un attore umano, un dispositivo automatizzato o un'attività dell'azienda) decide qualcosa o fa un'azione a cui l'azienda deve reagire attivando una determinata attività; ad esempio, "arriva il paziente", "arriva l'ordine", "il paziente è registrato", "lo studente è iscritto", "il triage è terminato", "la visita è finita" e così via; - può essere utile distinguere gli eventi di azione in <b>interni</b> ed <b>esterni</b> , a seconda che l'attore o uno degli attori interessati all'evento sia appartenente o meno all'azienda; gli eventi interni corrispondono sovente al termine di un'attività e all'inizio dell'attività successiva, mentre quelli esterni richiedono una opportuna <b>interfaccia</b> per poter essere rilevati e quindi trattati da
-------------------------	--

	un'attività del processo, come illustrato nel modello di implementazione di Figura 2.25; - agli eventi di azione è generalmente associata un'informazione che può comprendere uno o più oggetti (l'ordine di un cliente, il modulo del paziente, la fattura, la pratica e così via).
<b>Eventi temporali</b>	- accadono quando si verificano determinate scadenze temporali a cui il sistema deve reagire; - possono riguardare <u>istanti determinati</u> ("fine anno", "ore 8.30",...) oppure degli intervalli temporali, come <u>ritardi</u> ("dopo 2 ore") o <u>scadenze</u> ("entro dieci giorni").
<b>Eventi condizionali</b>	- sono riferiti ad attività che "interrogano" costantemente l'ambiente; ad esempio un sistema di sicurezza può generare eventi come "porta di sicurezza aperta", "allarme fumo attivato", e così via; - anche a questo tipo di eventi sono in genere associate delle informazioni (ad esempio il numero della stanza).

Tabella 2.8: Tipologie di evento

Nel caso del Pronto Soccorso non abbiamo eventi condizionali e l'insieme degli eventi di azione e temporali rilevanti è piuttosto semplice (Tabella 2.9).

<b>Eventi di azione</b>	<b>esterni:</b> in input - <u>arrivo</u> del paziente in output - <u>uscita</u> , <u>dimissione</u> , <u>ricovero</u> , <u>decesso</u> del paziente <b>interni:</b> riguardano la conclusione delle attività a cui corrisponde l'abilitazione delle attività successive
<b>Eventi temporali</b>	l'evento timer <i>laboratorio</i> rappresenta il ritardo necessario per ricevere i risultati degli esami prescritti al paziente

Tabella 2.9: Eventi del Pronto Soccorso

Osserviamo che gli eventi esterni in input sono particolarmente importanti perché determinano il carico di lavoro del processo, cioè quante istanze del processo il sistema deve trattare (in un certo tempo). Riprenderemo questa osservazione quando si discuterà la simulazione del modello del sistema in esame.

Per quanto riguarda la fase di implementazione il trattamento degli eventi dipende dall'ambiente in cui avviene l'esecuzione del processo. In un ambiente "manuale" sovente sono gli attori stessi che gestiscono l'avanzamento del processo, ad esempio è il paziente che si sposta dalla registrazione al settore triage con il suo modulo e abilita l'attività Eseguire triage "consegnando" il modulo all'infermiere di triage. In un ambiente automatizzato la "pratica" del paziente (interamente digitalizzata) viene spostata dalla stazione di lavoro dell'impiegato a quella dell'infermiere dal motore di workflow (utilizzando la rete interna dell'ospedale); quando il paziente arriva fisicamente nel settore di triage l'infermiere può "aprire" direttamente sulla propria stazione di lavoro il modulo del paziente e provvedere a inserire le informazioni di sua competenza.

## Bibliografia

- Object Management Group, Business Process Model and Notation (BPMN),

<http://www.omg.org/spec/BPMN/2.0.2>, December 2013.

- Object Management Group, Business Motivation Model (BMM), <http://www.omg.org/spec/BMM/1.2>, May 2014.
- Object Management Group, Unified Modeling Language™ (UML), <http://www.omg.org/spec/UML/2.4.1>, August 2011.
- D.M. Bridgeland, R. Zahavi, Business Modeling: A Practical Guide to Realizing Business Value, Morgan Kaufmann, 2008.
- J. Rumbaugh, I. Jacobson, G.Booch, Unified Modeling Language Reference Manual, Addison-Wesley, 2004.
- T. Allweyer, BPMN 2.0: Introduction to the Standard for Business Process Modeling, Herstellung und Verlag, 2010.

## 3. Il linguaggio BPMN

Come già accennato precedentemente per la descrizione dei processi aziendali useremo il linguaggio BPMN. Lo scopo primario dello sviluppo dello standard BPMN è quello di fornire una notazione che sia facilmente comprensibile da tutti gli utenti di business, sia da parte degli analisti che hanno il compito di elaborare i diagrammi relativi ai processi, sia da parte degli sviluppatori che devono mettere a punto la tecnologia che realizzerà tali processi sulla rete aziendale, sia da parte dei responsabili dei diversi processi i quali avranno il compito di gestirli e controllarne il funzionamento.

La caratteristica fondamentale per cercare di soddisfare questi requisiti è la forma grafica di BPMN, tale forma è una evoluzione dei comuni linguaggi usati per specificare i diagrammi di flusso (flowchart) e quindi i diagrammi di processo BPMN hanno un aspetto "amichevole". La notazione diventa complessa quando è necessario affrontare le particolarità di processi relativi a domini diversificati, in cui la semantica del sistema richiede sempre più elementi del linguaggio. E' comunque opinione comune che degli oltre 100 elementi del linguaggio solo un quinto, cioè una ventina, è sufficiente a modellare la grande maggioranza dei processi del mondo reale.

Un altro obiettivo di BPMN, è quello di permettere una implementazione diretta su opportuni *esecutori* (motori di workflow) oppure di facilitare la traduzione dei modelli in un linguaggio eseguibile, come il BPEL (Business Process Execution Language). Poiché questo testo non considera l'implementazione dei processi questo obiettivo non sarà ulteriormente approfondito.

### 3.1 Componenti Fondamentali

Per introdurre il linguaggio riprendiamo l'esempio del Pronto Soccorso (PS):

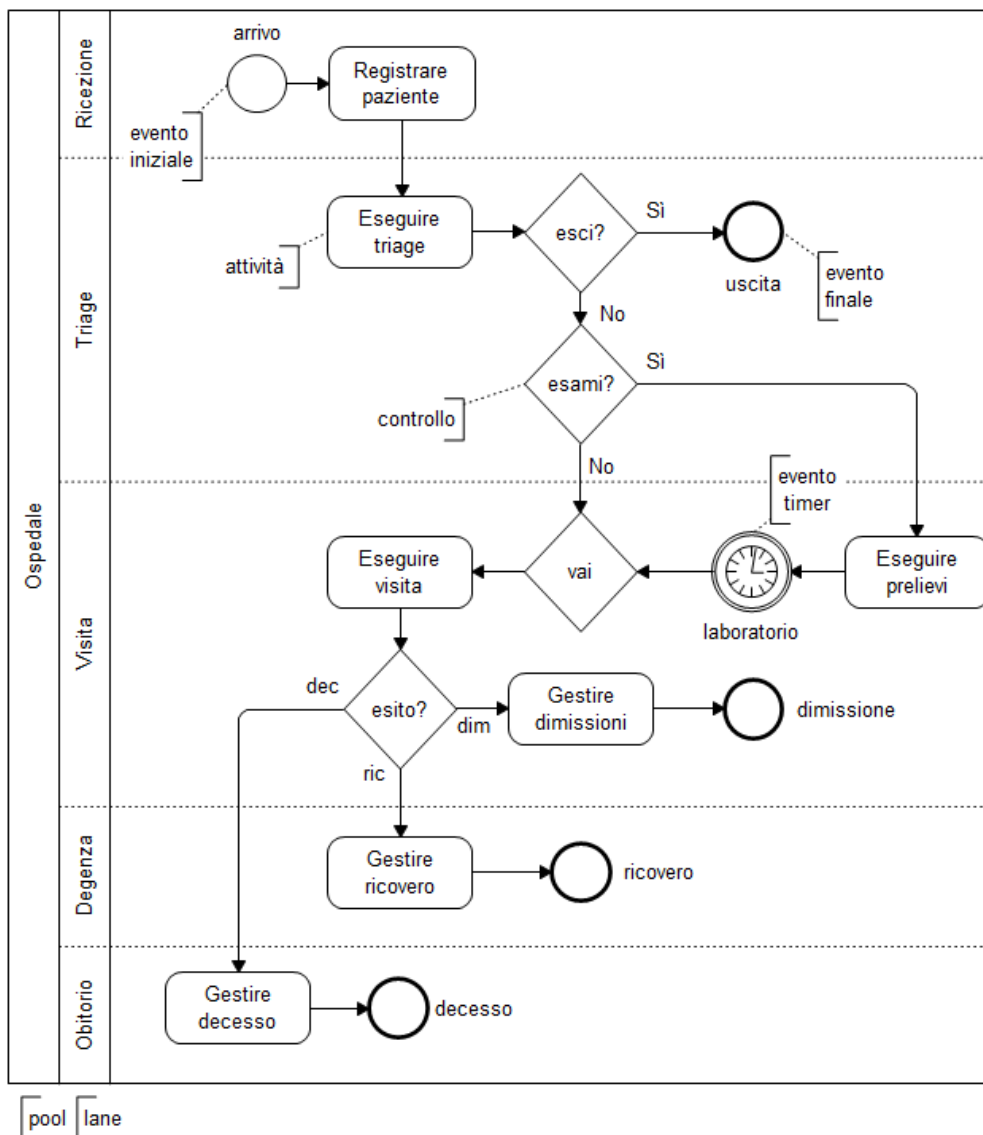


Figura 3.1: Diagramma di processo - Gestire Pronto Soccorso

L'intero processo PS è contenuto in un *pool*: Ospedale. Il pool, o partecipante, è il contenitore di un processo "interno" alla struttura a cui il nome del pool fa riferimento. Osserviamo che se consideriamo un solo processo non è necessario introdurre un pool: più pool sono necessari quando ci sono più partecipanti e bisogna modellare le comunicazioni fra gli stessi (prenderemo in considerazione successivamente questo caso).

Il pool Ospedale è partizionato in cinque *lane* corrispondenti alle strutture (unità organizzative o posti di lavoro) interessate dal Pronto Soccorso. Pool e lane sono anche chiamate **corsie**, nome che useremo quando non c'è la necessità di precisare meglio il partecipante. Le corsie possono far riferimento anche a **ruoli** aziendali, ad esempio la corsia Visita avrebbe potuto anche essere chiamata Medico; in ogni caso se un'attività è inserita in una corsia vuol dire che è eseguita da una risorsa gestita dall'unità organizzativa a cui il nome della corsia fa riferimento.

Il processo PS inizia con l'evento arrivo che esprime il verificarsi (in un certo istante) del fatto: un paziente arriva al Pronto Soccorso. Nel processo sono riportati quattro eventi finali: uscita, dimissione, ricovero, decesso, corrispondenti ai quattro possibili modi in cui il processo può terminare. Osserviamo che potrebbero esserci più eventi iniziali (nella

grande maggioranza dei casi è uno solo) o anche nessuno (in tal caso l'attività senza archi entranti sarà l'attività iniziale), così come potrebbero non esserci eventi finali, in tal caso le attività senza archi uscenti saranno le attività finali. Comunque la pratica migliore è la seguente (PM.1):

<b>PM.1</b>	In ogni processo interno ad un pool inserire almeno un evento iniziale e almeno un evento finale
-------------	--

Le attività del modello sono rappresentate mediante rettangoli a spigoli arrotondati con un nome che dovrebbe essere correlato al **cosa** l'attività dovrebbe fare e non al **come**. Riguardo ai nomi delle attività è utile tener presente la pratica PM.2:

<b>PM.2</b>	Per i <b>nomi</b> di attività usare una forma del tipo: <b>verbo + oggetto</b> dove il <b>verbo</b> può essere all'infinito (preferibilmente) o all'imperativo, l' <b>oggetto</b> deve riferirsi ad un'entità specifica del contesto in cui si opera (ad esempio un risultato che l'attività trasmette alle attività successive) e la forma deve esprimere il raggiungimento di un certo <u>obiettivo</u> o la realizzazione di un certo <u>risultato</u> al termine dell'attività. Inoltre i nomi non dovrebbero contenere congiunzioni (e o) e non dovrebbero contenere dettagli.
-------------	---

Le attività del processo PS seguono questa convenzione, per esempio al termine dell'attività Registrare paziente la registrazione è stata fatta; non sarebbe stato ragionevole chiamare l'attività "Registrazione" (manca la parte verbale), "Registrare" (manca l'oggetto) o "Inserire i dati del richiedente nel modulo" (troppo dettagliata), se è utile avere dei dettagli questi si possono inserire come commenti associati alla forma grafica dell'attività.

Se un'attività è seguita da un controllo, è da intendersi che la decisione sull'arco lungo cui proseguire è presa sulla base di una valutazione effettuata durante l'esecuzione dell'attività, per cui nel controllo si suppone che non vengano eseguite delle azioni. Può anche essere utile inserire esplicitamente prima del controllo un'attività di verifica di certe situazioni per rendere più esplicativo il diagramma.

Il controllo diventa quindi un semplice instradamento del flusso e richiede un tempo nullo, mentre le attività (in quanto insiemi di azioni) richiedono un tempo finito. Il nome del controllo può essere espresso come una forma interrogativa legata a degli obiettivi dell'attività; ad esempio, per l'attività Eseguire triage potremmo avere la descrizione illustrata in Tabella 3.1.

<b>Obiettivo</b>	<b>Cosa fare al termine dell'esecuzione</b>
O1: eseguire il triage del paziente	continua
O2: verificare se il paziente vuole uscire	«esci?» Si → vai ad <u>uscita</u> «esci?» No → continua
O3: decidere se sono necessari degli esami	«esami?» Si → vai ad <u>Eseguire prelievi</u> «esami?» No → vai ad <u>Eseguire visita</u>

Tabella 3.1: Descrizione dell'attività Eseguire triage

I **controlli** sono rappresentati graficamente come *rombi* e regolano l'instradamento delle istanze di processo; in particolare chiameremo **split** i controlli che separano il flusso di esecuzione in più cammini e **join** quelli che provvedono a ricongiungere i cammini stessi.

Nel processo PS abbiamo tre split, «esci?», «esami?» ed «esito?», ed un join «vai»; osserviamo che generalmente ai join non viene assegnato un nome in quanto il loro significato è evidente.

Gli split come quelli specificati nell'esempio sono di tipo XOR (exclusive or), cioè si segue un cammino o l'altro (*mutua esclusione*), è utile quindi nominare gli archi che si dipartono dallo split con le clausole che diventano vere o false dopo che lo split è stato valutato, ad esempio dopo lo split «esito?» se è vera la clausola dim (il paziente deve essere dimesso) si va all'attività Gestire dimissioni, se è vera la clausola ric si va all'attività Gestire ricovero, e così via. Questa interpretazione presuppone che nell'attività Eseguire visita siano predisposte delle azioni per permettere la valutazione delle clausole. Come abbiamo visto, nel modello di implementazione l'attività Eseguire visita sarà implementata mediante una procedura che, nel caso in esame, potrebbe contenere delle istruzioni del tipo:

.....

**logic** dim, ric, dec;

**if** «paziente da dimettere» **then** dim ← *true* **else** dim ← *false*;

**if** «paziente da ricoverare» **then** ric ← *true* **else** ric ← *false*;

**if** «paziente deceduto» **then** dec ← *true* **else** dec ← *false*;

.....

A questo punto nello split «esito?» la clausola dim può assumere la forma:

.....

**if** dim = *true* **then** «vai all'attività Gestire dimissioni»

.....

e analogamente per le altre clausole.

Nel modello è possibile introdurre delle **note** per specificare meglio una forma qualsiasi. Le note sono costituite da un testo inserito in una parentesi quadra aperta che è connessa alla forma da specificare mediante una linea tratteggiata. Nell'esempio di Figura 3.1 le note sono state usate per indicare i concetti fondamentali del linguaggio BPMN; nella figura le forme sono a sfondo bianco ma con lo strumento Process2013 lo sfondo può essere variamente tratteggiato e ogni forma può essere colorata, l'uso del colore non è specificato nella notazione BPMN (e non sarà usato in questo testo) ma può essere utile per accrescere la significatività del modello (ad esempio, attività critiche possono essere colorate in rosso).

## 3.2 Rappresentare i Controlli

I controlli sono usati per specificare lo split (separazione) e il join (ricongiungimento) dei cammini di flusso. Oltre ai controlli-XOR presenti nel processo PS in BPMN è possibile specificare altri 3 tipi di controllo: controlli-AND, controlli-OR e controlli complessi.

Come illustrato in precedenza la logica degli split-XOR richiede che una e solo una delle clausole di decisione risultino vere: questo vincolo deve essere assolutamente verificato. Le clausole sono in genere riportate sopra gli archi in uscita (che sono chiamati in tal caso archi condizionali) ed è anche possibile introdurre una clausola di default che viene selezionata se tutte le altre clausole sono false (l'arco corrispondente è sbarrato). Ad esempio, i due diagrammi in



Figura 3.2 sono equivalenti e gestiscono l'instradamento di ordini il cui totale è inferiore a 100 €, compreso fra 100 € e 1000 € o è superiore a 1000 €.

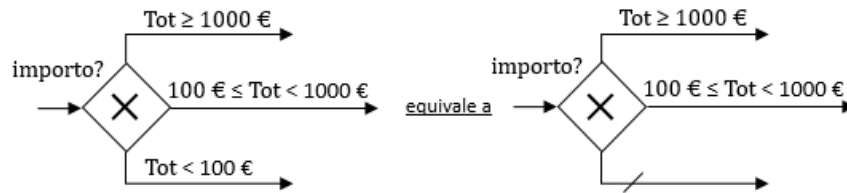


Figura 3.2: Semantica dei controlli: split-XOR

Come si può osservare, i tre intervalli sono mutualmente esclusivi e completi, cioè la loro unione ricopre tutti i possibili valori della variabile Tot (totale dell'ordine); possiamo esprimere questo fatto dicendo che i tre intervalli costituiscono una **partizione** del dominio di valori possibili per la variabile.

In BPMN i controlli di tipo XOR, come quelli incontrati finora, sono rappresentati con rombi vuoti o con un nome all'interno, è possibile anche inserire il simbolo X, in tal caso il nome del controllo può essere posto a fianco del simbolo.

Tramite uno split-AND il flusso di controllo viene separato in due o più flussi paralleli che devono essere riuniti mediante un join-AND. Le attività che si trovano sui cammini paralleli possono essere eseguite in un ordine qualsiasi o anche simultaneamente, questo implica che esse siano indipendenti, ad esempio gli oggetti prodotti da una delle attività non possono coincidere con quelli in ingresso all'altra. Graficamente split e join di tipo AND sono rappresentati inserendo un simbolo + nel rombo.

Poiché i tempi di esecuzione delle attività possono essere anche radicalmente differenti il join-AND che ricompone i cammini paralleli opera come un punto di **sincronizzazione**, cioè non si può proseguire nel processo se non quando entrambi i cammini sono terminati.

Questa semantica di controllo può essere specificata rappresentando l'istanza del processo, ad esempio quella relativa a un certo paziente, mediante un **token** (gettone) che percorre all'interno del modello un certo cammino (riprenderemo nel Capitolo 4 questo concetto) e che attiva le attività man mano che le incontra. In questo contesto un token che raggiunge uno split-AND viene duplicato in modo che ciascuno dei duplicati possa percorrere i cammini paralleli: questi cammini risultano quindi entrambi attivati (e questo implica l'esecuzione di tutte le attività che si incontrano lungo i cammini); al termine dei cammini i due token raggiungono il join-AND che provvede a sincronizzarli e quindi riunificarli in un solo token che procede. La Figura 3.3 mostra l'evoluzione dei token.

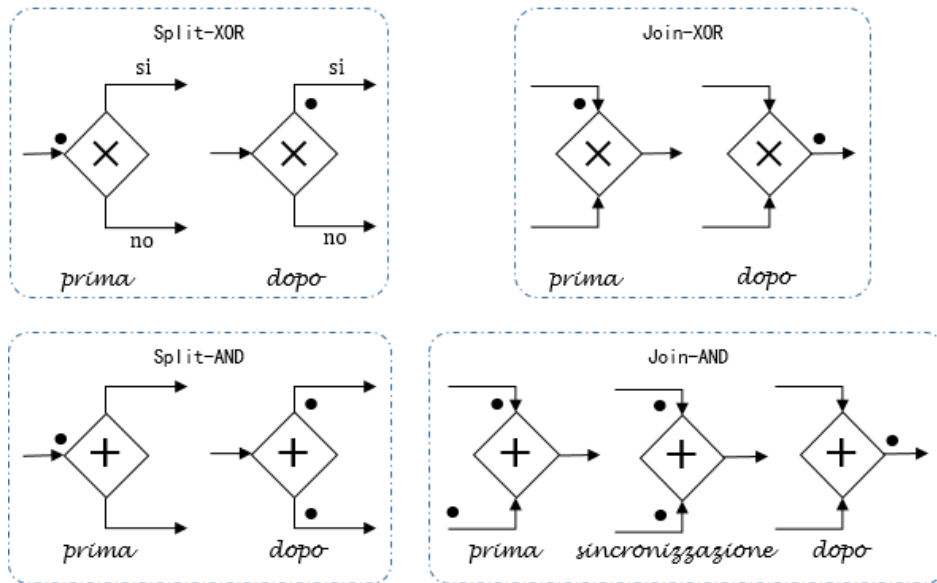



Figura 3.3: Semantica dei controlli: evoluzione dei token

Il terzo tipo dei controlli BPMN permette di specificare una semantica OR (inclusive or); se  $n$  sono gli archi uscenti da uno split-OR può essere selezionato un numero qualsiasi di archi da 0 a  $n$ . Osserviamo che benché logicamente sia possibile specificare 0 archi in uscita da uno split-OR questo caso non ha senso perché porterebbe ad un blocco del processo.

I controlli OR sono rappresentati graficamente con un cerchio  all'interno del rombo.

Situazioni in cui è utile usare uno split-OR non sono molto frequenti e in genere devono essere trattate con molta prudenza perché possono portare facilmente a seri problemi di interpretazione. Consideriamo ad esempio il caso seguente:

In un ambulatorio dietetico il medico dopo una visita può prescrivere al paziente delle vitamine, degli integratori, degli antiossidanti o una qualsiasi combinazione di queste tre classi di medicinali.

Il diagramma corrispondente a questo frammento di processo può essere quello indicato in Figura 3.4.

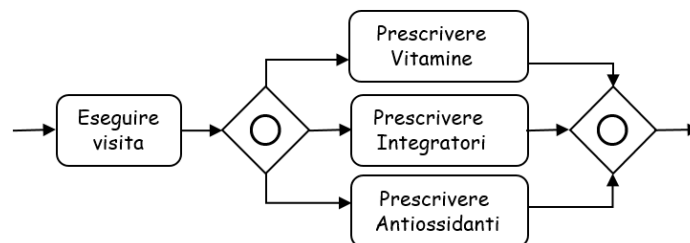


Figura 3.4: Semantica dei controlli OR

Osserviamo che il diagramma è apparentemente semplice ma effettivamente rappresenta 7 possibili cammini di esecuzione (il caso con 0 cammini non è considerato):

<b>C1</b>	<u>Prescrivere Vitamine</u>
<b>C2</b>	<u>Prescrivere Integratori</u>
<b>C3</b>	<u>Prescrivere Antiossidanti</u>
<b>C4</b>	<u>Prescrivere Vitamine + Prescrivere Integratori</u>
<b>C5</b>	<u>Prescrivere Vitamine + Prescrivere Antiossidanti</u>
<b>C6</b>	<u>Prescrivere Integratori + Prescrivere Antiossidanti</u>
<b>C7</b>	<u>Prescrivere Vitamine + Prescrivere Integratori + Prescrivere Antiossidanti</u>

Il controllo join-OR deve operare quindi come un controllo join-AND rispetto ai cammini C4, C5, C6 e C7, e come un controllo join-XOR rispetto ai cammini C1, C2 e C3; questo fatto fa comprendere che la semantica dei controlli può assumere una certa complessità da tener presente in fase di progetto.

Il modo più semplice di utilizzare i controlli OR è quello di utilizzare degli archi condizionali, ad esempio il caso:

Un'azienda spedisce i prodotti ai suoi clienti a partire da tre magazzini centrali: A, B e C, e può accadere che in un ordine siano richiesti prodotti da uno o più di questi magazzini.

La situazione può essere rappresentata con il diagramma di Figura 3.5 in cui il predicato  $loc=\alpha$  (con  $\alpha$  che può assumere i valori A, B o C) diventa vero se nell'ordine sono indicati prodotti localizzati nel magazzino specificato.

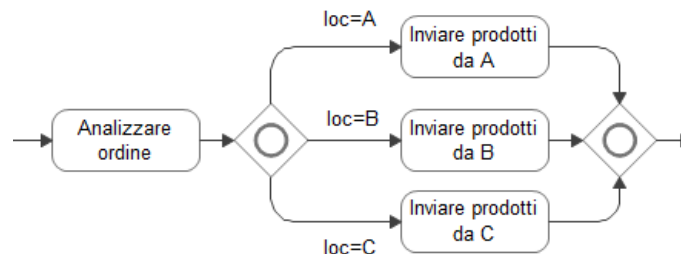


Figura 3.5: Semantica dei controlli OR

Possono quindi verificarsi tutti i casi previsti per uno split-OR a seconda che uno o più dei predicati assuma il valore vero.

I controlli possono annidarsi l'uno dentro l'altro ma in tal caso è necessario procedere con molta attenzione perché si possono generare dei diagrammi scorretti; ad esempio, se su uno o più cammini di Figura 3.5 si inseriscono controlli di tipo qualsiasi possono facilmente verificarsi condizioni di blocco del processo come illustrato in Figura 3.6 in cui se si attivano gli archi 1,2,3,4 il processo si blocca perché il join-OR finale attende due token per poter proseguire.

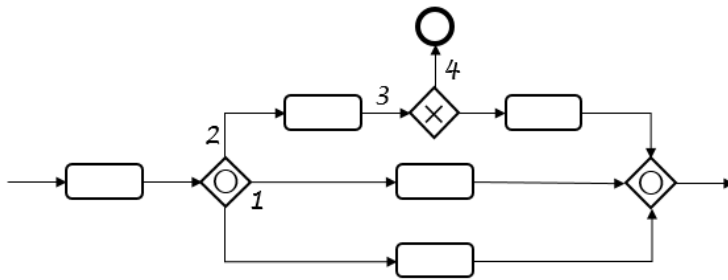


Figura 3.6: Esempio di blocco

Ovviamente possono esserci controlli annidati che non presentano problemi logici come i blocchi, in ogni caso è sempre utile cercare di strutturare questo tipo di diagrammi facendo in modo che ad ogni controllo di split corrisponda uno di join dello stesso tipo. Possiamo anche dire che il comportamento dei join deve essere “speculare” a quello degli split corrispondenti.

I diagrammi costruiti seguendo queste indicazioni sono strutturati e quindi più facili da interpretare applicando le regole di evoluzione dei token illustrate in Figura 3.3. In merito a queste regole osserviamo che generalmente sono attive diverse istanze di uno stesso processo, quindi contemporaneamente fluiscono molti token nel diagramma ed è necessario che ogni token abbia un identificatore primario a cui aggiungere un secondario quando il token si deve sdoppiare incontrando split di tipo AND e OR: in tal modo i join corrispondenti possono “assorbire” le copie opportune.

Il linguaggio BPMN permette una notazione abbreviata per i controlli introducendo la possibilità di avere più archi in uscita da e in ingresso a un’attività; in Figura 3.7 è illustrata l’interpretazione di questa notazione.

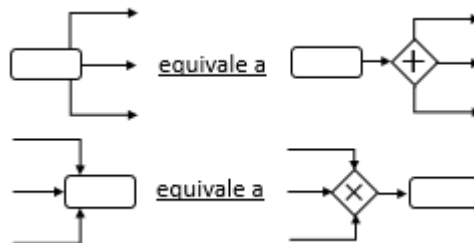


Figura 3.7: Archi multipli

Come risulta evidente si viene a creare un’asimmetria poco intuitiva fra come sono trattati gli archi multipli in ingresso (può arrivare un solo token = controllo XOR) e in uscita (tutti gli archi in uscita ricevono un token = controllo AND) e questo fatto può creare confusione in progettisti non esperti.

Sovente la semantica in ingresso/uscita di un’attività è di tipo AND-AND o XOR-XOR: in questi casi una rappresentazione corretta, ottenuta esplicitando i controlli, è mostrata in Figura 3.8.

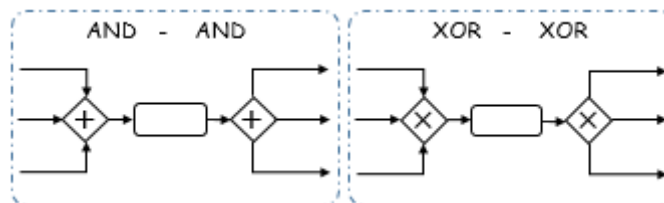


Figura 3.8: Semantiche di archi multipli

L'utilizzo della notazione abbreviata e i controlli annidati costituiscono delle potenziali fonti di errori concettuali, per cui potrebbe essere conveniente adottare le seguenti "buone" pratiche:

<b>PM. 3</b>	Separare la gestione del controllo dalle attività facendo in modo che ogni attività abbia un solo arco in ingresso e uno in uscita.
--------------	---

<b>PM. 4</b>	I controlli di tipo split devono avere un solo arco in ingresso e quelli di tipo join un solo arco in uscita. Ad ogni split deve corrispondere un join dello stesso tipo con un comportamento "speculare" per quanto riguarda il trattamento dei token.
--------------	--

Un ultimo tipo di controllo è costituito dai **controlli complessi**, controlli applicabili in situazioni in cui gli altri controlli sono difficili da utilizzare. Consideriamo ad esempio la situazione seguente:

Per l'assunzione di un ricercatore junior in un Centro Ricerche è richiesto il parere di due direttori di ricerca e di un professore universitario. La decisione può comunque essere presa dalla direzione non appena arrivano due pareri qualsiasi.
--

La situazione può essere modellata, come illustrato nel diagramma di processo di Figura 3.9, con un join-complesso (rappresentato con il simbolo \* inserito nel rombo).

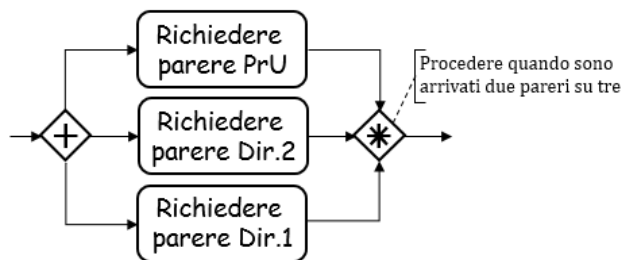


Figura 3.9: Controlli complessi

La regola per la gestione dei token in arrivo può essere qualsiasi e in genere viene specificata in una nota associata al join-complesso. Come mostra l'esempio l'uso di controlli complessi rende molto "leggibile" il processo, avremmo potuto modellare il caso anche con i controlli "tradizionali" (XOR, AND, OR) ma il risultato sarebbe stato un diagramma molto intricato che avrebbe "oscurato" la semantica del processo.

Osserviamo che è possibile inserire in un diagramma di processo degli split-complessi (in cui la regola di assegnazione dei token agli archi in uscita può essere qualsiasi) ma non approfondiremo l'argomento poiché questo tipo di controlli gioca un ruolo in genere poco rilevante nella modellazione dei processi.

### 3.3 Rappresentare le Collaborazioni

In BPMN è possibile rappresentare le interazioni fra processi di più partecipanti, ad esempio fra aziende diverse (interazioni Business-to-Business o B2B), fra clienti e aziende (Business-to-Consumer o B2C) e così via. Le interazioni

possono essere descritte a due diversi livelli di dettaglio come **collaborazioni** o come **coreografie**.

Le collaborazioni sono modellate rappresentando l'interazione fra due o più partecipanti come uno scambio di **messaggi**. Per comprendere questi nuovi concetti consideriamo (Figura 3.10) un esempio molto semplificato in cui due processi interni a due partecipanti (pool in BPMN), Candidato e Azienda, interagiscono tra loro scambiandosi dei messaggi. Un candidato compila un *Modulo* di assunzione che sottomette all'azienda, questa lo riceve e invia un *Invito* al candidato per un'intervista; l'intervista viene eseguita inviando un questionario con delle *Domande* e ricevendo delle *Risposte* dal candidato, quindi l'azienda invia il *Contratto* al candidato.

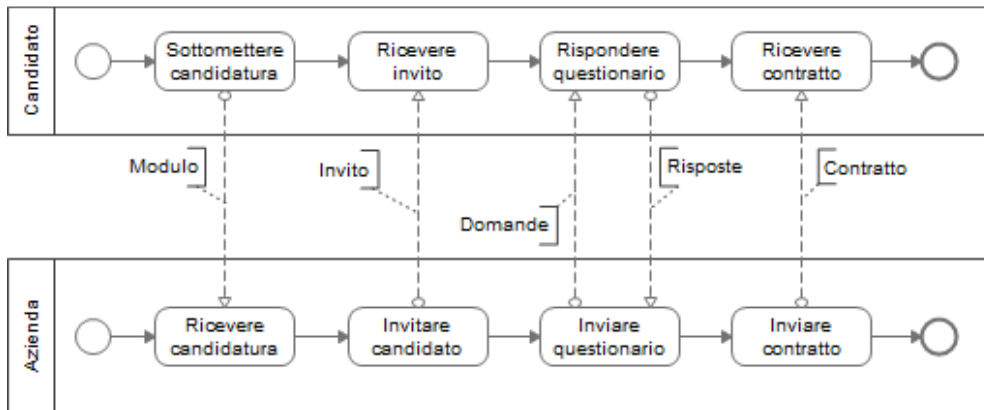


Figura 3.10: Esempio di collaborazione

Come mostra la Figura 3.10 i messaggi sono simili agli archi di controllo del flusso ma graficamente sono tratteggiati ed hanno un circoletto ad un estremo per individuare chi **invia** il messaggio e una freccia all'altro per specificare chi lo **riceve**.

In BPMN i messaggi sono usati *esclusivamente* per descrivere le interazioni fra i pool e *non* possono comparire all'interno dei pool, simmetricamente gli archi di controllo possono comparire solo all'interno dei pool e *non* possono descrivere le interazioni fra un pool e l'altro.

L'idea che sta alla base di questa scelta è che nei pool sono descritti i processi *interni* del partecipante (ad esempio, un'azienda) e quindi *non* devono necessariamente essere messi a disposizione del mondo esterno, come accadrebbe se dall'esterno si potesse accedere ai processi interni inserendosi nei flussi di controllo. In tal modo i processi sono *incapsulati* all'interno dei pool e interagiscono con l'esterno solo attraverso l'*interfaccia* definita tramite un insieme di messaggi. Per enfatizzare questa scelta è possibile eliminare il processo interno del pool e in tal caso i messaggi si fermano sul contorno del pool stesso, come mostrato in Figura 3.11; diremo in tal caso che il pool è *occultato*.

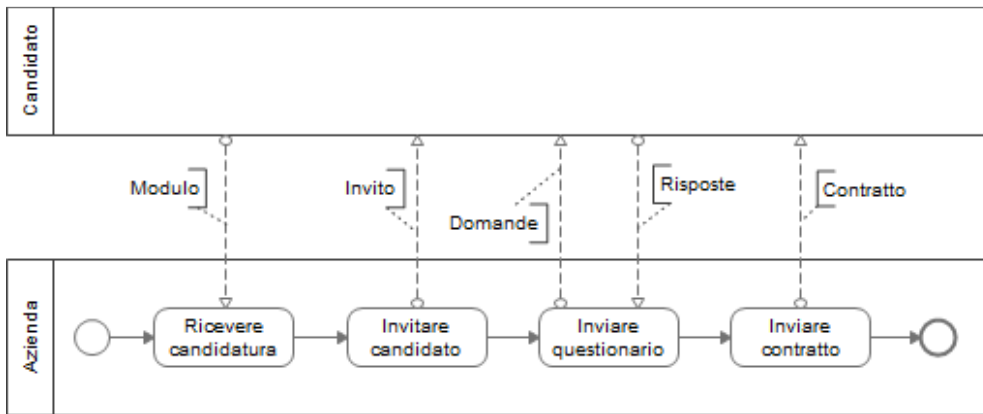


Figura 3.11: Collaborazione con un pool occultato

Questo diagramma rappresenta il punto di vista dell’Azienda, ovviamente potremmo considerare il punto di vista del Candidato e anche occultare entrambi i pool, ottenendo il diagramma di Figura 3.12.

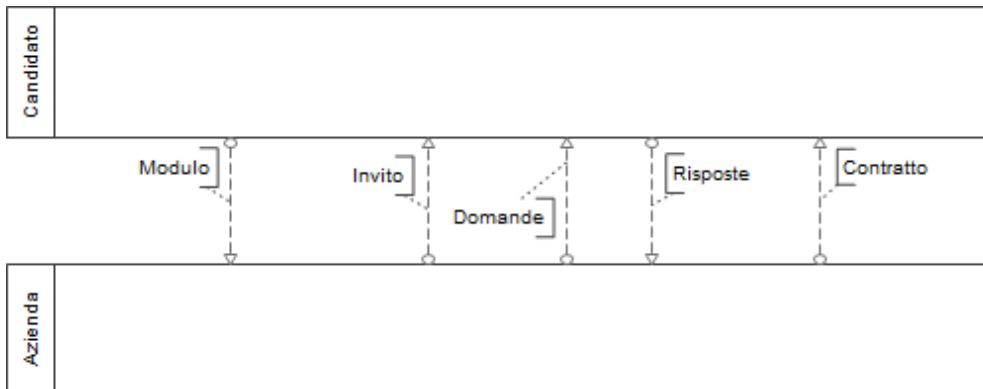


Figura 3.12: Collaborazione fra pool occultati

Il problema nella Figura 3.12 è che non è possibile capire a questo punto quali siano le dipendenze fra i messaggi, ad esempio non sappiamo se il contratto deve essere inviato dall’azienda prima o dopo l’arrivo del modulo compilato dal candidato. Queste dipendenze possono essere modellate specificando una **coreografia** fra i messaggi, cioè un diagramma che esplicita le relazioni fra i messaggi stessi. Considerando la natura introduttiva di questo testo la rappresentazione delle coreografie, del resto poco rilevante rispetto alla modellazione dei processi interni dell’azienda, non sarà presa in considerazione.

### 3.4 Rappresentare gli Eventi

Come abbiamo già visto nel Capitolo precedente un evento esprime il fatto che qualcosa di interesse dell’azienda è accaduto. Per modellare gli eventi due aspetti devono essere tenuti in considerazione: la causa (trigger) dell’evento ed il suo effetto sul processo.

Riguardo alle cause di un evento abbiamo già introdotto tre tipologie di eventi: eventi di azione, eventi temporali ed eventi condizionali. In questo capitolo introdurremo altri tipi come, ad esempio, gli errori.

Gli eventi possono essere **iniziali** (cerchi sottili ○), **intermedi** (cerchi doppi ◎) e **finali** (cerchi spessi ⊙). Nel diagramma di Figura 3.1 compaiono esempi di eventi appartenenti a tutti questi tipi ma un uso caratteristico degli eventi fa riferimento alla gestione dei messaggi, infatti un messaggio può essere visto come una comunicazione fra un attore che ad un certo istante *invia* (evento invio) e un altro attore che *riceve* il messaggio stesso (evento ricezione). Per illustrare questo uso riprendiamo l'esempio di collaborazione di Figura 3.10 riformulandolo con la gestione esplicita dei messaggi che in esso compaiono, come illustrato in Figura 3.13 (come mostra l'esempio è possibile rappresentare le corsie anche in verticale).

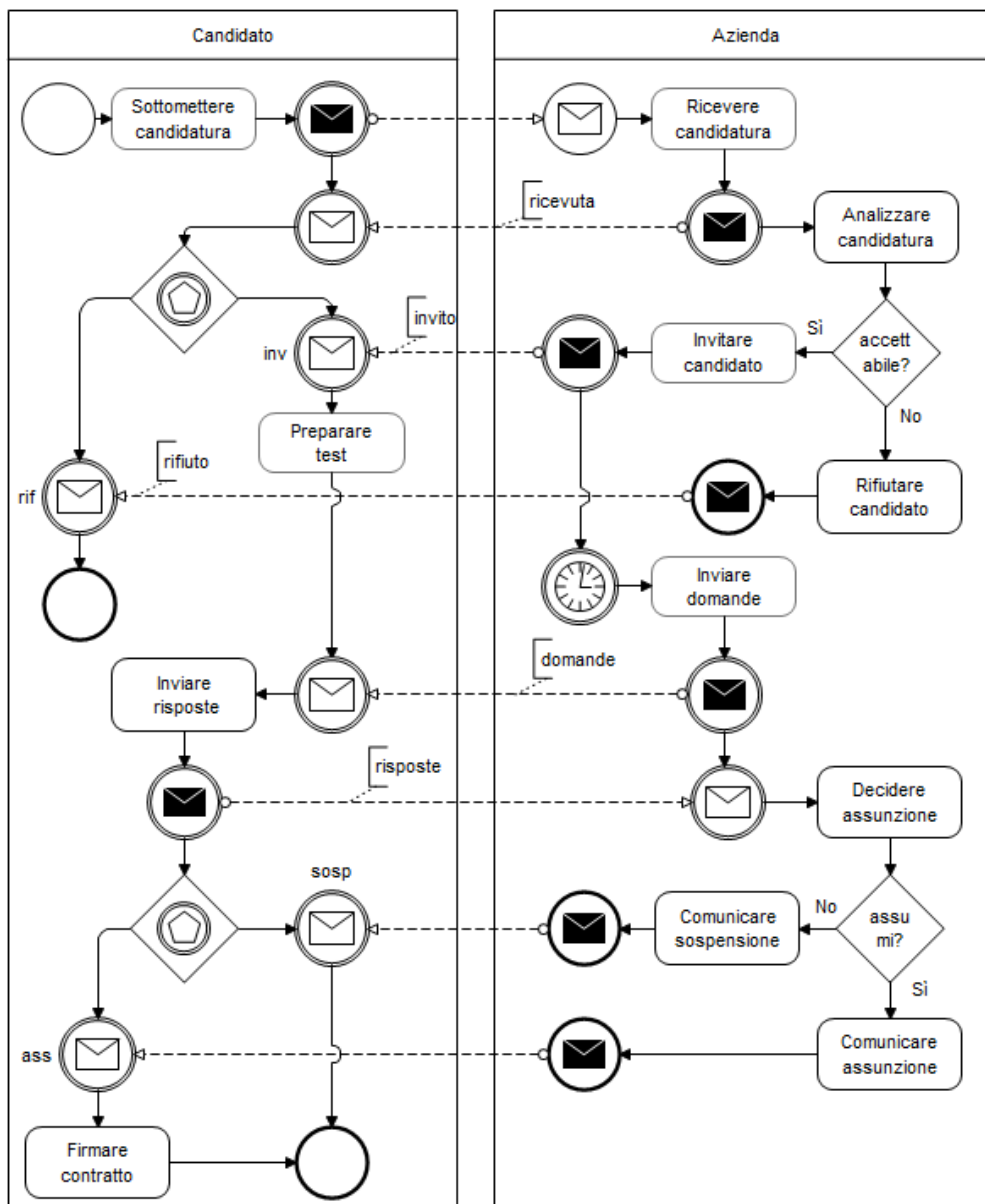



Figura 3.13: Diagramma del processo Esaminare candidato


Nel diagramma un messaggio viene rappresentato come un arco di collegamento fra un evento che *invia* (cerchio con busta nera all'interno) e un evento che *riceve* (cerchio con busta bianca all'interno).

Nell'esempio compaiono anche dei controlli gestiti tramite degli eventi e rappresentati graficamente mediante il



simbolo  split-XOR basato su eventi (di tipo intermedio); nel diagramma il Candidato può ricevere un rifiuto (evento *rif* - in tal caso termina il processo) oppure un invito (evento *inv* - in tal caso prepara il test), e considerazioni analoghe valgono successivamente per la sospensione o l'assunzione del candidato (eventi *sosp* e *ass*).

E' possibile anche introdurre controlli di tipo OR basate su eventi ma l'interesse per questo tipo di strutture è decisamente scarso.

Oltre all'evento iniziale generico  il linguaggio BPMN permette diversi tipi di eventi iniziali che sono illustrati in Figura 3.14.

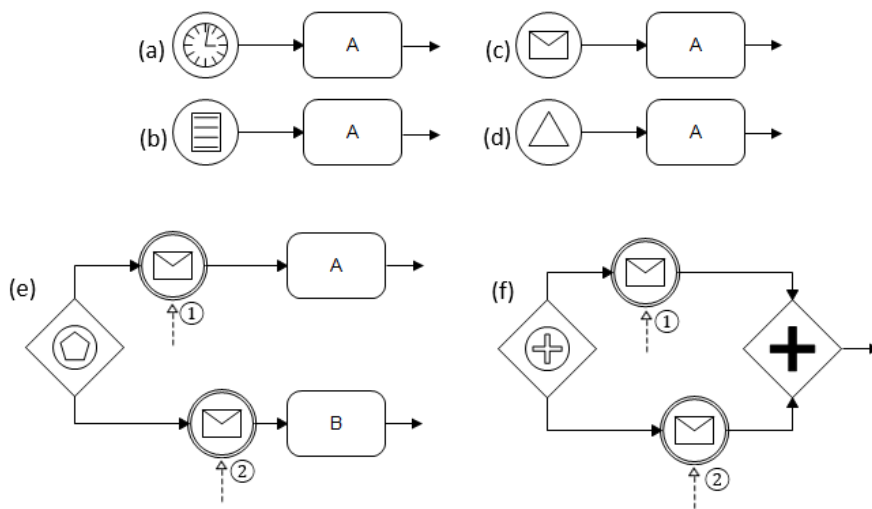


Figura 3.14: Eventi iniziali

Il significato dei diagrammi in Figura 3.14 è il seguente:

- (a) evento iniziale temporale: può specificare una certa data (ad esempio il 31/03/1993 alle 8 del mattino) oppure impostare un certo ciclo (ad esempio "ogni fine mese", "ogni lunedì alle ore 8" e così via);
- (b) evento iniziale condizionale (definito da una **regola**): si attiva quando è soddisfatta una certa condizione definita dall'analista (ad esempio "basso livello di scorta", "temperatura inferiore a 70°C", "il valore dell'azione è sceso sotto i 10€",...);
- (c) evento iniziale messaggio: il processo è attivato da un **messaggio** proveniente da un partecipante *diverso* dal ricevente (ad esempio "arriva un ordine del cliente") oppure generato nell'ambiente esterno;
- (d) evento iniziale segnale: il processo è attivato da un **segnale** proveniente dallo stesso partecipante o da un qualsiasi altro partecipante. Contrariamente ad un messaggio che è sempre inviato ad un determinato ricevente (in un partecipante *diverso*) un segnale è *pubblico* e può essere "letto" o ricevuto ovunque, anche nello stesso partecipante in cui il segnale è stato generato (ad esempio "è caduto il nodo x della rete") e in più punti diversi;
- (e) evento iniziale **multiplo** di tipo XOR: il processo è attivato dal verificarsi dell'evento ① (e si esegue l'attività A) oppure dell'evento ② (e si esegue l'attività B), ad esempio bisogna revisionare il motore se "scade il tagliando" oppure "si è verificato un incidente";
- (f) evento iniziale **multiplo** di tipo AND: il processo è attivato quando si sono verificati entrambi gli eventi ① e ②, ad esempio l'aereo può decollare se "il check del velivolo è positivo" e "OK dalla torre di controllo".

Osserviamo che in BPMN è possibile introdurre in un diagramma di processo un solo evento iniziale generico ○ (nello strumento Process2013 questa limitazione non esiste) però è possibile introdurre altri tipi di eventi iniziali, da soli o in combinazione tra loro e con quello generico; in tal modo possono essere modellati facilmente i casi (e) e (f).

Per quanto riguarda gli eventi finali accanto all'evento finale generico ○ possono essere presenti altri quattro tipi, illustrati in Figura 3.15.

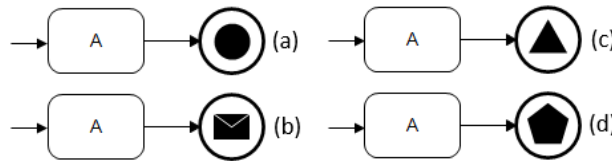


Figura 3.15: Eventi finali

Il significato dei diagrammi in Figura 3.15 è il seguente:

- (a) evento finale terminale: mentre l'evento finale generico ○ chiude solo il proprio cammino di esecuzione (assorbendo il token relativo) ed altri cammini possono continuare ad essere attivi l'evento terminale determina la conclusione non solo del proprio cammino ma dell'intero processo qualunque sia la situazione sugli altri cammini; ad esempio il processo Concessione di un prestito deve terminare se anche una sola delle situazioni "garanzie bancarie OK" e "situazione immobiliare OK" si conclude con un risultato negativo. Se l'evento terminale appare in un sotto-processo si termina il sotto-processo stesso e tutti i suoi sotto-processi ma non il processo a livello superiore;
- (b) evento finale messaggio: come risultato determina l'emissione di un messaggio verso un altro partecipante, ad esempio si avvisa il cliente che "il prodotto è stato spedito";
- (c) evento finale segnale: è analogo al precedente però il segnale è pubblico e chiunque può riceverlo, ad esempio si notifica a tutti una interruzione del servizio;
- (d) evento finale multiplo: è analogo ai precedenti ma adesso la fine del cammino ha diverse conseguenze, ad esempio si "notifica al cliente l'invio della fornitura" e si "invia un rapporto al manager".

Gli eventi intermedi ○ hanno un arco di controllo in ingresso e uno in uscita, e possono comparire ovunque nel processo. L'uso di eventi intermedi all'interno del processo di un partecipante X è giustificato nelle situazioni illustrate in Figura 3.16.

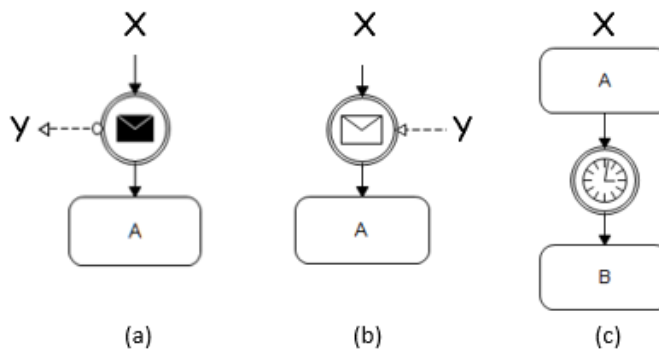






Figura 3.16: Eventi intermedi

- (a) un evento che è rilevante per un altro partecipante Y accade nel processo interno del partecipante X; quindi X invia un messaggio a Y e prosegue con l'attività A;
- (b) il partecipante X entra in attesa e per continuare con l'attività A attende un messaggio dall'altro partecipante Y (X si sincronizza con Y);
- (c) il partecipante X prima di continuare con l'attività B deve tener conto di un evento temporale (ad esempio un ritardo: "5 minuti", "1 giorno", o anche un certo istante di tempo).

Nei casi (a) e (b) di Figura 3.16 possiamo anche considerare eventi intermedi di tipo invia segnale  e ricevi segnale  (o invia multiplo  e ricevi multiplo ) per cui valgono regole analoghe a quelle viste per l'invio e il ricevimento di messaggi.

In Figura 3.17 è illustrato un esempio in cui compaiono diversi dei costrutti che abbiamo esaminato finora, questo processo Ordinare una pizza è un semplice esempio di processo "Business-to-Consumer" (B2C) in cui l'azienda (Pizzeria) e il consumatore (Cliente) sono modellati come due partecipanti e sono stati esplicitati i loro processi interni (orchestrazioni).

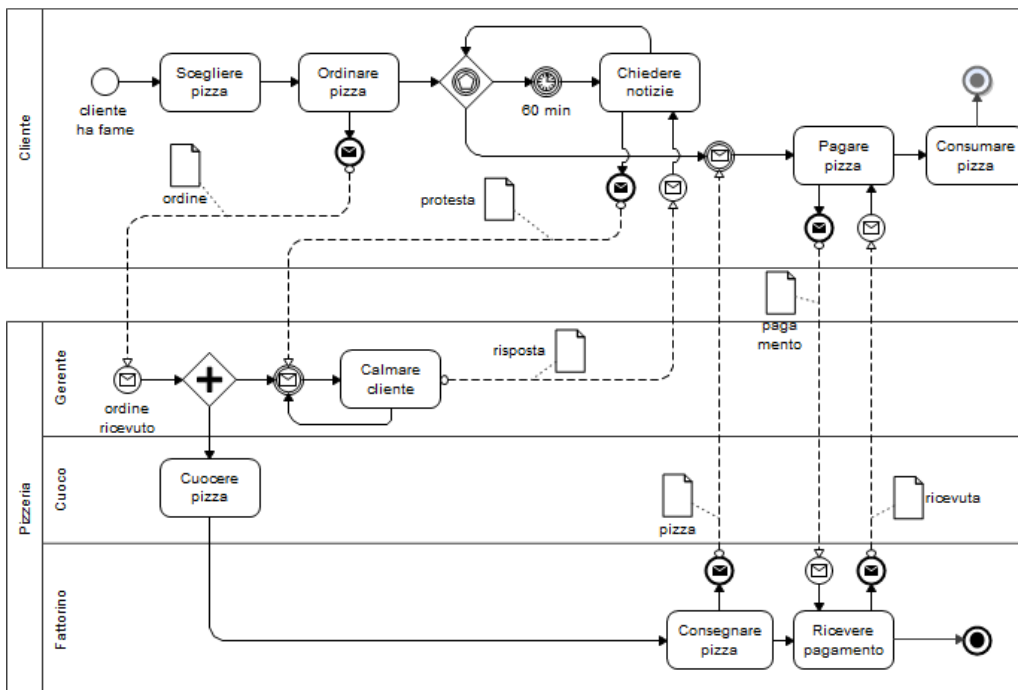


Figura 3.17: Processo Ordinare una pizza

Il cliente dopo aver scelto e ordinato la pizza si pone in attesa di due possibili eventi: 1) arriva la pizza (e quindi la paga e la consuma), 2) è passata un'ora e in tal caso telefona protestando al Gestore della Pizzeria (rimettendosi quindi in attesa).

Per quanto riguarda l'azienda, il Gestore riceve l'ordine del cliente e lo trasmette al Cuoco che prepara la pizza e l'affida al Fattorino per la consegna al cliente, il Fattorino quindi riceve il pagamento e rilascia la ricevuta.

I nodi terminale per entrambi i processi interni hanno il compito di chiudere i processi e rimuovere i token ancora

attivi (relativi al ciclo Calmare cliente in Pizzeria e Chiedere notizie in Cliente).

Ricordiamo che il linguaggio BPMN ammette una notazione compatta per rappresentare i messaggi scambiati fra le attività di due partecipanti distinti. La notazione prevede l'inserimento nell'attività che invia il messaggio e in quella che lo riceve dei simboli ✉ (invio) e ✉ (ricezione). La Figura 3.18 illustra l'equivalenza delle due rappresentazioni. Come risulta evidente dalla figura la rappresentazione compatta riduce il numero di forme nel diagramma di processo senza compromettere la sua comprensibilità.

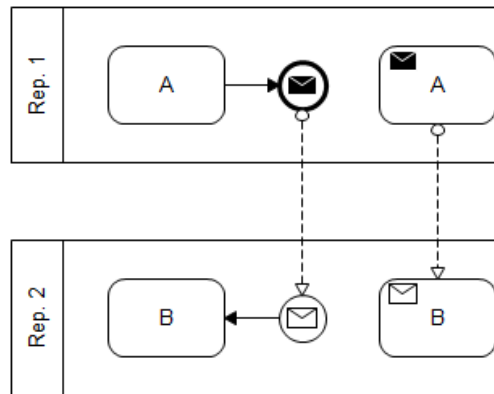


Figura 3.18: Rappresentazioni equivalenti di messaggi

## 3.5 Rappresentare le Attività

In BPMN esistono sostanzialmente due tipi di attività, attività *non ulteriormente dettagliate* che da ora in poi, per brevità, chiameremo semplicemente **attività**, e attività così complesse da richiedere un ulteriore livello di dettaglio che chiameremo **sotto-processi**. A sua volta un sotto-processo può essere rappresentato in forma *compressa* o *espansa*.

Per illustrare questi concetti riprendiamo l'esempio del Pronto Soccorso e supponiamo di descrivere più in dettaglio l'attività Eseguire triage che, anche nella realtà, è piuttosto complessa. La Figura 3.19 illustra la trasformazione dell'attività in un sotto-processo compresso.

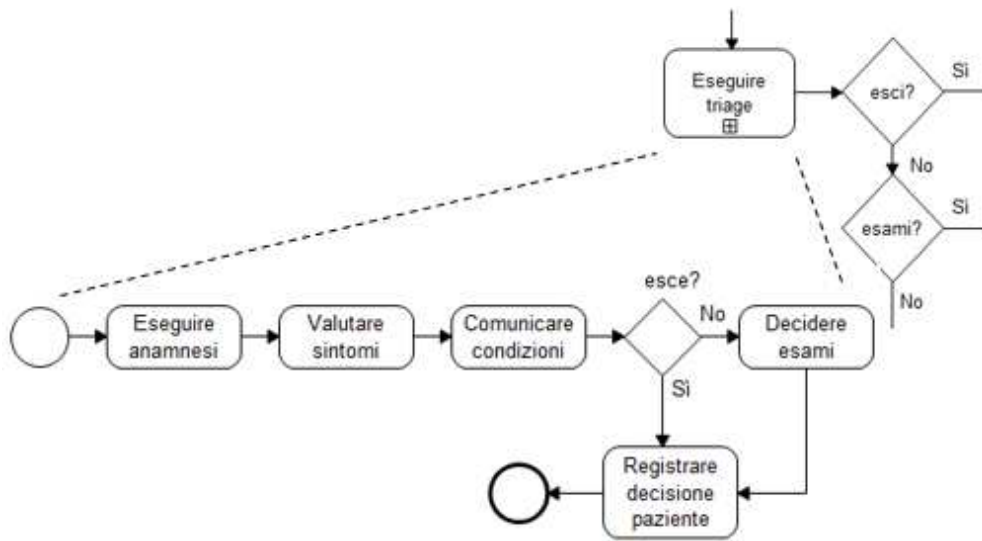


Figura 3.19: Sotto-processo Eseguire triage

A livello superiore la forma grafica del sotto-processo Eseguire triage è caratterizzata dal simbolo inserito inferiormente al ☒ rettangolo. Quando il flusso di controllo raggiunge la forma il controllo viene trasferito all'evento iniziale del sotto-processo che è specificato separatamente dal processo principale.

Poiché Eseguire triage è seguita da due controlli il sotto-processo deve essere interpretato nel seguente modo (o in un modo equivalente):

a) si creano due variabili di tipo logico (variabili che possono assumere solo due valori, *true* o *false*), ad esempio:

**logic uscita, esami;**

(queste variabili, così come ogni altro dato utile al processo, sono da considerarsi dati *globali*, cioè informazioni memorizzate in una *memoria di lavoro* e visibili da ogni punto del processo stesso);

b) l'infermiere che esegue il triage valuta i sintomi del paziente e gli assegna il colore, quindi gli comunica le condizioni per usufruire del servizio e il tempo previsto per la visita; il paziente decide se uscire o meno e l'attività Registrazione decisione paziente assegna di conseguenza alle variabili *esami* e *uscita* il valore *true* o *false*;

c) quando si ritorna nel processo principale i due controlli esci? ed esami? andranno a "leggere" le variabili e decideranno di conseguenza.

Come mostra questo semplice esempio approfondendo l'analisi di un'attività si arriva ad un punto in cui è possibile fare delle ipotesi su come implementare l'attività stessa in termini di una *procedura* scritta in certo linguaggio di implementazione (quello che è utilizzato dalla piattaforma su cui è implementato anche il motore di workflow).

Chiaramente questo livello di dettaglio è troppo alto mentre siamo ancora nella fase di analisi del sistema ma interesserà la fase implementativa in cui il processo ristrutturato deve essere tradotto in modo da poter essere eseguito dal motore di workflow.

Un sotto-processo può anche essere incorporato nel processo principale invece che specificato a parte (in questo caso si parla di sotto-processo *espanso*); nel nostro esempio possiamo avere la rappresentazione del processo Pronto Soccorso illustrata in Figura 3.20.

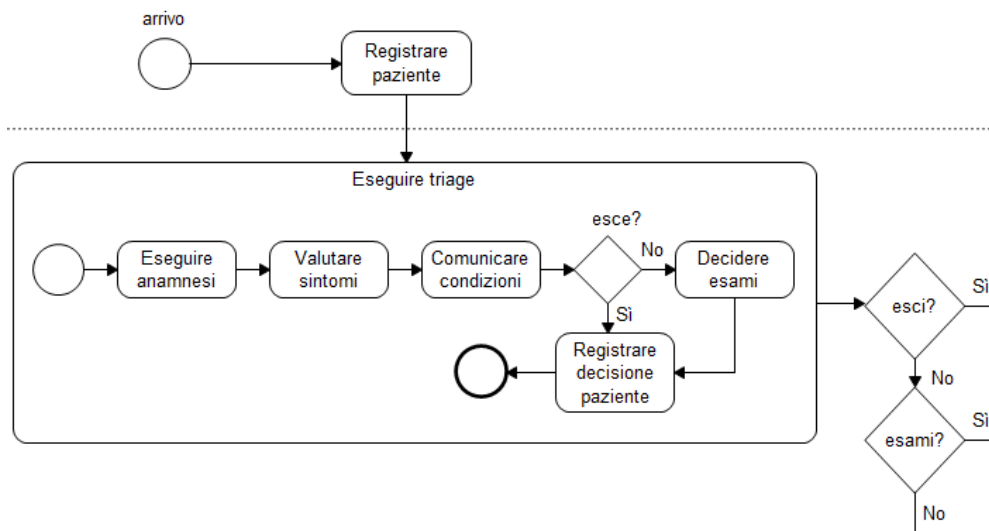



Figura 3.20: Sotto-processo Eseguire triage espanso

Una caratteristica dei sotto-processi espansi è che essi possono iniziare soltanto con un evento generico (ad esempio non si può iniziare con la ricezione di un messaggio).

Spesso le attività (e i sotto-processi) devono essere ripetuti più volte, questo comportamento è facilmente realizzabile con dei nodi condizione ma in BPMN è possibile usare una notazione abbreviata che consiste nell’inserire il simbolo  nella forma.

La Figura 3.21 illustra tre possibili rappresentazioni per un’attività A che deve essere eseguita 3 volte (c è una variabile contatore che deve essere inizializzata con c=1).

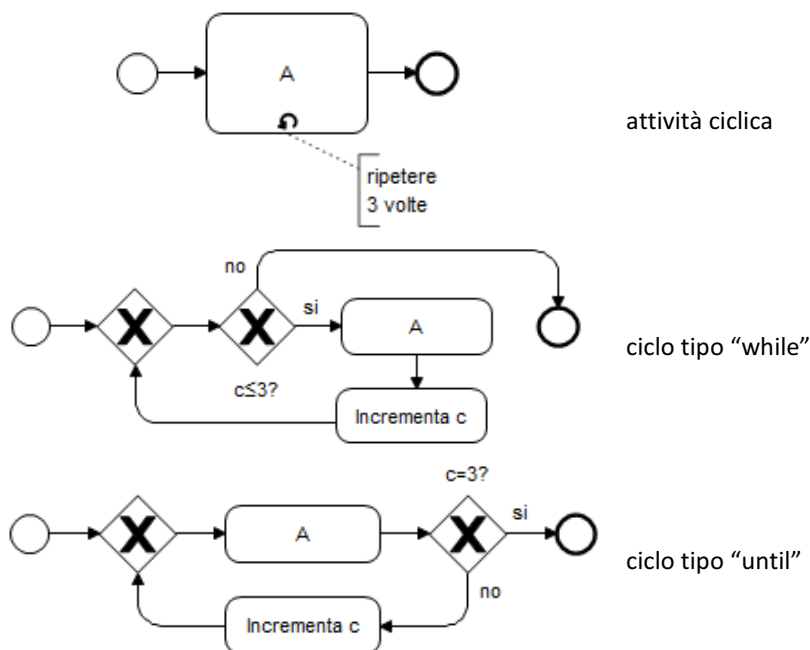


Figura 3.21: Cicli di attività

Come mostrato in figura è comunque utile inserire un commento sull'attività ciclica che indichi quante volte l'attività deve essere eseguita o qual è la condizione di uscita dal ciclo. Il numero di ripetizioni del ciclo può essere noto in fase di progetto oppure essere determinato all'atto dell'esecuzione. E' compito dell'analista specificare le modalità del ciclo mediante **attributi** (associati alla forma) che permettono di definire le regole di esecuzione; ad esempio negli attributi si può dichiarare se usare la tecnica *while* (la condizione di ciclo è testata prima che l'attività sia eseguita) o *until* (la condizione è testata dopo che l'attività è stata eseguita); queste informazioni saranno usate quando il modello sarà implementato. Anche i sotto-processi possono avere un comportamento ciclico che viene indicato con lo stesso simbolo.

Le attività (e i sotto-processi) possono avere un'altra modalità di esecuzione oltre a quella ciclica, è l'esecuzione multipla di istanze in cui l'attività deve essere eseguita più volte su istanze diverse di un processo (ad esempio la sede centrale di una banca deve effettuare una verifica su tutte le filiali). L'esecuzione multipla può avvenire in due modi:

- le istanze sono attivate in parallelo (si inserisce il simbolo **III** nella forma);
- le istanze sono attivate in sequenza (si inserisce il simbolo **≡** nella forma).

Nel caso di attivazione parallela dei parametri della forma permettono di definire la conclusione dell'attività (o del sotto-processo) nel suo complesso; ad esempio se l'attività consiste nel controllare in parallelo dei campioni si potrebbe decidere che l'attività termina quando una sola delle verifiche fornisce un risultato negativo.

Per illustrare questi casi consideriamo un semplice esempio:

Un'azienda deve ordinare un certo prodotto, allo scopo prepara una lista di possibili fornitori, richiede loro un'offerta e quindi seleziona l'offerta migliore. La ricerca dei fornitori può chiudersi quando ne sono stati trovati almeno 5.

La situazione può essere rappresentata con il diagramma di Figura 3.22.

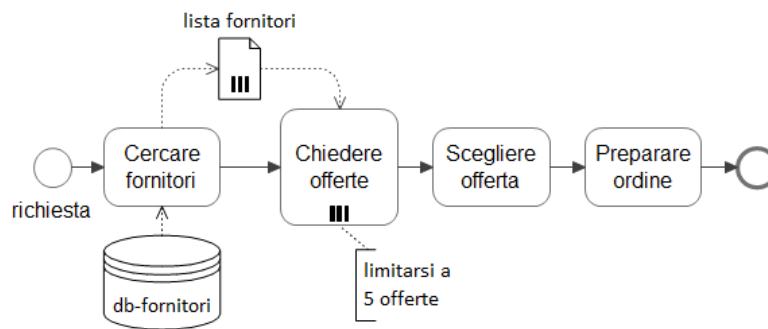


Figura 3.22: Attività a istanze multiple

Osserviamo che l'attività Cercare fornitori produce una **lista fornitori** di possibili fornitori di quel certo prodotto a partire dalla base dati **db-fornitori**; i simboli utilizzati per questi due oggetti rappresentano in BPMN dei manufatti e saranno discussi in seguito. L'attività Chiedere offerte è istanziata contemporaneamente N volte, se N è il numero di fornitori nella lista. L'annotazione *[limitarsi a 5 offerte]* termina l'attività quando sono arrivate 5 offerte (il come sarà realizzata la semantica specificata è demandato alla fase di implementazione). E' importante osservare che qui non si tratta di un ciclo, le N istanze sono indipendenti fra loro e quindi potrebbero essere eseguite in parallelo (se, ad esempio, ci fossero a disposizione le risorse necessarie).

I sotto-processi (e non le attività) ammettono una specifica parziale che viene indicata con il simbolo **~**, questi

sotto-processi sono detti **Ad-hoc** e la Figura 3.23 mostra un esempio in cui non è noto a priori l'ordine in cui saranno eseguite le attività A, B, C, D e neanche il numero di volte che saranno attivate, sappiamo solo che l'attività D deve essere eseguita dopo l'attività C. In ogni caso tutte le attività devono essere svolte prima che il processo possa proseguire e la presenza di un arco di controllo non implica che ci sia un'attività iniziale o finale.

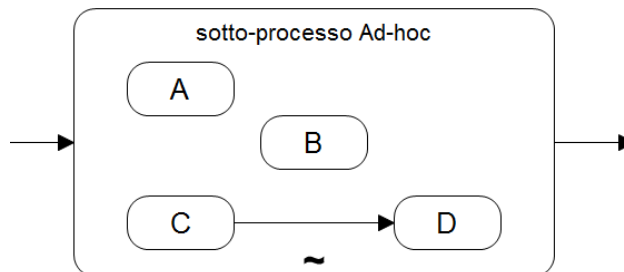


Figura 3.23: Sotto-processi Ad-hoc

Non essendoci un evento finale il sotto-processo Ad-hoc ha bisogno di un meccanismo per segnalare il completamento del sotto-processo; questo è ottenuto associando al sotto-processo un attributo logico di completamento che all'inizio ha il valore *false*, l'attributo diventa *true* solo quando una delle attività provvede a modificarlo al suo termine: a quel punto termina anche il sotto-processo. Se il processo è gestito da un motore di workflow in fase di esecuzione deve essere possibile specificare, in un qualche modo, una sequenza di esecuzione (ad esempio lasciando alla risorsa che esegue le attività la decisione su cosa fare in tempo reale).

La Figura 3.24 mostra un esempio di un sotto-processo di questo tipo.

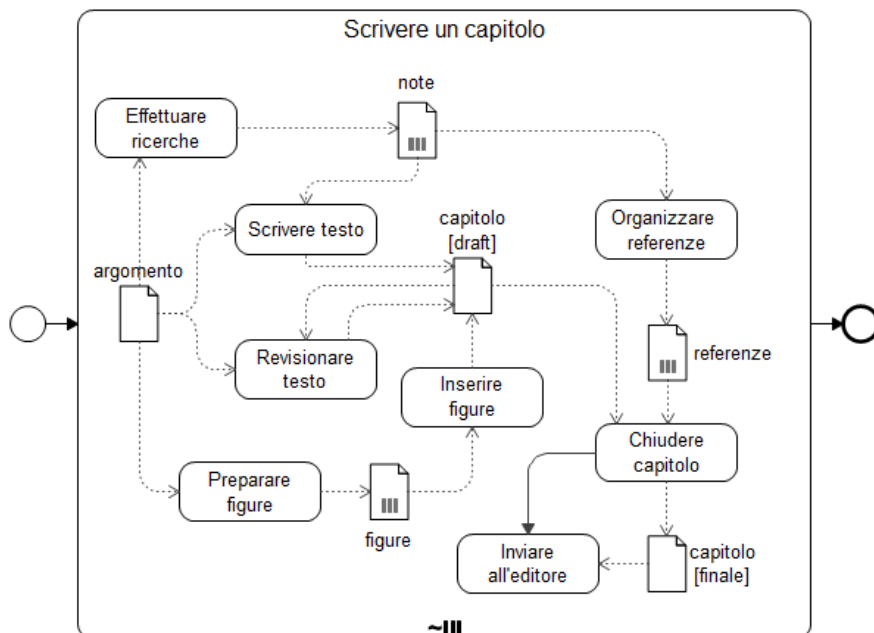


Figura 3.24: Sotto-processo Ad-hoc – Scrivere un capitolo

La scrittura del capitolo di un libro è tipicamente un processo in cui le varie attività sono svolte in un ordine qual-



siasi e ripetute un numero imprecisato di volte, inoltre in questo caso il sotto-processo è stato dichiarato a istanze multiple, cioè è specificata la possibilità di scrivere più capitoli contemporaneamente. All'atto dell'esecuzione sarà la risorsa, cioè lo scrittore, che deciderà cosa fare prima di chiudere il capitolo e inviarlo all'editore; in questo caso l'attività Inviare all'editore ha il compito di aggiornare l'attributo di completamento. Ricordiamo che gli archi diretti tratteggiati non hanno alcun significato di controllo dell'esecuzione del sotto-processo, indicano semplicemente che certi oggetti (come figure, referenze, ecc.) sono "prodotti" da un'attività e "utilizzati" in un'altra.

Durante l'esecuzione di attività o sotto-processi possono verificarsi delle condizioni poco probabili per cui non è conveniente, e spesso molto laborioso, appesantire la rappresentazione del processo per tener conto di queste **eccezioni**.

Il linguaggio BPMN fornisce diversi costrutti per trattare le eccezioni, un primo esempio è fornito dalle **scadenze** (o **time-out**); il frammento di processo di Figura 3.25 illustra un semplice caso in cui si devono analizzare delle pratiche, una pratica può essere accettata o rifiutata ma se dopo 20 giorni l'analisi non è terminata l'accettazione è automatica.

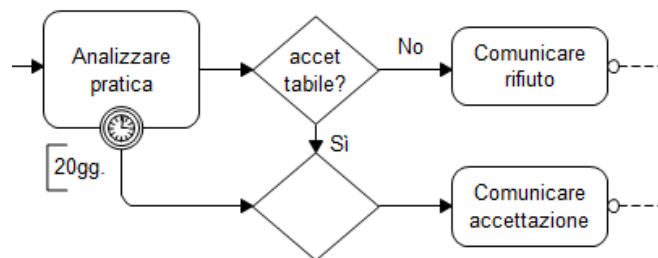


Figura 3.25: Rappresentare le eccezioni (scadenze)

Come si vede le **eccezioni** sono rappresentate mediante **eventi** messi sul *contorno* dell'attività (o del sotto-processo): l'attività ha due flussi di controllo in uscita, uno normale e uno di eccezione che viene seguito quando la condizione eccezionale (20gg.) diventa vera. Quindi alla partenza dell'attività diventa attivo anche il timer che controlla il passare del tempo e alla scadenza *interrompe* l'esecuzione dell'attività, questa termina senza emettere il token che invece è emesso lungo il flusso eccezionale. La scadenza può anche non interrompere l'esecuzione dell'attività, in tal caso l'evento timer ha il doppio bordo tratteggiato; la Figura 3.26 rappresenta il caso in cui si deve eseguire una spedizione e l'eccezione si riferisce al fatto che se accadono dei ritardi bisogna avvisare il cliente.

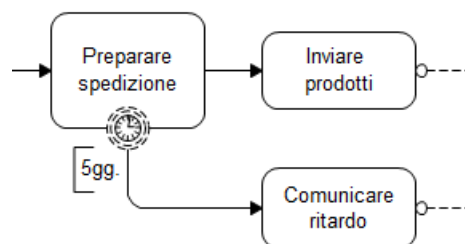




Figura 3.26: Rappresentare le eccezioni non-terminanti

Nell'esempio di Figura 3.26 se la preparazione termina prima di 5 gg. si prosegue inviando i prodotti, se si arriva alla scadenza dei 5 gg. si comunica il ritardo al cliente ma si continua a preparare la spedizione e quindi a inviare i prodotti: in tal caso abbiamo due cammini (e due token) che evolvono in parallelo.

La gestione di eccezioni terminanti e non-terminanti possono riguardare anche i messaggi in uscita da un'attività o

da un sotto-processo, in tal caso si inseriscono sul contorno i simboli  e  (un discorso analogo vale per i segnali).

Nell'esempio illustrato in Figura 3.27 una Agenzia Viaggi riceve una richiesta e prepara il viaggio richiesto; durante la preparazione può arrivare l'annullamento del viaggio (il messaggio annulla interrompe il sotto-processo) mentre il messaggio cambia modifica solo alcuni particolari (la preparazione continua), anche l'evento timer 5gg. non interrompe il sotto-processo e, presumibilmente, genera un avviso.

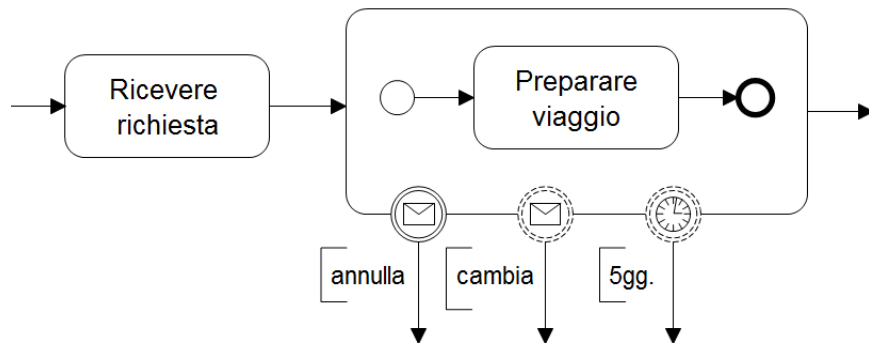



Figura 3.27: Eccezioni causate da messaggi

Un ultimo tipo di eccezione che può essere utile riguarda il fatto che durante l'esecuzione di attività (o sotto-processi) possono verificarsi degli **errori** che provocano l'interruzione dell'attività, in tal caso si inserisce sul contorno il simbolo . Nell'esempio di Figura 3.28 durante l'attività Eseguire misura può accadere che lo strumento di misura si guasti, in tal caso si procede lungo il flusso eccezionale. Osserviamo che gli errori si considerano sempre come eventi che interrompono l'esecuzione.

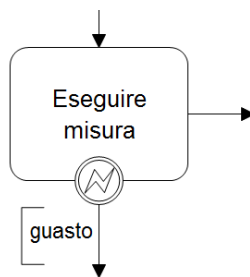


Figura 3.28: Eccezioni causate da errori

Se l'errore compare all'interno di un sotto-processo questo deve avere un nodo finale interno di tipo errore che viene trasmesso al livello superiore mediante l'eccezione; la Figura 3.29 illustra il caso seguente:

L'azienda fabbrica dei prodotti che devono essere collaudati prima del rilascio finale. Dopo aver recuperato il prodotto si procede alla sua analisi e si verificano i risultati, se questi sono plausibili il prodotto è rilasciato altrimenti si genera un *errore risultati*; se durante l'analisi lo strumento di misura si guasta (evento *errore strumento*) bisogna cercare di ripararlo e riprendere l'analisi; se lo strumento continua a non funzionare (al più si fa un tentativo di riparazione) si genera un *errore strumento guasto* e bisogna inviare lo strumento in fabbrica per la riparazione; in entrambi i casi di

uscita per errore il prodotto in analisi deve essere inviato ad un laboratorio esterno per il collaudo.

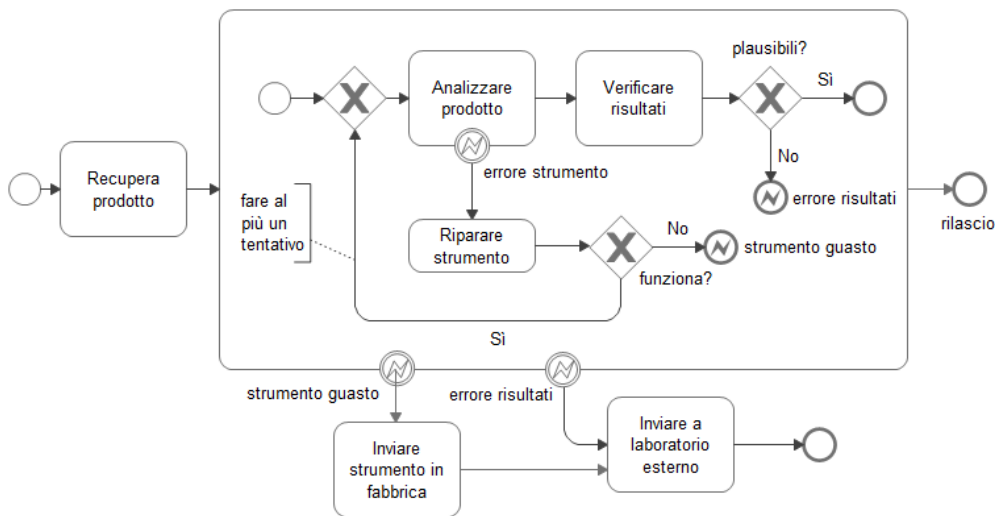


Figura 3.29: Sotto-processo con errori

Come mostra l'esempio è possibile trattare diversi tipi di errore all'uscita di un sotto-processo basandosi sui nomi degli eventi relativi.

### 3.6 Rappresentare i Dati e gli Elementi Ausiliari

Durante l'esecuzione di un processo possono essere manipolati una rilevante quantità di **dati** (informazioni, archivi, documenti, e così via). Finora non ci si è occupati dei dati ma al flusso di controllo è spesso associato un flusso di informazioni e l'obiettivo dei messaggi è anche quello di scambiare informazioni fra i vari partecipanti al processo.



nome  
[stato]

In BPMN i dati sono rappresentati da un simbolo rettangolare col bordo in alto a destra ripiegato ed è anche possibile specificare uno [stato]; lo stato è generalmente usato per descrivere come un dato viene modificato nel processo.

I dati possono essere o associati ai flussi di controllo (e corrispondono, anche "fisicamente", al trasferimento del dato) o specificati in ingresso/uscita dalle attività, come mostrato nell'esempio di Figura 3.30: gli **ordini** (in diversi stati: *nuovo*, *respinto* o *accettato*) sono o associati ai flussi (*ordine[nuovo]*) o sono generati da un'attività e usati da un'altra (*ordine[accettato]*, *ordine[respinto]* e *bolla*); ad esempio la *bolla* di accompagnamento dei prodotti è creata nell'attività Preparare spedizione ed è usata in Inviare prodotti. L'*ordine[accettato]* e la *bolla* saranno quindi registrati nel database *db-ordini*.

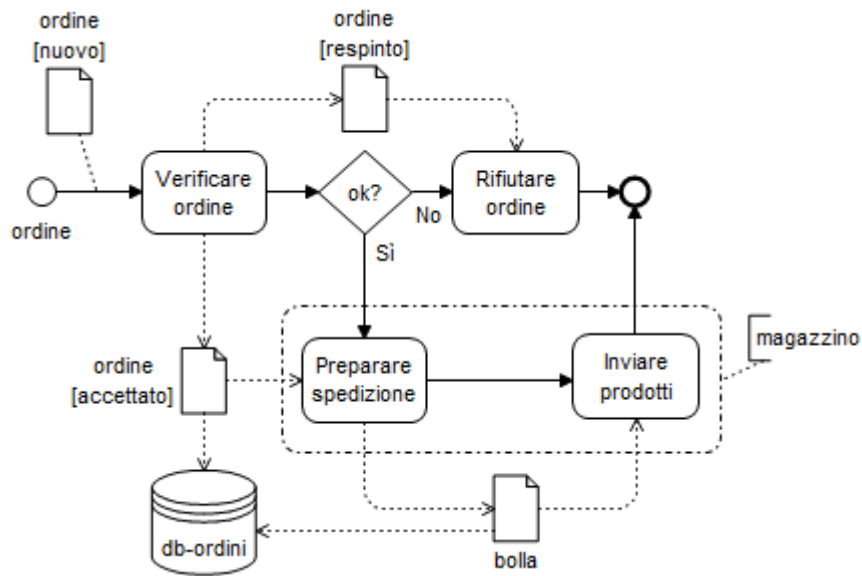


Figura 3.30: Rappresentazione dei dati e degli elementi ausiliari

La Figura 3.30 mostra anche altri due elementi ausiliari definiti dal BPMN:

1. una **annotazione** è costituita da un testo di commento inserito in una parentesi quadra aperta [ e connessa ad un elemento qualsiasi del processo mediante un arco (tratteggiato) di associazione (che esamineremo in seguito); il significato dell'annotazione può essere qualsiasi e in genere è usata per specificare meglio la semantica dell'elemento annotato;
2. il **gruppo**, costituito da un contorno tratteggiato ----- comprendente un insieme di elementi graficamente adiacenti del processo che per un qualche motivo si intende mettere in evidenza (nell'esempio l'annotazione [magazzino] potrebbe indicare che le attività incluse nel gruppo si svolgono nel magazzino aziendale).

Questi tre elementi ausiliari del BPMN (chiamati **artefatti**) possono essere *estesi* a cura dell'analista o del costruttore dello strumento utilizzato nella modellazione del processo. Ad esempio lo strumento Process2013 utilizzato in questo testo ha introdotto un certo numero di artefatti supplementari per rappresentare oggetti che si presentano frequentemente nei processi, artefatti descritti nella Tabella 3.2 ; alcuni di questi sono stati usati in Figura 3.30 e in diagrammi successivi.







	<u>deposito di dati</u> (ricorda il simbolo analogo usato nei tradizionali flow-chart per rappresentare una basedati);
	<u>oggetto o dato in input ad un'attività</u>
	<u>oggetto o dato in output da un'attività</u>
	<u>raccolta di oggetti</u> (possono essere liste o sequenze di oggetti o dati omogenei fra loro)
	<u>raccolta di oggetti in input ad un'attività</u>
	<u>raccolta di oggetti in output da un'attività</u>

Tabella 3.2: Artefatti in BPMN

Ovviamente tutti gli artefatti (e le eventuali estensioni) non hanno alcuna influenza sul flusso di controllo, infatti non è possibile inserire un artefatto in un arco di controllo!

Per la connessione degli artefatti agli elementi del processo a cui si riferiscono è stato utilizzato il terzo tipo connettori definiti in BPMN accanto ai connettori di *flusso*  $\longrightarrow$  ed ai connettori di *messaggio*  $\circ\text{-----}\triangleright$ , i connettori di *associazione* che possono essere semplici  $\text{-----}$  o direzionali  $\text{-----}\longrightarrow$ .

Discutendo della rappresentazione dei processi abbiamo implicitamente supposto che tutti i dati necessari al processo siano a disposizione in un qualche deposito a cui il processo può accedere, ma è importante tener presente che il processo è comunque strettamente correlato con la gestione dei dati che esso utilizza. Ad esempio se un'attività cerca di modificare un dato bisogna assicurarsi che il dato stesso sia stato precedentemente creato da una qualche attività del processo o sia già esistente e memorizzato in qualche archivio da cui è possibile estrarlo.

Tener conto dei dati gestiti dal processo è essenziale se siamo in fase di progetto e bisogna costruire il sistema informativo in grado di supportare il processo stesso. Un analogo discorso vale se siamo in una fase di ristrutturazione del processo, in questo caso la ristrutturazione potrebbe indurre modifiche nel sistema informativo, nuovi dati potrebbero diventare necessari e dati obsoleti potrebbero essere cancellati.

Se la ristrutturazione riguarda anche una radicale innovazione nel trattamento dei dati, ad esempio viene introdotto un trattamento automatizzato dei documenti per cui questi saranno gestiti in formato digitale, sia il processo che il sistema informativo ne risulteranno radicalmente modificati.

Riprenderemo questi argomenti in un successivo capitolo, per ora osserviamo che i dati comunque determinano dei **vincoli** sul processo; è evidente che se un documento (come la *bolla* di Figura 3.24) è in ingresso ad un'attività (Inviare prodotti) questa non può iniziare se il documento non è presente e quindi il documento **deve** essere prodotto da un'attività precedente (Preparare spedizione), o essere associato ad un messaggio in ingresso al processo o essere prelevato da un deposito di dati.

Nel caso della Figura 3.30 sono presenti due documenti, *ordine* e *bolla*, e devono quindi essere verificati i seguenti vincoli sulle attività:

- Verificare ordine: il documento *ordine[nuovo]* è associato all'evento iniziale *arrivo ordine* per cui è generato nel mondo esterno; l'attività può modificare lo stato del documento in [*accettato*] o [*respinto*];

- Rifiutare ordine: il documento *ordine[respinto]* è creato da un'attività precedente;
- Preparare spedizione : il documento *ordine[accettato]* è creato da un'attività precedente; l'attività crea il documento *bolla* ;
- Inviare prodotto: il documento *bolla* è creato da un'attività precedente.

Un uso importante delle annotazioni riguarda l'inserimento nel diagramma di processo di vincoli che saranno trattati in fase di implementazione del processo stesso. Un uso di questo tipo l'abbiamo visto, ad esempio, in Figura 3.21 con la nota *[ripetere tre volte]* e in Figura 3.29 con la nota *[fare al più un tentativo]*.

## Bibliografia

- T. Allweyer, BPMN 2.0: Introduction to the Standard for Business Process Modeling, Herstellung und Verlag, 2010.
- M. Owen, J. Raj, BPMN and Business Process Management: Introduction to the New Business Process Modeling Standard, Popkin Software, 2003.
- B. Silver, BPMN Method and Style: A Levels-based Methodology for BPM Process Modeling and Improvement Using BPMN 2.0, Cody-Cassidy Press, 2009.
- R.M. Dijkman, M. Dumas, and C. Ouyang. Semantics and analysis of business process models in bpmn. Information and Software Technology (IST), 50(12):1281-1294, 2008.
- S. White, D. Miers (2009). BPMN Modeling and Reference Guide Understanding and Using BPMN, Lighthouse Pt, FL: Future Strategies Inc.