

Giulio Destri

**INTRODUZIONE
AI SISTEMI
INFORMATIVI
AZIENDALI**



Indice

Indice	5
Prefazione	7
Struttura del testo	8
Ringraziamenti	9
Informazione ed organizzazioni: il sistema informativo	13
Introduzione	13
La realtà: sistemi e modelli	13
I sistemi informativi	16
Bibliografia.....	18
I processi aziendali	19
Un modello fondamentale: il processo aziendale.....	19
L'interno di un'organizzazione: i processi fondamentali.....	21
Il sistema azienda: visione per funzioni vs. visione per processi.....	24
Processi ed organizzazione aziendale	25
Bibliografia.....	27
La risorsa informazione.....	29
La risorsa informazione e le sue caratteristiche	29
Rappresentazione dell'informazione.....	30
Flussi informativi e flussi informatici	34
XML: l’“esperanto” elettronico	37
Bibliografia.....	39
Analisi di un processo aziendale	41
Analisi dell'interno di un processo: i diversi punti di vista.....	41
Analisi del processo come successione di attività.....	42
Analisi del processo come successione di casi d'uso (di strumenti).....	49
Analisi delle entità che prendono parte ai processi	53
Analisi delle interazioni fra gli elementi operanti entro un processo.....	61
Analisi del processo come successione di cambiamenti di stato	66
Un esempio completo di analisi	69
Una visione d'insieme: il legame fra le viste del processo	73
Bibliografia.....	76
Il sistema informatico entro il sistema informativo	77
Il sistema informatico.....	77
La struttura di un'applicazione software	82
Le reti entro i sistemi informatici	84
L'evoluzione tecnologica: dal monolite al middleware	89
Il mondo del client-server e le applicazioni multi-tier	93
L'evoluzione del Client-Server: il 3-tier e il multitier	96
Il problema delle compatibilità fra componenti ed interi applicativi	101
Panoramica su sistemi operativi maggiormente diffusi	103
Panoramica sulle tecnologie correnti per lo sviluppo di applicazioni.....	106
Le nuove soluzioni di integrazione: Service Oriented Architecture	109
Il Grid Computing	112
Bibliografia.....	114

Soluzioni informatiche per l'impresa.....	117
I sistemi integrati di gestione: gli ERP	117
Il Customer Relationship Management (CRM)	120
La Supply Chain Management (SCM).....	122
La business intelligence	124
Bibliografia.....	127
Le professionalità nei sistemi informativi	129
Le risorse umane ed il loro ruolo	129
I dettagli dei ruoli “canonici”	131
Alcuni esempi di organizzazioni “reali”	135
Bibliografia.....	138
La sicurezza informatica	139
Le problematiche della sicurezza informatica.....	139
La prevenzione di problemi di Safety	141
I problemi di security: le minacce umane alla sicurezza.....	149
I problemi di security: il ruolo dei virus.....	162
La protezione delle informazioni	168
L'identità elettronica	177
Applicazioni operative di crittografia ed autenticazione.....	182
La protezione dei sistemi.....	189
Gestire la sicurezza.....	196
Bibliografia.....	202
La gestione dei sistemi informativi	207
Strumenti per la pianificazione delle attività.....	207
Uno strumento per valutare i ricavi: il Return Of Investment (ROI)	209
Uno strumento per valutare i costi: il Total Cost of Ownership (TCO).....	210
Altri strumenti importanti per la gestione e pianificazione.....	213
Le politiche di gestione	215
Gestione corrente e gestione del cambiamento	216
La gestione della sicurezza.....	217
La gestione del progetto informatico	220
ICT e business: situazione corrente e possibili evoluzioni future	225
Bibliografia.....	228
Case study	229
Uno schema di analisi parziale e completa	229
Studio associato di professionisti	230
Agenzia di lavoro interinale	235
Azienda vinicola.....	239
Azienda vendita CD e libri con sito Web.....	243
Bibliografia.....	247

Prefazione

I sistemi informativi sono il cuore di qualsiasi azienda od organizzazione. Negli ultimi anni, sia per l’interconnessione sempre più presente fra le nazioni, dovuta alla globalizzazione, sia per l’avvento di nuove tecnologie, è in essi in atto una vera e propria rivoluzione. Ciò nonostante l’importanza di una loro conoscenza sufficientemente generale e completa, basata su approcci metodologici, è ancora troppo spesso trascurata, non solo entro i percorsi di studio universitario, ma anche tra molti professionisti dell’informatica.

Questo libro nasce con molteplici scopi.

In primo luogo, cercare di superare la contrapposizione tra chi affronta lo studio dei sistemi informativi eccessivamente da un punto di vista tecnico, cioè considerandoli come composti quasi solo dalla loro parte informatica, e chi invece ne considera solo gli aspetti umani e/o organizzativi. Il giusto punto di vista è considerarli un sistema unico, ma composto da diversi elementi, con diverse caratteristiche, che interagiscono fra di loro e devono produrre lo scopo complessivo del sistema. Per questo motivo, questo libro presenta i vari aspetti, dalla tecnologia ai processi, all’organizzazione ed ai ruoli delle risorse umane. Non manca entro il testo anche la definizione precisa dei concetti legati ad alcune sigle di uso corrente, non sempre adeguatamente tenuti in considerazione.

In secondo luogo, presentare un punto di partenza sintetico ma abbastanza completo per chi, provenendo da studi e/o facoltà tecniche o economiche, si avvicina per la prima volta allo studio dei sistemi informativi senza conoscenze precedenti. Si sottolinea che questo testo è un punto di partenza e in esso sono ampiamente presenti riferimenti bibliografici per ampliare le nozioni relative ai vari aspetti trattati.

In terzo luogo, presentare metodologie ed esempi applicabili praticamente non solo alle grandi aziende, ma anche alle piccole e medie imprese, vero cuore del tessuto produttivo italiano.

Il lavoro che ha portato al libro nasce da molteplici fonti:

1. dall’esperienza personale più che decennale di consulente e progettista nel contesto dei sistemi informativi di aziende grandi (come, ad esempio, il Gruppo Barilla, il Gruppo Telecom, il Gruppo CRIF), di aziende medie e piccole, di banche e di pubbliche amministrazioni locali;
2. dall’esperienza di alcuni anni di insegnamento nel corso di “Sistemi Informativi I” (nei primi anni denominato “Informatica in Azienda”) presso il Corso di Laurea in Informatica dell’Università di Parma e del corso di “Informatica per la Gestione d’Impresa” presso il Corso di Laurea Specialistica in Gestione d’Azienda dell’Università Cattolica di Piacenza;
3. dell’esperienza di docenza di “Internet Security” e “Electronic Payment Systems” entro il Master of Management in Network Economy (MiNE) dell’Università Cattolica di Piacenza /U.C. Berkeley, con allievi provenienti da tutte le parti del mondo, aventi alle spalle percorsi professionali molto diversi fra loro;

4. dall'esperienza di docenza di molti corsi aziendali e di recruitment di risorse umane per il settore ICT.

Sono state incluse anche nozioni imparate “sul campo”, mutuandole dall'esperienza di colleghi, con l'obiettivo di creare un testo che spiegasse chiaramente i concetti utili agli studenti universitari, con riferimento ad esperienze ritrovabili nel contesto economico aziendale italiano, tipicamente orientato alla piccola e media impresa. Sono presenti anche riferimenti a metodologie molto innovative, quali l'UML for Business, che oggi sta entrando a far parte delle metodologie di lavoro dei progettisti di sistemi informativi delle più grandi aziende come IBM e Microsoft, applicati ad un contesto di progetti anche piccoli. Alcuni case study, mutuati dall'esperienza, completano le parti teoriche del testo.

Struttura del testo

Il libro è suddiviso in dieci capitoli, ognuno dei quali è dedicato ad una tematica specifica. Il percorso cronologico dovrebbe essere affrontato nell'ordine, ma lettori già in possesso di conoscenze specifiche possono anche focalizzare l'attenzione solo su alcune parti.

Nel primo capitolo vengono definiti i concetti fondamentali relativi ai sistemi informativi ed alle loro componenti primarie e vengono forniti, mutuandoli dalla teoria dei sistemi, alcuni strumenti di analisi di base.

Nel secondo capitolo, applicando quanto visto nel primo, sono definiti i modelli fondamentali dei meccanismi che governano il funzionamento dell'azienda, in primo luogo i processi aziendali o processi business, con ampi riferimenti alla letteratura, ed alcuni esempi.

Nel terzo capitolo si focalizza l'attenzione sulla risorsa primaria trattata entro i sistemi informativi: l'informazione, analizzata sia da un punto di vista logico, sia da un punto di vista di contenuti ed effetti, sia, soprattutto, da un punto di vista tecnico legato al suo trattamento automatico.

Nel quarto capitolo vengono presentate alcune metodologie moderne di analisi dei processi, che devono consentire a operatori e, soprattutto, ai dirigenti di comprendere lo status corrente dei sistemi e di valutare se, quando e come inserire dei miglioramenti in essi. In particolare si pone enfasi sull'uso delle metodologie basate su UML, che stanno assumendo particolare importanza anche per l'azione di grandi operatori del mondo dell'informatica

Nel quinto capitolo è presentato l'aspetto tecnologico, ossia la componente dei sistemi informatici entro i sistemi informativi. Si parte da una definizione del ruolo della componente informatica, per analizzare l'evoluzione degli ultimi anni, dovuta alla massiccia introduzione delle reti entro l'azienda. Vengono presentate anche panoramiche di tecnologie informatiche utilizzate entro le aziende e indicate nuove metodologie, come la Service-Oriented Architecture (SOA), che stanno rivoluzionando i sistemi informativi.

Nel sesto capitolo viene presentata una panoramica delle categorie principali di software che nel corso degli anni'90 sono diventati patrimonio tipico di ogni azienda

medio-grande come gli ERP, i CRM, i software per la SCM, i data warehouse ed il loro uso nella business intelligence.

Nel settimo capitolo è presentata una breve panoramica delle professionalità richieste dal mercato legato ai sistemi informativi.

Nell'ottavo capitolo viene analizzato nei dettagli il problema della sicurezza informatica. Dopo avere definito l'insieme dei rischi legati a eventi accidentali e quelli legati ad azioni fraudolente di origine umana, vengono presentate le possibili contromisure e alcuni esempi delle loro applicazione a protezione dei sistemi. E' considerato anche l'effetto sulla evoluzione dei sistemi che l'inserimento delle misure di protezione ha avuto.

Nel nono capitolo sono presentati ed analizzati gli strumenti più importanti per la pianificazione e gestione dei sistemi informativi, come la WBS, i diagrammi PERT e Gantt, il ROI ed il TCO. La presentazione delle politiche di gestione, la loro applicazione alla sicurezza informatica ed una metodologia completa di conduzione dei progetti informatici, mutuata dallo standard dello Unified Process, completa la definizione. Il capitolo si chiude con una breve panoramica sullo "stato dell'arte" e sulle tendenze in atto nell'evoluzione dei sistemi informativi.

Nel decimo capitolo vengono presentate le linee guida dell'analisi dei sistemi informativi entro aziende. Alcuni esempi di analisi di sistemi, applicate a realtà tipiche del mondo italiano, quali le piccole aziende e gli studi tecnici, completano il tutto.

Il libro è accompagnato da una serie di slide, raggruppate in un file per ogni capitolo, usate nel corso di Sistemi Informativi I a Parma. Le slide sono reperibili presso il sito del Corso di Sistemi Informativi I a cui si arriva dalla home page di Informatica <http://informatica.unipr.it>, oppure presso il sito di Area Solutions Providers all'indirizzo http://www.areasp.com/formazione/documenti/slide_si.html.

Ringraziamenti

E' per me doveroso concludere questa premessa con il ringraziamento a tante persone che ho conosciuto in questi anni, colleghi, superiori, collaboratori ed anche allievi dai quali ho imparato tantissimo.

Vorrei ringraziare Oscar Figus, Cesare Chiodelli, Paolo Marenzoni, Andrea Gangini, Massimo Ponzoni, Simone Mainini, Marco Solci, Gianluca Golinelli, Max Bellomi, Ivan Makale, Mary Ercolini, Marco Barigazzi, Ivan Nebbi, Roberto Morpanini, Paola Foranzi, Laura Biasin, Andrea Gherardi, Francesco Fiorini, Alessio Zatti, Stefano Dolcini, Alberta Rossi, Roberto Bianchi, Andrea Gandini, Andrea Zanella, Corrado Ianelli, Giuseppe Tamborino e gli altri colleghi ed ex colleghi con cui abbiamo condiviso importanti esperienze,

l'Ing. Alessandro Galaverna, l'Ing. Massimo Moroni, il Dr. Andrea Pataccini, l'Ing. Stefano Pietroni, l'Ing. Giovanni Tortorici e la Dr.ssa Giovanna Verani di Barilla,

l'Ing. Riccardo Paini di IBM,

il Dr. Simone Ragnolini di Accenture,

l'Ing. Michele Vignali di BancaIntesa per il suo prezioso contributo al capitolo 5, la Dr.ssa Silvia Davoli di Procter&Gamble per il suo prezioso contributo ai cap. 2 e 6, l'Ing. Giuseppe Vannini, il Dr. Luciano Agrimonti e gli altri di SinfoPragma (Parma),

l'Ing. Giovanni Montomoli della F.I.A.S.A. di Parma,
il Dr. Aldo Bruschi, l'Ing. Sergio Badini e l'Ing. Cristina Massa di CRIF (Bologna),
il Dr. Fabrizio Cassoni e il Dr. Luigi Pugnetti di Symbolic (Parma),
il Dr. Andrea De Pasquale, il Dr. Enzo Bisotti e gli altri amici di SMART.IT
(Bologna),
il Dr. Pietro Rossi di Prometeia (Bologna),
Il Dr. Renzo Tavoni e il Dr. Afranio Montermini di Marina Rinaldi (Reggio Emilia)
il Professor Domenico Ferrari, il Professor Giancarlo Piacentini, il Professor Emilio
Rottoli, il Professor Roberto Berchi e gli altri colleghi del Master of MiNE,
la Dr.ssa Franca Cantoni, la Dr.ssa MariaCristina Piva, il Dr. Roberto Bernazzani, il
Dr. Eugenio Tacchini e gli altri colleghi dell'Università di Piacenza,
il Prof. Valentino Gandolfi, il Prof. Marco Riani, il Prof. Alessandro Zaccagnini, la
Prof.ssa Cristina Ziliani, Il Dr. Roberto Alfieri, il Dr. Roberto Covati, l'Ing. Monica
Mordonini e gli altri colleghi dell'Università di Parma,
il Dr. Vincenzo D'Andrea dell'Università di Trento, Il Dr. Francesco Virili
dell'Università di Camerino e gli altri docenti e ricercatori del gruppo italiano Sistemi
Informativi,
i miei allievi del Master of MiNE,
l'Ing. Manfredo Manfredi,
l'Ing. Giovanni Rimassa dell'INRIA,
l'Ing. Franco Venturi,
il Dr. Stefano Ferrari di Finanza&Software, il Dr. Alessandro Poli di
ProgettoSinergia... e tanti altri ancora.

Inoltre vorrei ringraziare il Professor Gianni Conte della Facoltà di Ingegneria di
Parma ed il Professor Gianfranco Rossi della Facoltà di Scienze di Parma per il loro
contributo diretto, sotto forma di correzioni, consigli e incoraggiamento, alla
realizzazione del libro. Infine ringrazio anche i miei genitori, Matilde Sarzi Sartori
Destri e Giovanni Destri, per il loro preziosissimo contributo alla correzione delle
bozze, ed i preziosi suggerimenti che mi hanno consentito di migliorare la chiarezza
della esposizione.

A mia madre, a mio padre, a zia Maria

A Irma

Informazione ed organizzazioni: il sistema informativo

Introduzione

Il mondo reale è per sua natura complesso e le organizzazioni umane lo sono in modo particolare. Per potere comprendere e gestire la realtà, in particolare quella delle imprese ed organizzazioni, quali per esempio pubbliche amministrazioni, enti sanitari, onlus, ecc... è indispensabile sviluppare metodologie di **modellazione** della realtà, atte a creare *modelli semplificati* della stessa, evidenziandone le caratteristiche di interesse in relazione allo scopo della analisi e al contesto.

I sistemi informativi sono oggi il cuore di qualsiasi azienda. Al loro interno la componente informatica (il sistema IT da Information Technology o ICT da Information and Communication Technology) ha proprietà particolari, che non possono essere ignorate anche da chi non è addetto direttamente alla tecnologia, in quanto influenzano profondamente tutte le problematiche relative alla gestione.

In questo capitolo vengono definite le tecniche di modellazione che saranno applicate nei capitoli successivi all'area dei processi aziendali, noti anche come processi business.

La realtà: sistemi e modelli

Un sistema è un insieme di elementi distinti, in relazione fra di loro secondo leggi ben precise, che concorrono al raggiungimento di un obiettivo comune, oppure di una evoluzione comune. I sistemi possono essere classificati in base alla loro origine, per esempio definendo l'esistenza di:

- Sistemi naturali
- Sistemi artificiali
- Sistemi misti.

I sistemi possono essere suddivisi in componenti, ciascuno dei quali ha una sua struttura e può essere un sistema esso stesso. La realtà può quindi essere vista come un macro-sistema, suddivisibile in tanti sottosistemi, aventi struttura più o meno complessa.

Per potere capire e padroneggiare la realtà, occorre modellizzarla. Un **modello**, ossia il prodotto del processo di modellizzazione o modellazione, è una *semplificazione* della realtà, che si ottiene riducendo le caratteristiche in esame e considerando solo quelle utili al fine dello scopo (es. progetto considerato/analisi in corso). In modo più formale, possiamo dire che "Il modello è un'entità M, concreta od astratta, che abbia la proprietà di essere utilizzata per simulare e quindi, entro certi limiti, spiegare, comprendere, il comportamento di un'entità P, il prototipo, anch'essa concreta o astratta" [Orlandi 1998]. In pratica un modello è una rappresentazione del sistema stesso che, pur avendo forma e natura diverse da esso, ne conserva ed evidenzia in modo analogico alcune caratteristiche particolarmente significative per l'analisi. Ad esempio, possiamo considerare i modelli matematici dell'atmosfera usati per le previsioni meteorologiche, i modelli statistico-matematici usati per le previsioni degli

andamenti nei mercati azionari, i modelli iconici usati nei progetti degli edifici e dei componenti meccanici (es. CAD). In particolare i modelli matematici possono essere definiti in modo formale come “un insieme completo e coerente di equazioni matematiche che si presume corrispondente a una qualche entità, detta prototipo del modello” [Aris 1994].

I modelli ci aiutano a “visualizzare” un sistema come è o come vorremmo che fosse, ci permettono di specificare la struttura o il comportamento di un sistema e *documentano* le decisioni che abbiamo preso nel corso della loro definizione. Per esempio, se consideriamo l’insieme dei parametri che servono a definire e descrivere un cliente nel settore dell’industria cosmetica o nel settore bancario, essi conducono a modelli completamente diversi del cliente, ottimizzati e dipendenti dallo scopo per cui il processo di modellazione viene fatto.

Per potere arrivare ai modelli si possono identificare almeno i seguenti passi:

- Definire l’obiettivo della modellazione
- Identificare il sistema e le parti interessanti ed i suoi confini (boundary)
- Definire i vincoli
- Generare un modello di massima che ponga in evidenza le relazioni fra le parti del sistema
- Formalizzare completamente il sistema, raffinando il modello per passi successivi
- Usare il modello (es. per una simulazione).

Per sistemi reali non banali, spesso non si realizza un solo modello, ma un piccolo insieme di modelli, che possono essere costruiti e studiati separatamente, ma che sono strettamente intercorrelati. Poiché la mente umana può elaborare contemporaneamente solo un numero limitato di cose, se il modello non è ben fatto, si possono avere troppi dettagli, alcuni dei quali sfuggono all’attenzione dell’osservatore, o pochi dettagli, con conseguente mancanza di conoscenza e imprecisione del modello stesso.

L’analisi può essere compiuta in approccio Top-down, compiendo una scomposizione di un sistema per passi successivi in sottosistemi sempre più elementari o in approccio Bottom-up, attraverso la costruzione di un sistema complesso per composizione successiva di sistemi elementari. Molto spesso gli analisti esperti usano entrambi gli approcci sullo stesso sistema reale. Inoltre il modello deve essere valido per il contesto in cui si opera e, nel caso serva come base per un confronto di idee, deve anche essere adattato all’interlocutore del momento e le sue parti denominate con un linguaggio appropriato al contesto della comunicazione. Vanno quindi presi in considerazione aspetti diversi in momenti diversi e a diversi livelli di dettaglio. L’uso dei modelli è stato introdotto ormai da decenni nelle discipline scientifico-economiche, ed è stato codificato nella Teoria dei Sistemi o Scienza dei Sistemi, ove si afferma che “per quanto sia complesso o diversificato il mondo della nostra esperienza, potremo sempre trovare in esso vari tipi di organizzazione, che possono essere descritti per mezzo di principi comuni” [PCE 2002]. L’approccio sistemico si distingue dal più tradizionale approccio analitico, in quanto privilegia l’interazione e la connessione delle differenti componenti di un sistema.

La Teoria dei Sistemi è un componente fondamentale anche nelle tecniche di ottimizzazione come la Ricerca Operativa, ossia la disciplina che concerne l'utilizzazione del metodo scientifico nei processi decisionali [AIRO 2001].

Un modello estremamente utile è il cosiddetto modello a “scatola nera” (black-box), rappresentato in figura 1.1, in cui un osservatore esterno non conosce la struttura interna di un sistema, ma ne rappresenta il comportamento, osservando ciò che entra nel sistema (input) e ciò che ne esce (output).

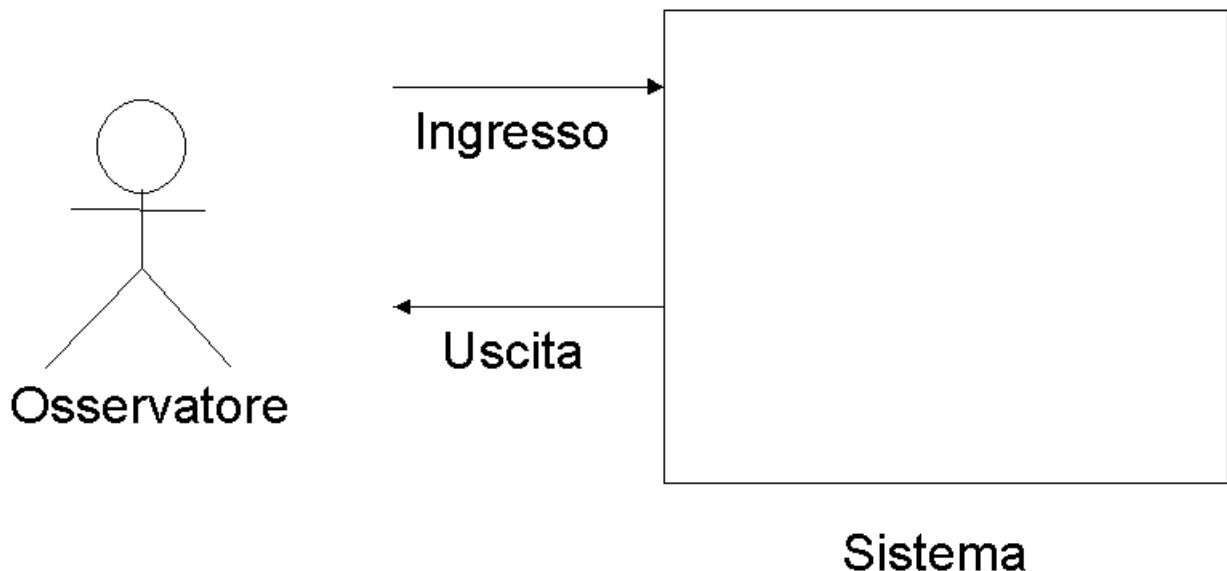
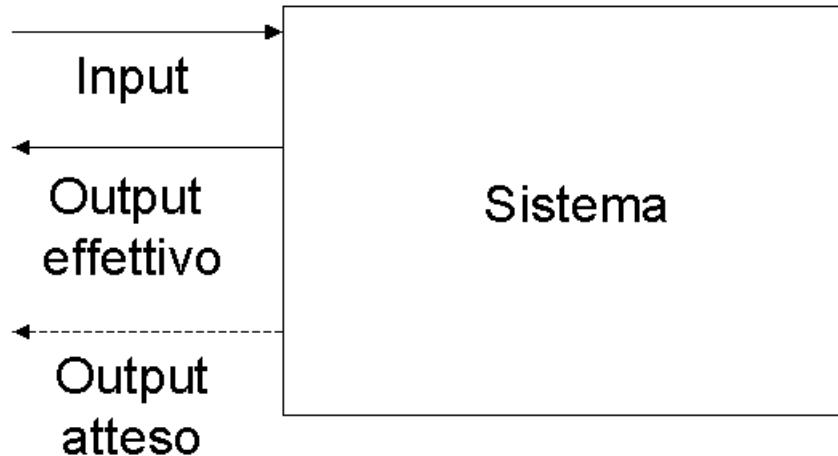


Figura 1.1: Il modello a scatola nera di un sistema, di cui non è nota, o non si intende considerare, la struttura interna, ma solo ciò che entra e ciò che esce, ossia le sue comunicazioni con il mondo esterno.

Come definito anche in uno dei documenti ISO sulla certificazione di qualità [ISO544 2000], due ulteriori parametri descrittivi di fondamentale importanza per qualsiasi sistema di interesse in ambito aziendale o comunque in ambito economico, sono, come rappresentato in figura 1.2:

- **L'Efficienza**, definita come il rapporto fra l'output effettivo di un sistema e il suo input, ovvero il rapporto fra i risultati raggiunti e le risorse utilizzate;
- **L'Efficacia**, definita come il rapporto fra l'output effettivo di un sistema e il suo output atteso, ovvero la sua capacità di raggiungere i risultati desiderati.



EFFICIENZA = Output effettivo/Input

EFFICACIA = Output effettivo/Output atteso

Figura 1.2: Efficienza ed Efficacia di un sistema, per esempio un sistema di produzione.

I sistemi informativi

L'impresa è un sistema ed ha una struttura, fortemente dipendente dal settore in cui opera. Entro l'impresa le varie componenti interagiscono fra loro, scambiandosi anche informazioni. In ogni momento il management o dirigenza può avere bisogno di conoscere lo stato dell'impresa. A queste esigenze di comunicazione e controllo deve rispondere il sistema informativo.

Il sistema informativo può essere definito come “l'insieme di persone, apparecchiature, procedure aziendali il cui compito è quello di produrre e conservare le informazioni che servono per operare nell'impresa e gestirla” (M. De Marco in [De Marco 2000]). Esso corrisponde al termine inglese “Information System”. Pertanto un sistema informativo si suddivide in:

- **Risorse umane** (con organizzazione, ruoli, esperienze, ecc...)
- **Risorse tecnologiche** (sistema informatico, indicato solitamente in inglese con “IT System”)
- **Risorse organizzative** (procedure, regolamenti, workflow, ecc...).

Anche il sistema informativo è un sistema e quindi composto da un insieme di elementi in relazione fra di loro, secondo leggi ben precise. Pertanto non è corretto considerare solo gli aspetti tecnologici di un sistema informativo, esso va considerato nel suo insieme. Il sistema informativo a sua volta è il componente principale che entro l'azienda permette il funzionamento dei processi business (o processi aziendali) su cui le operazioni manageriali ed operative dell'azienda si fondano, che saranno trattati nel prossimo capitolo.

Un concetto fondamentale da tenere presente è quello di **granularità**, ossia praticamente di scala del componente o sotto-sistema su cui si focalizza l'attenzione in un determinato istante, entro il sistema complessivo, e che viene considerato come unità del proprio processo elaborativo. Questo concetto viene rappresentato graficamente in figura 1.3, dove è mostrato il rapporto fra componenti del sistema informatico e tutto il sistema azienda, a diversi livelli di scala e quindi di granularità.

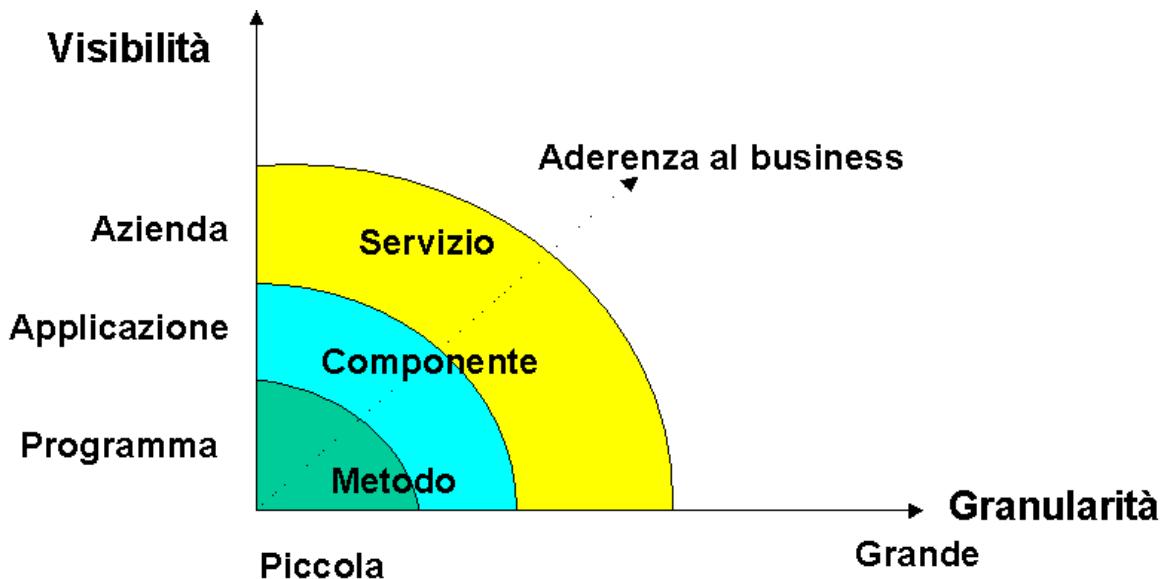


Figura 1.3: Rappresentazione grafica del rapporto fra singole componenti del sistema informatico e componenti dell'azienda e granularità.

La maggior parte dei concetti enunciati non si applicano solo alle imprese, ma ad ogni organizzazione (pubblica amministrazione, ospedali, onlus...). Perciò spesso nell'ambito di questo testo i termini impresa, azienda ed organizzazione saranno usati come sinonimi. Il sistema informatico (detto anche sistema tecnico) si riferisce a “una specifica combinazione di macchine e metodi impegnato nella produzione di un certo risultato” (Sproull e Goodman in [GS 1990]).

In generale la tecnologia può influenzare ma non determinare la scelta delle macchine e dei metodi che costituiscono il sistema informatico. A sua volta, un sistema informatico può essere un supporto appropriato per più tipi di tecnologia, anche in relazione ai diversi requisiti dipendenti dal contesto. La risorsa tecnologica è comunque un componente fondamentale per tutto il sistema informativo. Sono possibili diverse “modalità di interazione” per il rapporto tra tutto il sistema informativo e la tecnologia, e la conseguente evoluzione del primo, come qui elencato secondo la classificazione proposta da Markus e Robey in [MR 1988].

- **Technological imperative:** una nuova disponibilità IT “impone” il cambiamento
- **Organizational imperative:** nuove necessità organizzative impongono il cambiamento

- **Emergent perspective:** l'interazione con una nuova tecnologia conduce al cambiamento; ossia il cambiamento organizzativo emerge da una complessa interazione tra l'IT, gli utenti e l'intero contesto.

In contesti diversi quindi diversi interventi saranno considerati più o meno efficaci nel produrre effetti desiderabili o nell'accrescere la loro probabilità.

Negli ultimi anni in particolare stanno emergendo nuove necessità per i sistemi informativi aziendali e per la loro componente informatica. Acquisizioni e fusioni di aziende, nuovi business e la conseguente necessità di rapide riconfigurazioni stanno diventando una necessità assoluta per le aziende medio-grandi. Per questo sono in corso di definizione standard tecnologici per arrivare alla integrazione rapida e totale dei componenti informatici interni all'azienda. Ma a questo deve accompagnarsi una migliore visione delle necessità di business dell'azienda stessa, con nuove metodologie di analisi, che saranno trattate nei prossimi capitoli.

Domande

1. Cos'è un sistema? In che modo la nozione di sistema è utile?
2. Cos'è un modello e come aiuta a capire la realtà?
3. Cos'è il sistema informativo di un'azienda e quali sono le sue componenti?
4. Che differenza esiste tra il sistema informativo e il sistema informatico?
5. Cosa significa il concetto di granularità?
6. Come può evolvere il sistema informativo?

Bibliografia

[AIRO 2001] Associazione Italiana di Ricerca Operativa, su Web <http://www.airo.org>

[Aris 1994] R. Aris - *Mathematical Modelling Techniques* – Ed. Dover Publications, New York, 1994

[De Marco 2000] M. De Marco - *Sistemi Informativi Aziendali* - Franco Angeli Edizioni, Milano, 2000

[GS 1990] P.S. Goodman, L.S. Sproull et Al. - *Technology and Organization* – Ed. Jossey-Bass, San Francisco, 1990

[ISO544 2000] *L'Approccio per Processi e la ISO 9001:2000, documento ISO TC 176 SC2 N544 R3*, su Web <http://www.aicq.it/vision2000/Approccio.pdf>

[MR 1988] M.L. Markus and D. Robey -*Information technology and organizational change: causal structure in theory and research* - Management Science, vol.34, pp.583-598, 1988

[Orlandi 1998] T. Orlandi - *Linguistica, sistemi, modelli* - in Atti del Convegno: Il ruolo del modello nella scienza e nel sapere, Accademia dei Lincei, Roma, 27-28 ottobre 1998, su Web <http://rmcisadu.let.uniroma1.it/~orlandi/modello.html>

[PCE 2002] F. Heylighen, C. Joslyn, e V. Turchin per il Principia Cybernetica Project, su Web <http://pespmc1.vub.ac.be>

I processi aziendali

Un modello fondamentale: il processo aziendale

Un processo aziendale (noto anche come business process o processo business) può essere definito come “un insieme organizzato di attività e di decisioni, finalizzato alla creazione di un output effettivamente domandato dal cliente, e al quale questi attribuisce un valore ben definito” (E. Bartezzaghi in [BSV 1999]). In altre parole, si può dire che un processo business è un insieme di attività (sequenze di decisioni e azioni) che l’organizzazione svolge per realizzare un risultato definito e misurabile (prodotto o servizio), che trasferisce valore al fruitore del prodotto o servizio (cliente) e che contribuisce al raggiungimento della missione dell’organizzazione (si veda [Pierantozzi 1998]). Il concetto di **valore** è fondamentale, in quanto uno degli scopi fondamentali della modellazione tramite processi delle attività aziendali è proprio un ausilio alla misurazione del valore prodotto.

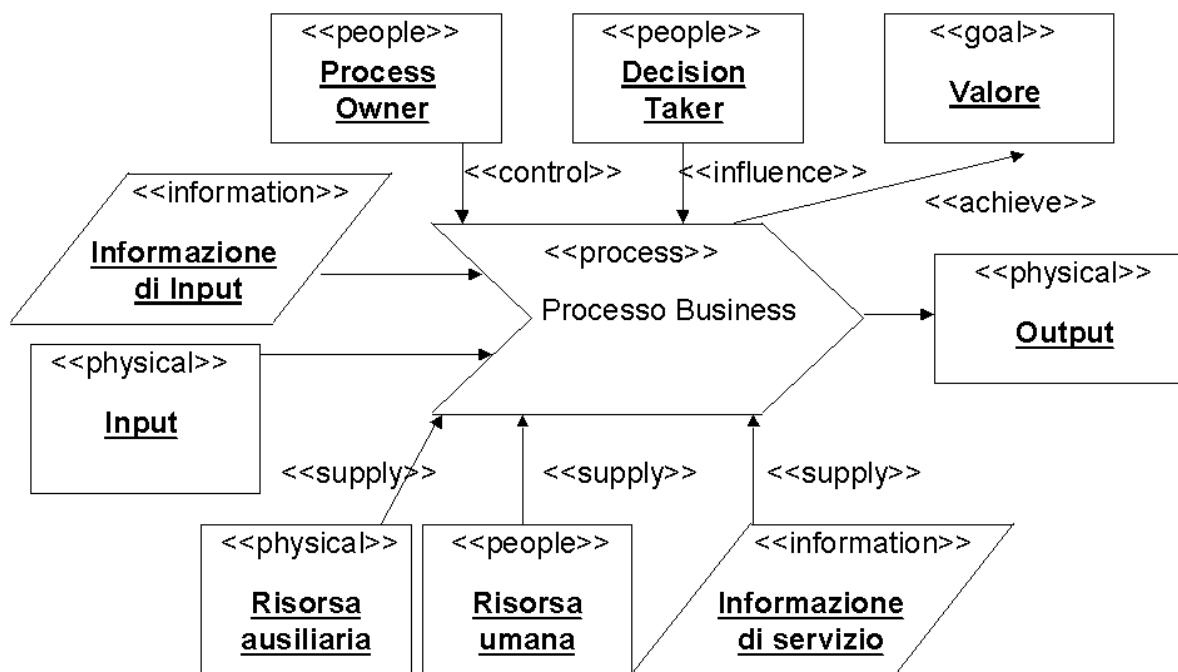


Figura 2.1: Schematizzazione di un generico processo business; viene usata la simbologia di UML for Business (definita in [EP 2000]) con gli stereotipi, identificati dai simboli <<...>>, a rappresentare le categorie e le relazioni; in particolare, i romboidi identificano elementi “immateriali” di informazione.

Da un punto di vista formale quindi un processo business, come mostrato in figura 2.1, viene identificato da vari parametri ed entità. Per definire (e quindi conoscere) un processo, è necessario identificare con precisione tutte le sue componenti, qui di seguito riportate (si veda [EP 2000]).

1. Input del processo, ossia le entità che vengono trasformate dal processo stesso, di tipo materiale (es. le materie prime di un processo di produzione), o immateriale (informazioni di vario tipo);
2. Output del processo, ossia i prodotti del processo stesso (beni materiali, servizi, informazione);
3. Risorse ausiliarie per il processo, ossia entità che contribuiscono al funzionamento del processo stesso, ma che non sono trasformate dal processo stesso (ad esempio, in un processo di produzione appartengono a questa categoria le macchine utensili, in un processo amministrativo appartengono a questa categoria i PC e il software gestionale che su di essi opera);
4. Risorse umane che compiono il processo (ad esempio operai nel processo di produzione, impiegati nel processo amministrativo);
5. Risorse organizzative che impongono regole e vincoli per il funzionamento del processo;
6. Risorse umane che possono prendere decisioni sul funzionamento del processo (decision takers);
7. Risorse umane che sono responsabili per il processo, a cui è affidato il compito di sovraintendere al processo stesso per farlo funzionare al meglio;
8. Costi del processo, dovuti a tutte le componenti del processo stesso (input, energia, manutenzione delle risorse ausiliarie, risorse umane coinvolte);
9. Destinatario dell'output, ovvero il cliente del processo;
10. Valore aggiunto che il processo genera, definito attraverso la qualità dell'output, per la quale il cliente del processo è disposto a pagare, generando quindi l'utile del processo che ne rende possibile il funzionamento; strettamente associato al valore è l'obiettivo del processo.

In questa accezione il processo viene visto come scatola nera, in cui entrano input e risorse ausiliarie e da cui escono i prodotti finiti. Nel caso si debba esaminare la struttura interna del processo, si giunge alla sua scomposizione in **sottoprocessi**, **attività** e **azioni** od **operazioni**, queste ultime definibili come attività *atomiche* e quindi *non ulteriormente scomponibili*. Sono anche possibili diverse scomposizioni in cui l'attenzione non viene posta solo sulla sequenza di attività ma anche su altri elementi, come, per esempio, attori esterni. Tali scomposizioni saranno esaminate in dettaglio nel capitolo 4.

Un altro punto di vista, strettamente collegato, è la visione analitica del processo (si veda [BFM 2001]), secondo la quale:

- I processi sono formati da **attività**, collegate nel tempo e nello spazio e svolte dalle risorse dell'azienda (uomini e mezzi)
- Partendo da **input** definiti, le attività producono un **output** utilizzabile da **clienti**
- Le attività possono essere ulteriormente scomposte in **azioni** o **operazioni** (atomiche, non ulteriormente scomponibili).

Quindi un Business process viene definito come tupla

BP(A,I,O,C) dove

- A = attività, formate da una serie di azioni fisiche o decisioni manageriali
- I = input del processo, formati da materie prime o risorse aziendali (uomini e mezzi)
- O = output del processo, formato da beni materiali o immateriali, servizi
- C = clienti, destinatari dell'output del processo.

L'output di un processo può poi costituire l'input di un processo successivo così come l'input di un processo può essere l'output di quello precedente. Da quanto detto si può rilevare come all'interno dell'azienda stessa esista una catena di clienti-fornitori da soddisfare. Il cliente infatti, non necessariamente deve essere esterno, e cioè acquirente di beni e servizi in cambio di denaro, ma può essere altresì un'unità organizzativa dell'impresa stessa che utilizza il risultato finale di un processo come input necessario per lo svolgimento di altri processi aziendali.

L'interno di un'organizzazione: i processi fondamentali

Indipendentemente dallo scopo, dal settore di appartenenza, dal fatto di essere un'impresa od una onlus o un ente pubblico, in ogni organizzazione è possibile individuare alcuni processi fondamentali. La catena del valore di Porter (definita in [PM 1985]), visualizzata graficamente in figura 2.2, rappresenta una classificazione dei processi, modellizzando il funzionamento dell'intera azienda come una successione di processi.

I processi sono suddivisi in

- **buy side**, come acquisti/approvigionamenti, ossia processi il cui input proviene dai fornitori;
- **inside**, ossia aventi sia input sia output interni all'azienda, che possono essere ulteriormente suddivisi tra **processi primari**, che sono direttamente legati alla produzione del valore del core business dell'azienda e **processi ausiliari** o **secondari** (ad esempio, amministrazione e gestione delle risorse umane), che non generano direttamente un valore, ma producono quei servizi senza i quali l'organizzazione non potrebbe operare;
- **sell side**, il cui output (prodotti o servizi, come l'assistenza) è rivolto direttamente ai clienti esterni all'azienda.

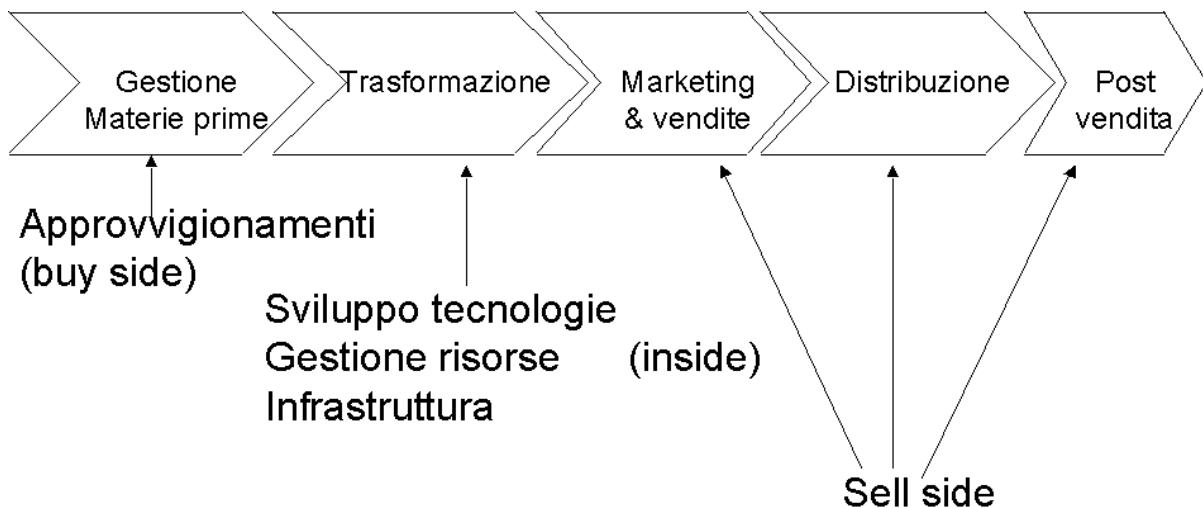


Figura 2.2: La catena del valore di Porter nel caso di un’azienda di produzione

La Piramide di Anthony [Anthony 1965] è un’altra importante classificazione e suddivide i processi in:

- **Processi direzionali** (indicati anche come strategici): concorrono alla definizione degli obiettivi strategici;
- **Processi gestionali** (indicati anche come amministrativi o manageriali): traducono gli obiettivi strategici in obiettivi economici e ne controllano il raggiungimento;
- **Processi operativi**: concorrono alla attuazione degli obiettivi.

Per meglio chiarire la classificazione di Anthony è importante vedere alcuni esempi di processi classificati come direzionali/gestionali/operativi in vari ambiti.

Contesto 1: amministrazione comunale

Esempi di processi operativi: contabilizzazione dei pagamenti dei cittadini, manutenzione delle strade

Esempi di processi gestionali: controllo dei pagamenti, solleciti, confronti mensili tra entrate previste ed effettive, monitoraggio dell'inquinamento

Esempi di processi direzionali: verifica dei costi e dei ricavi relativi ai servizi sociali, definizione di nuove tariffe, piani regolatori

Contesto 2: banca

Esempio di processi operativi: gestione movimenti dei conti correnti

Esempio di processi gestionali: revisione degli scoperti

Esempio di processi direzionali: verifica dell’andamento di un servizio, decisione di aprire nuovi servizi

Contesto 3: azienda

Esempio di processi operativi: registrazione costi delle commesse

Esempio di processi gestionali: controllo scostamenti settimanali tra preventivo e consuntivo

Esempio di processi direzionali: scelta delle aree di mercato più convenienti

Anche i dati oggetto dei processi possono essere suddivisi fra direzionali, gestionali ed operativi. Il seguente esempio, tratto dal contesto aziendale, ma applicabile anche a molti altri contesti, chiarifica il concetto.

Dati operativi: importi di versamenti, ore di presenza dei dipendenti

Dati di controllo (gestionali): saldi mensili, lavoro mensile di ciascun reparto

Dati di pianificazione (direzionali): dati macroeconomici, indicatori generali, dati di budget

Esistono criteri per distinguere tra le tre categorie di dati e processi durante l'analisi dei processi nei contesti dei vari tipi di organizzazione:

- **Tempo:** lunga/media/bassa durata
- **Orizzonte temporale:** passato+futuro/presente
- **Livello organizzativo:** Top management/quadri, intermedi/operativi
- **Importanza della decisione:** alta/media/bassa
- **Livello di aggregazione:** alto/medio/basso

Dai processi del modello di Porter derivano alcuni macro processi con le loro proprietà specifiche, presenti in forma più o meno ampia in quasi tutte le organizzazioni:

- **Ciclo attivo**, ossia l'insieme di tutte le attività legate alla vendita ai clienti, e quindi generanti il fatturato dell'azienda; il singolo processo di vendita è il componente principale del ciclo attivo;
- **Ciclo passivo**, ossia l'insieme di tutte le attività legate all'acquisto di prodotti e/o materie prime per il processo produttivo, o di servizi da rivendere, e quindi generanti le spese di approvvigionamento dell'azienda;
- **Gestione magazzino;**
- **Produzione;**
- **Amministrazione e gestione costi;**
- **Manutenzione impianti ed attrezzature;**
- **Gestione risorse umane;**
- **Gestione globale costi ed attività;**
- **Gestione della qualità;**
- **Marketing**, strettamente connesso con il ciclo attivo, ma anche con la produzione;
- **Gestione delle relazioni col cliente** (noto anche come Customer Relationship Management o CRM), che comprende l'assistenza post vendita e che deve operare a stretto contatto con il Marketing;
- **Gestione della catena di approvvigionamento** (meglio noto con l'acronimo SCM dalle parole inglesi Supply Chain Management), insieme dei processi di gestione aziendale che consentono di ottimizzare la consegna di prodotti, servizi ed informazioni dal fornitore al cliente, sia dal lato acquisti, sia

internamente, sia dal lato vendite, attraverso un uso appropriato delle tecniche di logistica.

L'insieme di questi processi viene compiuto dalle risorse umane presenti nelle varie divisioni aziendali, ossia nei dipartimenti dell'azienda specializzati per funzione (ad esempio, ufficio vendite, ufficio acquisti, ufficio del personale). Molti processi (i cicli attivo e passivo in primis) sono trasversali a più divisioni, e devono quindi essere definiti nella maniera più chiara possibile per potere essere efficienti, permettendo l'individuazione di eventuali inefficienze o colli di bottiglia. Inoltre i processi sono strettamente interconnessi, ad esempio le informazioni prodotte in output da alcuni di essi sono spesso input di altri processi. Il valore aggiunto, per l'impresa, nell'utilizzare una visione per processi piuttosto che per funzioni risiede sostanzialmente nell'obiettivo generale di creazione del valore e dell'analisi di efficienza dei processi che conducono alla creazione di tale valore. Una visione per processi sembra facilitare la realizzazione di obiettivi di profitto e di monitoraggio più efficace delle performance di costo, tempo e qualità. Tali meccanismi di controllo infatti consentono di far funzionare meglio i processi creando soddisfazione al cliente e quindi valore per l'impresa.

Il sistema azienda: visione per funzioni vs. visione per processi

Le funzioni sono aggregazioni di uomini e mezzi necessari per lo svolgimento di attività della stessa natura (D. Pierantozzi). In un'impresa organizzata per funzioni le attività simili, che assolvono cioè la stessa funzione, che richiedono le stesse competenze e che utilizzano lo stesso tipo di risorse e di tecnologie, vengono raggruppate in un'unità organizzativa sotto un'unica responsabilità (E. Bartezzaghi). Esempi sono la funzione acquisti, vendite, produzione, amministrativa, ecc... L'intera azienda viene dunque suddivisa in unità organizzative funzionali, ciascuna delle quali potrà poi suddividersi in reparti e/o uffici, a seconda delle esigenze; ad esempio, la funzione amministrativa si può suddividere in ufficio contabilità, ufficio clienti, ufficio fornitori, ecc..., la funzione produzione può suddividersi in reparto assemblaggio, reparto confezioni, reparto controllo qualità, ecc...

Le funzioni raggruppano quindi attività che hanno la stessa natura mentre i processi sono formati da attività anche di diversa natura, ma che sono finalizzate al raggiungimento dello stesso output. Si può affermare dunque che i processi aziendali "tagliono trasversalmente" le strutture organizzative, perché richiedono il contributo di diverse unità funzionali: un processo "attraversa" più funzioni o analogamente più funzioni concorrono alla realizzazione di un unico processo.

Un esempio può rendere maggiormente comprensibili questi concetti. Consideriamo il processo di gestione degli ordini. In una visione per processi lo possiamo scomporre nelle seguenti attività:

- Ricezione dell'ordine
- Inserimento dell'ordine nel sistema informativo
- Valutazione dell'ordine (analisi posizione cliente, controllo del suo fido...)
- SE il cliente non rientra entro parametri "accettabili", blocco dell'ordine
- ALTRIMENTI generazione della distinta di prelievo

- Prelievo dal magazzino
- Controllo qualità
- Pesatura e imballo
- Generazione bolla/fattura
- Spedizione

Si noti che questa rappresentazione testuale, per quanto chiara, potrebbe essere visualizzata in modo più immediatamente evidente con una rappresentazione semi-grafica come quelle di UML for business, come si vedrà in seguito.

Questa successione di attività deve essere “mappata” sulle divisioni funzionali dell’azienda, ovvero sulle persone che le formano, per esempio:

- gli *addetti amministrativi* si occupano della ricezione dell’ordine e del suo inserimento nel sistema informativo;
- dopo di ché *l’ufficio amministrativo, commerciale, recupero crediti* valuteranno la situazione e la solvibilità del cliente e decideranno se rendere esecutivo l’ordine o se bloccare il tutto; se l’ordine procede, i *magazzinieri* e il *reparto spedizioni* si occuperanno di prelevare i beni e prepararli per la loro spedizione
- infine *l’ufficio amministrativo* provvederà a generare la fattura o la bolla e ad aggiornare la contabilità aziendale.

Il valore aggiunto, per l’impresa, nell’utilizzare una visione per processi piuttosto che per funzioni, risiede sostanzialmente nell’obiettivo generale di **creazione del valore**. I processi devono comunque sempre essere “mappati” su divisioni aziendali e persone, ovvero sulle risorse umane ed organizzative che li realizzano, come spiegato nel paragrafo seguente.

Processi ed organizzazione aziendale

I processi sono composti di attività, la cui successione genera un flusso di attività e deve essere soggetta ad una opportuna organizzazione, con assegnamenti della esecuzione di ciascuna fase od attività ad una divisione funzionale ed eventualmente, in casi di estrema precisione, anche con gli assegnamenti di una singola attività elementare o di una azione, ad una singola risorsa umana [BFM 2001].

Il flusso di attività, ovvero la sequenza delle attività attraverso cui è svolto il processo determina la durata di quest’ultimo, attraverso la somma delle durate delle singole attività componenti. La modellazione del processo può essere più o meno precisa, con la sequenza attività, gli attori coinvolti ed i vincoli (temporali o logici) da rispettare.

Le informazioni associate alla successione di attività possono essere schematizzate come segue:

- Attività:
 - Tipologia di attività (es. trasformazione, trasporto, ecc...)
 - Durata
 - Volumi
 - Risorse ausiliarie e Tecnologie coinvolte
- Sequenza delle attività:
 - Alternative nella sequenza
 - Natura del flusso di collegamento (fisico, informativo o entrambi)

- Attori
 - Tipologia attori
 - Azioni svolte sulle attività del flusso
- Eventi
 - Tipologia evento (scadenza, messaggio, ecc...)
 - Conseguenza evento sull’attività (avvia, ferma, modifica ecc...)
- Oggetti coinvolti
 - Natura (fisico, informativo o entrambi)
 - Profilo temporale (es. informazione permanente o temporanea)

L’associazione tra processo ed organizzazione richiede poi la conoscenza di

- Organigramma: gerarchia delle responsabilità e delle autorità nell’organizzazione
- Proprietà logiche: mandato, compiti, processi
- Proprietà quantitative: organici della struttura, volumi di lavoro
- Efficienza e sua misurazione

L’organigramma deve definire la gerarchia delle responsabilità a vari livelli di dettaglio e può essere corredata dalle Tabelle delle proprietà, comprendenti la descrizione del mandato, l’elenco dei compiti assegnati, l’elenco dei processi svolti, gli organici ed i volumi di lavoro. Il metodo LRC (Linear Responsibility Charting), evoluzione del metodo omonimo introdotto negli Stati Uniti negli anni’50, prevede una specificazione dei ruoli e delle strutture nei processi, attraverso una visione tabellare della responsabilità organizzativa, che integra l’organigramma ed incrocia le attività o fasi del processo con le strutture organizzative o con loro parti (anche singole risorse umane).

In figura 2.3 viene rappresentato un esempio di mappatura delle varie attività di gestione di un ordine su un organigramma. L’analisi di un processo può avere obiettivi anche molto diversi fra loro e prendere in esame caratteristiche molto diverse dei processi stessi. Si rendono necessari allora strumenti di ausilio per lo studio di queste diverse caratteristiche. I diagrammi di UML for Business, che pongono in evidenza le diverse caratteristiche, possono essere un valido aiuto e saranno presentati nel capitolo 4. Inoltre è importante evidenziare il fatto che i processi hanno sempre un contenuto informativo, sia al proprio interno sia come input ed output verso l’esterno. Un esempio di contenuto informativo può essere dato dalla bolla di accompagnamento di un carico di materie prime in arrivo, ossia l’input di un processo buy-side di approvvigionamento. Diviene allora fondamentale definire esattamente le caratteristiche della “risorsa informazione”, prima di procedere nella analisi e comprensione dei processi. Tale risorsa viene analizzata nel prossimo capitolo.

	Fliale	Dir. Com.	Dir. Distr.	Magazzino	Sped.	Contab. Cli.
Ricezione ordine	E	D	I	I		A
Evasione ordine da magazzino	I		D	E	I	
Spedizione			I	A	E	I
Fatturazione			I			E

D = decide/approva E = esegue I = è informato A = assiste

Figura 2.3: Esempio di LRC, tratto da [BFM 2001].

Domande

1. Cos'è un processo business e quali sono le sue caratteristiche?
2. Quali sono le informazioni necessarie se si vuole conoscere completamente un processo business?
3. Cos'è la catena del Valore di Porter? Perchè è così importante nel contesto di un'azienda?
4. Cosa si intende per processi primari e processi secondari o ausiliari?
5. Classificazione gerarchica dei processi: cos'è e come è strutturata la Piramide di Anthony?
6. Come avviene solitamente la mappatura dei processi sulle strutture aziendali?
7. Cos'è e a cosa serve la LRC?
8. Cosa sono il ciclo attivo e il ciclo passivo?

Bibliografia

[Anthony 1965] R. Anthony - *Planning and Control Systems: A Framework for Analysis*. – Ed. Harvard Business Review, 1965

[BFM 2001] G. Bracchi, C. Francalanci, G. Motta - *Sistemi Informativi e aziende in rete* – Ed. McGraw-Hill Italia, Milano, 2001

[BSV 1994] E. Bartezzaghi, G. Spina, R. Verganti - *Nuovi Modelli d'impresa e tecnologie d'integrazione* – Franco Angeli Edizioni, 1994

[BSV 1999] E. Bartezzaghi, G. Spina, R. Verganti - *Organizzare le PMI per la crescita. Come sviluppare i più avanzati modelli organizzativi: gestione per processi, lavoro per progetti, sviluppo delle competenze* - Ed. Il Sole 24 Ore, 1999

[Davenport 1993] , T.H. Davenport - *Process Innovation* - Ed. Harvard Business School Press, Boston, MA, 1993

[De Marco 2000] M. De Marco - *Sistemi Informativi Aziendali* - Franco Angeli Edizioni, Milano, 2000

[EP 2000] H.E. Eriksson, M. Penker - *Business Modeling with UML* - Ed. Wiley and Sons, 2000

[HC 2004] M. Hammer and J. Champy - *Reengineering the Corporation : A Manifesto for Business Revolution - Updated Edition* - Ed. Collins, 2004

[LL 2004] K Laudon, J Laudon – *Management dei sistemi informativi* – Ed. Pearson Education Italia, Milano, 2004

[Malhotra 1998] Y. Malhotra - *Business Process Redesign: An Overview* - IEEE Engineering Management Review, vol. 26, no. 3, Fall 1998, su Web <http://www.kmbook.com/bpr.htm>

[Pierantozzi 1996] D. Pierantozzi - *Riduzione del tempo di risposta al mercato e creazione di valore* - Finanza, Marketing e Produzione, n°4 1996, pag. 201.

[Pierantozzi 1998] D. Pierantozzi - *La gestione dei processi nell'ottica del valore: Miglioramento graduale e reengineering. Criteri, metodi, esperienze* - Ed. Egea, Milano, 1998

[PM 1985] M.E. Porter, V.E. Millar – How Information Gves You Competitive Advantage – Harward Business Review 63 (4), pp.149-161, 1985

[Porter 1985] M.E. Porter - *Competitive Advantage* – The Free Press, New York, 1985

La risorsa informazione

La risorsa informazione e le sue caratteristiche

Il lavoro del sistema informativo ha come oggetto l'**informazione**. I processi aziendali usano al proprio interno e scambiano fra loro informazione. L'informazione ha caratteristiche particolari che contribuiscono a rendere il sistema informativo diverso dai tradizionali settori tecnici di un'azienda.

L'informazione è la principale risorsa scambiata, selezionata ed elaborata nelle attività gestionali di coordinamento e controllo (C. Ciborra, si veda [Ciborra 2002]). In ogni caso un qualunque compito nell'ambito di un'organizzazione, operativo o no, ha un contenuto gestionale e, in quanto tale, elabora informazione (P. Maggiolini, si veda anche [CP 1994]).

L'informazione è una risorsa immateriale (o intangibile) e costituisce la radice di ogni altra risorsa organizzativa immateriale come conoscenza, esperienza individuale, esperienza organizzativa.

L'informazione non viene distrutta dall'uso (non-depletable), permette la creazione di nuova conoscenza (self-generating), non è facilmente misurabile o divisibile o appropriabile e può essere soggetta a obsolescenza (si pensi, ad esempio, alle informazioni relative al mercato azionario dello stesso giorno dell'anno di trent'anni fa, non più utili per gli scambi azionari di oggi).

La costruzione dell'informazione ed il suo uso entro l'azienda può avvenire attraverso quattro stadi successivi qui di seguito presentati (secondo la classificazione di G. Bellinger, N. Shedroff ed altri, per la trattazione completa della quale si consigliano [BCM 1994] e [Shedroff 1994])

- **Dati:** i dati sono materiale informativo grezzo, non (ancora) elaborato dal ricevente, e possono essere scoperti, ricercati, raccolti e prodotti. Sono la materia prima che abbiamo a disposizione o produciamo per costruire i nostri processi comunicativi.

Esempio: l'insieme dei valori dei dati di accesso ad un determinato portale Web.

- **Informazione:** l'informazione viene costruita dai dati elaborati cognitivamente, cioè trasformati in un qualche schema concettuale successivamente manipolabile e usabile per altri usi cognitivi. L'informazione conferisce un significato ai dati, grazie al fatto che li pone in una relazione reciproca e li organizza secondo dei modelli. Trasformare dati in informazioni significa organizzarli in una forma comprensibile, presentarli in modo appropriato e comunicare il contesto attorno ad essi.

Esempio: il risultato dell'analisi dei dati di accesso al portale Web.

- **Conoscenza:** la conoscenza è informazione applicata, come un senso comune, o non comune, che "sa" quando e come usarla. E' attraverso l'esperienza che gli esseri umani acquisiscono conoscenza. E' grazie alle esperienze fatte, siano esse positive o negative, che gli esseri umani arrivano a comprendere le cose. La conoscenza viene comunicata sviluppando interazioni stimolanti, con gli

altri o con le cose, che rivelano i percorsi nascosti e i significati dell'informazione in modo che possano essere appresi dagli altri. La conoscenza è fondamentalmente un livello di comunicazione partecipatorio. Dovrebbe rappresentare sempre l'obiettivo a cui tendere, poiché consente di veicolare i messaggi più significativi.

Esempio: azioni di marketing svolte sulla base delle informazioni tratte dall'analisi dei dati di accesso al portale Web

- **Saggezza:** la saggezza è verità “eterna” distillata dalla conoscenza. L'informazione costituisce lo stimolo di un'esperienza, mentre la saggezza può derivare dalla comprensione del messaggio che acquisiamo attraverso l'esperienza. La saggezza è il livello di comprensione più indefinito e più intimo. Essa è una sorta di “meta-conoscenza” di processi e relazioni che viene acquisita attraverso l'esperienza. E' il risultato di contemplazione, valutazione, retrospezione e interpretazione - tutti processi estremamente personali. Non è possibile creare la saggezza allo stesso modo di come vengono creati i dati e le informazioni, e non è possibile condividerla con gli altri, come invece avviene per la conoscenza. E' soltanto possibile creare esperienze che siano in grado di offrire opportunità e descrivere dei processi.

Esempio: regole di azione e di uso dello strumento Web estratte dalla conoscenza guadagnata dall'esperienza.

“La conoscenza consiste di fatti, verità, credenze, prospettive e concetti, giudizi e aspettative, metodi e saper fare” (K.M. Wiig, si vedano anche [Wiig 1994] e [Wiig 2000])

Processi gestionali efficienti sono in grado di instaurare circoli virtuosi di generazione di conoscenza e arricchimento dell'informazione disponibile. Tali circoli, in linea teorica, generano un aumento delle prestazioni dei processi gestionali e dell'organizzazione in toto.

Ma una quantità molto grande di informazione non necessariamente genera conseguenze positive. Infatti, si definisce “overload o sovraccarico di informazione” (si veda, ad esempio, [Heylighen 1999]) l'aumento incontrollato di informazione complessivamente disponibile, che comporta un maggiore sforzo per filtrare l'input costituito da “dati grezzi”, estraendone le informazioni utili. Questo fenomeno si realizza spesso quando avviene un abuso da parte dei processi organizzativi della possibilità di creare nuova informazione, a partire da quella in input. L'avvento di Internet ed il suo uso non regolamentato per la ricerca di dati entro le aziende spesso ha contribuito alla creazione di overload d'informazione e al boom dell'uso di strumenti software di ausilio alla gestione della conoscenza.

Rappresentazione dell'informazione

Per esistere nel mondo fisico, l'informazione deve essere rappresentata in modo fisico. L'informazione può essere rappresentata come variazioni di grandezze fisiche entro opportuni supporti fisici, come per esempio colori su carta, livelli di tensione entro circuiti elettronici. Per potere essere immagazzinata e trasmessa l'informazione necessita sempre di supporti fisici: i supporti di immagazzinamento possono essere

suddivisi fra archivi cartacei ed archivi informatici, mentre i supporti di trasmissione possono essere suddivisi fra “canali tradizionali” come posta, fax, telex, e sistemi digitali più moderni (EDI, Internet...). In ogni caso l’archivio informatico richiede l’esistenza di supporti hardware adeguati (computer, insiemi di dischi, nastri...) e il canale di trasmissione digitale richiede la presenza di sistemi di comunicazione (cavi o supporti wireless, apparati di rete, software di gestione della rete...).

In una accezione informatica i dati divengono sottocomponenti di informazione, che possono essere rappresentati in forma digitale, attraverso opportune codifiche. In realtà spesso nel contesto informatico, in modo impreciso, il termine dati viene usato direttamente come sinonimo di informazione, mentre, come visto, i dati sono la materia prima “grezza” da cui viene tratta l’informazione attraverso un processo di elaborazione.

In generale, da un punto di vista informatico l’informazione viene rappresentata (e la sua quantità misurata) come insiemi di **byte**. Si ricordi che un byte è un insieme di 8 bit, che, mentre il bit esprime semplicemente una informazione si/no, può esprimere $2^8=256$ valori. Le codifiche associate agli standard assegnano significati particolari a tali valori, per esempio il codice ASCII (American Standard Code for Information Interchange) associa ai valori da 0 a 255 rappresentati dai byte le lettere (maiuscole e minuscole) dell’alfabeto latino internazionale, le lettere accentate, le cifre da 0 a 9, i segni di interpunkzione, parentesi, simboli matematici ed altro, oltre che caratteri di controllo che rappresentano, ad esempio, l’“a capo” alla fine di una riga.

Il codice ASCII è in realtà un insieme di vari codici diversi, in quanto, pur essendo univoca l’assegnazione dei caratteri per i codici con il numero inferiore a 127, non è univoca quella per i numeri compresi fra 128 e 255, il cosiddetto codice ASCII esteso. Infatti l’alfabeto latino, usato nella scrittura di molte lingue nel mondo, presenta una grande quantità di varianti grafiche: si va dalle semplici vocali accentate (accento grave à, acuto á, circonflesso â, dieresi ä, tilde ã) a lettere modificate (lettere con barrette, cediglie, segni), lettere speciali usate solo in una lingua, segni di punteggiatura particolari (il punto interrogativo ed il punto esclamativo capovolti usati nello spagnolo), simboli di valuta e così via, senza considerare poi che gran parte di questi segni presentano le due forme maiuscola e minuscola.

Le varianti sono talmente numerose che i 128 byte della tabella estesa non sono purtroppo sufficienti a rappresentarle tutte, per questo motivo esistono diverse estensioni della tabella ASCII: lo standard ISO 8859 prevede 15 diverse estensioni, comprese quelle per gli alfabeti diversi dal latino, ma esistono anche ulteriori estensioni, non riconosciute dall’ISO, e create, per esempio, dalla Microsoft per i sistemi Windows o dalla Apple per i Macintosh. La tabella ASCII estesa tipicamente utilizzata in Italia è quella dell’Europa occidentale, creata per le lingue germaniche e neolatine (escluso il rumeno). Altre estensioni usate in Europa sono la Centro Europea per i paesi dell’Europa orientale (lingue slave, ungherese, rumeno), la Turca, la Cirillica e la Greca.

Questa coesistenza fra diverse versioni del codice ASCII produce spesso discordanze nella visualizzazione dei file di testo. Sarà capitato a molti di aprire un file di testo o ricevere una E-mail e trovare segni apparentemente privi di senso al posto di tutte le

lettere accentate. Ciò accade perché chi ha scritto il testo stava usando una tabella estesa diversa da quella usata dal computer del ricevente, che quindi interpreta alcuni byte del file in modo diverso.

Il codice ASCII standard è rappresentato in figura 3.1, mentre l'ASCII esteso in uso in Italia viene rappresentato in figura 3.2.

Codice	Carattere		Codice	Carattere		Codice	Carattere		Codice	Carattere
0	Null		32			64	@		96	`
1	Start of heading		33	!		65	A		97	a
2	Start of text		34	"		66	B		98	b
3	End of text		35	#		67	C		99	c
4	End of transmit		36	\$		68	D		100	d
5	Enquiry		37	%		69	E		101	e
6	Acknowledge		38	&		70	F		102	f
7	Bell		39	'		71	G		103	g
8	Backspace		40	(72	H		104	h
9	Tab orizzontale		41)		73	I		105	i
10	Fine linea (LF)		42	*		74	J		106	j
11	Tab verticale		43	+		75	K		107	k
12	Form Feed		44	,		76	L		108	l
13	Ritorno carrello (CR)		45	-		77	M		109	m
14	Shift out		46	.		78	N		110	n
15	Shift in		47	/		79	O		111	o
16	Data link escape		48	0		80	P		112	p
17	Device ctrl 1		49	1		81	Q		113	q
18	Device ctrl 2		50	2		82	R		114	r
19	Device ctrl 3		51	3		83	S		115	s
20	Device ctrl 4		52	4		84	T		116	t
21	Neg. Acknowledge		53	5		85	U		117	u
22	Synchronous idle		54	6		86	V		118	v
23	End trans. Block		55	7		87	W		119	w
24	Cancel		56	8		88	X		120	x
25	End of medium		57	9		89	Y		121	y
26	Substitution		58	:		90	Z		122	z
27	Escape		59	;		91	[123	{
28	File separator		60	<		92	\		124	
29	Group separator		61	=		93]		125	}
30	Record separator		62	>		94	^		126	~
31	Unit separator		63	?		95	_		127	Del

Figura 3.1: Il codice ASCII standard, con i valori da 0 a 127. I primi 32 codici sono segnali di controllo, pensati per le telescriventi per cui l'ASCII fu inizialmente creato. Alcuni sono ancora in uso entro il PC, come per esempio i codici 10 e 13 che servono per l'a capo nei testi. Il carattere 32 è lo spazio.

Codice	Carattere		Codice	Carattere		Codice	Carattere		Codice	Carattere
128	€		160			192	À		224	à
129			161	í		193	Á		225	á
130	,		162	¢		194	Â		226	â
131	f		163	£		195	Ã		227	ã
132	"		164	¤		196	Ä		228	ä
133	...		165	¥		197	Å		229	å
134	†		166	:		198	Æ		230	æ
135	‡		167	§		199	Ç		231	ç
136	^		168	"		200	È		232	è
137	%		169	©		201	É		233	é
138	Š		170	ª		202	Ê		234	ê
139	<		171	«		203	Ë		235	ë
140	Œ		172	¬		204	Ì		236	ì
141			173			205	Í		237	í
142	Ž		174	®		206	Î		238	î
143			175	-		207	Ï		239	ï
144			176	°		208	Ð		240	ð
145	'		177	±		209	Ñ		241	ñ
146	,		178	²		210	Ò		242	ò
147	"		179	³		211	Ó		243	ó
148	"		180	'		212	Ô		244	ô
149	•		181	µ		213	Õ		245	õ
150	—		182	¶		214	Ö		246	ö
151	---		183	.		215	×		247	÷
152	~		184	,		216	Ø		248	ø
153	™		185	¹		217	Ù		249	ù
154	š		186	º		218	Ú		250	ú
155)		187	»		219	Û		251	û
156	œ		188	¼		220	Ü		252	ü
157			189	½		221	Ý		253	ý
158	ž		190	¾		222	Þ		254	þ
159	ÿ		191	¸		223	Ը		255	Ӧ

Figura 3.2: Il codice ASCII esteso in uso in Italia, comune con altri paesi dell'Europa Occidentale.

Per ovviare al problema dei differenti formati è stato creato un nuovo standard internazionale detto Unicode, definito dalla Unicode Consortium e dalla International Organization for Standardization (ISO 10646), che rappresenta i caratteri usando 2 byte (16 bit).

Con 2 byte il numero di combinazioni possibili diventa $256 \times 256 = 65.536$, perciò Unicode supporta 65.536 diversi segni. Si riescono così a rappresentare non solo tutte le varianti dell'alfabeto latino, ma anche tutti gli altri alfabeti (greco, cirillico, arabo, ebraico, hindi, thai, ...), oltre all'insieme degli ideogrammi cinesi e giapponesi (che sono alcune decine di migliaia, anche se poi ne vengono effettivamente utilizzati solo

poche migliaia). Lo standard definitivo è però ancora in corso di definizione. Lo svantaggio dell'Unicode, rispetto all'ASCII, è che le dimensioni dei file di testo risultano comunque raddoppiate (vengono usati 2 byte per carattere, invece di 1 solo). Un altro esempio di codifica di informazione è quella delle immagini in toni di grigio, nelle quali, nella cosiddetta codifica diretta, al valore 0 viene assegnato il nero ed al valore 255 il bianco, mentre i valori intermedi corrispondono ai vari toni di grigio, come rappresentato in figura 3.3.

Esistono standard pubblici per l'espressione di dati ed informazioni come insiemi di byte, riconosciuti da molti produttori di strumenti informatici, come, per esempio, il sopra descritto codice ASCII per i testi semplici, il formato MP3 per la musica, il formato TIFF per le immagini, il formato ebXML per documenti commerciali complessi. Ma spesso i produttori di software hanno anche sviluppato rappresentazioni proprietarie dei dati, come ad esempio il formato .doc di Microsoft Word, il formato .xls di Microsoft Excel, o i tracciati proprietari per la rappresentazione degli archivi di programmi gestionali.

Ancora oggi il problema della conversione dei formati di rappresentazione dati, fra applicazioni o sistemi diversi, per permettere loro di comunicare informazione, è critico entro i sistemi informatici di qualunque azienda od organizzazione e può generare costi notevoli o vincoli e criticità nei processi cui l'informazione è associata.



Figura 3.3: Immagini ad 8 bit, in cui la luminosità di ogni pixel (elemento fondamentale componente l'immagine stessa) è rappresentata con un valore da 0 a 255. In particolare, a sinistra è la codifica diretta (0=nero, 255=bianco) e a destra l'applicazione della visualizzazione secondo la codifica inversa (0=bianco, 255=nero) alla stessa immagine, che risulta visualizzata in negativo.

Flussi informativi e flussi informatici

Si definiscono **flussi informativi** i flussi di informazioni che vengono trasferite tra diverse componenti di un'impresa o tra l'impresa e i propri clienti. Possono avvenire attraverso diversi mezzi fisici di comunicazione:

- Voce, Telefono
- Fax
- Posta cartacea

- EDI (Electronic Data Interchange), Internet, reti informatiche

Esempi di flussi informativi possono essere l'aggiornamento del contenuto di portali Web, le notizie in arrivo e in partenza dalle agenzie di stampa, gli scambi azionari, le transazioni elettroniche bancarie, i documenti associati alla compravendita di prodotti e servizi (richiesta d'ordine, offerta, conferma d'ordine...), il trasferimento di progetti alla produzione, la spedizione di dati sanitari tra ospedali ecc...

Come già in precedenza accennato, ad ogni processo sono associati uno o più flussi informativi, che trasmettono le informazioni tra le varie attività che formano il processo stesso e sono necessari per il suo funzionamento.

Quando i flussi informativi vengono realizzati attraverso un canale digitale, trasferendo dati (e informazione) direttamente tra strumenti informatici, senza intervento diretto di operatori umani, se non con funzione di verifica e controllo, prendono il nome di **flussi informatici**.

Ai flussi informativi viene spesso associato il concetto di **workflow**, definito come l'insieme degli aspetti operativi di una procedura, ossia di un componente di un processo business che specifica come si esegue un'attività od una sua parte. Il workflow, di solito non tradotto con il corrispondente italiano di flusso di lavoro, definisce come i compiti sono strutturati, chi li compie, qual'è il loro ordine relativo, come sono sincronizzati, come è fatto il flusso di informazioni che supporta i compiti e come essi possono essere tracciati permettendo anche di identificare il punto di evoluzione del processo.

Da un punto di vista informatico i dati sono una rappresentazione della realtà di interesse e devono essere organizzati, classificati e archiviati in modo da poter essere facilmente reperiti e trasformati da “fatti grezzi” in informazioni utili a significative decisioni. I dati informatici rendono persistenti (ossia registrano e rendono permanenti, a meno di una esplicita cancellazione o della distruzione fisica del supporto di immagazzinamento) i risultati delle elaborazioni.

Le applicazioni informatiche (dette anche applicazioni software o programmi applicativi) elaborano dati, visti come componenti di informazione ed in genere possono trasformare le informazioni. I dati in ingresso in un'applicazione sono detti dati di Input, mentre il prodotto dell'elaborazione che esce dall'applicazione sono i dati di Output. I dati devono durare oltre l'applicazione che li ha generati/acquisiti, per potere entrare nel patrimonio informativo dell'azienda.

Lo svolgimento dei processi aziendali è basato sui flussi di dati, ovvero sull'insieme dei flussi di informazione gestiti all'interno di una organizzazione. La gestione della informazione comporta Generazione, Memorizzazione, Elaborazione ed Uso dei dati che compongono l'informazione, e si può fare in modo automatico (coinvolgendo quindi le tecnologie informatiche) o manuale. In realtà la maggior parte delle attività che compongono i processi comprendono sia trattamenti automatici sia trattamenti manuali dei dati.

Un flusso informativo, una volta individuati i nodi che attraversa (che possono corrispondere alle attività o, più frequentemente, alle divisioni funzionali dell'azienda che eseguono le attività), può essere visto come una composizione di flussi elementari.

In generale, definendo due nodi, i flussi elementari possono essere suddivisi nelle tipologie seguenti (si vedano [OHE 1999] e [Chappell 2004] per approfondimenti).

- **Invio semplice**, in cui il mittente invia una informazione al destinatario che non manda indietro nulla; un esempio di questa interazione può essere il trasferimento di un documento da un ufficio all’altro, una volta completata la procedura relativa a tale documento da parte del primo ufficio;
- **Richiesta/risposta sincrona** (synchronous request/response): in questo caso l’informazione associata alla richiesta viene inviata dal richiedente (indicato anche come chiamante o mittente) al destinatario (indicato anche come chiamato o richiesto) ed il mittente rimane in attesa, bloccato, in quanto ha bisogno, per procedere nella propria attività, dell’informazione contenuta nel messaggio di risposta che il destinatario gli deve ritornare; un esempio di questo tipo di interazione può essere la richiesta di consegna di un lotto di materie prime da un reparto di produzione al magazzino;
- **Richiesta/risposta asincrona**: in questo caso il mittente invia un messaggio di richiesta al destinatario e poi non si ferma ad attendere, ma prosegue nella sua attività senza aspettare la risposta, che gli giungerà in modo asincrono dopo un certo tempo; un esempio di questa interazione è dato da una pratica che deve subire una verifica da parte di un secondo ufficio, prima che il primo possa proseguire nella sua elaborazione: il primo ufficio passa la pratica e poi prosegue la propria attività su altri documenti, la pratica viene messa in coda nel secondo e dopo un certo tempo sarà verificata e ritornata al primo ufficio, entro il quale poi rimarrà per un certo tempo in coda, prima che venga ripresa l’elaborazione su di essa;
- **Ricezione semplice**: in questo caso il ricevente, magari già impegnato su altre attività riceve l’informazione (fatto talvolta indicato come evento o evento di ricezione) e, immediatamente (reazione sincrona) o dopo un certo tempo, reagisce a tale evento variando la propria attività (ad esempio, elaborando l’informazione ricevuta); un esempio di questo tipo di interazione è rappresentato dalla segretaria che riceve una telefonata ed interrompe la propria attività corrente per rispondere (reazione sincrona), mentre il caso asincrono è rappresentato da qualcuno che pone una pratica sulla scrivania della segretaria, che, ultimato quanto sta facendo, prenderà in esame la pratica stessa, per apporvi un timbro e portarla poi in un altro ufficio.

I tipi di interazione sopra descritti, combinati tra loro in vario modo, compongono flussi informativi anche complessi. Le interazioni asincrone presuppongono l’esistenza di buffer o code temporanee per l’informazione, ove l’informazione stessa rimane memorizzata, in attesa che il nodo destinazione proceda al suo trattamento o alla sua elaborazione.

Per potere trasmettere, memorizzare e recuperare velocemente l’informazione è necessario che questa venga trasmessa, memorizzata e trattata in formato digitale. Per l’impresa è vantaggioso che quanto più possibile dei flussi informativi suddetti avvengano come flussi informatici, in cui le informazioni sono in formato digitale e scorrono in tale formato. Ogni conversione in formato fisico (es. stampa) di una informazione (ad esempio una fattura) produce un costo ed un rallentamento; una

successiva reimmissione del dato in formato digitale richiede tempo e potenzialmente è soggetta ad errori (ad esempio, si pensi all'inserimento dei dati di un documento stampato entro il proprio sistema informatico, con la possibilità di errori di battitura). E' per questo che sempre più i flussi informativi vengono implementati come flussi informatici.

Il passaggio di informazioni associate ad un processo può avere punti critici e di inefficienza, come ad esempio le code di ricezione di documenti viste sopra. Per rendere più rapidi ed efficienti i processi è necessario conoscere ed analizzare nel modo più chiaro possibile le attività e gli altri componenti dei processi stessi, con i loro legami logici e le informazioni che tra essi fluiscono. Tale analisi sarà oggetto del prossimo capitolo.

XML: l’“esperanto” elettronico

Come si è visto nei precedenti paragrafi, il problema della rappresentazione elettronica dell’informazione e, soprattutto dell’interscambio dati fra applicazioni e sistemi diversi, è critico per qualunque azienda od organizzazione.

Per ovviare a questo problema sono state proposte molte soluzioni, negli ultimi anni sta prendendo piede l’uso dell’eXtended Markup Language (XML).

L’XML è un metalinguaggio, ovvero un linguaggio per definire altri linguaggi, basato su tag, ossia particolari parole chiave racchiuse fra i caratteri ASCII ‘<’ e ‘>’, con rilevanza semantica. Rispetto al suo “parente” HTML, che usa tag appositi solo per definire l’aspetto grafico di un documento, XML consente di definire una struttura avente contenuto semantico隐含的 entro i documenti. In pratica quindi un documento XML è un file di testo (in formato ASCII o Unicode) dove, accanto alle informazioni, sono presenti anche metainformazioni (costituite dai tag) che ne definiscono un significato. Ciò consente di interpretare facilmente le informazioni anche a programmi diversi da quelli che hanno creato il file.

XML definisce una sintassi rigorosa per la successione dei tag, per cui si parla di documenti “well formed” quando rispettano tale sintassi. Inoltre, normalmente, si distinguono due categorie di documenti:

1. documenti di definizione, ossia che definiscono il formato di un documento XML vero e proprio, per i quali si può usare il formato Data Type Definition o DTD (purtroppo non basato su XML e quindi in progressivo abbandono proprio per questo motivo) o il formato schema (basato anche esso su XML e quindi elaborabile con gli stessi strumenti che leggono XML);
2. documenti veri e propri con un contenuto di informazione che, non avendolo di solito incluso, hanno il riferimento, di solito in forma di indirizzo Web, dello schema o del DTD che li definisce.

L’XML non riguarda la rappresentazione visiva di un documento, ma solo il suo contenuto semantico. Un linguaggio basato su XML, l’eXtended Stylesheet (XSL) serve ad associare ad un documento XML una rappresentazione grafica, ovvero a trasformare un documento XML in ogni altro tipo di documento (es. doc di word 2000, foglio excel, pagina HTML ecc...). Un esempio di documento XML è rappresentato in figura 3.4, mentre nelle figure successive sono il DTD e lo schema che lo definiscono.

L'uso dell'XML per definire tipi di dati è in continua espansione. Solo per citare alcuni esempi, il nuovo formato di rappresentazione dei documenti di Microsoft Office, denominato OpenXML [OpenXML 2005], ed anche il "rivale" formato OpenDocument, sostenuto tra gli altri da IBM [OpenDoc 2005] si basano su XML. Anche le più moderne tecnologie di integrazione fra i sistemi informatici, sia internamente all'azienda, sia esternamente, come per esempio Web Services, Service Oriented Architecture e Grid Computing, si basano, parzialmente o completamente, sull'uso di XML come componente interno. Tali tecnologie saranno trattate nel capitolo 5. Inoltre esistono veri e propri database XML, come ad esempio Tamino di Software AG (si veda [Tamino 2005]).

Non si deve pensare che XML consenta di rappresentare solo dati di testo, anche dati multimediali o file binari di ogni tipo possono essere inclusi (in modo analogo a quanto avviene con gli allegati della posta elettronica) entro un file XML, o essere associati ad esso.

Lo svantaggio evidente di XML è l'incremento delle dimensioni dei file dovuto ai tag, che, a parità di contenuto informativo vero e proprio, può raggiungere anche il 500% delle dimensioni di un file a "tracciato compatto" cioè che rappresenta al suo interno le sole informazioni in formato digitale, senza alcuna indicazione sulla interpretazione della rappresentazione stessa.

Per approfondimenti sull'XML si consigliano, ad esempio, i siti [XML.com 2005] e [XML.org 2005].

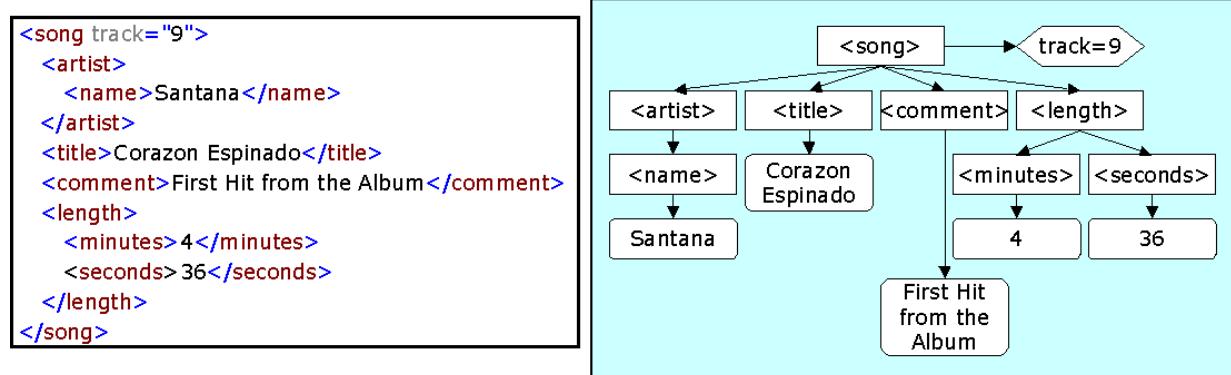


Figura 3.4: un documento XML rappresentante una canzone ed una sua visione grafica ad albero che segue la cosiddetta modalità DOM (esempio tratto da [Tamino 2005]).

```

<!ELEMENT artist (#PCDATA)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT year (#PCDATA)>
<!ELEMENT comment (#PCDATA)>
<!ELEMENT length (#PCDATA)>

<!ELEMENT song (artist?, title, year?, comment?, length)>
<!ELEMENT CD song+>

```

Figura 3.5: il DTD che definisce il documento rappresentato in figura 3.4

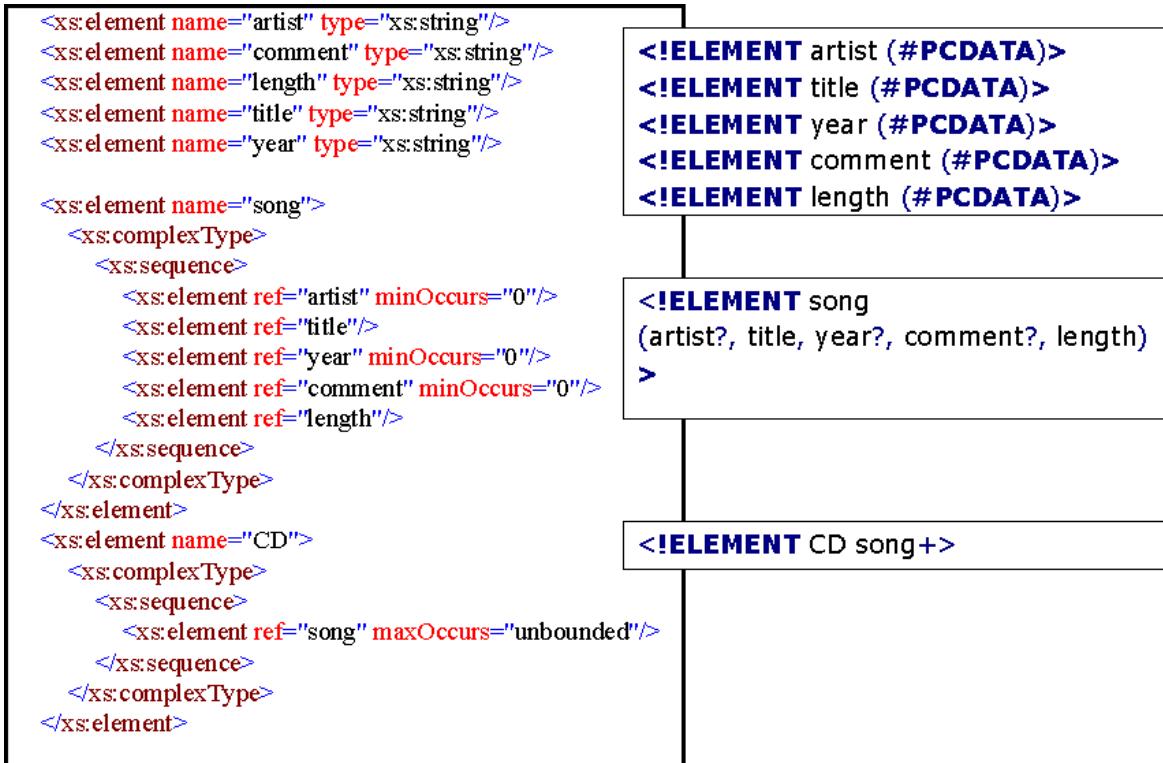


Figura 3.6: lo schema XML che definisce il documento rappresentato in figura 3.4, con a fianco il DTD corrispondente. Nonostante le ridondanze lo schema, basato su XML, risulta preferibile.

Domande

1. Come si può definire la risorsa informazione?
2. Come funziona il ciclo di vita teorico dell'informazione?
3. Cosa si intende per flusso informativo? E cos'è invece un flusso informatico?
4. Cosa si intende per overload informativo?
5. Come viene rappresentata l'informazione nel mondo digitale? Che problematiche sono associate a tale rappresentazione? Come si possono prevenire tali problematiche da un punto di vista organizzativo?
6. Cos'è l'XML? Perché può essere utile entro i sistemi informativi?

Bibliografia

[ACPT 2002] P. Atzeni, S. Ceri, S. Paraboschi, R. Torlone - Basi di Dati: Modelli e Linguaggi di Interrogazione – Ed. McGraw-Hill Italia, Milano, 2002

[ACFPT 2003] P. Atzeni, S. Ceri, P. Fraternali, S. Paraboschi, R. Torlone - Basi di Dati: Architetture e Linee di Evoluzione – Ed. McGraw-Hill Italia, Milano, 2003

[BCM 1994] G. Bellinger, D. Castro, A. Mills - Data, Information, Knowledge, and Wisdom - su Web <http://www.systems-thinking.org/dikw/dikw.htm>, 1994

[Chappell 2004] D.A. Chappell – *Enterprise Service Bus* – Ed. O'Reilly, 2004

[Ciborra 2002] C. Ciborra - *The Labyrinths of Information: Challenging the Wisdom of Systems* - Oxford University Press, Oxford. 2002

[CP 1994] C. Francalanci, P. Maggiolini - *Measuring the Impact of Investments in Information Technologies on Business Performance* - HICSS (4) 1994, pp. 600-609

[ebXML 2005] sito Web degli standard ebXML, <http://www.ebxml.org/>

[Heylighen 1999] F. Heylighen - Change and Information Overload: negative effects - su Web <http://pespmc1.vub.ac.be/CHINNEG.html>, 1999

[OASIS 2005] sito Web del consorzio OASIS, <http://www.oasis-open.org/home/index.php>

[OHE 1999] R. Orfali, D. Harkey, J. Edwards - Client/Server Survival Guide - Wiley 3rd edition, 1999

[OpenDoc 2005] Il formato OpenDocument del consorzio OASIS, su Web http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=office

[OpenXML 2005] Il formato OpenXML di Microsoft, su Web <http://www.microsoft.com/office/preview/developers/fileoverview.mspx>

[Shedroff 1994] N. Shedroff - *Information Interaction Design: A Unified Field Theory of Design* - su Web <http://www.nathan.com/thoughts/unified/index.html> e capitolo del libro R. Jacobson (Editor) - *Information Design* - The MIT Press, 1999

[Tamino 2005] sito Web del database XML Tamino, <http://www.softwareag.com/corporate/products/tamino/default.asp>

[Wiig 1994] K.M. Wiig - *Comprehensive Knowledge Management* - su Web http://www.krii.com/downloads/comprehensive_km.pdf

[Wiig 2000] K.M. Wiig - *The Intelligent Enterprise and Knowledge Management* - su Web http://www.krii.com/downloads/intellig_enterprise%20&%20km.pdf

[XML.com 2005] sito Web con catalogo dei tool per l'uso di XML, <http://www.xml.com/>

[XML.org 2005] sito Web degli standard XML, <http://www.xml.org/>

Analisi di un processo aziendale

Analisi dell'interno di un processo: i diversi punti di vista

L'obiettivo dell'analisi dei processi è arrivare a realizzare la loro gestione ottimale, in modo da ridurre i loro costi ed aumentare il valore da essi prodotto. Questa gestione prende anche il nome di **Business Process Management (BPM)**, e deve necessariamente partire da un chiaro ed efficace percorso di modellazione dei processi (Business Process Modeling, si ricordi che però l'acronimo BPM solitamente indica il management), si veda [EP 2000].

Qualora l'analisi dello status esistente di un processo ne evidenzi troppi limiti o criticità, può diventare necessaria una trasformazione del processo steso, per giungere all'obiettivo di ridurne i costi e/o aumentarne il valore associato. Tale trasformazione prende il nome di **Business Process Reengineering (BPR)** ed è stata definita per la prima volta all'inizio degli anni '90 da Hammer del MIT (si veda [Hammer 1990]), come metodologia per superare i problemi legati all'uso non ottimale delle risorse informatiche entro le aziende. Infatti nel corso della prima informatizzazione delle aziende, avvenuta tra gli anni '60 e gli anni '80 nei vari paesi industrializzati, molto spesso gli strumenti informatici furono inseriti senza cambiare le procedure operative (ad esempio, inserendo semplicemente il computer come strumento di videoscrittura al posto delle macchine da scrivere) con la conseguenza di non migliorare, e in alcuni casi addirittura di peggiorare, l'efficienza dei processi. Questa fase fu indicata anche come **meccanizzazione** dei sistemi informativi. Oggi il BPR è una metodologia di lavoro ampiamente usata nelle grandi organizzazioni e si propone di migliorare l'efficienza dei processi attraverso la loro analisi, l'individuazione delle attività critiche o inefficienti e l'adozione di opportune azioni correttive (si vedano [HC 2004], [BFM 2001] e [LL 2004] per approfondimenti).

Un processo può avere una struttura interna anche molto complessa. Entro un processo avvengono trasformazioni, le entità in gioco sono collegate fra di loro e con gli attori interni ed esterni al processo, e possono essere condivise fra più processi. L'analisi del processo deve considerare le varie caratteristiche del processo stesso allo scopo di condurre a modelli utili. L'uso di modelli grafici e semi-grafici può essere di grande aiuto nello svolgimento dell'analisi stessa.

Esistono molti metodi di analisi dei processi (si veda [LL 2004]). Un metodo di analisi abbastanza innovativo è quello basato su UML for Business, definito in [EP 2000]. Unified Modeling Language (UML) è un linguaggio semi-grafico unificato per la modellazione di concetti, entità, funzionalità, processi e relazioni che fra essi intercorrono (si veda [Fowler 2003] per approfondimenti). UML nasce nel 1997, unificando precedenti sintassi di modellazione (Booch, OMT, OOSE) ed oggi è uno standard internazionale gestito dal consorzio OMG (si veda [OMG.org 2005]). Dopo il 1997 le estensioni standard di UML si susseguono e anche l'uso viene esteso alle varie fasi di realizzazione dei sistemi IT, comprendendo anche la fase di analisi dei processi business (indicata anche come "analisi business"). Infatti nel tempo è emersa la necessità di un linguaggio chiaro comune a tutti i componenti del sistema informativo, dal business alla tecnologia.

Il linguaggio UML serve a descrivere, in modo grafico e “compatto”

- I requisiti utente (use case), ossia, in pratica, le funzionalità che un sistema IT deve rendere disponibile ad un utente
- Le componenti dei sistemi
- I dati in esse contenuti
- Le azioni da esse svolte
- Le relazioni che fra loro intercorrono (il processo in cui esse operano).

UML for business nasce col proposito di usare le stesse notazioni di UML “tecnologico” per esprimere i concetti del business, sia per uso diretto (ovvero per uso di analisi business), sia per uso indiretto (in vista di una successiva implementazione di sistemi IT di ausilio al business). E' da poco stato rilasciato lo standard 2.0, anche se molti strumenti informatici di ausilio all'analisi UML si rifanno ancora allo standard 1.5 o anche addirittura alla versione 1.3.

Una caratteristica molto importante in UML for business è che l'uso dei vari diagrammi UML consente, nel corso dell'analisi di un processo, di focalizzare l'attenzione sui vari elementi costitutivi il processo stesso e sui vari punti di vista che si possono seguire. Al di là della simbologia usata dai singoli diagrammi è importante capire il singolo punto di vista espresso da ciascuno di essi. Un termine talvolta usato è **vista**, traduzione dell'inglese **view**, per focalizzare l'attenzione sul fatto che un diagramma esprime solo alcune caratteristiche ovvero definisce un modello che proietta e rende visibili solo alcune caratteristiche, ossia quelle che costituiscono la vista che si vuole esaminare, del sistema reale sotto esame, in questo caso il processo business.

I diagrammi che trovano uso in UML for Business sono solo un sottoinsieme dei diagrammi tecnici, e precisamente:

- **Diagramma di Attività** o **Activity Diagram**, che definisce il processo come successione di attività;
- **Diagramma dei casi d'uso** o **Use Case Diagram**, che definisce il processo come successione di fasi di interazione fra un operatore umano, che identifica un ruolo, ed uno strumento, oppure fra due strumenti o apparati automatici;
- **Diagramma delle Classi** o **Class Diagram**, che identifica le entità coinvolte nei processi o scambiate fra più processi ed i loro legami logici; un derivato da questo è il **Diagramma degli Oggetti** od **Object Diagram**, che definisce legami logici non fra entità astratte come persone o impiegati generici, ma tra entità concrete, chiaramente identificate;
- **Diagramma di Sequenza** o **Sequence Diagram** e **Diagramma di Collaborazione** o **Collaboration Diagram**, che esprimono legami di interazione dinamica fra le entità;
- **Diagramma di Stato** o **Statechart Diagram**, che descrive il processo come successione di stati.

Analisi del processo come successione di attività

Come già definito nel capitolo 2, un processo è formato da sottoprocessi o fasi, formate da attività, a loro volta formate da azioni od operazioni. L'Activity Diagram

evidenzia le componenti del processo o dell’attività sotto forma di successione logico-temporale di azioni, eventualmente riportando entità usate o modificate nelle azioni stesse, ma sempre da un punto di vista di successione di azioni. Le azioni sono viste in primo piano come componenti del processo, senza evidenziare chi le compie o come vengono compiute.

In modo più preciso, possiamo dire che gli Activity Diagram:

- Sono una evoluzione dei diagrammi di flusso o flow-chart
- Rappresentano una procedura o un workflow mostrando l’evoluzione di un flusso di attività
- Ogni attività è definita come un’evoluzione continua, non necessariamente atomica, di uno stato

Gli elementi di base degli Activity Diagram sono i seguenti.

- **Activity (Attività):** Esecuzione non atomica entro un sistema dotato di stati. Può essere scomposta in azioni.
- **Action (Azione):** Operazione atomica eseguibile che produce come risultato un cambiamento nello stato di un sistema o il ritorno di un valore.
- **Action State (Stato di azione):** Uno stato che rappresenta l’esecuzione di un’azione (atomica), tipicamente l’invocazione di una operazione.
- **Activity State (Stato di attività):** Stato composito, in cui il flusso di controllo è formato di altri stati di attività e stati di azione. Non è atomico, il che significa anche che può essere interrotto, rimanendo “congelato” in un certo punto della sua evoluzione. Può anche essere ulteriormente scomposto in altri diagrammi di attività.
- **Transition (Transizione):** Rappresenta il flusso di controllo fra due attività, che mostra il percorso da un action o activity state al successivo action o activity state.
- **Object Flow (Flusso di oggetti):** Rappresenta uno o più oggetti (un’entità) coinvolti nel flusso di controllo associato con un activity diagram.
- **Object State (Stato di oggetto/i):** Una condizione o situazione operativa nella vita di un oggetto (un’entità) durante la quale l’oggetto soddisfa certe condizioni, compie certe attività o attende certi eventi.
- **Swimlane o Corsia:** Una suddivisione per l’organizzazione di responsabilità per le attività. Non ha un significato fisso, ma spesso corrisponde alla unità organizzativa entro un business model (es. ufficio acquisti, vendite...). Viene rappresentata graficamente con una linea tratteggiata che divide in sezioni il diagramma.

I simboli grafici elementari che compongono un diagramma di attività sono riportati in figura 4.1.

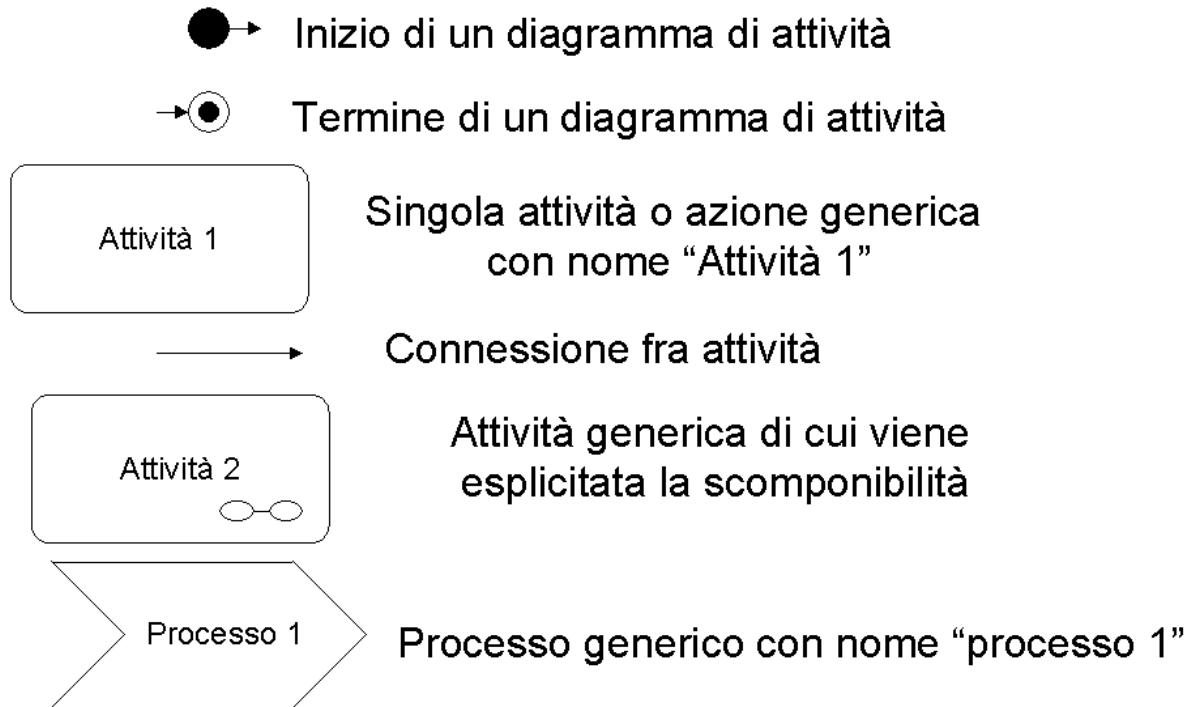


Figura 4.1: Gli elementi costitutivi fondamentali di un diagramma di attività. Eventuali oggetti (entità) sono rappresentati come rettangoli con gli spigoli normali (non arrotondati). La connessione può indicare legame di successione temporale, vincolo od anche semplicemente collegamento logico. Azioni ed attività possono essere indicate con lo stesso simbolo.

Oltre che per indicare il processo, i simboli poligonali con la “freccia” entrante od uscente, sono usati per esplicitare la presenza di comunicazioni esplicite (scambi di informazioni, sincronizzazione) tra attività, ovvero scambio di informazione. In figura 4.2 viene mostrata questa simbologia.

Come nei flow-chart o diagrammi di flusso da cui i diagrammi di attività derivano, i legami di sequenza temporale che legano fra di loro le attività in un diagramma possono essere di vari tipi, rappresentando diversi legami del mondo reale. Il più semplice è senza dubbio la sequenza semplice, rappresentata in figura 4.3, in cui due attività sono semplicemente consecutive.

Altri tipi di legame logico-temporale sono rappresentati nelle figure da 4.4 a 4.8, dove troviamo percorsi di biforcazione (in base ad una condizione il processo prosegue con attività diverse), punti di attesa e sincronizzazione (finché non sono state completate tutte le attività collegate non può avere luogo l’attività successiva a tutte) ed iterazioni, ossia ripetizioni multiple della stessa attività.

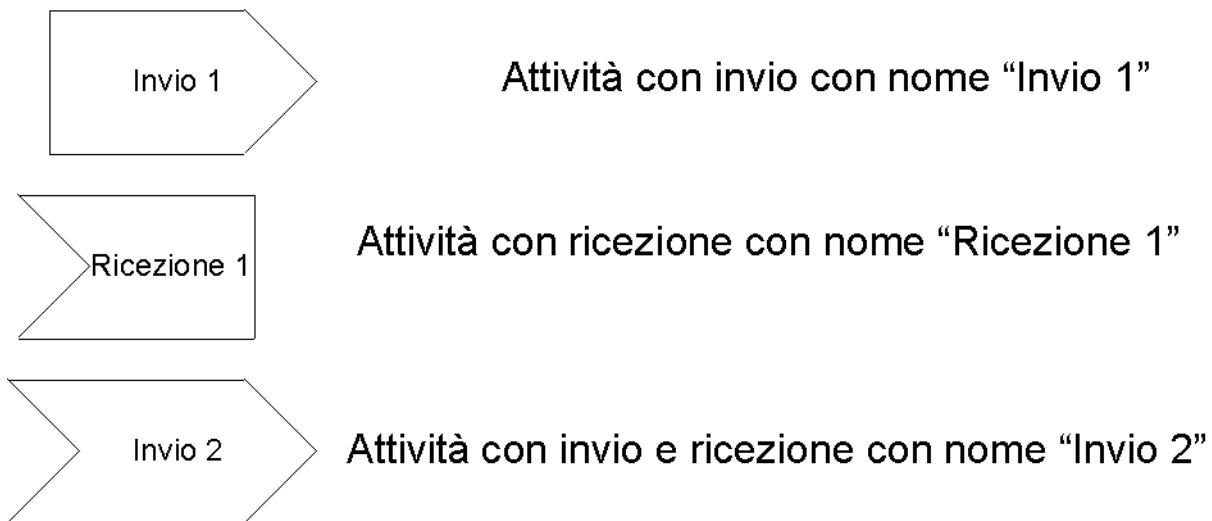
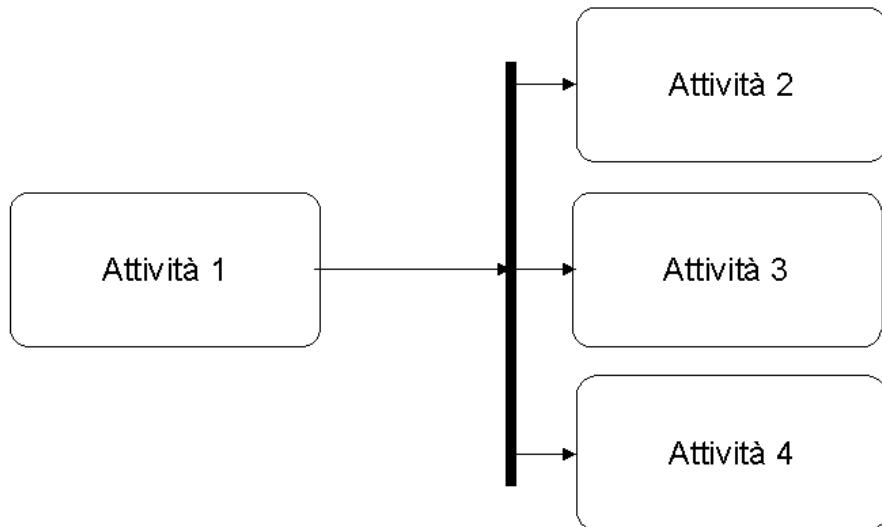


Figura 4.2: Esplicitazione di scambio di informazione e sincronizzazione tra attività e processi.



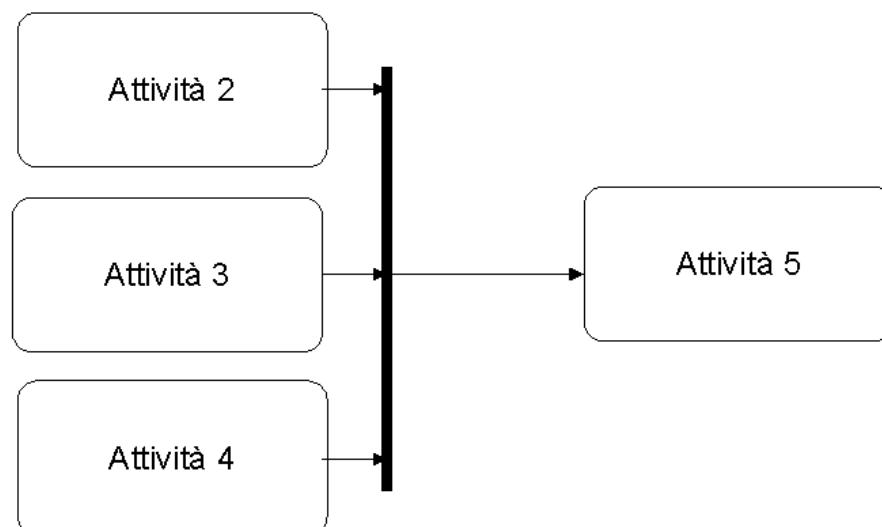
Un’attività (Attività 2) viene eseguita dopo la fine della
Precedente (Attività 1)
(Single Thread)

Figura 4.3: L’attività 2 viene eseguita dopo la fine della precedente attività 1.



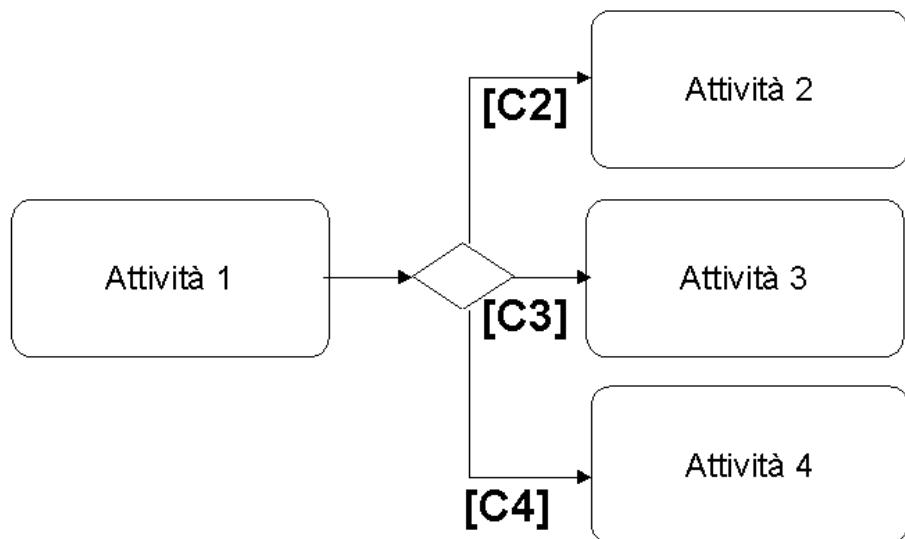
Un singolo flusso di attività si divide in più flussi, consentendo l'esecuzione simultanea di più attività
(Multiple Thread)

Figura 4.4: And-split.



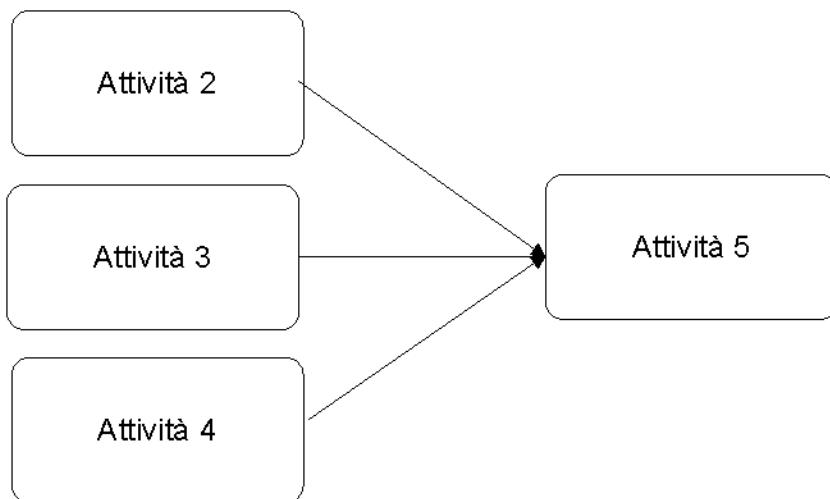
Due o più flussi di attività convergono in uno solo
E' un punto di sincronizzazione per il workflow: non si va avanti finché non sono terminate tutte le attività precedenti

Figura 4.5: And-join.



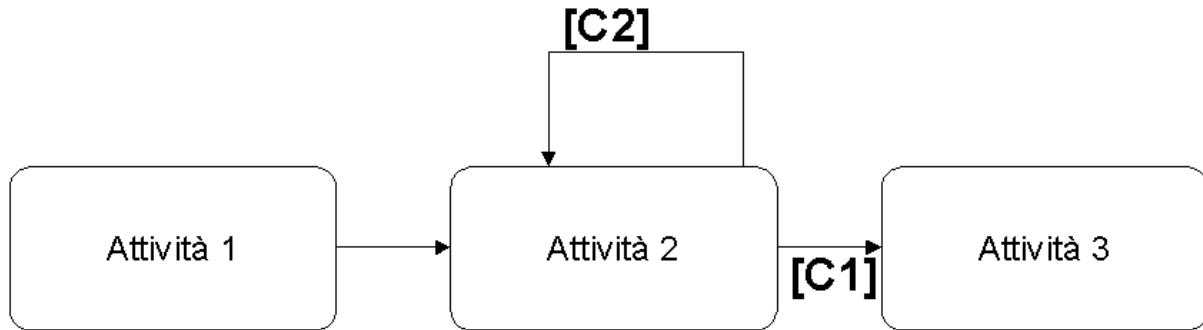
Un singolo flusso di attività prosegue per uno dei cammini in base al verificarsi delle condizioni di transizione, indicate fra parentesi quadre

Figura 4.6: Or-split.



Un punto dove due o più flussi di attività ri-convergono in uno solo ovvero hanno tutti Attività 5 come elemento successivo

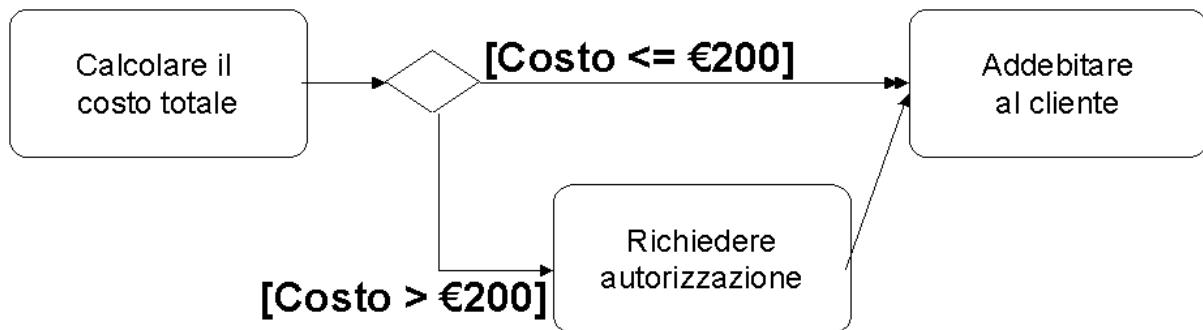
Figura 4.7: Or-join.



Un'attività (Attività 2) viene ripetuta più volte, in base al verificarsi o meno di opportune condizioni di controllo

Figura 4.8: Iterazione.

Per meglio chiarire il tutto vediamo ora alcuni esempi semplici di attività reali. Il primo esempio in figura 4.9 riporta una regola di business, in cui se il costo dell'acquisto è maggiore di 200 Euro si deve chiedere l'autorizzazione ad un dirigente prima di potere completare la procedura con l'addebito al cliente e l'emissione della fattura.



Se il costo totale è maggiore di 200€, bisogna chiedere l'autorizzazione prima di addebitarlo al cliente.

Figura 4.9: Esempio di regola di business espresso con un diagramma di attività.

Un altro esempio viene riportato in figura 4.10, ove viene rappresentato un processo di produzione con le due macro-attività di produzione vera e propria e di verifica del prodotto, esplicitando anche entità che influenzano o sono oggetto delle varie attività. Le entità “concrete” vengono rappresentate con i rettangoli, mentre le entità di

informazione vengono rappresentate con i romboidi. Gli stereotipi, racchiusi tra i simboli <>, indicano una categoria generale cui le entità appartengono.

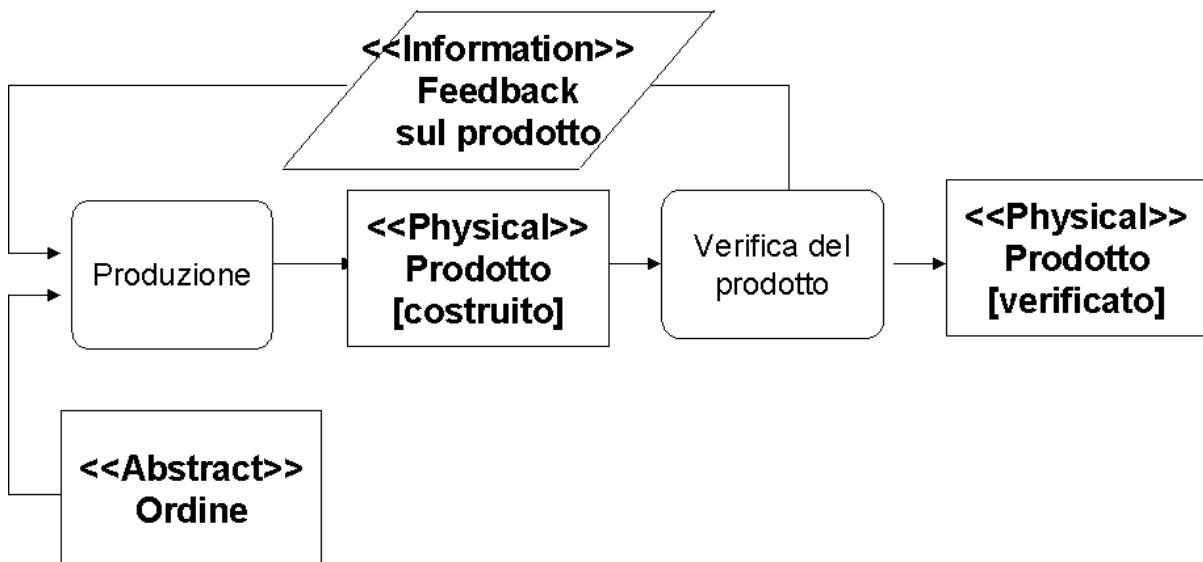


Figura 4.10: Esempio di diagramma di attività con flusso di oggetti; è chiara l'influenza che le informazioni di ritorno hanno poi sul processo di produzione.

Analisi del processo come successione di casi d'uso (di strumenti)

Nei diagrammi di attività l'enfasi viene posta sulle attività, ma senza specificare di solito chi le compie, o comunque come le compie, ossia anche se sono presenti come oggetti le rappresentazioni di attori, non viene esplicitata l'azione compiuta dagli attori, ovvero l'interazione fra più entità presenti allo scopo di realizzare l'attività. Per esplicitare questa informazione serve un punto di vista diverso, che viene espresso dagli Use Case Diagram (diagrammi dei casi d'uso), entro i quali le singole attività sono “aperte” per essere viste come interazioni fra un attore (che nella maggior parte dei casi identifica un operatore umano) ed un sistema, come per esempio un computer od una macchina con cui l'operatore deve interagire per svolgere il compito associato all'attività in esame. Nell'ambito dei sistemi informativi nella maggior parte dei casi il sistema identifica un sistema informatico. Il dettaglio dei singoli casi d'uso diviene una successione di interazioni elementari (richieste) compiute dall'operatore, soggetto attivo della situazione, e dalle conseguenti risposte del sistema.

Entrambi i diagrammi visti concentrano l'attenzione sul punto di vista dell'azione o dell'interazione fra un utente ed un sistema. Ma il secondo, in un ottica di processo, è successivo al primo in quanto richiede per la sua realizzazione, di avere definito un'azione sotto forma di interazioni e quindi di avere identificato un sistema che svolge le azioni, i suoi confini e l'operatore (ovvero il ruolo da esso espresso) che, interagendo con il sistema, ovvero dando comandi al sistema stesso, gli fa compiere le azioni che compongono l'attività in esame.

Vediamo ora di definire in modo più preciso cosa rappresenta un caso d'uso ed il relativo diagramma. In modo formale possiamo definire un caso d'uso o use case

come: una **sequenza di transazioni**, eseguita da un **attore** in interazione col **sistema**, la quale fornisce un **valore misurabile** per l'attore. Risulta chiaro quindi il legame fra il caso d'uso e l'attività.

Lo use case è in pratica un contratto, che descrive l'interazione fra due entità che interagiscono fra loro, consentendo di stabilire con precisione:

- Servizi forniti
- Servizi richiesti
- Utenti abilitati
- Vincoli nell'erogazione.

Per individuare un caso d'uso il primo passo è quello di trovare un confine preciso (boundary) per il sistema/sottosistema/componente che si sta analizzando (e questo dovrebbe risultare dall'analisi dell'attività che si sta "aprendo" per arrivare al caso d'uso). Una volta definito il confine si può stabilire cosa fa il sistema rispetto all'esterno e identificare attori e use case. In pratica quindi un caso d'uso è

- Una sequenza di transazioni in dialogo col sistema
- Comporta sempre uno o più attori
- Rappresenta **cosa** (non come) il sistema offre all'attore
- Mappato alle attività di business.

Uno use case rappresenta una situazione tipica di utilizzo del sistema e comprende in sé vari flussi possibili di esecuzione. Uno use case rappresenta un'importante parte di funzionalità, completa dall'inizio alla fine. Un attore rappresenta un'entità esterna al sistema (una persona, un altro sistema software, un componente hardware) che interagisce col sistema. Un attore individua un ruolo piuttosto che un'entità fisica, per individuare il quale si possono usare le domande seguenti:

- chi ha bisogno del sistema?
- chi userà le funzionalità principali?
- chi dovrà manutenere e amministrare il sistema?
- di quali dispositivi (fisici, hardware, software...) il sistema ha bisogno?
- con quali altri sistemi il sistema dovrà comunicare?

Poi, per ognuno degli attori precedentemente identificati, si può rispondere alle seguenti domande:

- quali funzioni l'attore richiede al sistema?
- l'attore ha bisogno di leggere o scrivere o immagazzinare informazioni nel sistema?
- l'attore deve ricevere notifiche di eventi dal sistema?

Attori e use case sono sempre collegati fra loro, un attore isolato non può interagire col sistema, mentre uno use case isolato non fornisce alcuna funzionalità all'esterno (intendendo funzionalità che abbiano un senso all'esterno).

Un attore può essere:

- **attivo**, ovvero inizia uno use case
- **passivo**, ovvero partecipa a uno use case, ma non lo inizia.

Vediamo ora la sintassi grafica degli Use Case Diagram. Nella figura 4.11 viene rappresentato un caso d'uso generico, ove l'attore è l'omino stilizzato, il sistema è

esplicitato con un rettangolo e il caso d'uso con un'ellisse. Il caso d'uso nelle fasi successive di analisi viene espanso nella sequenza delle transazioni, ovvero di singole azioni od operazioni, non ulteriormente scomponibili, che lo compongono. Rispetto ad una attività però qui viene esplicitato chi fa che cosa. Un esempio di questa espansione viene riportato nelle figure 4.12 e 4.13, relative alla interazione fra un cliente ed un bancomat.

L'uso dei diagrammi serve, una volta identificati i singoli casi d'uso, a esplicitare sia le relazioni con gli attori, sia soprattutto eventuali relazioni di dipendenza o successione temporale che esistano tra i vari casi d'uso. In relazione al diagramma delle attività, se la costruzione di quest'ultimo è stata fatta con precisione, è molto probabile che a ciascuna attività venga a corrispondere esattamente un singolo caso d'uso e viceversa. Pertanto l'analisi può procedere dal processo alle attività e da queste ai casi d'uso, percorso normalmente usato dagli analisti di processo, oppure in modo inverso, percorso spesso usato dai progettisti informatici. Un esempio di relazione di successione temporale (“uso”) tra casi d'uso è rappresentato in figura 4.14, dove entrambi i casi d'uso del bancomat hanno una prima fase comune che viene estratta e posta in evidenza: l'autenticazione.

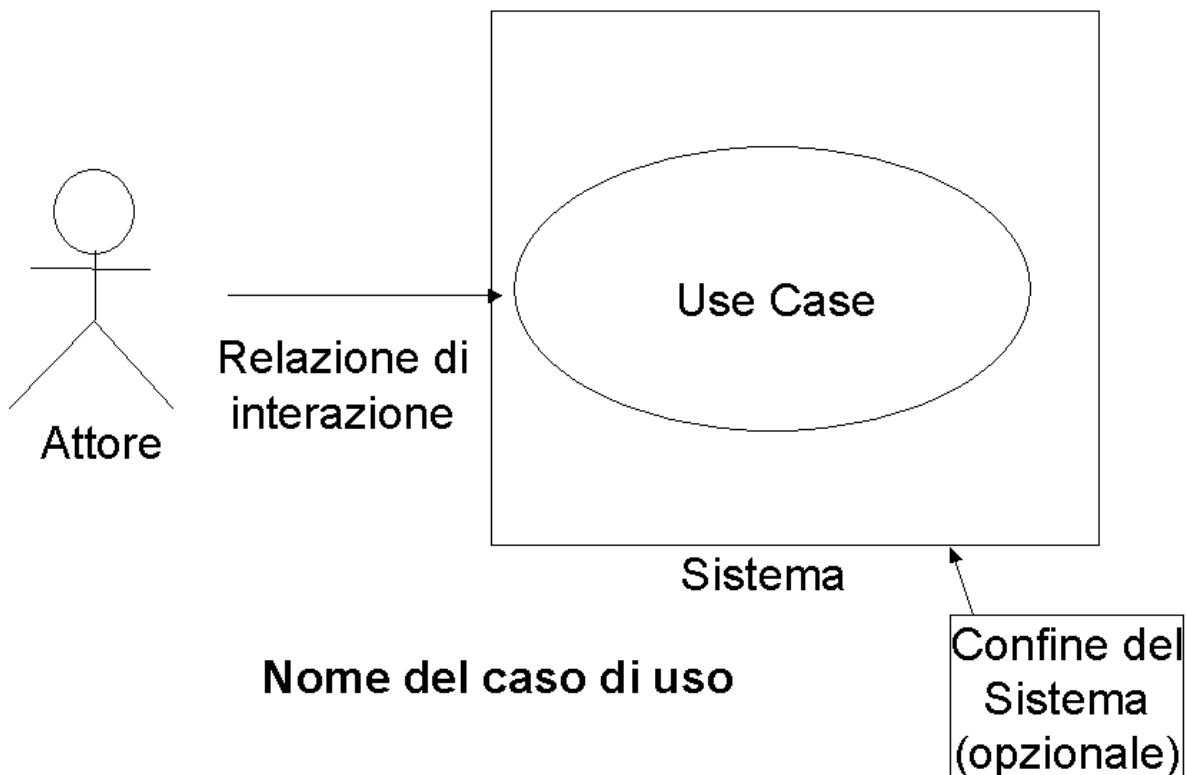
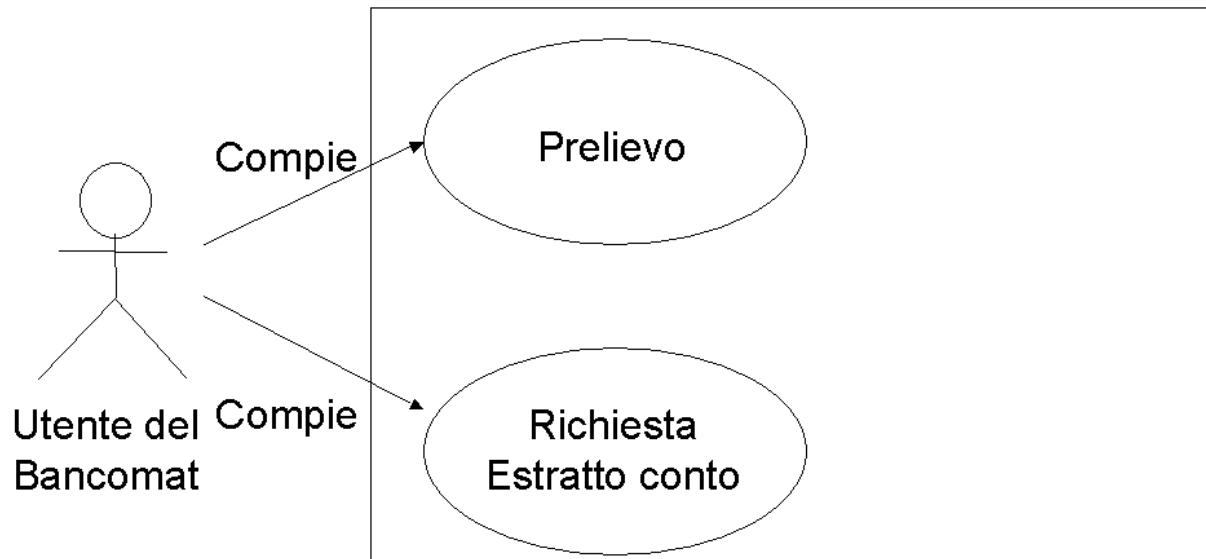


Figura 4.11: Gli elementi basilari di un Use Case Diagram.



Caso di uso: interazioni con il bancomat

Figura 4.12: L'interazione tra utente e bancomat, in cui sono presenti due possibili casi d'uso.

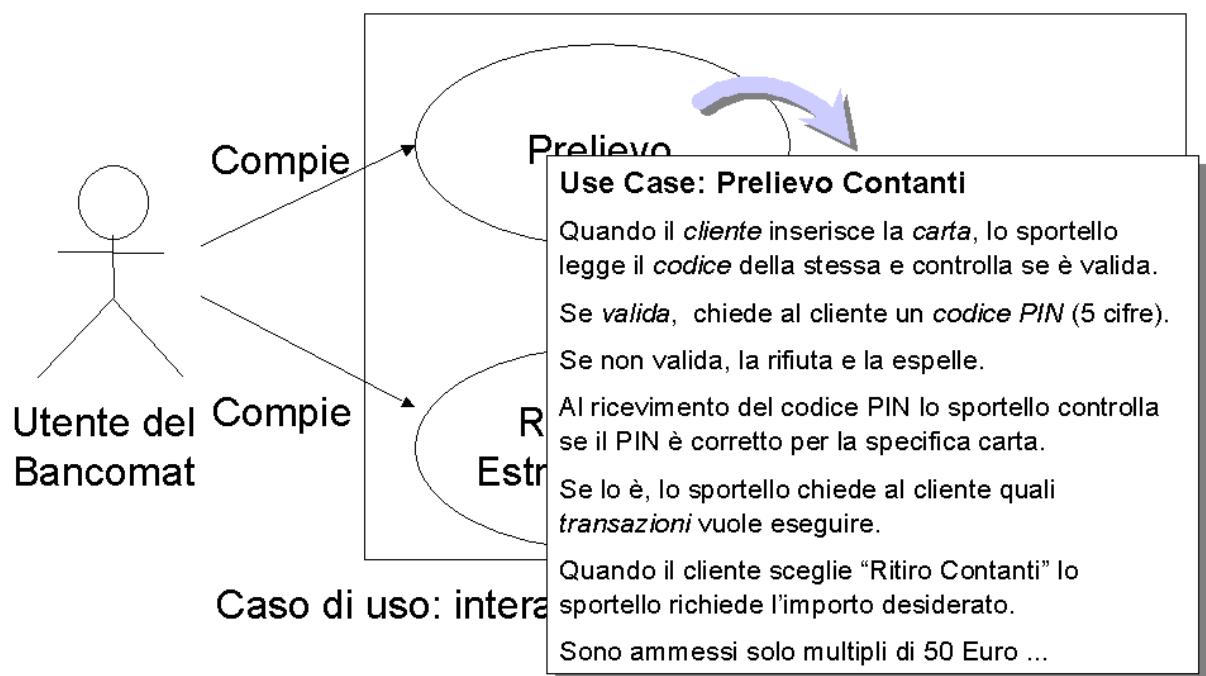


Figura 4.13: Espansione del caso d’uso “prelievo contanti” nelle sue azioni elementari che lo compongono.

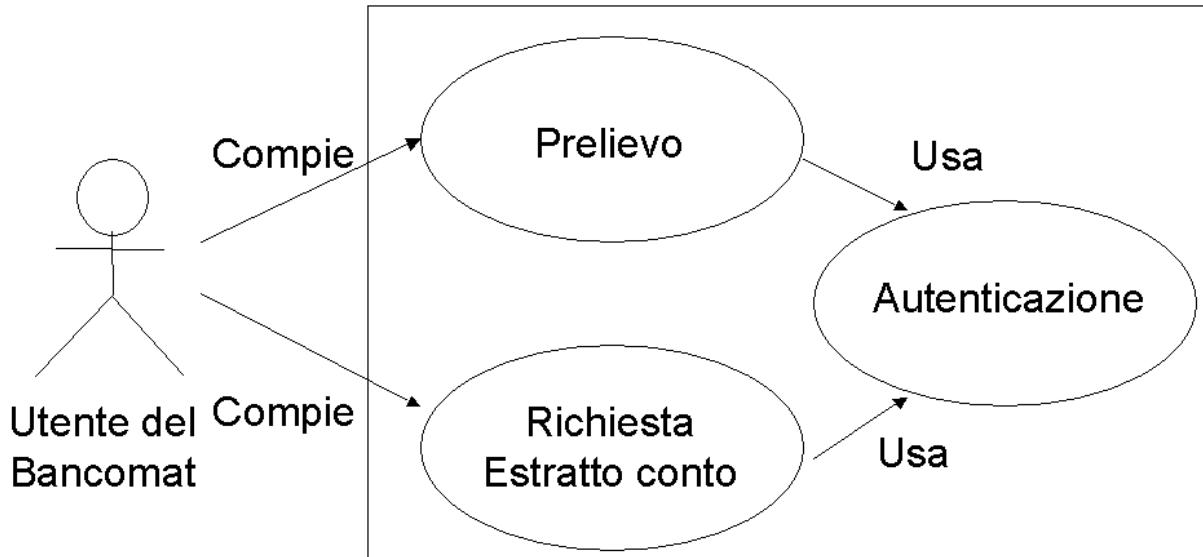


Figura 4.14: Relazione tra i due casi d'uso e la comune fase di autenticazione, che in una sequenza temporale, ovvero vista come attività dentro un processo, deve precedere entrambi. Il diagramma di attività corrispondente avrebbe un or-split come percorso.

Analisi delle entità che prendono parte ai processi

Nei diagrammi visti precedentemente non viene focalizzata l'attenzione direttamente sulle entità che sono oggetto (o soggetto) delle attività, se non come elementi che fanno parte del flusso o come concetti entro le descrizioni associate ai singoli diagrammi. Tali entità possono inoltre essere comuni a più attività, ovvero essere trasformate dalle varie attività (si pensi ad esempio alle materie prime, trasformate in semi lavorati e poi in prodotti finiti durante le varie fasi del processo produttivo), o essere l'output di un'attività e l'input di quella seguente. In generale le attività possono avere fra loro legami logico sequenziali che non risultano direttamente dai diagrammi sinora visti.

Per focalizzare l'attenzione su questo punto di vista serve un approccio diverso, quello espresso dai Class Diagram (diagrammi delle classi). Tali diagrammi identificano le entità, espresse come modello formale associato al concetto da esse espresso (ad esempio, le persone, i prodotti, gli ordini, le fatture) e le relazioni o associazioni che fra esse intercorrono, che identificano i rapporti che legano le entità, comprensivi dei legami numerici, definiti dalle molteplicità (ad esempio un'auto ha 4 ruote, un'azienda ha molti dipendenti, un'auto ha in un determinato momento un solo proprietario). Talvolta si rende necessario indicare non entità astratte, ma istanze concrete di tali entità (ad esempio non un docente generico, ma il professor Giulio Destri, non un dirigente generico, ma il direttore del personale Dr. Guido Ventura, non un prodotto generico, ma il sapone Mantovani). I diagrammi relativi a questo caso particolare si definiscono Object Diagram, sono simili ai Class Diagram, ma per ogni entità concreta al loro interno, definiscono il nome univoco dell'oggetto (ovvero dell'oggetto del mondo reale modellato) e la sua classe di appartenenza (ossia la categoria astratta cui esso appartiene, definita con il nome della classe di appartenenza). Per definire i Class Diagram è necessario anche definire la terminologia del sistema reale in analisi (o in

progetto), identificando con precisione le entità (persone, ruoli, luoghi, oggetti materiali, eventi, strutturazioni ecc...) coinvolte nel sistema del mondo reale (ovvero del dominio di business) che hanno importanza per il sistema informativo che lo dovrà gestire e per le componenti informatiche al suo interno. I Class Diagram sono una evoluzione dei diagrammi Entità-Relazione e vengono usati anche per progettare le Basi di Dati atte a contenere e rendere persistenti (ossia permanenti sino a che non vengono cancellati esplicitamente) i dati aziendali che compongono l'informazione entro l'azienda.

Nei diagrammi delle classi i componenti elementari sono le classi stesse, rappresentate come rettangoli se identificano elementi del mondo reale, anche astratti, e come romboidi se identificano componenti di informazione, che, in una accezione informatica, esistono quindi entro sistemi informatici. Per completezza va detto che l'uso dei romboidi non è universalmente accettato. Le classi possono essere rappresentate a vari livello di dettaglio, come mostrato nella figura 4.15, dove vediamo le classi come semplici elementi senza una struttura esplicitata, le classi come appartenenti a categorie, rappresentate dagli stereotipi, la classi con esplicitati i propri attributi, ovvero elementi caratterizzanti, e con esplicitati anche i metodi, ovvero i "servizi", le azioni che le entità che le classi rappresentano sono in grado di compiere dietro indicazioni provenienti dall'esterno.

I diagrammi delle classi esprimono anche relazioni di tipo logico, di dipendenza, di derivazione e di inclusione, attraverso opportune simbologie. Il legame generale che due classi possono avere tra loro prende il nome di **associazione**.

Uno schema generale dell'associazione fra classi viene rappresentato nella figura 4.16, dove troviamo anche altri concetti importanti:

- **derivazione** o **ereditarietà**, rappresentati dal legame con il triangolo vuoto, che esprime la specializzazione di una classe in un'altra, ad esempio la persona (generica) si specializza nel dipendente che a sua volta si specializza nell'impiegato, nell'operaio e nel dirigente; se percorsa in senso inverso questa è invece una generalizzazione;
- **molteplicità**, che indica un valore numerico, ossia una cardinalità associata ad un legame, per esempio un'auto ha quattro ruote (più il ruotino di scorta), una bicicletta ha due ruote e così via; tale informazione viene indicata dai numeri vicini al legame.

Esempi di associazioni sono rappresentati nelle figure 4.17, dove un'auto è associata a 4 pneumatici, un motore ed una persona (il proprietario) e 4.18, dove viene introdotto il concetto di classe di associazione, ossia una classe che esprime e "da dignità di entità" ad un legame logico, come la proprietà di un auto, caratterizzata dalle date di inizio e date di fine.

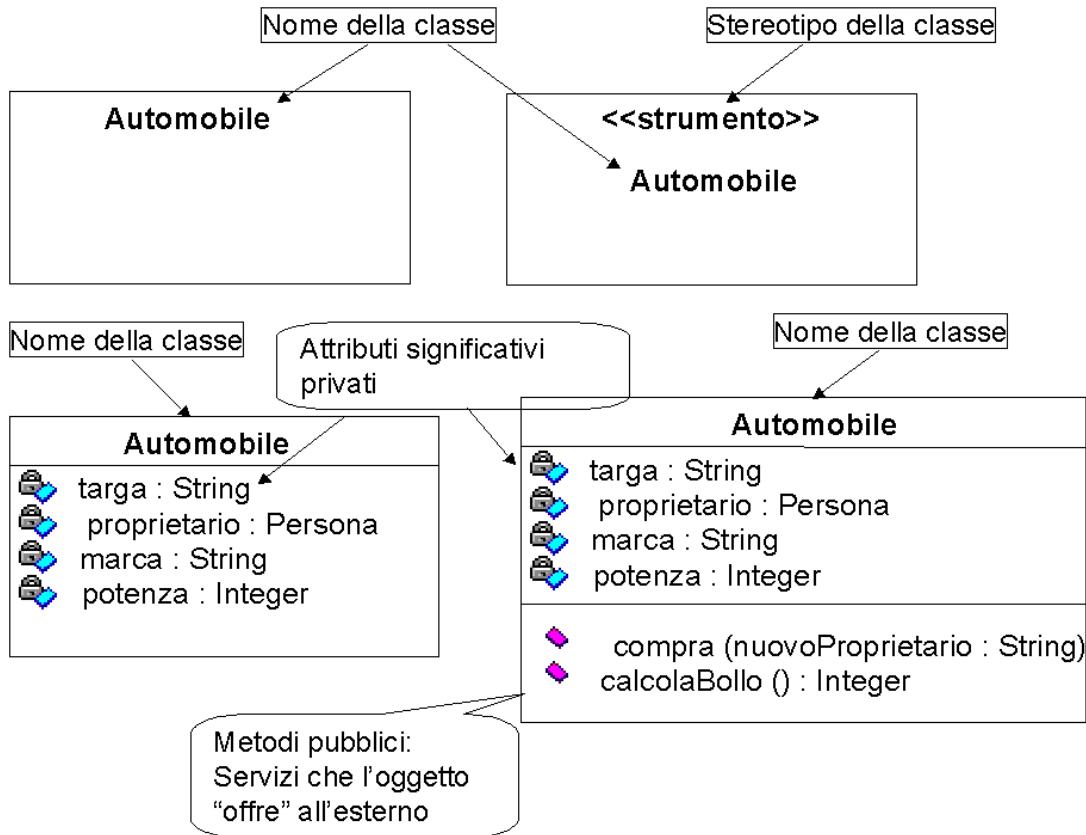


Figura 4.15: Vari modi di rappresentare una classe entro un Class Diagram, a vari livelli di dettaglio.

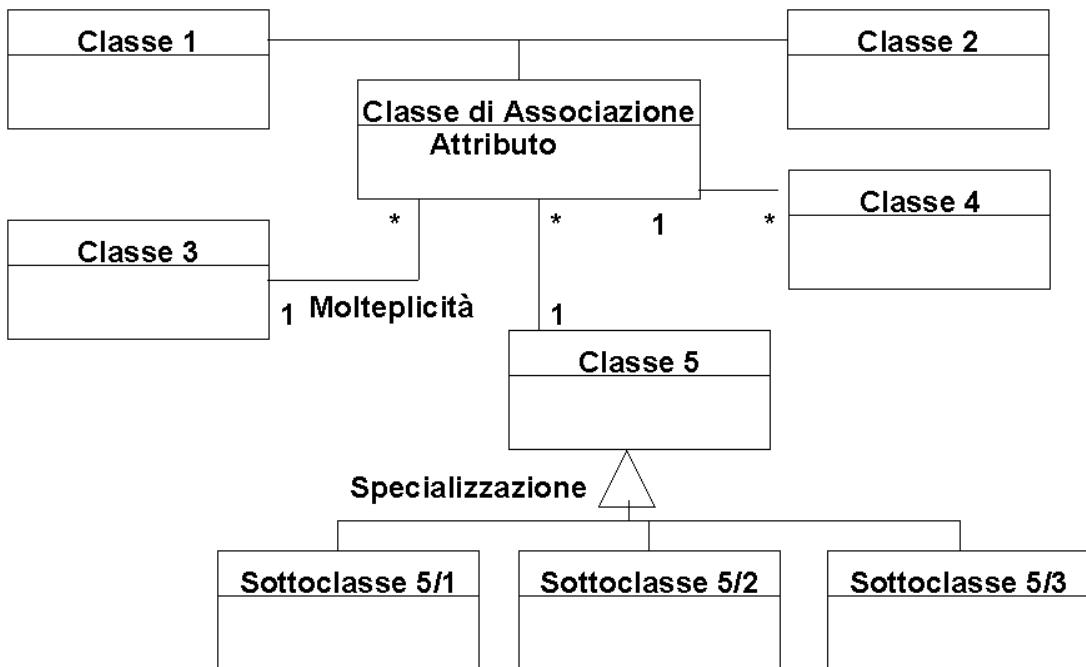


Figura 4.16: Un esempio di class diagram con associazioni, molteplicità e specializzazioni.

Nella figura 4.19 invece vengono rappresentate le relazioni in base al legame logico (non necessariamente espresso da un'entità vera e propria come la classe di associazione) che esse rappresentano ed al significato che esso assume. In particolare è importante notare che il verso del legame espresso dalla relazione può essere rovesciato, con conseguente variazione del concetto associato al legame stesso.

Modellando entità reali complesse è possibile che anche i loro attributi diventino complessi. Attributi complessi e strutturati dovrebbero essere modellati come oggetti a loro volta. Questo conduce alla relazione di inclusione, espressa mediante le due forme alternative:

- **composizione** (rappresentata graficamente con un rombo o diamante nero nella linea che esprime la relazione), in cui gli elementi inclusi non hanno vita propria al di là della classe che li include e quindi, se l'oggetto di tale classe viene eliminato, anche tutti gli elementi inclusi lo sono;
- **aggregazione** (rappresentata graficamente con un rombo o diamante bianco nella linea che esprime la relazione), in cui gli elementi inclusi hanno comunque una vita propria e l'eliminazione dell'oggetto includente non elimina gli oggetti inclusi.

Un esempio di uso simultaneo di entrambe le tipologie di inclusione è presentato in figura 4.20: l'eliminazione di un certo ordine elimina anche le righe ordine che lo compongono, ma non elimina i prodotti che esse comprendono.

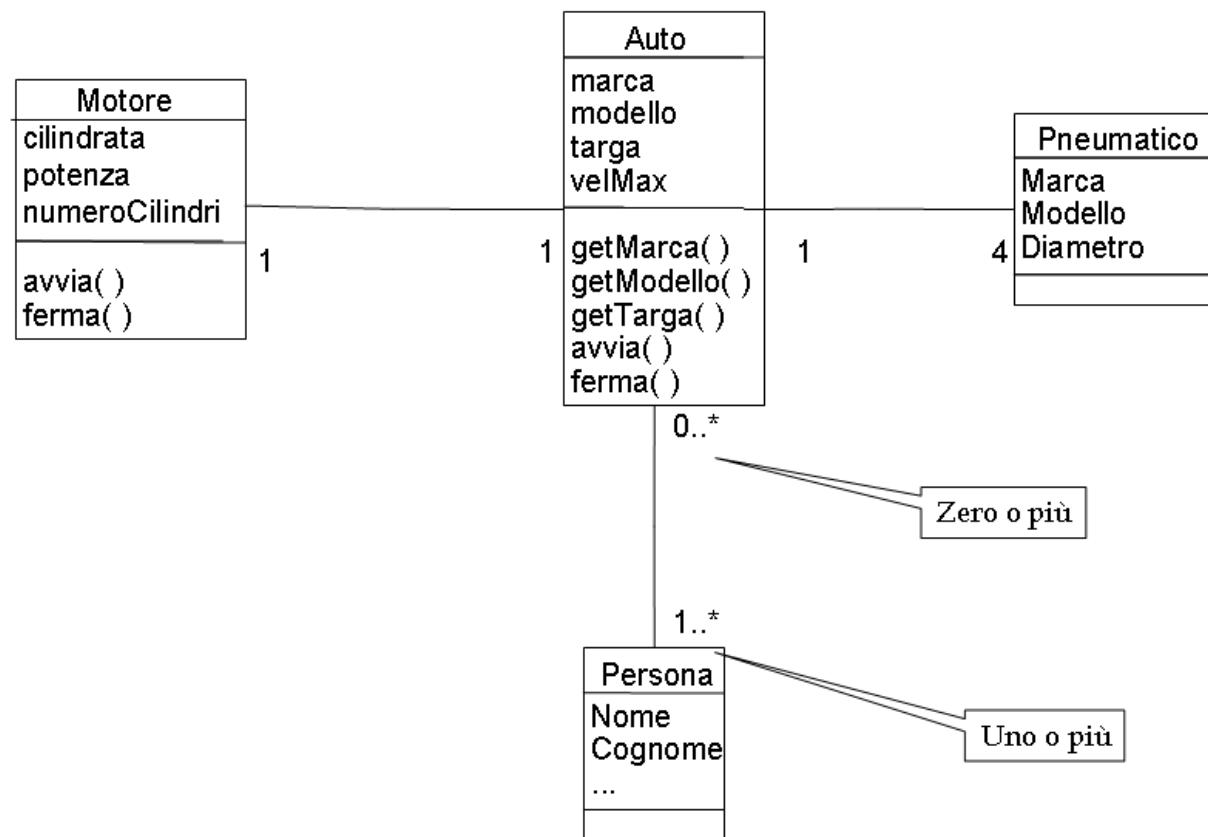


Figura 4.17: La classe auto e le classi motore, pneumatico e persona ad essa associate; si notino le molteplicità con intervalli nell'associazione auto-persona.

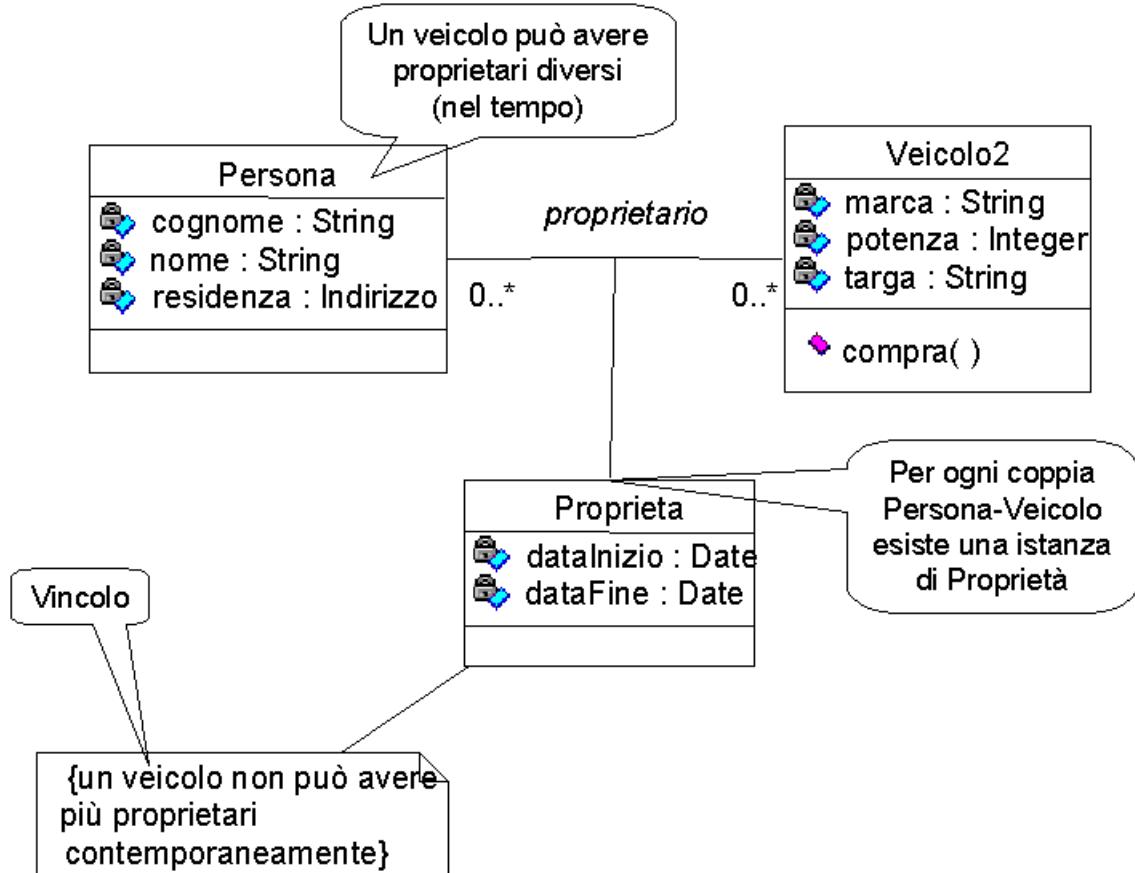


Figura 4.18: La classe di associazione proprietà ed i vincoli cui essa è soggetta.

Come detto in precedenza l'object diagram è una rappresentazione di oggetti concreti, istanze “reali” di concetti più o meno astratti. In figura 4.22 viene rappresentato l’uso combinato di un class diagram per rappresentare un modello di struttura organizzativa tipica di una media impresa e di un object diagram per l’applicazione di questo modello astratto ad un’azienda reale, con la cosiddetta istanziazione delle classi in oggetti. In pratica possiamo definire l’istanziazione come il passaggio contrario all’astrazione: in quest’ultima si passa da elementi concreti del mondo reale come il Prof. Giulio Destri, il Prof. Gianfranco Rossi ecc... alla definizione della categoria di insegnante universitario, mentre nella istanziazione si prende un appartenente a tale categoria. In termini di algebra la classe può rappresentare un insieme (l’insieme di tutti gli insegnanti universitari), l’oggetto un elemento di tale insieme. Il simbolo dell’object diagram, rappresentato il figura 4.21, è simile al rettangolo del class diagram, ma viene indicato il nome dell’oggetto a lato della classe di appartenenza.

L’uso della classe nell’object diagram e l’uso dell’ereditarietà nel class diagram sono diversi dall’uso dello stereotipo. Quest’ultimo rappresenta l’appartenenza dell’oggetto in questione ad una categoria particolare cui il creatore del diagramma associa un significato “assoluto”, condiviso con i lettori del diagramma, che non richiede quindi la definizione di una classe per spiegare tale significato.

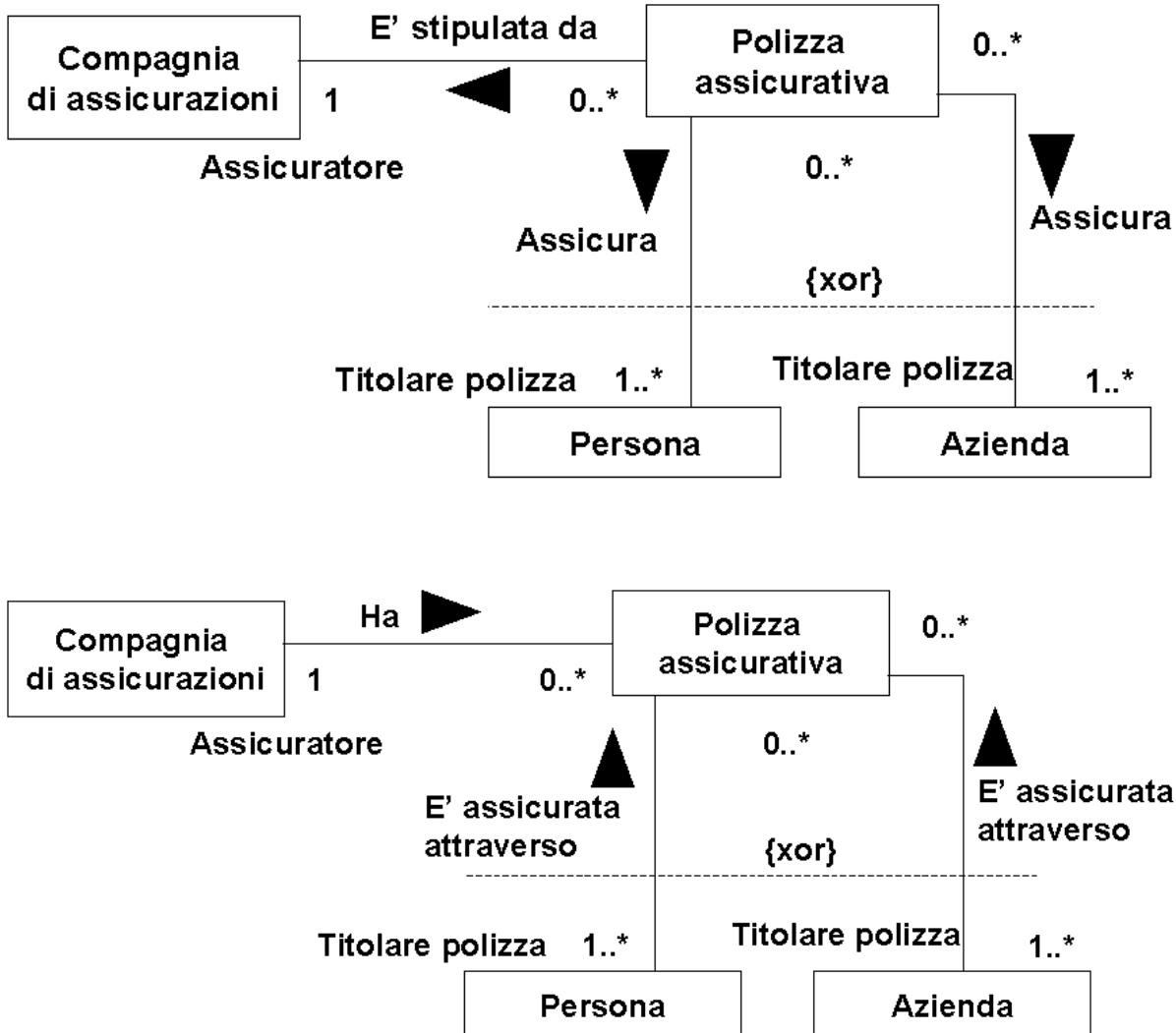


Figura 4.19: Relazioni logiche espresse graficamente attraverso le associazioni tra classi; si noti il verso delle associazioni ed il significato ad esso associato (esempio tratto da [EP 2000]).

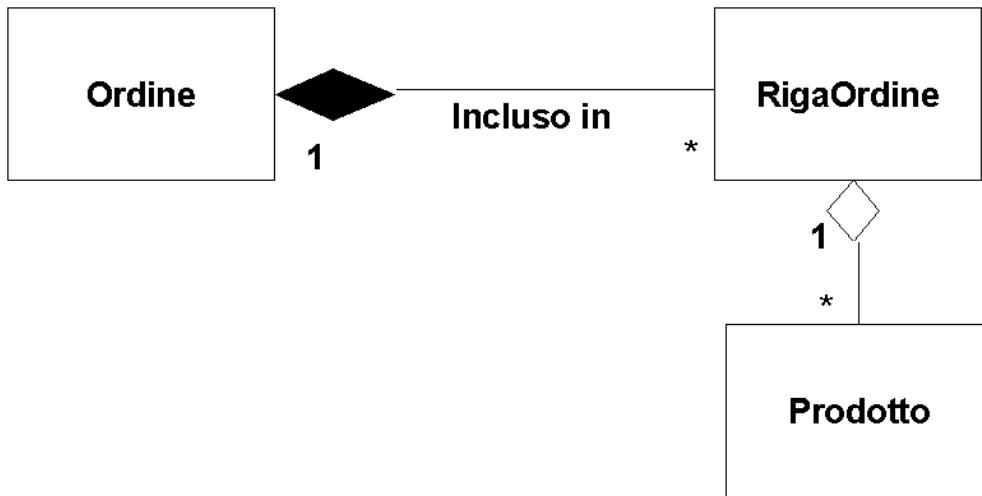


Figura 4.20: Esempio di aggregazione (tra riga dell'ordine e prodotto) e composizione (tra ordine e riga dell'ordine) usati simultaneamente nello stesso class diagram.

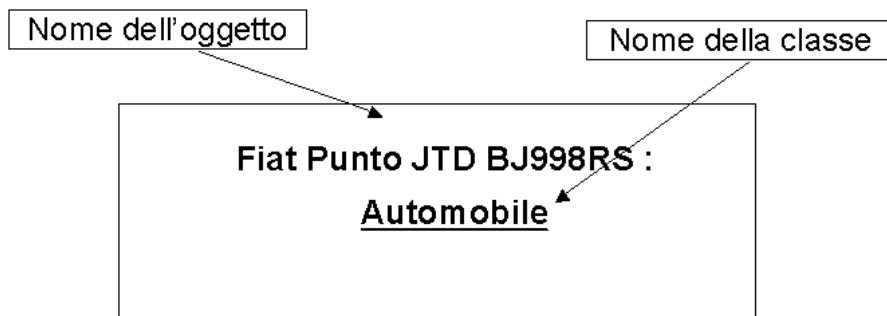
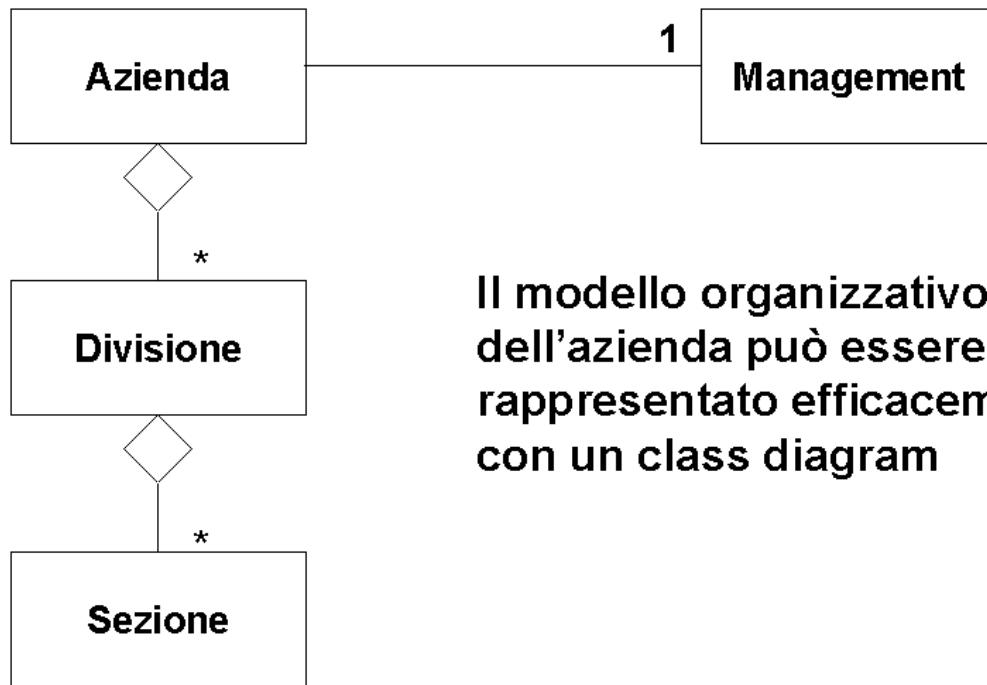


Figura 4.21: Sintassi di un object diagram, con nomeoggetto: nomeclasse, a indicare un oggetto con nomeoggetto, la cui classe è nomeclasse.



Il modello organizzativo dell'azienda può essere rappresentato efficacemente con un class diagram

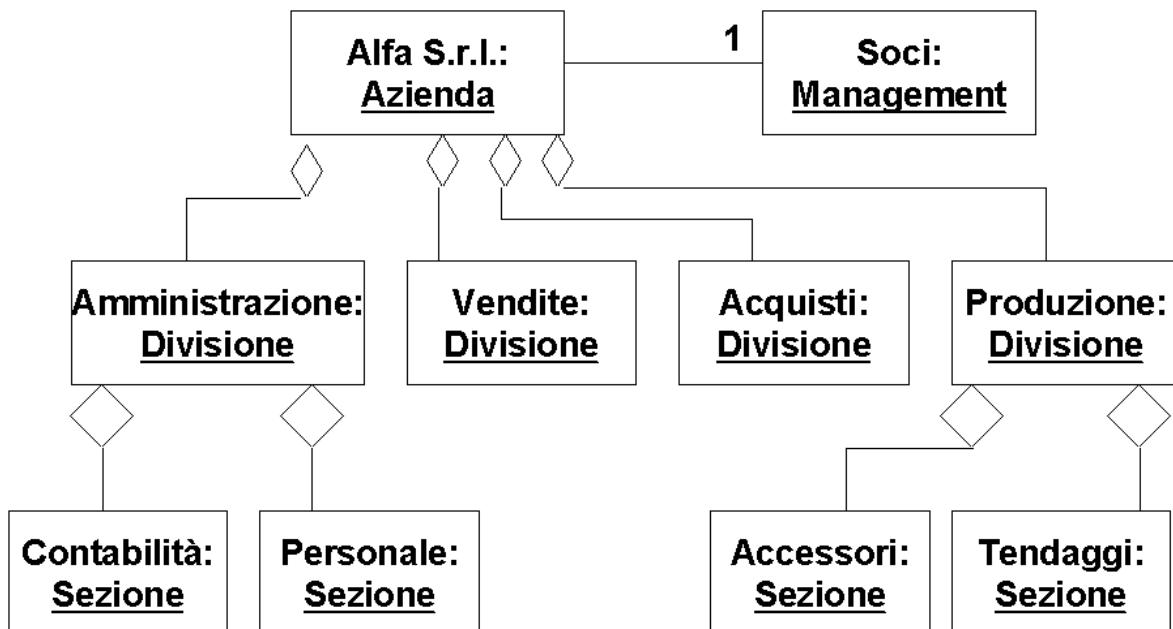


Figura 4.22: Il class diagram in alto rappresenta il modello generale di un'azienda; l'object diagram in basso è una sua istanziazione particolare che rappresenta l'azienda Alfa S.r.l., specializzata nella produzione di accessori per la casa e tende, il cui management è formato dai 5 soci (capitale paritario).

In pratica passare dalle entità del mondo reale, ossia il cosiddetto dominio di business, alle classi significa seguire un percorso logico di analisi, che conduce alle classi a partire dalle entità del mondo reale, attraverso fasi di astrazione, raccolta di caratteristiche comuni e di formalizzazione. Il percorso viene rappresentato in figura 4.23.

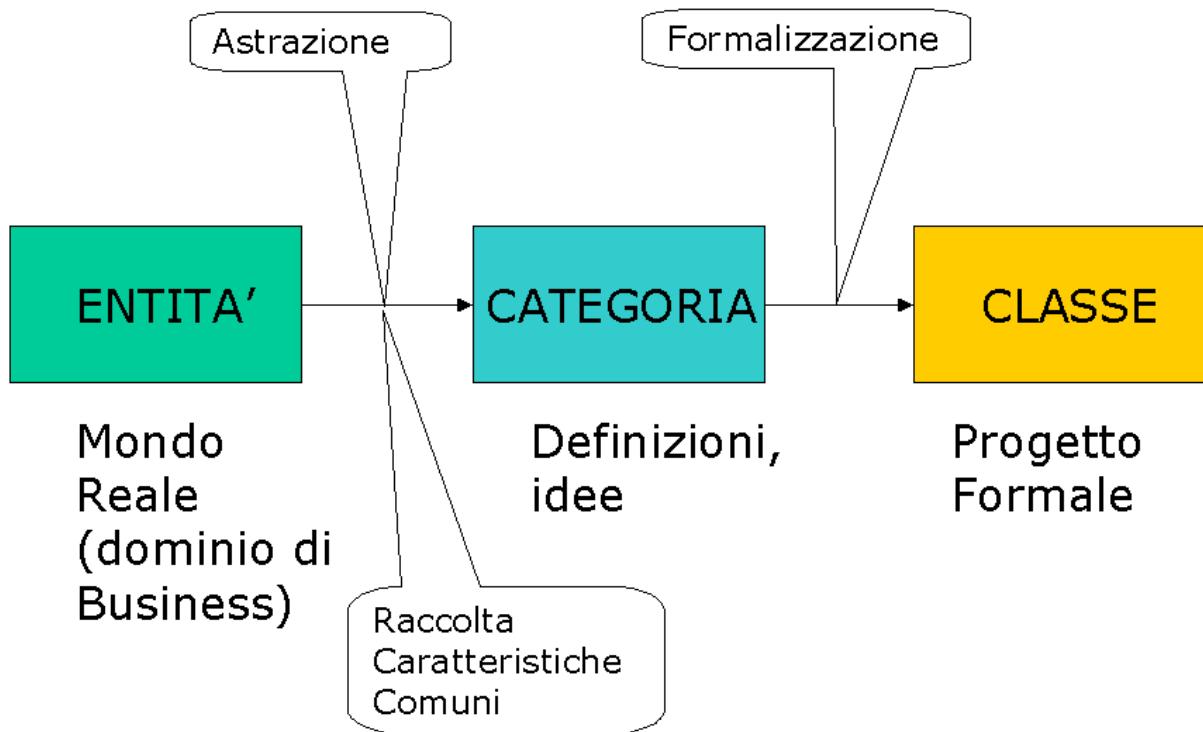


Figura 4.23: Il percorso logico di analisi che conduce dalle entità alle classi.

Analisi delle interazioni fra gli elementi operanti entro un processo

Un limite dei Class e Object Diagram è quello di esprimere solo legami “statici” fra entità, senza porre enfasi sulle interazioni dinamiche che fra esse avvengono. D'altronde, gli Use Case Diagram descrivono interazioni solo ad un livello molto elevato di astrazione, trascurando volutamente molte caratteristiche interne dei sistemi. Inoltre non rappresentano le entità su un piano paritetico, ponendo enfasi sul ruolo dell'attore. Per questo, per valutare con chiarezza un comportamento dinamico vengono aggiunti i diagrammi di interazione, suddivisi in Sequence Diagram o diagrammi di sequenza e Collaboration Diagram o diagrammi di collaborazione, che definiscono le interazioni fra le varie entità definite dai Class Diagram.

Le interazioni sono praticamente dei flussi informativi che scorrono tra le classi e possono rappresentare comandi, ovvero invocazioni di servizio fatte da una classe all'altra, oppure trasferimento del controllo del processo dall'una all'altra classe o oggetto.

I due tipi di diagrammi sono semanticamente equivalenti, ma il diagramma di sequenza ordina temporalmente la sequenza dei messaggi e questo lo rende uno strumento migliore per la comprensione dei vincoli temporali, mentre il diagramma di collaborazione mette in evidenza i legami di dipendenza tra le classi, e quindi tra le entità che esse rappresentano, per l'esecuzione delle attività.

Gli elementi sintattici di base dei Sequence Diagram è riportata nelle figure 4.24, 4.25 e 4.26. Si noti che i diagrammi sono formati da un insieme di oggetti sotto i quali stanno gli assi temporali diretti verso il basso. I messaggi che gli oggetti si scambiano sono espressi tramite le frecce e in corrispondenza alla durata dell'azione svolta dagli oggetti in rispondenza ai messaggi sta il rettangolo bianco sull'asse. Gli oggetti possono talvolta anche essere direttamente sostituiti dalle classi, indicando che in quel momento viene rappresentato il ruolo della categoria e non del singolo componente della categoria. I messaggi che rappresentano i flussi informativi che gli elementi del diagramma si scambiano possono essere di vari tipi, in riferimento alle tipologie di flussi elementari definite nel capitolo 3, come mostrato nella figura 4.26.

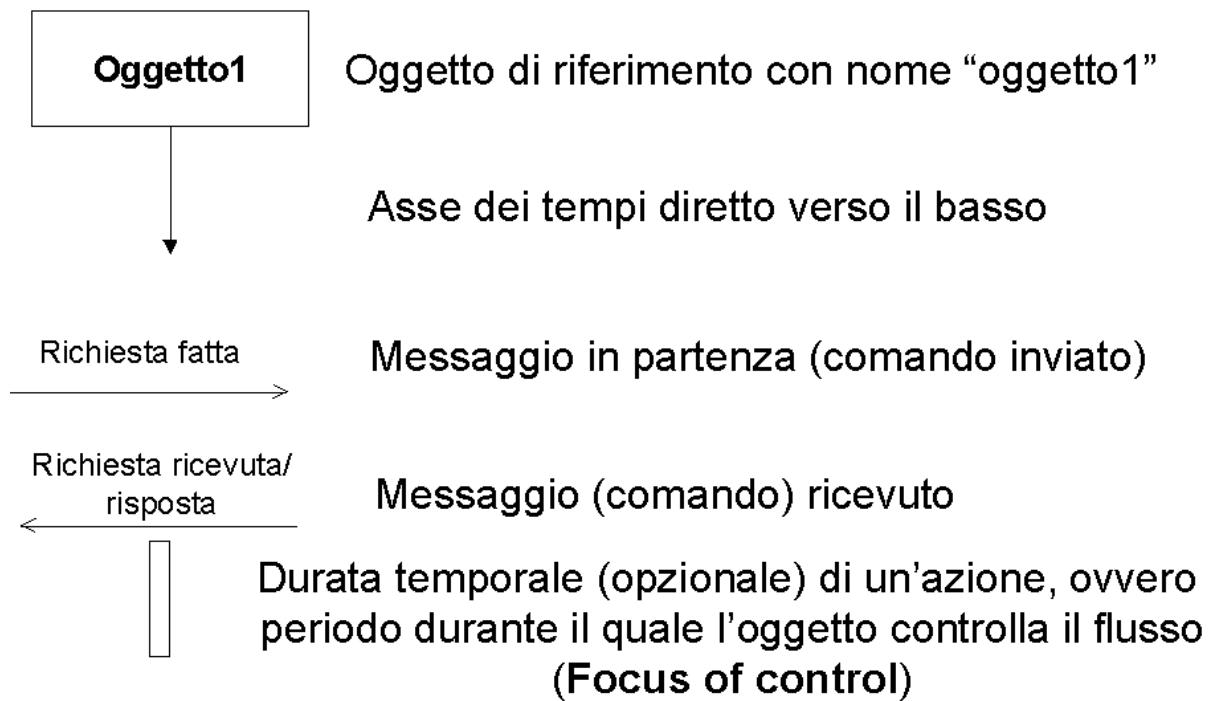


Figura 4.24: Gli elementi base che formano i diagrammi di sequenza.

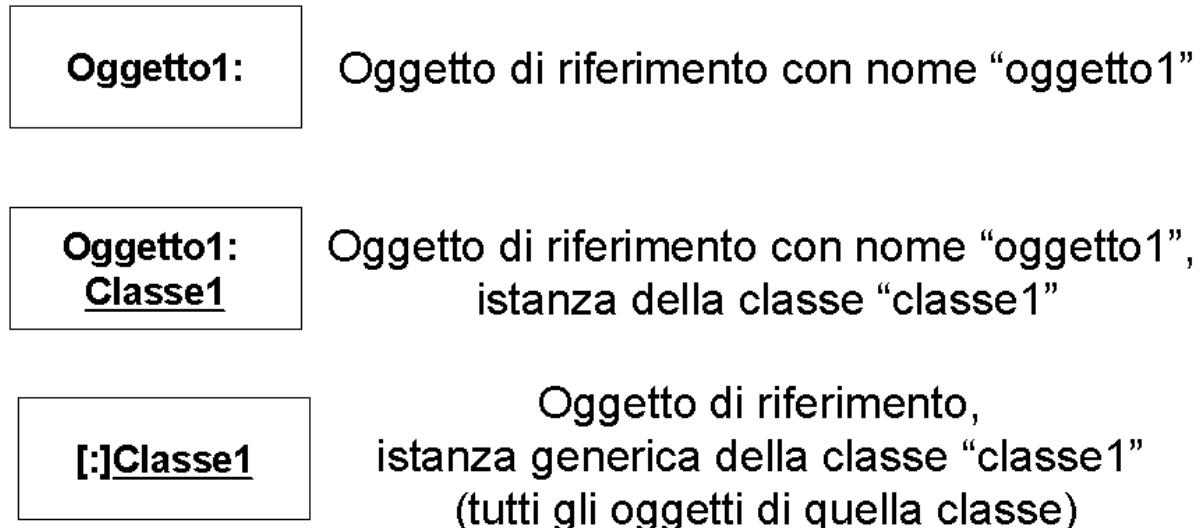


Figura 4.25: Diversi tipi di oggetto entro i diagrammi di sequenza.

- **Semplice:** il controllo è passato dal chiamante al ricevente
- **Sincrono:** il controllo è passato dal chiamante al ricevente ed il primo attende che il secondo gli restituisca il controllo
- **Asincrono:** il chiamante trasmette un segnale al ricevente ma prosegue poi nelle proprie azioni senza attendere il secondo che può o meno ritornare informazioni

Figura 4.26: Diversi tipi di messaggi entro i diagrammi di sequenza; per completezza occorre dire che non sempre questi messaggi

Esempi di uso di diagrammi di sequenza sono riportati nelle figure 4.27 e 4.28, dove vengono rappresentate le successioni di scambi informativi sotto forma di messaggi che avvengono nel corso dell’attività “creazione di una offerta commerciale” tra cliente e fornitore e “uso del bancomat” tra cliente, interfaccia utente del bancomat e dispositivo di distribuzione banconote considerato come entità distinta dal bancomat. In quest’ultimo esempio si noti la biforcazione del percorso dei messaggi in base al riconoscimento corretto del codice utente da parte del bancomat.

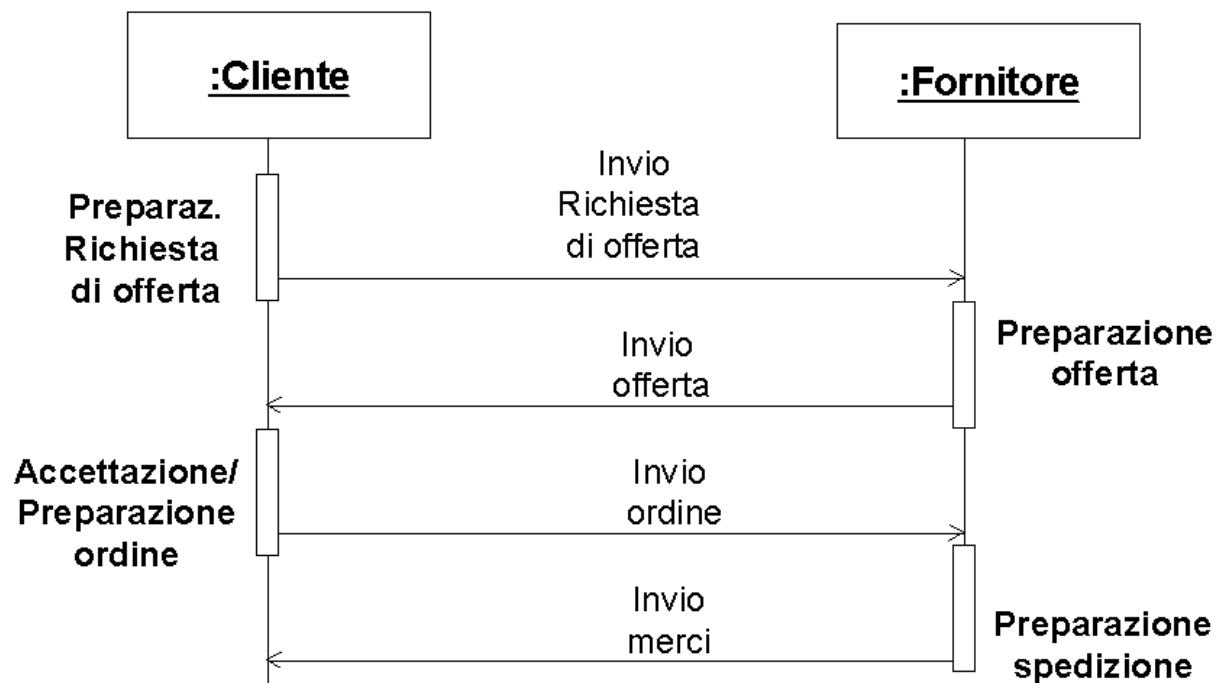


Figura 4.27: Rappresentazione dell'attività di “creazione di una offerta” con il sequence diagram che evidenzia il flusso informativo che intercorre tra la categoria dei clienti e quella dei fornitori.

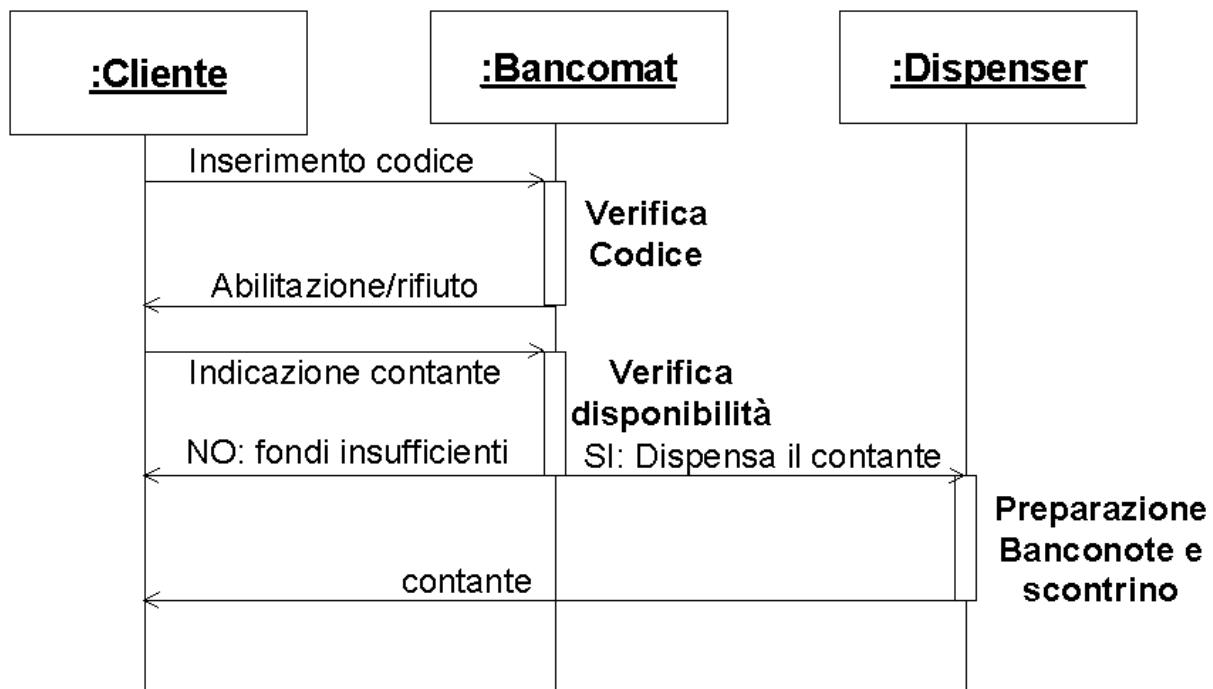


Figura 4.28: Rappresentazione dell'interazione fra cliente e l'insieme di bancomat (la sua interfaccia utente) e dispenser (il distributore meccanico delle banconote).

I collaboration diagram, pur rappresentando sostanzialmente la stessa informazione dei sequence diagram, pongono l'enfasi sui legami che i flussi informativi associati ai messaggi creano e non sulla loro sequenza temporale. Quindi i diagrammi di collaborazione enfatizzano le relazioni fra oggetti (ovvero l'organizzazione strutturale), i diagrammi di sequenza enfatizzano la sequenza temporale delle comunicazioni. La sequenza dei messaggi è meno evidente che nel diagramma di sequenza, mentre sono più evidenti i legami tra gli oggetti. Gli elementi sintattici sono praticamente gli stessi dei sequence diagram salvo che nel nome e descrizione del messaggio deve essere sempre presente anche il numero di sequenza del messaggio stesso, per potere conservare l'informazione relativa alla sequenza temporale.

I diagrammi di collaborazione vengono usati prevalentemente in fase di progetto, quelli di sequenza in fase di analisi, perché sono più comprensibili da parte del committente (cliente, esperto del dominio). In ogni caso si ricordi che i due diagrammi sono isomorfi, è possibile cioè trasformare uno nell'altro.

In figura 4.29 viene rappresentato l'esempio del bancomat di figura 4.28 usando il collaboration diagram.

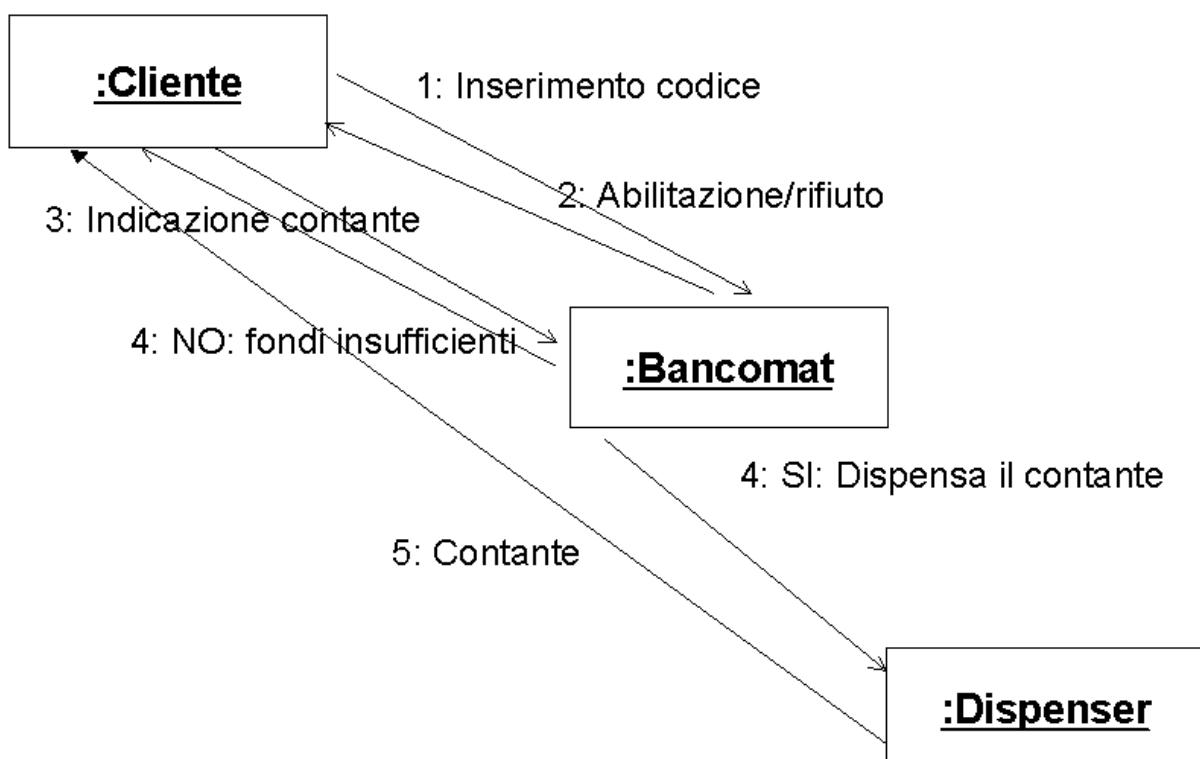


Figura 4.29: L'esempio del bancomat di figura 4.28 rappresentato con il collaboration diagram.

Un'altra potenzialità importante di questi diagrammi è che possono essere usati (soprattutto il Collaboration Diagram) anche per meglio visualizzare la mappatura dei processi sulle suddivisioni funzionali dell'azienda, attraverso un matching delle attività espresse da un activity diagram su classi ciascuna delle quali rappresenta una divisione funzionale dell'azienda, aiutando a definire con chiarezza i flussi informativi

associati al processo che devono passare tra le varie divisioni funzionali. Un esempio di questo uso viene rappresentato nel case study vendita su Web nel capitolo 10.

Analisi del processo come successione di cambiamenti di stato

Gli Statechart Diagram (diagrammi di stato) esprimono una informazione duale a quella degli Activity Diagram, focalizzando l'attenzione non sulle azioni che avvengono ma sul cambiamento di stato di una particolare entità coinvolta nelle azioni stesse (ad esempio l'offerta di vendita passa dallo stato di "sottoposta" a quello di "accettata" e poi a quello di "spedizione in atto").

I diagrammi di stato possono essere usati per descrivere il comportamento nel tempo di un particolare elemento come un oggetto (ovvero una singola entità) o un intero sottosistema, ovvero l'evoluzione di una interazione. In pratica essi descrivono sequenze di stati ed azioni attraverso cui l'elemento considerato passa durante la propria vita reagendo a eventi discreti (segnali, chiamate a funzionalità...). L'enfasi è posta sugli stati e non sulle azioni.

I simboli di base dei diagrammi di stato sono molto simili a quelli dei diagrammi di attività, e sono rappresentati in figura 4.30 e 4.31 dove vengono mostrati sia stati semplici, sia stati che racchiudono al loro interno delle attività, ossia all'ingresso o all'uscita dei quali devono avvenire eventi che sono in realtà attività vere e proprie.

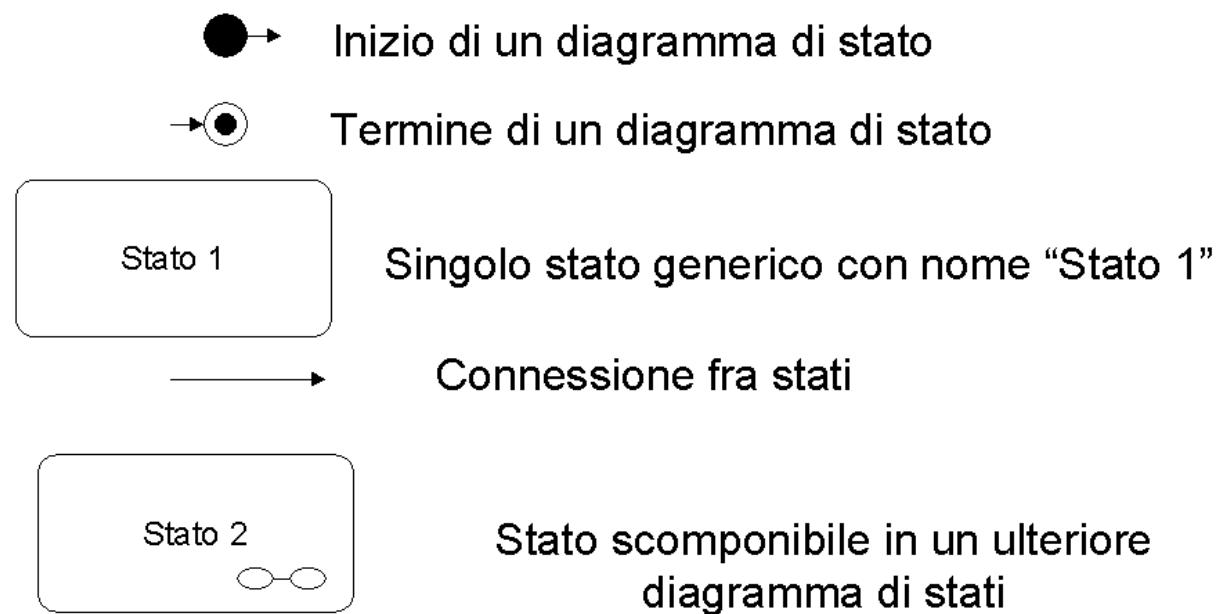


Figura 4.30: Gli elementi di base di un diagramma di stato.

Nei diagrammi di stato sono presenti anche biforazioni o riunioni delle transizioni da uno stato ad un altro, come rappresentato nelle figure 4.32 e 4.33. Esempi di uso dei diagrammi di stato sono invece rappresentato nelle figure 4.34 e 4.35. Nella prima vediamo un cammino semplice attraverso cui passa l'evoluzione di una fattura nel corso della sua "vita operativa" dentro un processo aziendale. Nella seconda vediamo invece l'evoluzione di un ordine di borsa, con diversi percorsi evolutivi possibili.

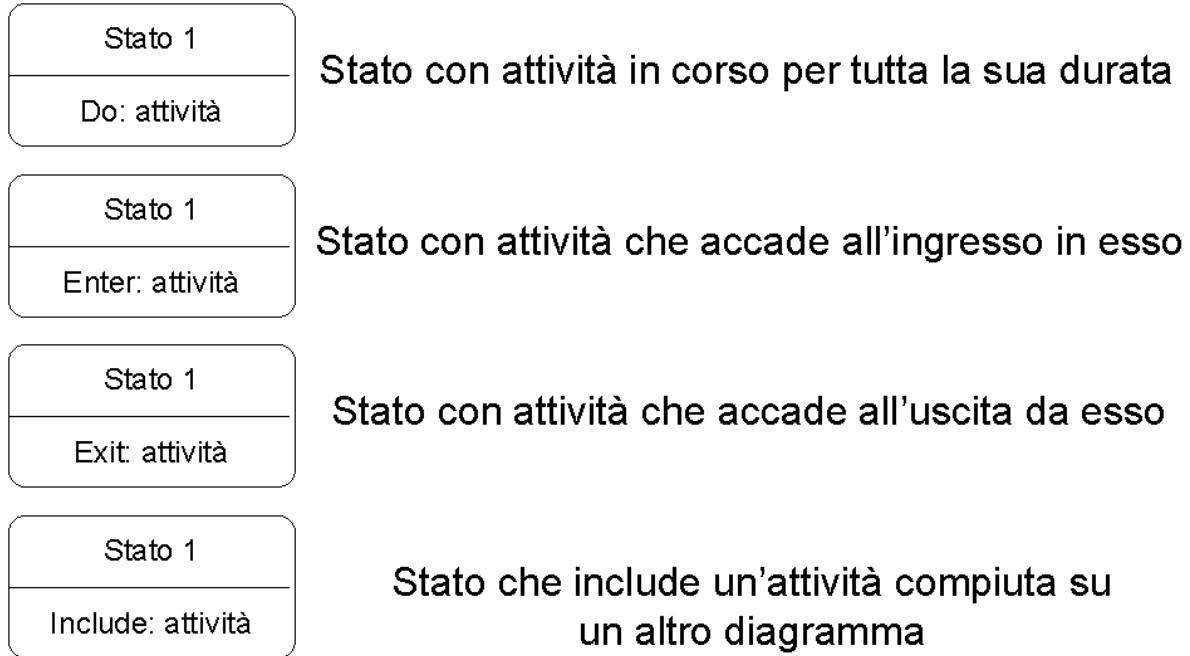


Figura 4.31: Elementi di base i diagrammi di stato composti, che evidenziano la presenza di attività associate allo stato.

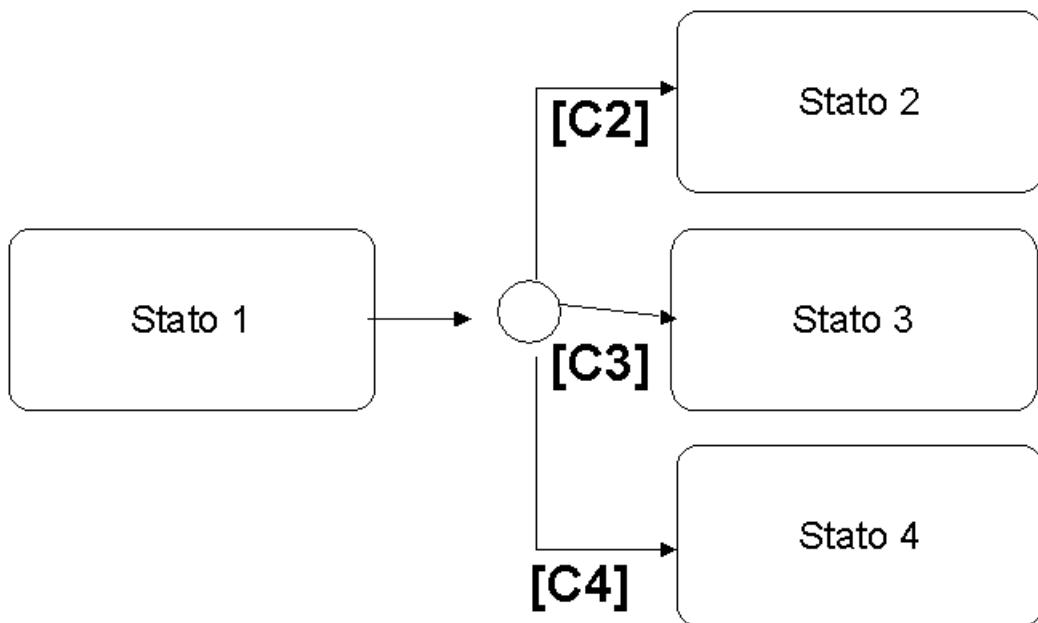


Figura 4.32: Punto di scelta dinamica per un cambiamento di stato, dove in base alle condizioni, rappresentate tra le parentesi quadre, si passa ad uno degli stati. Forma equivalente: le frecce partono direttamente da Stato 1.

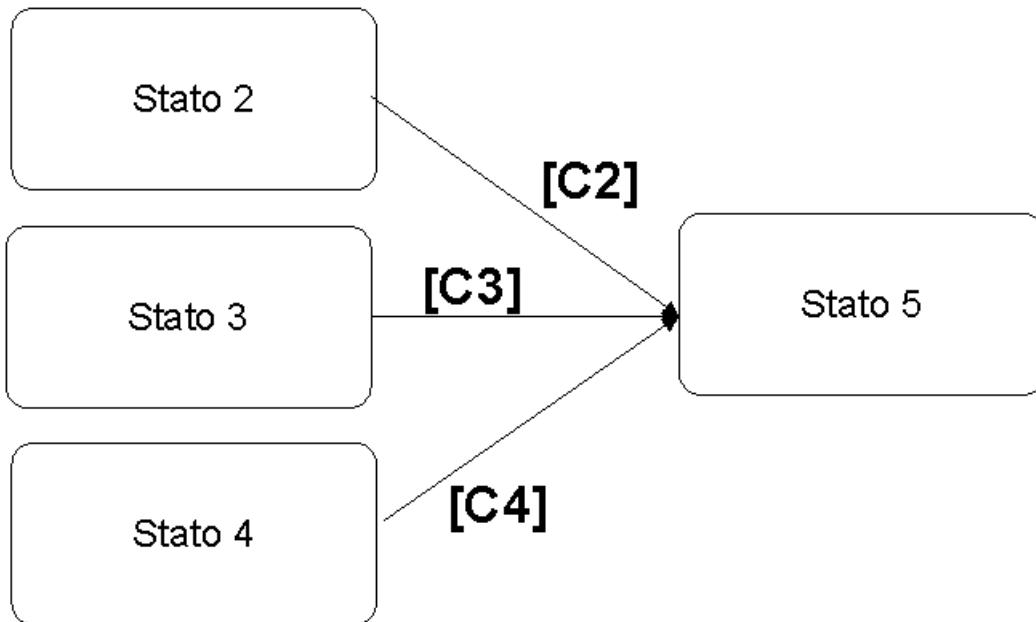


Figura 4.33: Punto di giunzione per una transizione di stato, dove il successivo di tutti e 3 gli stati, seguendo le condizioni, è Stato 5

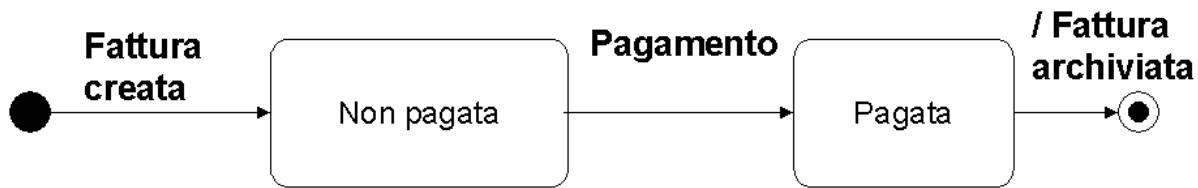


Figura 4.34: Percorso evolutivo, ovvero la successione degli stati attraverso cui passa una fattura nel corso della sua “vita operativa” nell’azienda. Non sono evidenziati i percorsi alternativi che potrebbero esistere. Gli eventi che scatenano il cambiamento di stato, come per esempio il pagamento, sono rappresentati direttamente con il loro nome.

I diagrammi di stato possono quindi mostrare l’evoluzione di un documento, rappresentando graficamente la procedura di workflow associata alla catena di elaborazione/approvazione/gestione/archiviazione di quel particolare tipo di documento, non dal punto di vista delle azioni da compiere, per le quali la rappresentazione dovrebbe essere fatta tramite un diagramma di attività, ma dal punto di vista del documento stesso. Lo stesso potrebbe valere per la rappresentazione di un processo di lavorazione esaminato non dal punto di vista delle azioni ma da quello del pezzo meccanico che subisce le lavorazioni stesse.

In generale comunque il diagramma di stato esprime un tipo di rappresentazione duale rispetto a quello del diagramma di attività e sicuramente meno frequentemente necessaria per le analisi di processo.

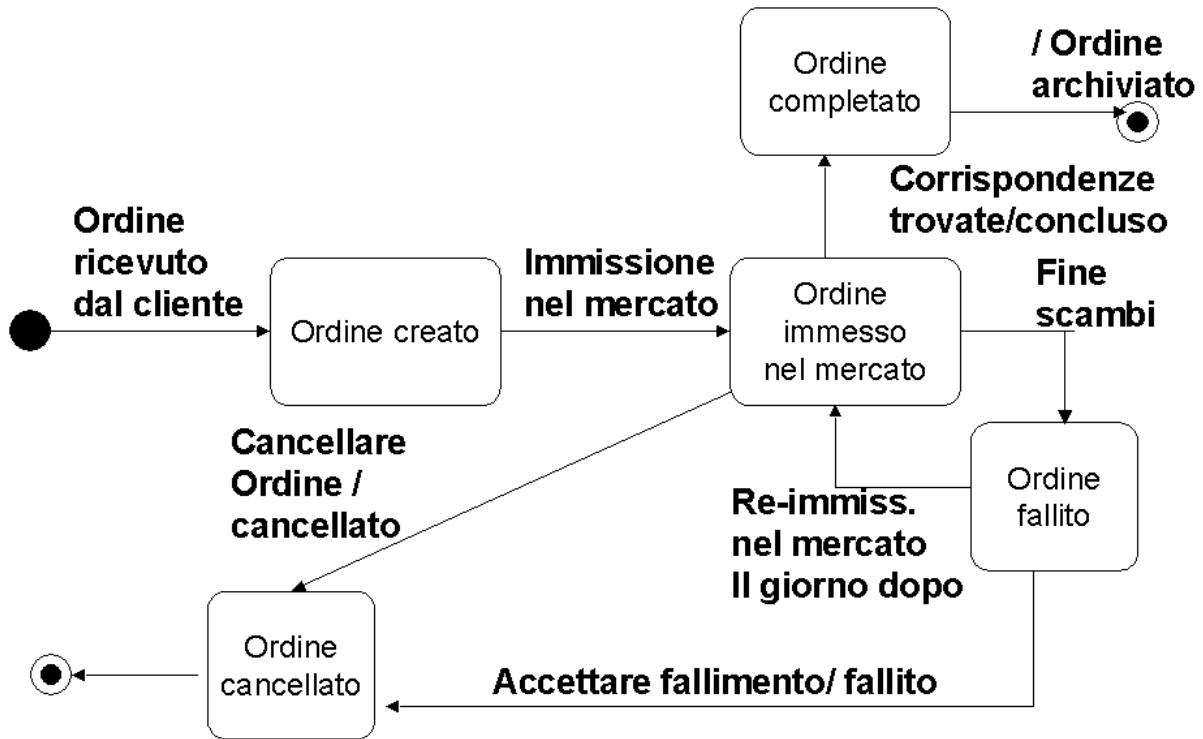


Figura 4.35: Successione degli stati attraverso cui passa un ordine di borsa quando viene ricevuto dal broker, esempio tratto da [EP 2000].

Un esempio completo di analisi

Un esempio utile del mondo reale è l'esecuzione della spesa. Applicando ad esso i vari punti di vista, ovvero le varie viste, finora esaminati, nelle figure seguenti vengono rappresentati i risultati.

Una visione ad alto livello del processo è rappresentata in figura 4.36, dove sono presenti semplicemente il cliente, visto come attore, e il sistema supermercato.

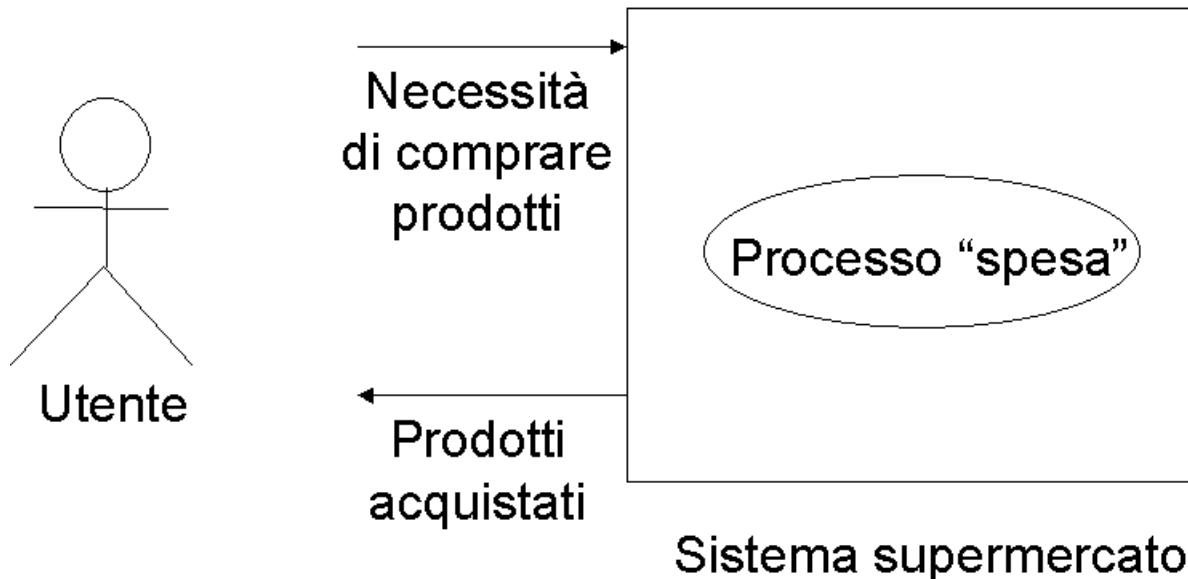


Figura 4.36: Visione astratta di alto livello del processo “fare la spesa”.

Nella figura 4.37 viene rappresentato un primo activity diagram del processo, che lo suddivide in tre fasi possibili, ponendo enfasi con un punto di attesa il fatto che si dovrebbe terminare la preparazione della lista prima di iniziare l’acquisto dei prodotti. A fianco è il corrispondente use case diagram in cui le tre attività vengono a corrispondere a tre casi d’uso.

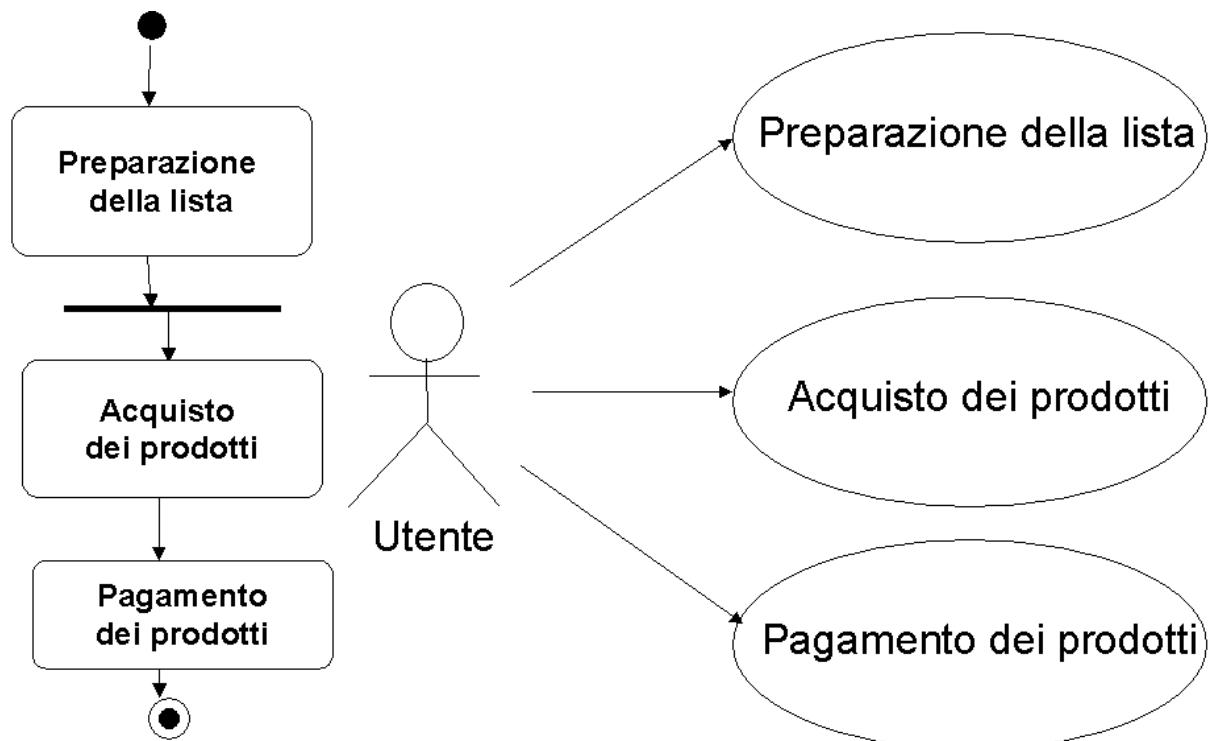


Figura 4.37: Rappresentazione del processo “fare la spesa” con un activity diagram ed il corrispondente use case diagram.

Una variazione interessante nel processo è rappresentata in figura 4.38, dove il sistema di figura 4.36 è “incapsulato” entro un nuovo confine e un attore esterno chiede all’attore interno di fare la spesa, senza magari nemmeno conoscere i dettagli interni del processo (per esempio a che negozio l’utente va per eseguire la spesa). Nella figura 4.39, con l’inserimento di una swimlane o corsia, anche il diagramma delle attività viene adattato alla nuova situazione, dividendo le attività in base a chi le compie, ovvero mappando le attività che compongono i processi sugli operatori umani che le devono effettivamente compiere.

Ma la spesa è interessante anche per le entità in gioco, che derivano dall’analisi delle componenti interne delle tre macro attività individuate nella prima scomposizione del processo. Per esempio, scomponendo l’attività del pagamento dei prodotti si giunge allo use case diagram rappresentato in figura 4.40, dove troviamo, accanto all’utente che è un attore attivo, ossia inizia lo use case, anche un attore passivo, cioè l’operatore di cassa, che usa il sistema, cioè il registratore di cassa per il computo del costo totale cui seguirà il pagamento vero e proprio dei prodotti. Anche altri casi d’uso del sistema registratore di cassa sono indicati. Dall’analisi delle entità in gioco poi deriva il class diagram di figura 4.41 ed i corrispondenti sequence diagram di figura 4.42 e collaboration diagram di figura 4.43.

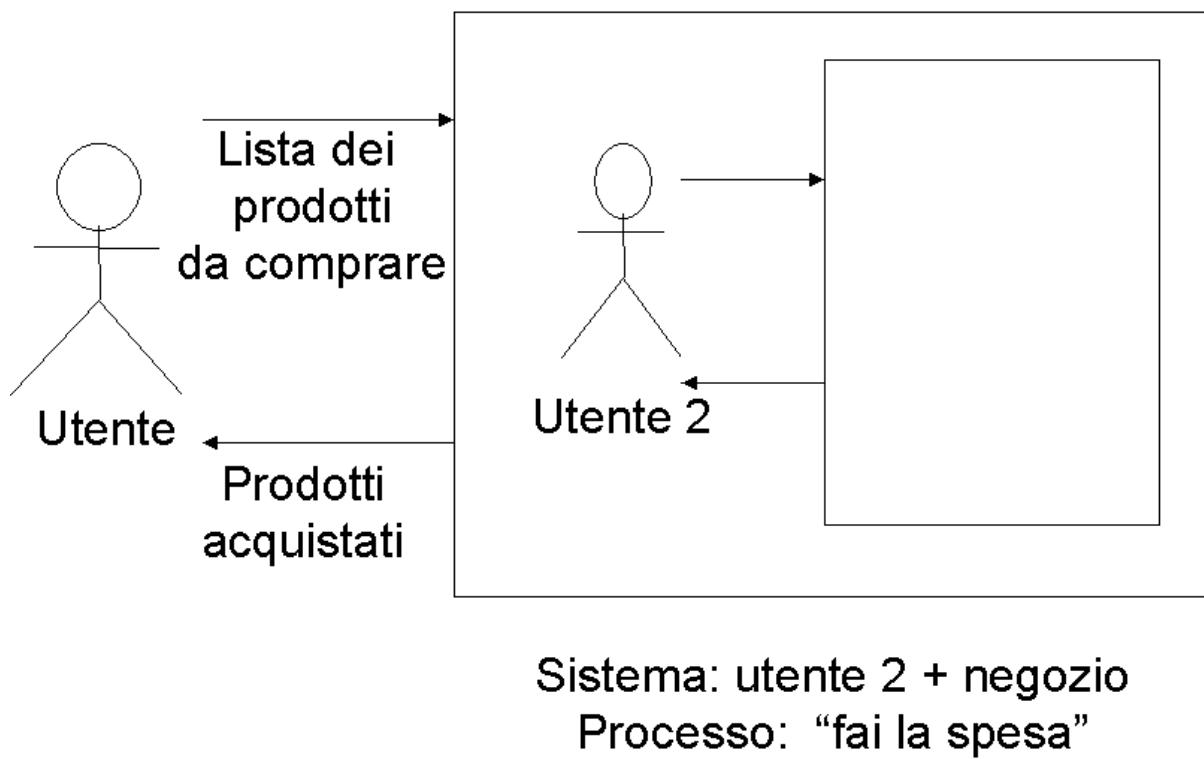


Figura 4.38: Inserimento di un nuovo utente (capo) che chiede al precedente (esecutore) di fare la spesa.

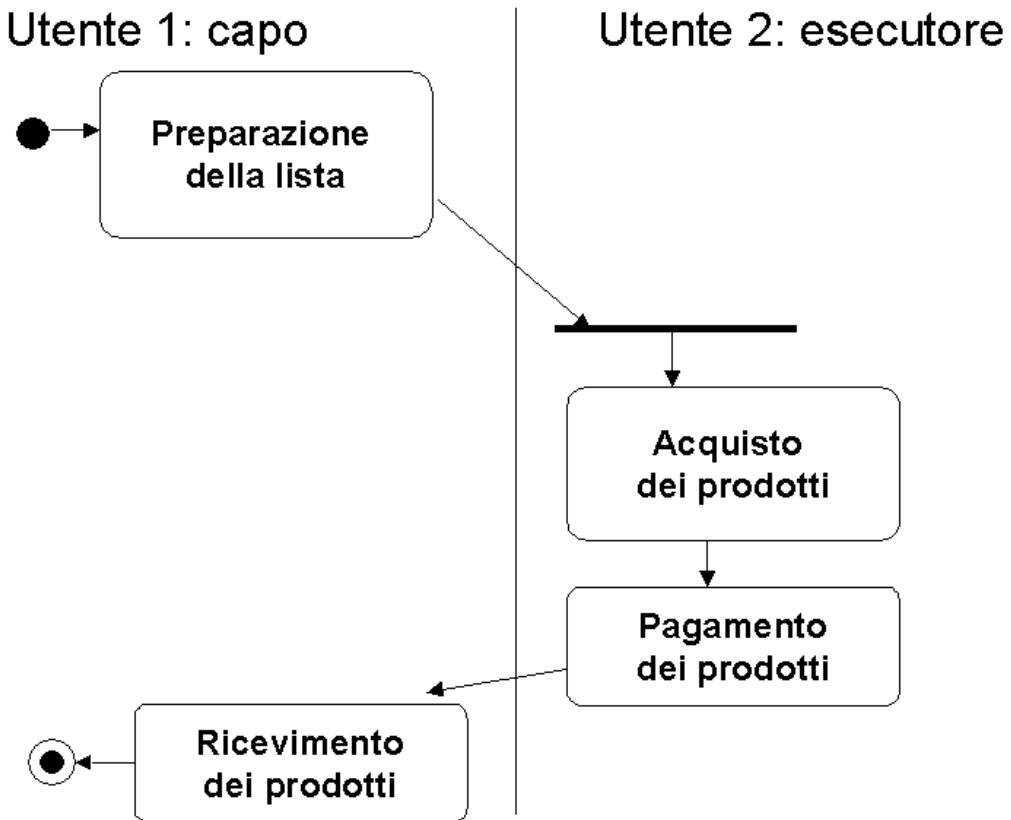


Figura 4.39: Mappatura delle attività del processo con la suddivisione dei compiti tra i due operatori.

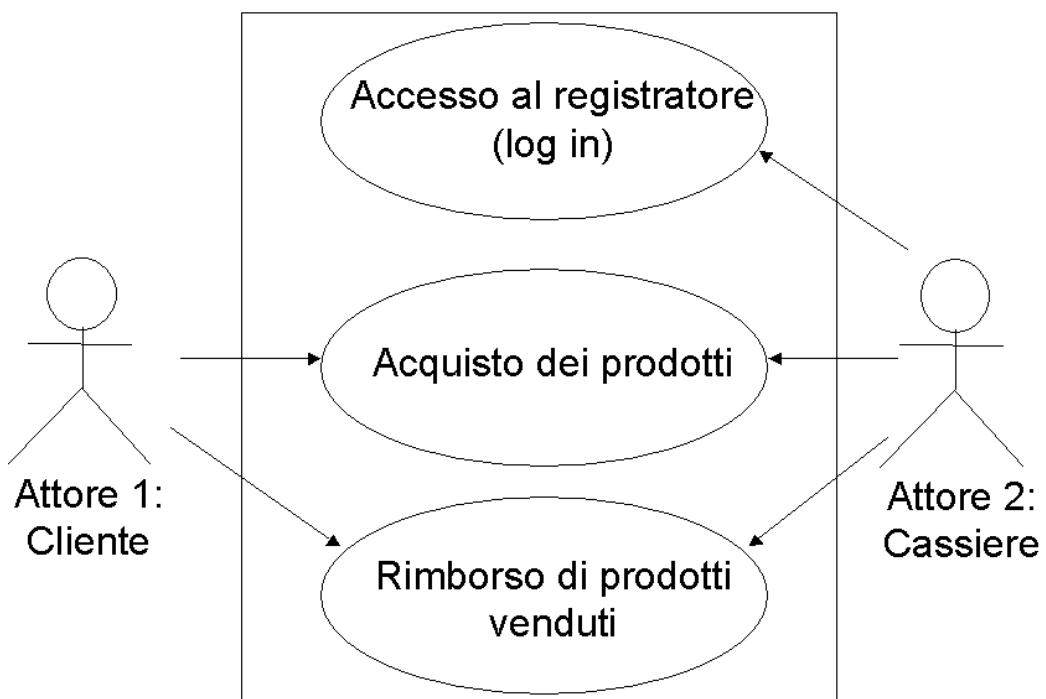


Figura 4.40: Scomposizione dell'attività di acquisto prodotti, con l'inserimento di altri casi d'uso possibili per il sistema registratore di cassa.

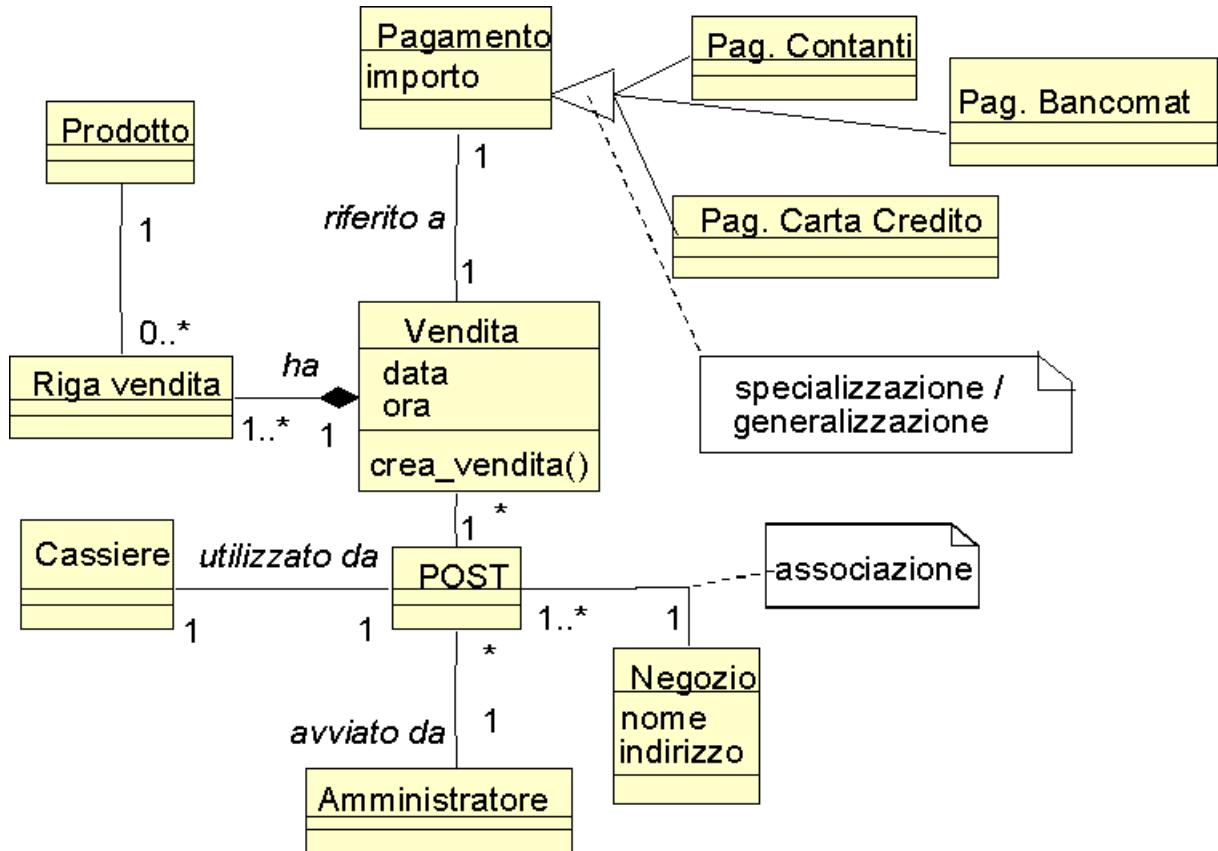


Figura 4.41: Class diagram che indica le entità principali della spesa; il POST (acronimo di Point Of Sale Terminal) è il registratore di cassa.

Una visione d'insieme: il legame fra le viste del processo

Il legame fra le viste rappresentate dai diagrammi e le informazioni ad esse associate viene espresso nella figura 4.44, dove si pone in evidenza che cosa significano i vari diagrammi e come possa essere necessario, in base all'analisi che si sta compiendo, porre l'accento sull'una o sull'altra delle caratteristiche, ovvero esaminare l'una o l'altra delle possibili viste del processo. Il processo viene ad essere il sistema e i diagrammi sono modelli che servono a focalizzare l'attenzione su alcune caratteristiche del sistema stesso.

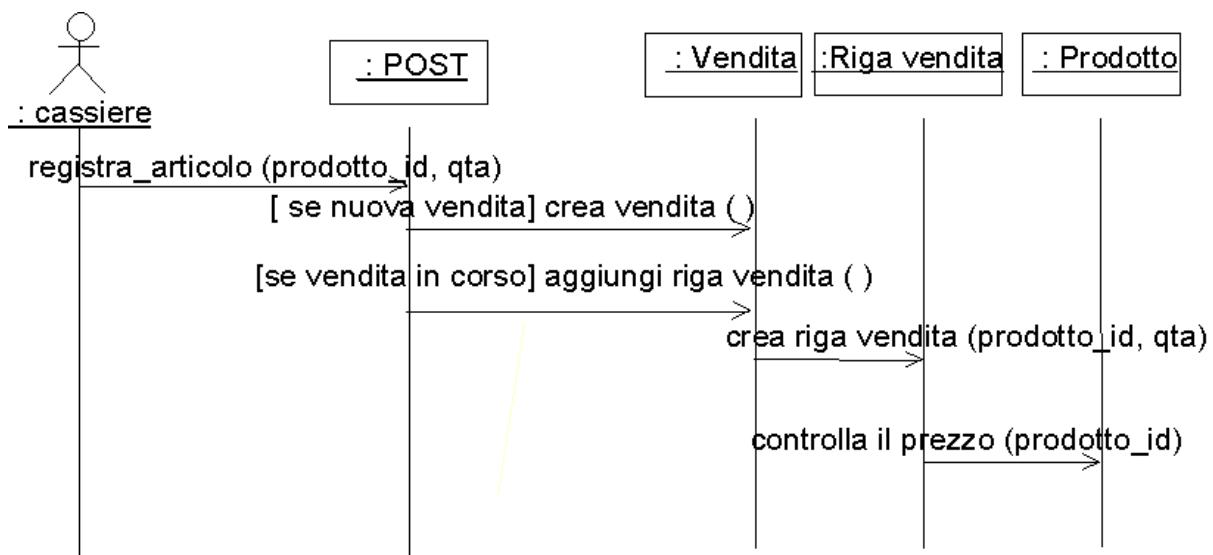


Figura 4.42: Sequence diagram che indica la sequenza temporale di scambio informazioni tra le entità della spesa.

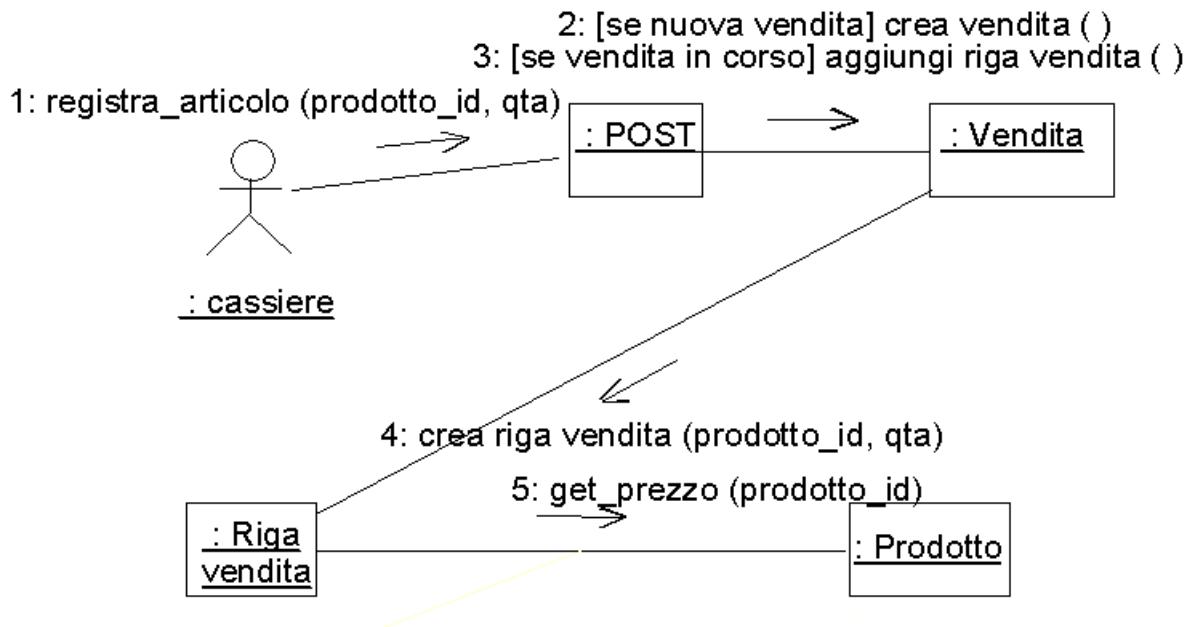


Figura 4.43: Collaboration diagram che indica lo scambio di informazioni tra le entità della spesa.

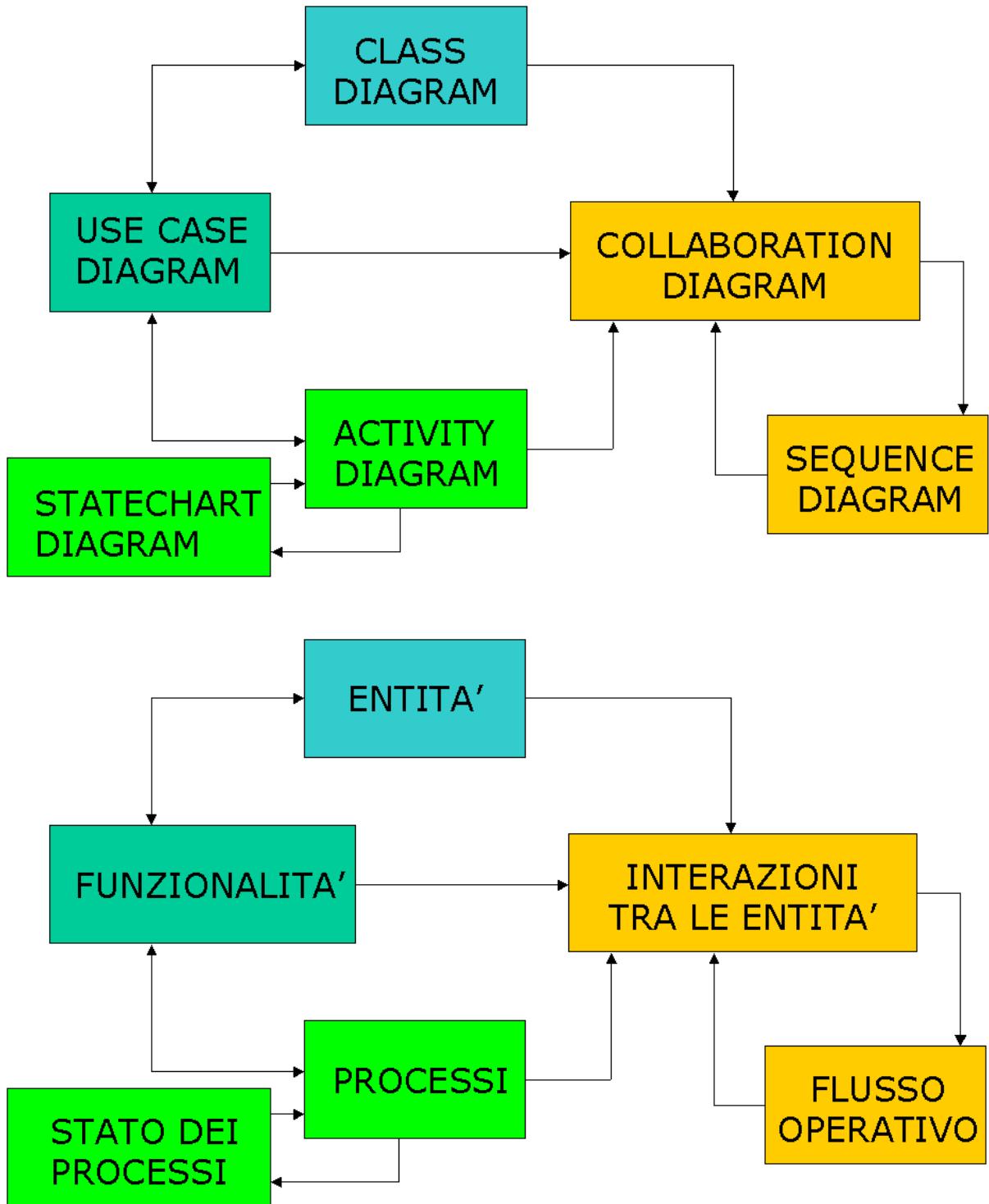


Figura 4.44: I diagrammi UML usati in UML for Business, le relazioni che tra essi intercorrono ed i concetti che essi esprimono.

Domande

1. Cos'è il Business Process Management (BPM)?
2. Cosa significa modellizzare un processo business?
3. Cosa si intende per Business Process Reengineering (BPR)? Che significato ha nel contesto di una evoluzione aziendale?
4. Cos'è il linguaggio UML e come può essere utile per descrivere i processi?
5. Cosa significa descrivere il processo come successione di attività?
6. Cosa significa descrivere il processo come successione di fasi di interazione fra elementi in esso coinvolti?
7. Perchè è utile definire le entità coinvolte in un processo e i loro legami logici?
8. Perchè è utile inserire l'elemento dinamico descrivendo le interazioni fra le entità?
9. Cosa significa descrivere il processo come successione di stati?
10. Qual'è il legame fra i vari modelli?

Bibliografia

[BFM 2001] G. Bracchi, C. Francalanci, G. Motta - *Sistemi Informativi e aziende in rete* – Ed. McGraw-Hill Italia, Milano, 2001

[BSV 1999] E. Bartezzaghi, G. Spina, R. Verganti - *Organizzare le PMI per la crescita. Come sviluppare i più avanzati modelli organizzativi: gestione per processi, lavoro per progetti, sviluppo delle competenze* - Ed. Il Sole 24 Ore, 1999

[Davenport 1993] , T.H. Davenport - *Process Innovation* - Harvard Business School Press, Boston, MA, 1993

[De Marco 2000] M. De Marco - *Sistemi Informativi Aziendali* - Franco Angeli Edizioni, Milano 2000

[EP 2000] H.E. Eriksson, M. Penker - *Business Modeling with UML* - Ed. Wiley and Sons, 2000

[Fowler 2003] M. Fowler - *UML Distilled: A Brief Guide to the Standard Object Modeling Language, Third Edition* – Ed. Addison-Wesley, 2003

[Hammer 1990] M. Hammer - *Reengineering Work: Don't automate, obliterate* - Harvard Business Review, Jul/Aug 1990, pp 104-112

[HC 2004] M. Hammer and J. Champy - *Reengineering the Corporation : A Manifesto for Business Revolution* - Updated Edition, Ed. Collins, 2004

[LL 2004] K Laudon, J Laudon – *Management dei sistemi informativi* – Ed. Pearson Education Italia, Milano, 2004

[Malhotra 1998] Y. Malhotra - *Business Process Redesign: An Overview* - IEEE Engineering Management Review, vol. 26, no. 3, Fall 1998, su Web <http://www.kmbook.com/bpr.htm>

[OMG.org 2005] Object Management Group su Web <http://www.omg.org>

Il sistema informatico entro il sistema informativo

Il sistema informatico

Il sistema informatico si compone, come detto, delle risorse tecnologiche atte al trattamento, elaborazione e memorizzazione dell'informazione nei formati digitali definiti in precedenza.

Il sistema informatico quindi raccoglie, elabora, archivia, scambia informazione mediante l'uso delle tecnologie proprie dell'Informazione e della Comunicazione (ICT): calcolatori, periferiche, mezzi di comunicazione, programmi. Nel seguito vengono presentate alcune definizioni, estremamente importanti, relative al sistema informatico.

Si definisce **programma software**, o semplicemente **programma**, un'applicazione software avente una sua identità precisa (ad esempio Word, programma per videoscrittura, o Excel, programma per funzionalità di foglio elettronico). Un insieme di applicazioni può andare a formare un sistema software adatto allo svolgimento delle attività di uno o più processi business.

Si definisce **processo informatico** un programma software in esecuzione, associato normalmente allo svolgimento delle attività di uno o più processi business. Il processo informatico usa varie risorse (percentuale del tempo della CPU, memoria RAM, disco entro un computer e canali di comunicazione via rete) per svolgere il proprio compito.

Si definisce **DBMS** (Data Base Management System) un sistema software che standardizza l'accesso dei processi ai dati, offrendo delle interfacce generalizzate che permettono:

la condivisione dei dati da parte dei processi informatici

l'indipendenza dei dati rispetto ai processi

Si definisce **database** o **base di dati** un insieme di archivi di dati gestiti contemporaneamente in modo efficiente ed unitario dal DBMS. Spesso il termine database viene usato anche come sinonimo di DBMS e non solo di base di dati.

Si definisce **transazione** un'unità logica di elaborazione, cioè una sequenza di operazioni che hanno un effetto globale sul database, vista come un insieme atomico, che completa con successo o fallisce, senza nessuna possibilità intermedia. Una transazione può essere pensata anche come *l'insieme delle operazioni informatiche attraverso cui si realizza una azione od operazione, cioè il componente minimo, non ulteriormente scomponibile, di un processo business*. Una transazione gode delle proprietà ACID (Atomicity Consistency Isolation Durability):

- **Atomicity (Atomicità)**: tutte le operazioni della sequenza terminano con successo (commit) oppure, se anche una sola di esse fallisce, l'intera transazione viene abortita (abort);
- **Consistency (Consistenza)**: una transazione è una trasformazione corretta dello stato del database, vale a dire, al termine di ogni transazione il database deve trovarsi in uno stato consistente;

- **Isolation (Isolamento)**: l'effetto di esecuzioni concorrenti di più transazioni deve essere equivalente ad una esecuzione seriale delle stesse. Quindi, transazioni concorrenti non devono influenzarsi reciprocamente;
- **Durability (Durabilità)**: gli effetti sulla base di dati prodotti da una transazione terminata con successo sono permanenti, cioè non sono compromessi da eventuali malfunzionamenti.

Il sistema informatico può essere suddiviso in molti modi e la sua struttura può essere studiata da vari punti di vista, nel presente paragrafo viene presentata principalmente una scomposizione tecnico-funzionale.

Una stratificazione tecnica suddivide il sistema informatico in quattro livelli:

- **Livello applicativo**, composto dagli applicativi software con cui gli utenti interagiscono e che effettuano le operazioni sui dati, ovvero le operazioni vere e proprie di trattamento dell'informazione; gli utenti non addetti alla tecnologia di norma vedono solo questo livello, che ha a sua volta una struttura interna piuttosto complessa che sarà esaminata nel seguito;
- **Livello del software infrastrutturale**, composto dai software di infrastruttura, che offrono servizi ai software dei livelli soprastanti, come, ad esempio, i database server o i Web server e dai sistemi operativi che consentono ai calcolatori di funzionare e gestire i vari programmi in esecuzione;
- **Livello dell'hardware**, formato dall'hardware dei computer, suddivisi tra le postazioni dei singoli utenti, chiamate anche workstation o computer client (o semplicemente client), di solito costituite da PC, e i computer che ospitano dati e servizi condivisi fra i vari utenti, chiamati computer server (o semplicemente server);
- **Livello della rete**, costituito dalla infrastruttura di rete, formata dai cavi e altre strutture di collegamento e dagli apparati di trasmissione (router, hub ecc...).

Il livello applicativo a sua volta può essere suddiviso in quattro livelli:

- Livello **strategico**, che tratta l'informazione destinata ai Top manager;
- Livello **manageriale**, che tratta l'informazione gestita dai Middle manager;
- Livello di **gestione conoscenza**, che tratta l'informazione gestita dagli Operatori di gestione conoscenza;
- Livello **operativo**, che tratta l'informazione gestita dagli addetti operativi.

Tale suddivisione può anche essere pensata come un'evoluzione della stratificazione dei processi vista nel modello della Piramide di Anthony presentato nel capitolo 2. E' necessario ora espandere ogni livello, valutandone caratteristiche, input, output ed utenti tipici (si vedano [BFM 2001] e [LL 2004] per approfondimenti). E' importante tenere presente che la strutturazione presentata definisce un modello generale che, in funzione della struttura organizzativa interna (quindi delle risorse organizzative), del tipo di applicazioni software in uso (quindi delle risorse tecnologiche) e dell'esperienza delle risorse umane assumerà caratteristiche specifiche all'interno di ogni azienda.

I sistemi di **livello operativo** costituiscono di solito il componente quantitativamente più presente del sistema informatico. Supportano la registrazione delle attività

elementari e delle transazioni che si svolgono nell'azienda, come, per esempio, vendite, incassi, depositi contante, paghe. Il loro scopo principale è quindi supportare le attività routinarie e registrare il flusso delle transazioni entro l'azienda, al livello operativo. Il loro componente fondamentale sono i **Transaction Processing Systems (TPS)**, chiamati anche On-line Transaction Processing (sottointeso systems) (**OLTP**), che svolgono e registrano le transazioni di routine necessarie per le attività quotidiane dell'azienda, come per esempio:

Registrazione ordini
Prenotazioni alberghiere
Documentazione spedizioni
Calcolo stipendi

Riassumendo quindi le caratteristiche degli applicativi del livello operativo, possiamo definire:

- **Input:** transazioni, eventi
- **Elaborazioni svolte:** Aggiornamenti, inserimenti, ordinamenti, unioni, produzione elenchi
- **Output:** documenti, report, liste, riepiloghi
- **Utenti tipici:** personale operativo, supervisori.

I sistemi di **gestione conoscenza** sono suddivisi in due componenti:

1. **sistemi per l'ufficio**, che aumentano la produttività dei lavoratori su documenti e dati; in questa categoria rientrano funzioni di:

Elaborazione testi (ad esempio, MS Word e Writer di OpenOffice)
Fogli elettronici (ad esempio, MS Excel e Calc di OpenOffice)
Database personali (ad esempio, MS Access o la rubrica di Windows), ovvero archivi ordinati di dati destinati alle esigenze di una sola persona
Minireport
Desktop publishing (ad esempio, MS PowerPoint o Impress di OpenOffice)
Trattamento elettronico documenti

Per gli applicativi per l'ufficio, possiamo definire:

- **Input:** documenti, programmi, posta elettronica
- **Elaborazioni svolte:** gestione documenti, programmi, comunicazioni
- **Output:** documenti, programmi, posta elettronica
- **Utenti tipici:** impiegati di ogni livello.

2. I sistemi di gestione conoscenza veri e propri, meglio noti come **Knowledge Working Systems (KWS)**, che supportano i lavoratori della conoscenza o knowledge workers nella creazione di nuova conoscenza e chi gestisce i dati ad integrare nuova conoscenza nelle proprie attività, oltre che a gestire i flussi di dati. Esempi di questa categoria sono:

Sistemi di progetto (civile, meccanico, chimico, informatico...), CAD, CASE
Archivi di progetti e conoscenza

Per gli applicativi di gestione conoscenza possiamo definire:

- **Input:** specifiche progettuali, elementi di conoscenza
- **Elaborazioni svolte:** modellazione, simulazioni

- **Output:** modelli, progetti, grafici
- **Utenti tipici:** professionisti, staff tecnico, progettisti...

I **sistemi di supporto dell'attività manageriale** favoriscono le attività di Controllo e Monitoraggio e le attività Decisionali ed Amministrative dei middle manager.

Il loro obiettivo è fornire un quadro preciso di “Come sta andando l’azienda?”

Di solito tendono a fornire report periodici e sono composti dai **Management Information Systems (MIS)**, che servono principalmente le funzioni di pianificazione e controllo, supportano le decisioni a livello manageriale e, di solito, traggono i dati dalle applicazioni dei livelli sottostanti.

Per gli applicativi di supporto all’attività manageriale possiamo definire:

- **Input:** riepilogo dati sulle transazioni, volumi (alti) di dati, semplici modelli
- **Elaborazioni svolte:** report di routine, modelli semplici, analisi di basso livello
- **Output:** riepiloghi, report delle eccezioni
- **Utenti tipici:** middle manager.

“In mezzo” fra il livello strategico e quello manageriale (possono essere considerati come appartenenti ad entrambi i livelli) sono i **Decision Support Systems (DSS)**. Rispondono anche essi alle esigenze del livello manageriale dell’azienda e aiutano a prendere decisioni in contesti non “unici” o nuovi per l’azienda. Si applicano anche a problemi per i quali la soluzione può non essere completamente nota in anticipo.

Per i DSS possiamo definire:

- **Input:** Bassi volumi di dati, grandi DB ottimizzati per lo scopo, modelli analitici e strumenti di analisi dei dati
- **Elaborazioni svolte:** simulazioni, analisi, correlazioni, interattività
- **Output:** report speciali, proposte, analisi delle decisioni, risposte alle interrogazioni
- **Utenti tipici:** professionisti, manager di staff, middle e top manager.

I DSS possono essere ulteriormente suddivisi in tre categorie:

- Sistemi *programmabili*, che incorporano la conoscenza di esperti umani e richiedono la fase di programmazione, ovvero di “trasferimento della conoscenza degli esperti umani” al loro interno;
- Sistemi *automatici*, che usano tecniche varie (reti neurali, algoritmi genetici, correlatori statistici ecc...) per giungere a una “decisione”; richiedono la fase di “addestramento”, ovvero la presentazione di una grande mole di esempi di ingressi con le corrispondenti uscite già note per potere “imparare” da tali esempi, estrapolando da essi e definendo le regole attraverso cui proporre le decisioni;
- Sistemi *ibridi*, che cioè usano caratteristiche di entrambe le categorie precedenti.

I **sistemi di supporto delle attività strategiche** aiutano i senior manager ad affrontare i problemi strategici e a valutare le tendenze a lungo termine, interne ed esterne all'azienda

Il loro obiettivo è rispondere a domande del tipo “Quale sarà lo status fra 1 anno?”

Sono formati dagli **Executive-Support Systems (ESS)**

Rispondono alle necessità di livello strategico delle aziende e riguardano decisioni non di routine che richiedono valutazioni, giudizi, conoscenze approfondite, creando un ambiente generalizzato di calcolo e comunicazione.

Per i sistemi di supporto delle attività strategiche possiamo definire:

- **Input:** dati aggregati, interni ed esterni
- **Elaborazioni svolte:** simulazioni, grafici, interattività
- **Output:** proiezioni, risposte alle interrogazioni
- **Utenti tipici:** senior manager.

Nella figura 5.1 sono rappresentate le relazioni ed i flussi informativi che intercorrono fra i sistemi sopra definiti.

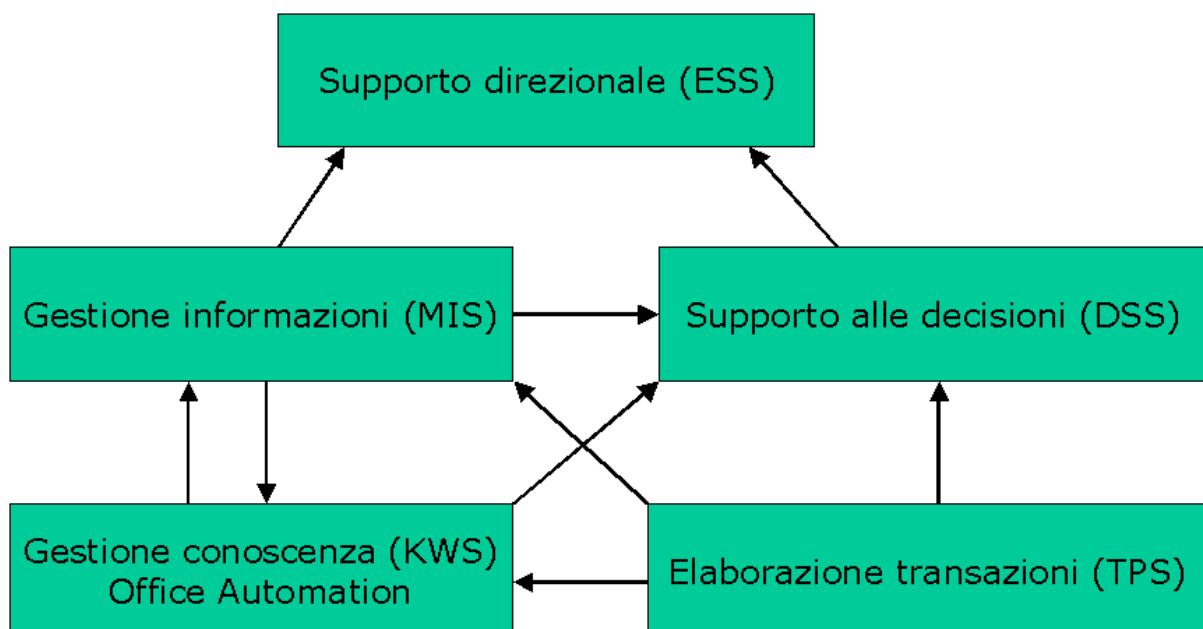


Figura 5.1: Le relazioni ed i flussi informativi che intercorrono fra i vari componenti dello strato applicativo del sistema informatico (fonte [LL 2004]).

Questa strutturazione nelle grandi imprese può anche essere applicata alle varie divisioni funzionali, come per esempio

- Vendite e Marketing
- Acquisti
- Produzione
- Gestione finanziaria
- Contabilità

- Risorse umane

realizzando funzionalità specifiche entro ogni divisione.

La strutturazione presentata è spesso corrispondente alla storia della informatica in azienda: nell'azienda strutturata per funzioni ogni reparto procede alla “meccanizzazione”, ossia all'inserimento semplice delle tecnologie IT entro le funzioni del lavoro, per proprio conto e le applicazioni specialistiche non sempre prevedono una facile integrazione tra di loro, creando il cosiddetto problema delle “isole informatiche” (talvolta detti silos informatici), cioè sistemi informatici diversi in ogni reparto, che usano formati di rappresentazione dei dati diversi, costringendo i responsabili IT alla creazione di apposite interfacce di comunicazione, che effettuino la traduzione dei dati tra i vari formati.

L'azienda moderna è sempre più strutturata per processi, con i flussi informatici che coinvolgono diverse divisioni, per cui i sistemi informatici collegati ai processi devono essere flessibili e riprogrammabili rapidamente seguendo l'evoluzione dei processi business stessi. Occorre l'integrazione dei sistemi o l'acquisizione di un sistema nuovo integrato, con l'unificazione delle basi di dati dei vari applicativi entro un database comune.

Il passaggio ai sistemi integrati, avvenuto a partire degli anni'90, ha condotto alla creazione di nuove grandi applicazioni software, modulari ed estese molto spesso a tutta l'azienda:

- Applicazioni gestionali integrate (ERP)
- Applicazioni per la gestione dei processi di fornitura, trasporto, vendita (SCM)
- Applicazioni per la gestione delle relazioni con i clienti (CRM)
- Applicazioni per la gestione della conoscenza (KMS e Business intelligence).

Prima di procedere con l'esame di alcune di queste grandi categorie di applicazioni software, che sarà compiuto nel prossimo capitolo, ora è necessario esaminare le relazioni che intercorrono entro il sistema informatico tra gli strati software del livello applicativo e i livelli sottostanti (software infrastrutturale e sistemi operativi, hardware e reti).

La struttura di un'applicazione software

Le applicazioni software dentro i sistemi informativi sono suddivisibili nelle seguenti categorie:

- **Interattive:** applicazioni che prevedono che un operatore umano interagisca col programma, dando a questo i comandi che esso deve eseguire e ricevendo da esso i risultati delle operazioni compiute; molte delle applicazioni rientrano in questa categoria, esempi della quale sono le applicazioni di MS-Office, i programmi gestionali delle anagrafiche clienti, prodotti, fornitori, il browser per la navigazione in Internet;
- **Macchina-macchina:** due o più applicazioni comunicano tra loro, senza intervento diretto (se non per le fasi iniziali di configurazione e test) di operatori umani; esempi di queste applicazioni sono i sistemi bancari che effettuano i processi di clearing, ossia gli scambi di fondi in formato elettronico fra due banche;

- **Batch:** un'applicazione inizia a momenti predefiniti, legge dati, li elabora, li salva e poi termina; esempi di queste applicazioni sono i generatori di report delle banche che ogni trimestre producono gli estratti conto da inviare ai clienti.

In ogni applicazione software interattiva usata entro i sistemi informativi (e in parte anche nelle applicazioni della categoria macchina-macchina), è riconoscibile una struttura a strati che corrisponde al seguente modello:

- **Interfaccia utente (tipicamente grafica)**, noto anche come **presentation layer**: strato che si occupa della presentazione dei dati in formato alfanumerico e/o grafico all'utente e gestisce in generale l'interazione con l'utente stesso, interpretando i comandi trasmessi dall'utente attraverso la tastiera, il mouse od altre periferiche (es. penna ottica), trasmettendoli allo strato inferiore, ricevendo da questo le risposte (es. dati elaborati) e visualizzandoli all'utente (per la categoria macchina-macchina questo è lo strato di comunicazione fra le due applicazioni, che effettua le conversioni fra i formati di rappresentazione dei dati usati entro le due applicazioni);
- **Regole funzionali (logica business)**, noto anche come **business rule layer**: insieme di funzioni (ossia sottoprogrammi che svolgono un compito ritornando direttamente un valore risultante) e procedure (sottoprogrammi che svolgono un compito senza ritornare direttamente un valore esplicito) che compiono le operazioni, ossia vengono attivate, in base ai comandi ricevuti dal livello precedente, ossia i comandi degli utenti; ciascuna procedura o funzione implementa in modo comprensibile ed eseguibile dal calcolatore una o più regole corrispondenti ad una operazione che fa parte di un processo business; nel caso che la struttura del programma sia ad oggetti, le procedure e le funzioni sono raggruppate entro unità logiche, dette appunto oggetti, ciascuna delle quali può essere la proiezione nel dominio dell'applicazione software di un'entità del dominio business, ossia del mondo dell'azienda, oppure un modulo originato da un'analisi tecnica che raggruppa funzioni simili;
- **Dati**, noto anche come **data layer**: i dati sono gli elementi finali su cui si deve agire e che devono essere memorizzati in supporti permanenti, come dischi, nastri, CD/DVD-ROM o altro; i dati durano oltre i programmi e anzi spesso sono trattati da più programmi diversi, essendo parte del flusso informativo associato al processo entro cui i programmi operano svolgendo ciascuno una delle attività del processo; nella maggior parte dei programmi moderni i dati sono memorizzati in database entro appositi DBMS, che ne permettono la memorizzazione permanente sicura ed allo stesso tempo la condivisione fra più applicazioni diverse.

Questa strutturazione, che nei primi sistemi era unicamente logica, oggi è spesso anche fisica: grazie all'avvento delle reti, gli applicativi client-server sono composti di due o tre componenti che svolgono le funzioni sopra descritte. Pertanto, per capire l'evoluzione compiuta negli ultimi anni è indispensabile comprendere il ruolo delle reti entro i sistemi informatici moderni.

Le reti entro i sistemi informatici

I sistemi informatici oggi sono basati sul concetto di rete. Il termine **rete** è nato per indicare in modo generico un **collegamento** tra **due apparecchiature** (**sorgente e destinazione**) attraverso un **mezzo trasmisivo** per effettuare una **trasmissione di informazioni**.

All'inizio le reti erano costituite essenzialmente da **terminali** remoti collegati a unità centrali (**mainframe**) mediante linee telefoniche o telegrafiche; l'uso di terminali remoti per l'elaborazione era noto come **teleprocessing**; la potenza di elaborazione era concentrata in un punto (**architettura centralizzata** o master/slave). La realizzazione di questo tipo di reti era legata a **soluzioni proprietarie** (una soluzione si dice proprietaria quando la realizzazione dipende dal costruttore ed è incompatibile con scelte di costruttori diversi; le specifiche costruttive in questo caso non sono quasi mai pubbliche).

Attualmente per rete di calcolatori si intende un insieme di **computer** indipendenti, cioè che possono lavorare autonomamente, ma collegati tra loro in modo da potersi scambiare informazioni (**architettura distribuita**).

Inoltre si è sentita la necessità di realizzare **sistemi aperti** che consentano di collegare e utilizzare prodotti di costruttori diversi; ciò rende necessario lo sviluppo di **standard** comuni. In particolare oggi lo standard sui cui poggiano le funzionalità di basso livello, ossia in pratica la trasmissione di flussi di byte tra un computer ed un altro o da un computer a molti altri, è il **TCP/IP**, ossia il protocollo nato per la rete Internet. Circa il 95% delle reti a livello mondiale usa ormai questo protocollo, rendendo possibile la interconnessione totale fra computer a livello mondiale. Purtroppo questo non significa necessariamente anche trasmissione di dati ed informazioni, in quanto il protocollo *garantisce solo la trasmissione di byte*, mentre, come già detto, *l'informazione che essi esprimono dipende dalla rappresentazione utilizzata*. Molto spesso, quando attraverso la rete due applicazioni colloquiano fra di loro, *si può rendere necessario l'inserimento di una opportuna interfaccia che effettui la conversione fra le rappresentazioni dei formati dei dati che le due applicazioni usano al loro interno*.

Il collegamento di computer in rete offre maggiore affidabilità e minor costo rispetto all'uso di mainframe e terminali. Un fattore importante a favore delle architetture distribuite è la loro migliore **scalabilità**, cioè la possibilità di aumentare le risorse della rete in base alle necessità, con un costo inferiore per le nuove risorse inserite nei sistemi distribuiti rispetto al costo del potenziamento dei sistemi centralizzati.

Le reti permettono tra l'altro:

- di **condividere risorse** (per esempio i dati in un file, una base di dati, una stampante, un fax...);
- di **comunicare** tra persone lontane (usando posta elettronica, videoconferenze ecc.); a tal proposito si ricordi il recente boom della telefonia via Internet, resa possibile dal protocollo Voice-Over-IP (VoIP, si veda in proposito [VoIP 2006]) e dalle sue implementazioni commerciali come, ad esempio, il programma gratuito Skype [Skype 2006];

- di **utilizzare servizi** di vario tipo come consultazione di informazioni, commercio elettronico, applicazioni di tele-medicina e così via; i servizi a cui si può accedere in rete vengono detti anche servizi telematici; la telematica è una disciplina che nasce dai rapporti tra scienza delle telecomunicazioni e informatica e si occupa dell'elaborazione a distanza delle informazioni.

Di solito nelle architetture distribuite si usano sistemi di tipo **client/server**. I **server** sono computer su cui operano applicazioni che mettono a disposizione delle risorse o dei servizi. I **client** sono computer che chiedono ai server di accedere a una risorsa o di eseguire un certo lavoro. Il server esegue il lavoro e restituisce la risposta. In una rete in genere ci sono pochi server, più potenti, condivisi, e molti client poco potenti e meno costosi. Comunque il ruolo non è così netto; uno stesso computer può fungere da client o da server in situazioni diverse, o anche da server e da client contemporaneamente. Un esempio di ruolo contemporaneo di client e server è costituito dai sistemi Peer-to-Peer (P2P), in cui due computer comunicano tra di loro permettendo la condivisione di dati (tipicamente file) ed entrambi svolgono tutti e due i ruoli (si veda [P2P 2006] per approfondimenti).

I termini server e client sono usati anche in relazione con le applicazioni software: un programma server è in attesa di richieste attraverso la rete che i programmi client gli inviano.

Ogni computer della rete può essere usato in modo autonomo e deve avere un proprio sistema operativo. Per il collegamento in rete deve inoltre utilizzare un opportuno **software di rete**.

Attualmente tutti i sistemi operativi maggiormente diffusi sul mercato sono dotati delle funzioni necessarie per il collegamento in rete, cioè integrano il software di rete necessario per la comunicazione tra computer attraverso il protocollo TCP/IP.

Un'ulteriore evoluzione delle reti sono i **sistemi completamente distribuiti** (chiamati anche semplicemente sistemi distribuiti, anche se questo termine è spesso usato come sinonimo di reti client-server); un sistema distribuito è una rete di computer che usa un sistema operativo e altri software infrastrutturali in grado di rendere trasparente all'utente l'esistenza di molteplici computer autonomi. Con una rete gli utenti devono esplicitamente collegarsi a un certo computer per usarne i file o richiedere elaborazioni e occuparsi della gestione della rete; in un sistema distribuito è tutto fatto automaticamente dal sistema operativo e dagli altri software deputati a tale compito: l'utente utilizza il sistema distribuito in modo trasparente, cioè non sa dove si trovi in esecuzione il programma o dove si trovino i file utilizzati.

Un sistema distribuito può essere dotato di un vero e proprio sistema operativo distribuito; il sistema operativo in questo caso è un insieme di programmi distribuiti, cioè dispersi ed eseguiti sui vari computer. Un esempio di sistema distribuito, in rapida diffusione negli ultimi anni, è il Grid Computing, che sarà trattato al termine del capitolo.

In base alla dimensione le reti si possono dividere in

- locali (**LAN** - Local Area Network), estese ad un edificio od un gruppo di edifici, senza attraversare suolo pubblico;
- metropolitane (**MAN** - MetropolitanArea Network), estese ad una città;

- geografiche (**WAN** - WideArea Network), estese ad aree geografiche.

Le reti locali sono reti private all'interno di un singolo edificio o edifici vicini, di dimensione al più di qualche chilometro; non possono attraversare suolo pubblico. Hanno velocità tra 10 Megabit/s (Mbps) e 1 Gigabit/s (Gbps), anche se è agli inizi la diffusione degli standard a 10 Gbps, basso ritardo (decine di microsecondi) e basso tasso di errore.

Le reti metropolitane possono coprire un gruppo di uffici o una città e possono essere private o pubbliche.

Le reti geografiche coprono una grande area geografica, una nazione, un continente o l'intero pianeta. I computer che eseguono programmi per gli utenti (server host) sono collegati da una sottorete di comunicazione costituita da linee di trasmissione ed elementi di commutazione ed instradamento dei dati, noti come **router**. Quando un computer vuole comunicare con un altro, bisogna individuare un percorso per raggiungere il destinatario (routing); i router collegano due o più linee di trasmissione; quando i dati arrivano su una linea di ingresso, il router deve scegliere una linea di uscita per farli proseguire. Reti locali o metropolitane possono collegarsi a reti più ampie mediante un router o una serie di router.

Il collegamento di due o più reti si chiama internet. La comunicazione tra reti di tipo diverso è un problema noto come internetworking. Internet (con la I maiuscola) indica la rete planetaria che connette tutte le sottoreti fra loro.

La forma che assume la sottorete di comunicazione tra le varie stazioni (cioè i computer ad essa connessi) viene detta topologia di rete.

Le reti geografiche collegano punti precisi in zone geografiche diverse e non possono quindi generalmente rispettare una disposizione particolare; la topologia di queste reti viene detta a maglia.

Nelle reti locali tutte le stazioni sono a breve distanza tra loro e possono quindi essere collegate in modo opportuno, rispettando una particolare disposizione (per esempio, a bus, stella o anello).

La topologia è importante per aspetti come l'instradamento dei messaggi e la tolleranza ai guasti, inoltre influenza l'organizzazione logica dei protocolli di livello superiore.

Per meglio valutare l'impatto sui sistemi informatici delle reti occorre analizzare più in dettaglio gli aspetti tecnici della comunicazione, in particolare partendo dalla struttura delle reti (si vedano [Tanenbaum 2003] e [GMN 1995] per ulteriori approfondimenti).

Per potere realizzare la funzione di trasmettere byte in flusso bidirezionale e con controllo degli errori da un computer ad un altro, la rete compie in realtà numerose azioni diverse fra di loro. Tali azioni vengono suddivise in strati in base ad uno schema teorico standardizzato, il cosiddetto modello ISO-OSI, che segmenta la rete in sette livelli.

1. **Livello Fisico**, che definisce le caratteristiche del canale fisico che deve trasmettere il segnale fisico attraverso cui si propagano i bit e, in parte, la codifica dei bit stessi attraverso le variazioni di tale segnale. Nei sistemi moderni sono possibili molti diversi supporti fisici per trasmettere segnali, ad esempio:

- a. Doppino telefonico, costituito da una coppia di fili di rame intrecciati con opportune caratteristiche di impedenza; nato agli inizi della era della telefonia per trasmettere la voce; oggi viene usato anche per la trasmissione di dati con standard come l'ADSL o l'HDSL;
 - b. Cavo coassiale, costituito da cilindri conduttori concentrici separati da strati isolanti; viene usato per trasmettere segnali televisivi ma anche per trasmissioni dati (ad esempio con i vecchi standard Ethernet 10baseT);
 - c. Cavo a 4 coppie RJ-45; contiene l'equivalente di 4 doppini telefonici indipendenti ed è il più usato nelle reti locali;
 - d. Cavi in fibra ottica, che consentono enormi capacità di trasmissione, ovvero i più alti valori di numero di bit trasmessi per secondo (sino a alcune decine di gigabit/s); sono molto usati nelle reti geografiche e anche in alcune reti metropolitane;
 - e. Sistemi wireless a corto raggio; sono basati su raggi infrarossi (ad esempio lo standard IRDA) o segnali radio (es. lo standard Bluetooth); consentono integrazioni punto-punto fra dispositivi vari (ad esempio, fra palmare o telefono cellulare e PC, o tra cellulare e sistema viva-voce) ed sono efficaci in spazi limitati (es. qualche metro al massimo);
 - f. Sistemi wireless LAN; sono basati su onde radio e consentono di realizzare reti locali, in spazi che vanno sino a un centinaio di metri, senza collegamento fisico fra i computer che ne fanno parte; le velocità di trasmissione sono buone e giungono sino a circa 100 Mbit/s per lo standard Wi-Fi (da Wireless Fidelity) IEEE 802.11g oggi in commercio; velocità ancora superiori sono previste dai nuovi standard oggi in studio;
 - g. Sistemi wireless su scala geografica; sono anch'essi basati su onde radio e realizzano reti geografiche via radio; esempi sono il GSM dei telefoni cellulari, gli standard GPRS/EDGE/UMTS per le trasmissioni dati via cellulare e lo standard Wi-MAX per la realizzazione di reti metropolitane wireless;
 - h. Collegamento radio via satellite;
 - i. Collegamento ponte radio terrestre (es. HamRadio);
 - j. Cavo elettrico, ove il segnale viene inviato in modo da non interagire con la trasmissione di elettricità; vari sono gli standard, per esempio, Modbus Powerline, usato dall'Enel per il telecontrollo dei contatori elettronici.
2. **Livello data link** che definisce i controlli da fare sulla conversione bit-segnale e viceversa, che raggruppa i bit in pacchetti e ne governa il trasferimento da sorgente a destinazione, oltre a definire gli standard per gli indirizzi che identificano univocamente sorgente e destinazione. Molti sono gli standard che operano in questo livello, coprendo in parte anche il livello 1, per esempio Ethernet (IEEE 802.3 e derivati, che consente velocità teoriche sino a 10GB/s), token ring, ADSL, HDSL, ATM, Frame Relay, Wi-MAX, Wi-Fi, EDGE ecc....). Ovviamente si richiede che sorgente e destinazione operino con lo stesso protocollo.

3. **Livello rete**, che consente di trasferire pacchetti di dati attraverso reti eterogenee, quindi con diversi standard per Data Link e diversi supporti fisici, garantendo comunicazioni punto-punto o uno-a-molti. Lo standard usato nella quasi totalità dei casi è l'Internetworking Protocol (IP).
4. **Livello trasporto**, che serve a suddividere blocchi di dati (ad esempio dei file) in più pacchetti, stabilendo per essi un numero d'ordine e verificando il corretto arrivo (con eventuale richiesta di ritrasmissione dei pacchetti mancanti), permettendo così la ricomposizione del blocco dati originale a destinazione. Lo standard usato a questo livello nella quasi totalità dei casi è il Transport Control Protocol (TCP). Una utile analogia per capire in pieno il funzionamento del meccanismo è la rete stradale ed autostradale italiana. I dati che viaggiano su Internet possono essere pensati come degli insiemi di merci, suddivisi fra vari camion, che viaggiano secondo percorsi indipendenti. Si supponga, ad esempio, di prelevare del file da un server di Parigi e trasferirli a Parma: i dati che si vogliono prelevare sono suddivisi in un certo numero di “camion” diversi, ognuno dei quali segue un percorso potenzialmente diverso lungo la “rete stradale ed autostradale” (l’insieme di mezzi fisici su cui “viaggia” Internet). Ovviamente i “camion” (detti tecnicamente pacchetti) che viaggiano su autostrada (collegamenti con capacità superiore, come le reti di fibre ottiche) arriveranno più in fretta di altri. Arrivati a destinazione, i “contenuti dei camion” (chiamati spesso tecnicamente carico utile dei pacchetti) saranno ricomposti e l’utente otterrà l’insieme dei dati che gli interessano, senza doversi in alcun modo interessare a questo meccanismo di funzionamento, così come non è necessario, per l’utente della televisione, ricordare in ogni momento come funziona la televisione. L’unica cosa che potrà influenzare direttamente il lavoro dell’utente sarà la velocità risultante del collegamento, dovuta al percorso compiuto, cioè al fatto che i “camion” abbiano viaggiato su “autostrada” (ossia su una rete a larga banda, per esempio a 100 Mbit/s) o su una “stradina di campagna” (per esempio un collegamento a 64 Kbit/s). Il risultato finale del lavoro congiunto dei quattro strati ora descritti è quindi un canale (il cosiddetto **socket**) con la capacità di trasmettere, in modo trasparente per i livelli superiori, flussi bidirezionali di byte tra due computer o altri dispositivi, collegati ad una rete TCP/IP come Internet.
5. **Livello sessione**, che garantisce un controllo di flusso per le informazioni che scorrono sotto forma di byte lungo il canale, dal momento in cui esso viene stabilito al momento in cui viene chiuso al termine della comunicazione.
6. **Livello presentazione**, che definisce come procedere ad eventuali conversioni di formato qualora i computer in connessione fra loro usino formati diversi per la rappresentazione dei dati, come, per esempio, un testo ASCII su un PC, che deve diventare un testo EBCDIC su un mainframe IBM.
7. **Livello applicazione**, ovvero l’applicazione che effettua la comunicazione (ad esempio, un client di posta come Outlook dal lato client e un server di posta elettronica dal lato server). In realtà spesso i compiti assegnati a questi tre livelli dallo standard vengono in realtà svolti tutti dall’applicazione, mentre gli altri

livelli sono svolti dal software TCP/IP, modulo del sistema operativo, e dal software che gestisce l'hardware del dispositivo fisico (es. la scheda) di rete.

Un ultimo aspetto che ha caratterizzato l'evoluzione delle reti negli ultimi anni è quello relativo alla sicurezza, che sarà trattato in modo esteso nel capitolo 8.

L'avvento delle reti e dei protocolli non proprietari ma aperti ha reso possibile l'evoluzione che ora vedremo.

L'evoluzione tecnologica: dal monolite al middleware

La manutenzione è sempre stata uno dei maggiori problemi per tutti coloro che si occupano di software. La manutenzione è la fase che segue l'entrata in servizio di un software, che svolge un compito entro un sistema informatico. Il termine manutenzione viene tipicamente usato per descrivere due attività distinte:

- la prima (**manutenzione evolutiva**) rappresenta i cambiamenti che il software deve subire per adattarsi alle nuove specifiche (per esempio, nuove funzionalità o il porting su nuovo hardware);
- la seconda (**manutenzione ordinaria**) è la rimozione di errori che in realtà non avrebbero dovuto essere presenti, ma che sono sfuggiti alle fasi di test prima del rilascio del software stesso.

Da studi effettuati si può osservare che la maggior parte del costo del software, durante la sua vita operativa, è interamente causato dalla manutenzione. Si può capire dunque come nel corso degli anni si siano sviluppate delle **architetture software** sempre più flessibili che potessero rappresentare dei veri e propri investimenti a lungo termine, riducendo i costi di manutenzione. Anche le architetture hardware sottostanti al software hanno subito l'influenza di quest'ultimo.

Il modello a Mainframe, storicamente il primo modello di sistema informatico aziendale interattivo, prevede che tutto deve essere centralizzato con il Mainframe a rappresentare la cosiddetta unità centrale, spesso chiamato anche "il cervellone". In molte aziende, specialmente banche, il sistema informatico è ancora organizzato, almeno parzialmente, secondo questo modello. A questa unità centrale sono collegati un numero anche molto grande di terminali, di solito aventi lo schermo nero a fosfori verdi. I terminali di questo tipo più diffusi sono gli IBM 3270 per i sistemi Mainframe IBM. Questi terminali sono detti anche "stupidi" sia perché possiedono poca memoria sia perché sono soggetti passivi, privi di sistema operativo, indipendenti da ciò che accade all'interno del Mainframe, e adatti solo a stabilire una sessione di lavoro con esso. In pratica è come se fossero un video ed una tastiera connessi al calcolatore attraverso un cavo molto lungo. L'unico vantaggio è che possono essere collegati a qualunque Mainframe senza richiedere cambiamenti di configurazione. Tutti i mutamenti del software o delle configurazioni degli utenti hanno luogo esclusivamente sul Mainframe e tutti i terminali "subiscono" automaticamente le modifiche.

Col passare degli anni, la manutenzione di questi grandi sistemi di calcolo è divenuta tutt'altro che semplice, ed ha condotto al concetto di **legacy**; con questo termine si intende "un oggetto di cui non si può più fare a meno, ma che nessuno riesce più a controllare pienamente". Molto spesso, associata alla legacy c'è poi una situazione di

software ormai inadeguato, che più che subire degli aggiornamento è soggetto continue correzioni, spesso non documentate, che rendono le ulteriori manutenzioni più difficili di giorno in giorno.

E' necessario tenere conto del fatto che i progetti dei Mainframe e dei loro sistemi operativi e programmi applicativi erano inizialmente fatti molto bene e rappresentavano il meglio dell'esperienza corrente del momento, seppure già alla base troppo rigidi. Ma nel corso dei decenni la gestione dell'insieme dei programmi applicativi (migliaia di righe di codice) passò da chi lo aveva creato ai cosiddetti "manutentori", operatori spesso intrappolati tra la pressione degli utenti che volevano sempre nuove funzionalità o semplificazioni di quelle esistenti in tempi brevi e la complessità e rigidezza del sistema stesso. Il risultato era la perdita del controllo completo del sistema, ossia l'aumento della legacy con la scrittura di centinaia di righe di codice non documentate.

Purtroppo il problema del legacy è sempre più presente nei sistemi informatici e ne sono esempio tutti quegli strumenti che, passando da una versione all'altra, cambiano le loro funzionalità a tal punto che tutte le cose che si sono prodotte con la versione precedente diventano obsolete, o l'uso di programmi scritti in linguaggi ormai quasi sconosciuti. In particolare in anni recenti è stato di scottante attualità il problema dell'anno 2000, generato dal fatto che nella maggior parte dei sistemi informatici solo 6 cifre venivano usate per indicare la data (secondo la nota rappresentazione gg/mm/aa), con la conseguente interpretazione dell'anno 2000 come 1900 o peggio. Il fatto che spesso sia stato necessario richiamare in servizio i programmatore originali (magari già in pensione) per la correzione dei programmi rappresenta la prova più evidente della perdita di controllo del codice stesso.

Il Mainframe monolitico "puro" era destinato a scomparire sia a causa del legacy sia a causa di problemi di costo di manutenzione e scalabilità: all'aumentare del numero dei terminali, per mantenere le stesse prestazioni l'unità centrale richiedeva espansioni estremamente costose (un raddoppio del numero dei terminali produceva più che un raddoppio dei costi).

La soluzione di questi problemi venne trovata nel ribaltare la situazione ovvero nel diminuire il carico computazionale del server e dotare invece il terminale (o meglio la postazione di lavoro) di funzionalità più "intelligenti".

Più formalmente, si potrebbe dire che questa architettura è la logica estensione alla programmazione modulare, il cui presupposto di base è la modularizzazione del software, ovvero la separazione di grossi stralci di codice in tante parti (moduli) che consentono uno sviluppo più facile e una migliore manutenzione. L'evoluzione successiva ha poi condotto al modello client-server definito all'inizio del presente capitolo.

Esiste molta confusione sul termine client-server, in quanto molte architetture che sono per molti aspetti differenti (perché per esempio forniscono ai client servizi diversi) vengono comunque fatte rientrare nello schema client/server. In generale nel modello client/server troviamo che un **programma server** opera su un **computer server (host)**, un **programma client** opera su una **postazione client (workstation)**. Le più

comuni funzionalità server oggi presenti nei sistemi informatici sono elencate qui in seguito.

File Server e Print Server

Questo tipo di servizio è uno dei più antichi e permette, per esempio, di accedere in lettura e/o scrittura a un file che non risiede fisicamente sul disco del computer che si sta usando, ma su una apposita **condivisione** del disco del computer che funge da file server. Un esempio tra i più noti è il servizio di condivisione cartelle di Windows, un sistema indispensabile per condividere file, documenti, configurazioni e molto altro. Il print server è un servizio analogo che consente di condividere fra più postazioni una stampante collegata fisicamente al computer che svolge il ruolo di print server. Praticamente tutti i sistemi operativi, freeware o proprietari oggi integrano servizi di file e print server.

Domain Server

Costituisce la centralizzazione della gestione accessi e privilegi, che gestisce in un unico punto sia l'accesso alla rete, sia l'accesso a singole risorse presenti entro la rete (ad esempio le cartelle condivise di un file server, cui solo utenti con particolari autorizzazioni hanno diritto di accedere). Praticamente tutti i sistemi operativi, freeware o proprietari oggi integrano servizi di domain server. Esempi di questo tipo di servizio sono le Active Directory dei sistemi Microsoft Windows 2000/2003, OpenLDAP del mondo Unix/Linux, RACF del mondo Mainframe IBM.

Database Server

E' sostanzialmente un **DBMS** (Data Base Management System), che consente la gestione centralizzata dei dati, condivisi fra tante postazioni client. In questo caso il Client di solito invia delle richieste che sono per lo più messaggi in formato SQL (Structured Query Language), ossia il linguaggio tipico dei DBMS relazionali, al Server. Il Server elabora questi messaggi e produce come risultato un set di dati che spedisce come risposta al Client, oppure, nel casi di istruzioni di modifica dei dati (inserimento, aggiornamento, cancellazione) il Server effettua la modifica dei dati, notificando poi al Client l'azione svolta.

Il mercato dei database server è suddiviso fra Oracle Server di Oracle Corporation (USA), IBM DB2 di IBM, SQL Server di Microsoft e altri minori (Sybase, CA Ingres ed altri), più alcuni prodotti freeware (MySQL, PostgreSQL ed altri).

Web Server

Il servizio Web è basato su particolari computer presenti nella rete, indicati come **nodi Web**, che rendono disponibili le informazioni in essi contenute sotto forma di **pagine ipertestuali**, contenenti **documenti multimediali** (ossia composti di testi, immagini fisse, filmati, suoni ecc...)

Entro il WWW ci si sposta nella rete come in un **ipertesto**, ossia selezionando semplicemente particolari "parole chiave" (chiamate tecnicamente **link**), "evidenziate" rispetto al resto del testo. Si può quindi "navigare" attraverso Internet semplicemente seguendo le parole chiave che collegano un documento ad un altro, indipendentemente

dalla collocazione fisica dei documenti stessi, che possono essere contenuti nel computer che si sta usando, o situati fisicamente in una banca dati, ad esempio, in Australia. Solo il tempo di accesso sarà maggiore.

Una ricerca di informazione eseguita in questo modo non richiede assolutamente la conoscenza, da parte dell'utente, della esatta ubicazione fisica dei documenti nella rete. Come già detto sopra, i documenti di WWW sono *iper testi multimediali*, quindi contengono testi, immagini, filmati, animazioni, suoni, voci ecc... ed è sufficiente selezionare la parola chiave (od anche l'immagine chiave o l'oggetto chiave entro una immagine) desiderata per raggiungere il documento legato a tale parola.

Il mercato dei Web Server è suddiviso fra Apache Web Server, del Consorzio Apache (ente di sviluppo indipendente, finanziato da aziende come IBM, Oracle, Sun, HP), che detiene il 62% del mercato (direttamente o con prodotti da esso derivati), MS Internet Information Server (IIS) con circa il 28% del mercato ed altri minori.

Il mercato del Web client (browser) è suddiviso fra MS Internet Explorer (circa 85% del mercato), Mozilla Firefox (circa 8% del mercato), Opera di OperaSoftware (Norvegia, circa il 3%), Safari di Apple (circa il 2%), più altri minori.

Terminal Server

I servizi di questo tipo consentono di accedere ad una macchina remota come se si fosse collegati direttamente alla sua console, attivando i processi direttamente su di essa e sfruttando quindi direttamente le sue risorse, in primo luogo la CPU. Si suddividono in due grandi categorie:

- i servizi di terminale a riga comando (simile quindi al prompt di MS-DOS presente in Windows), che aprono una interfaccia utente di tipo testuale come telnet (tipico del mondo UNIX, e possibile anche in Windows, sebbene non usato normalmente), tn3270 (tipico dei mainframe IBM), tn5250 (tipico dei sistemi IBM AS/400);
- i servizi di terminale grafico, tipici del mondo Windows ma possibili anche in UNIX, che, entro una finestra della macchina client, mostrano una finestra grafica o l'intero desktop della macchina remota, esattamente come se si fosse fatto l'accesso direttamente su di essa; accanto al protocollo di Microsoft terminal service, presente nel sistema operativo Windows, sono usati altri protocolli come VNC (multipiattaforma), Citrix Winframe, ecc...

Groupware Server

Questo servizio consente di mettere in contatto (tipicamente in modo asincrono, ma talvolta anche sincrono) più persone, per fare in modo che si scambino messaggi, documenti, immagini etc, come ad esempio è realizzato dai newsgroup di Internet. L'esempio più noto di groupware server è il mail server o server di posta elettronica, che predispone le caselle elettroniche degli utenti, nonché la possibilità per essi di spedire i messaggi. Evoluzione di esso sono le raccolte integrate di posta e documenti vari (es. MS Exchange Server, Lotus Notes). Altri esempi sono i Server di sviluppo per il lavoro in gruppo (es. CVS, RCCS, MS VisualSourceSafe).

Administration/monitoring server

Sono sistemi che offrono sorveglianza più o meno automatica dei sistemi di domain server, dei computer server (es. processi, uso disco, uso memoria), della rete (sniffer, analizzatori di traffico), dei servizi presenti nella rete (applicazioni, database). Tipicamente sono presenti esclusivamente nei sistemi informatici di grandi aziende od organizzazioni.

Application Server (Java)

Sono una categoria relativamente nuova, ospitano applicazioni che offrono i servizi infrastrutturali richiesti per operare da applicativi Java, tipicamente aventi una interfaccia utente Web, e ne garantiscono quindi robustezza e scalabilità. Per esempio, il framework ERP Oracle Fusion è basato sull'application server Oracle iAS. Il mercato di questi server è dominato da IBM WebSphere e BEA WebLogic, cui seguono il già citato Oracle iAS, Jboss, Sun iPlanet e altri minori.

Altri Server

Esistono molti altri servizi organizzati in architettura client-server, come i Transaction Server o gli Object Server, ma di solito essi sono integrati entro applicativi complessi client-server multi-tier (detti anche multilivello, cioè con più di due livelli di processi cooperanti attraverso la rete) di cui si parlerà in seguito.

Il mondo del client-server e le applicazioni multi-tier

In quasi tutte le applicazioni si possono riconoscere le tre differenti componenti, l'interfaccia utente, le business-rules (o business-logic) e i dati, come precedentemente visto. Se dunque si considera un ambiente tipicamente orientato alle basi di dati, possiamo schematizzare i tre livelli appena indicati come in figura 5.2.

Nelle applicazioni 2-Tier i primi due componenti (interfaccia e business-logic) sono unificati e il terzo è separato e rappresenta i servizi a cui accedere (per esempio, una base di dati).

Lo schema 2-Tier ha alcuni aspetti positivi: è di facile comprensione e quindi di facile implementazione, inoltre tale tecnica è particolarmente utile per piccole applicazioni dipartimentali, ossia specifiche di divisioni dell'azienda, come l'ufficio vendite o l'ufficio acquisti.

La gran parte delle istituzioni indicano la soluzione 2-Tier come la migliore, infatti in questo modo si sfruttarebbero al meglio il cosiddetti "cheap MIPS" dei PC (ovvero la loro potenza di calcolo a basso costo). Uno sguardo più approfondito verso le applicazioni commerciali indica invece una soluzione di diversa natura, infatti le applicazioni 2-Tier hanno un problema di scalabilità.

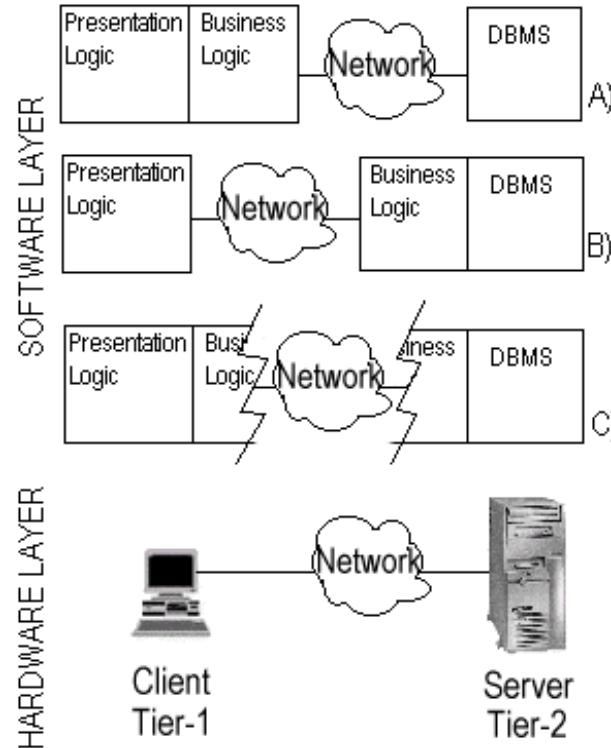


Figura 5.2 - Possibili configurazioni di un sistema client-server basato sul modello 2-Tier.

La prima osservazione che si può muovere si riferisce sempre alla figura 5.2, come si vede, esistono vari casi (A, B, C) che rappresentano come una applicazione 2-Tier può essere scritta. Purtroppo la gran parte delle applicazioni rientrano nel caso C), dove non c'è una separazione netta fra server e client e ogni tier fa una parte di quello che dovrebbe fare e una parte di quello che non dovrebbe fare.

Inoltre le applicazioni di grandi dimensioni, che tipicamente necessitano di complesse elaborazioni, alti volumi di transazioni e/o un gran numero di client, tendono a raggiungere presto la saturazione e questo può avvenire per causa di uno o più di questi tre problemi:

- Performance e Scalabilità;
- Complessità;
- Supportabilità.

Alcuni esempi illustrano meglio questi tre punti fondamentali.

Performance e Scalabilità

Questo limite è rappresentato per lo più dal numero di utenti che contemporaneamente useranno il sistema che si progetta. Gli utenti saranno dieci oppure saranno un migliaio? La risposta del sistema sarà accettabile anche quando ci saranno cinquecento utenti collegati? Tutte queste sono domande alle quali bisogna dare una risposta quando si progettano applicazioni su larga scala.

Un esempio che può illustrare la situazione è la costruzione di una applicazione di fatturazione. Si abbiano tanti client e un server sul quale risiede un database che contiene tutte le informazioni necessarie per stilare una fattura. Se i computer su cui girano i client sono veloci si potrebbe pensare di far spedire tutti i dati dal server ai client e far elaborare ogni fattura direttamente a ciascun client. In tal modo l'elaborazione da eseguire sul server calerebbe tantissimo (visto che ci sono moltissimi client), tuttavia, se il collo di bottiglia fosse rappresentato dalla rete, la scelta si rivelerebbe errata, in quanto tutti i client dovrebbero comunque attendere molto tempo, poiché i dati da trasmettere sono tanti e la rete è lenta, di conseguenza si dovrebbero far creare tutte le fatture al server e trasmettere solo i risultati ai client.

La complessità

La complessità è la complessità della applicazione, misurata come numero di algoritmi, oggetti e funzionalità. Nella maggior parte delle applicazioni 2-Tier l'interfaccia e la logica con cui si accede al database sono molto vincolate le une alle altre e la modifica di un componente causa una inevitabile modifica di un altro.

La complessità e le dimensioni della applicazione stessa salgono anche a fronte di un piccolo cambiamento, come potrebbe essere quello dell'aspetto dell'interfaccia utente. Altro problema ben più serio è rappresentato dal fatto che a causa dello stretto legame tra i vari componenti (interfaccia utente, business logic e data access logic) il programmatore deve essere un esperto di un gran numero di campi; qualità difficilmente riscontrabile nel programmatore medio. Sicuramente è molto più conveniente rendere modulare il codice e consentire allo sviluppatore di focalizzare la sua esperienza sul campo che conosce meglio.

La supportabilità

Ultimo non certo per importanza è il problema della supportabilità, può essere anche considerato come una conseguenza del problema della complessità.

Si consideri l'applicazione 2-Tier, supponendo di dover creare due interfacce grafiche diverse (a causa, per esempio, delle diverse richieste dei molteplici utenti); a questo punto le soluzioni sono due: o si scrivono due nuove applicazioni oppure si decide di ricostruire la presente perché supporti la possibilità di avere diverse interfacce grafiche; in ogni caso i costi e i tempi si dilatano.

Un problema ancora più serio è quello riguardante il cambiamento di funzionalità del DBMS: se la data access logic è cablata nella applicazione, diventa necessaria riscriverla per intero e poiché la struttura della applicazione non è modulare, questo si ripercuote in costi di aggiornamento elevatissimi.

In conclusione l'utilizzo di architetture 2-Tier e dei RAD (Rapid Application Development) tools, che tendono a generare programmi dalla struttura rigida, è indicato solo per applicazioni semplici che si prevede rimarranno sempre semplici.

L'evoluzione del Client-Server: il 3-tier e il multitier

La maggior parte delle applicazioni Client-Server sono di tipo dipartimentale, sono scritte secondo il modello 2-Tier e sono state progettate per supportare al massimo poche decine di utenti connessi simultaneamente e per eseguire funzioni non critiche. La gran parte di queste applicazioni sono state scritte per mezzo degli ambienti di sviluppo più in voga al momento (es. Visual Basic, .NET, Delphi, Visual C++, AcuCobol, Java). Incoraggiati dal successo iniziale di questo tipo di applicazioni, produttori di software ed aziende utilizzatrici hanno pensato di espanderne le funzionalità e conseguentemente di aumentarne la complessità, da questo punto in poi molte delle applicazioni 2-Tier hanno incominciato dimostrare i loro limiti e gli sviluppatori si sono accorti che quel tipo di architettura non era più adatta a supportare (e sopportare) il peso delle nuove funzionalità proposte dai manager.

Le architetture 3-Tier o a 3 livelli aiutano le aziende ad ammortizzare i loro investimenti creando applicazioni che effettivamente ripagano a lungo termine.

Tuttavia scrivere applicazioni di questo tipo è più complesso rispetto alle applicazioni a 2 livelli e alcuni dei tools che sono stati venduti nel passato, anche se supportano la stesura di questa categoria di applicazioni, sono ancora inadeguati e non forniscono tutti i servizi necessari per supportare un ambiente di calcolo distribuito. E' necessario poi tenere presente il futuro della applicazione al variare delle versioni del tool di sviluppo con cui essa è stata costruita. Come si vede dalla figura 5.3, se ci si riferisce ad una applicazione in termini logici, si osservano tre strati o tier:

- i servizi di presentazione (presentation services);
- i servizi dedicati all'elaborazione (process services);
- i servizi legati ai dati (data services).

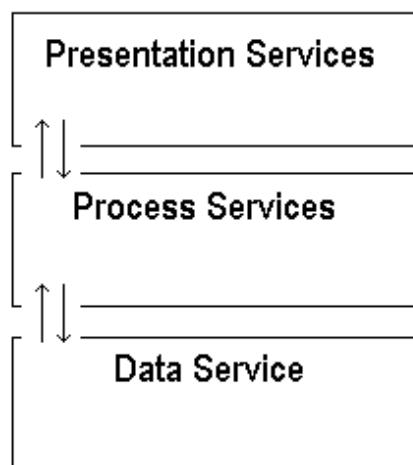


Figura 5.3 - I componenti logici di una architettura 3-Tier.

Il presentation service gestisce l'interfaccia utente verso il sistema. Questo tipo di servizio può essere realizzato come una tradizionale interfaccia grafica (classico programma che utilizzi le librerie grafiche proprie del sistema operativo su cui si sviluppa) oppure anche sotto forma di terminali (i classici IBM 3270 citati all'inizio

del capitolo); inoltre potrebbe anche venire realizzato in tecnologia Web nel qual caso l’interfaccia grafica sarebbe rappresentata dal Web browser.

Il process service, conosciuto anche come application service o business service, agisce come una sorta di buffer tra il livello superiore e quello inferiore. In una applicazione 3-Tier tutti gli accessi ai dati da parte degli utenti avvengono per mezzo del livello di presentazione, questo livello dunque incorpora tutte quelle elaborazioni degli input utente come controlli sulla validità delle richieste, analisi e autenticazione di password per accedere ai dati, il controllo del numero di utenti per non superare il limite di sovraccarico del sistema, nonché la possibilità di consentire l’automazione di eventi utili per l’amministratore di sistema.

Infine il data service rappresenta di solito il database vero e proprio, quindi sia i dati propriamente detti sia la logica che consente di aggiornarli, cancellarli e modificarli.

Bisogna osservare che la struttura che si è considerata fino ad ora è una struttura logica, ossia una “guida” per organizzare e scrivere un’applicazione client-server a tre livelli, come in figura 5.3. Esiste però anche una struttura fisica (spesso chiamata deployment architecture) associata a quella logica, che descrive dove i vari strati software (presentation, process, data) devono essere eseguiti o installati. Tale architettura è indicata in figura 5.4.

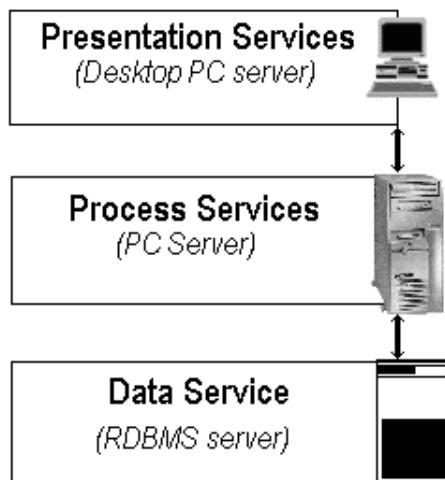


Figura 5.4 - I componenti fisici (deployment architecture) di una architettura 3-Tier.

Come si può osservare, non esiste una grande differenza tra la struttura fisica e quella logica e questo crea una sinergia intrinseca che porta molti effetti benefici.

La chiave della programmazione 3-Tier è quella di assicurare una netta separazione tra i livelli logici all’interno dell’applicazione; si è sicuri di aver scritto una applicazione 3-Tier corretta quando la sostituzione di uno qualunque dei 3 livelli non comporta la modifica degli altri due; quindi se si fosse scritta una applicazione con l’interfaccia utente (presentation services) in Windows, qualora se ne volesse scrivere una per Linux, questo non deve comportare alcuna modifica allo strato software sottostante (process services). Questo concetto è molto semplice tuttavia nasconde molte insidie per un programmatore. Per rendersi conto di questo basti pensare ad una banale applicazione il cui unico scopo è ricercare dei nomi in un database; per quanto

riguarda il primo livello (presentation layer) questo viene fatto con una interfaccia grafica che genera delle richieste (query) dai dati che inserisce l'utente, ebbene già così abbiamo legato il presentation layer al data layer, infatti il formato delle richieste, se non gestito opportunamente dal process layer, fa sì che qualora si cambi DBMS si debba anche modificare il presentation layer per uniformare il formato delle query a quello del nuovo database. Quindi senza neanche scrivere una linea di codice, si è già sbagliato il progetto intero. Ovviamente questo è solo un esempio molto semplificato. Fino a questo punto si è descritto come i vari componenti devono essere separati sia fisicamente che logicamente, tuttavia nulla si è detto circa il come i vari strati software devono o possono comunicare.

Ciò che collega i vari livelli (questo vale anche per il caso 2-Tier) è il **middleware**, gran parte delle applicazioni 3-Tier hanno livelli che comunicano per mezzo di vari tipi di middleware (ad esempio, socket, TLI, CPIC/APPC, NetBIOS e Named Pipes) con caratteristiche ben definite e diverse che rappresentano un vincolo per il funzionamento degli strati software che li usano (si veda [Tanenbaum 2003] per approfondimenti).

Dal punto di vista logico, ovvero della pura stesura del software, i benefici che una struttura di questo tipo porta sono i seguenti:

- scalabilità: le applicazioni 3-Tier scalano molto meglio di quelle 2-Tier;
- flessibilità: i servizi corrispondenti ai vari strati software possono venire aggiornati e/o sostituiti senza problemi, inoltre possono venire aggiornati e modificati da specialisti e non da programmatore, senza dover conoscere l'intera applicazione;
- minor costo a lungo termine: permettono infatti di sfruttare meglio la tecnologia già esistente, se per esempio è più familiare per l'utente usare un Web browser, basterà riscrivere solo il presentation layer;
- maggiori servizi: l'inserimento di nuovi servizi risulta semplificata dalla presenza del process layer;
- aumento della competitività: in generale viene aumentata la velocità di produzione in quanto ogni strato può crescere indipendentemente.

Inoltre dal punto di vista fisico, per il fatto di eseguire i vari tier su architetture hardware diverse, l'architettura porta i seguenti benefici:

- flessibilità di configurazione: si è liberi di decidere l'hardware più adatto per il livello opportuno (tipicamente il presentation layer va su un PC, il process layer potrebbe essere eseguito su una architettura multiprocessore, nota anche come SMP);
- espandibilità: alcune compagnie potrebbero utilizzare dei database gateway sul livello fisico intermedio (Middle-Tier) per consentire agli utenti un accesso trasparente a diverse basi di dati; ad esempio, il database gateway EDA/SQL traduce le API dei Client in sintassi appropriate per accedere a più di 50 diverse basi di dati SQL e non-SQL;
- trasparenza: se per qualche motivo divenisse necessario cambiare l'indirizzo di rete della macchina su cui risiede il database, i Client non si accorgerebbero di nulla perché sarebbe il Middle-Tier che si occuperebbe di questo;

- distribuzione dei servizi: è possibile distribuire i servizi su più piattaforme che appartengono al Middle-Tier, sarebbe come se i Client si dovessero collegare al Middle-Tier A per richiedere informazioni statistiche e al Middle-Tier B per richiedere informazioni amministrative; questa operazione di distribuzione dei dati risulta molto semplificata nel caso 3-Tier e aiuta ad avere una migliore distribuzione del carico di rete e del flusso di dati.

I grandi applicativi ERP, SCM, CRM rientrano quasi tutti nella categoria architettonica a 3-tier.

Un'evoluzione dell'architettura multi-tier è il mondo delle applicazioni Web, in cui il livello finale dell'interfaccia utente è costituito dal browser Web, ovvero da pagine HTML, e lo strato immediatamente sottostante è un server Web che provvede alla distribuzione di tali pagine, che spesso sono generate dinamicamente da appositi applicativi, eseguiti da estensioni del Web server o dagli Application Server presentati sopra.

In generale si può definire la seguente struttura per un'applicazione Web:

- **Web Client**, il browser, che visualizza la pagina HTML, ma che può eseguire anche istruzioni in linguaggio JavaScript in essa contenute;
- **Logica di presentazione**, costituita da tali pagine HTML, aventi lo scopo di visualizzare dati e richieste all'utente;
- **Logica di interazione lato client**, formata dalle istruzioni JavaScript contenute nelle pagine HTML, che può produrre veri e propri programmi applicativi client (detti anche client leggeri o lightweight client), come si può fare ad esempio con lo standard Asynchronous JavaScript and XML, meglio noto come AJAX, una tecnica per sviluppare applicazioni web interattive e dinamiche basata su combinazioni di HTML, JavaScript e XML (si veda [AJAX 2006] per approfondimenti);
- **Connessione HTTP**, il canale su cui viaggiano i dati;
- **Web Server**, che gestisce il canale HTTP;
- **Logica di azione lato server**, formata dalle Web form destinazione delle richieste dell'utente e che richiama le funzioni di libreria sottostanti; nel mondo Java questo livello è formato tipicamente dai Servlet, nel mondo .NET 2.0 dalle componenti reattive (i metodi richiamati dagli eventi) delle Web Form;
- **Logica funzionale**, insieme delle funzioni di libreria che accedono ai dati e compiono su di essi le operazioni richieste dall'utente attraverso la logica di azione;
- **Server di applicazione**, che ospita i programmi e le librerie costituenti logica di azione e logica funzionale; nel mondo Java questo è un Application Server o un Servlet Engine, nel mondo .NET è l'estensione ASP.NET del Web Server Microsoft IIS;
- **Database server**, che ospita direttamente i dati, o **data provider**, che costituisce comunque l'interfaccia di accesso ai dati presenti entro altre componenti del sistema informatico, come l'insieme dei connettori rappresentati in figura 5.7.

Esempi di server ed applicazioni Web sono presentati nelle figure seguenti.

Un'evoluzione delle applicazioni Web sono i Web Service, descritti al termine del capitolo.

HTTP è un protocollo request/response

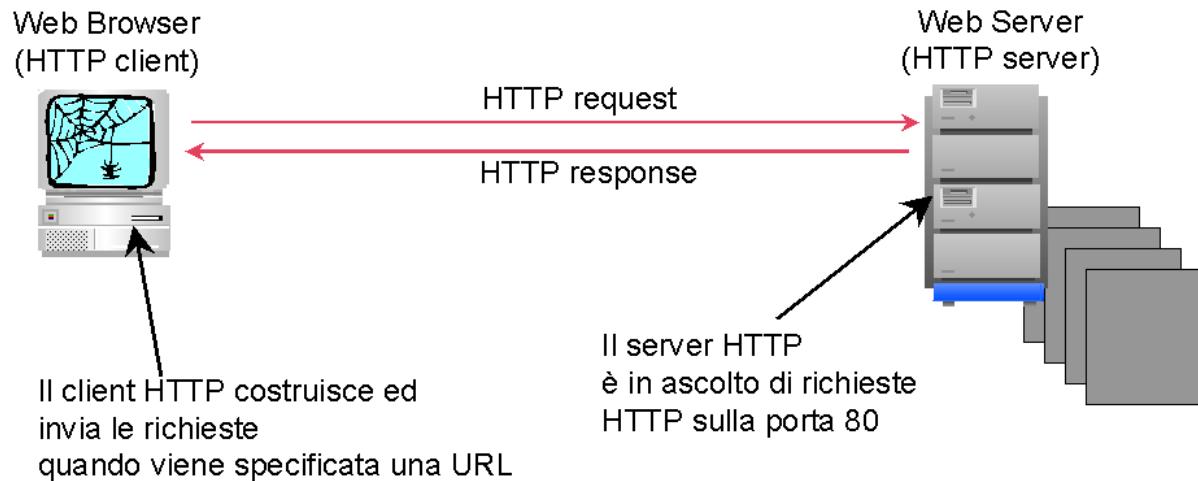


Figura 5.5: Un server Web statico, che distribuisce i documenti presenti nel suo disco già sotto forma di file.

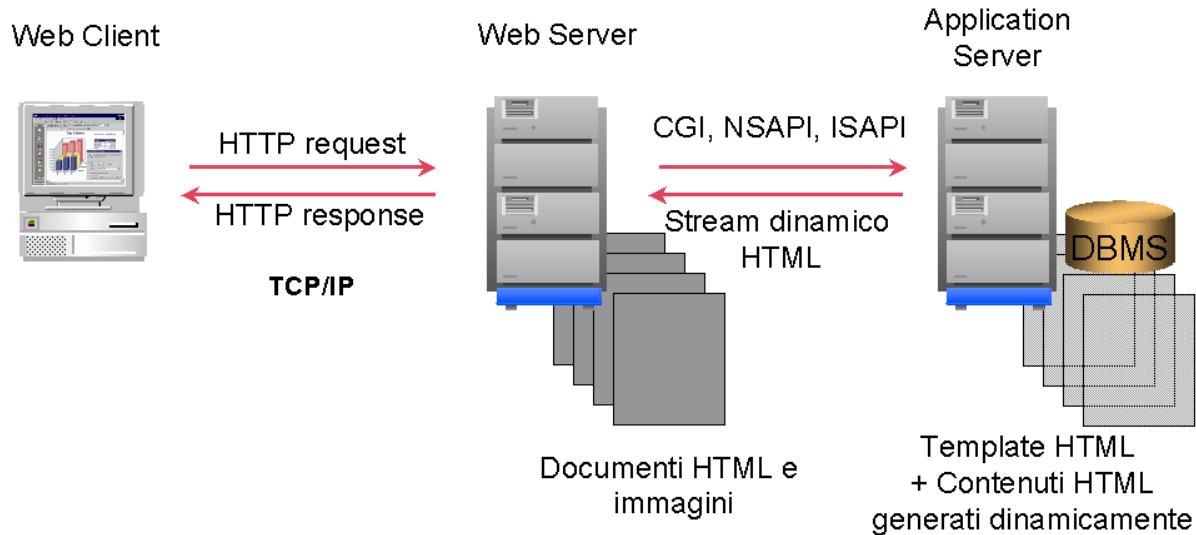


Figura 5.6: Un server Web dinamico, in cui parte delle pagine HTML viene generata dinamicamente a partire dai contenuti presenti nel database e formattata graficamente in base a modelli detti Template HTML.

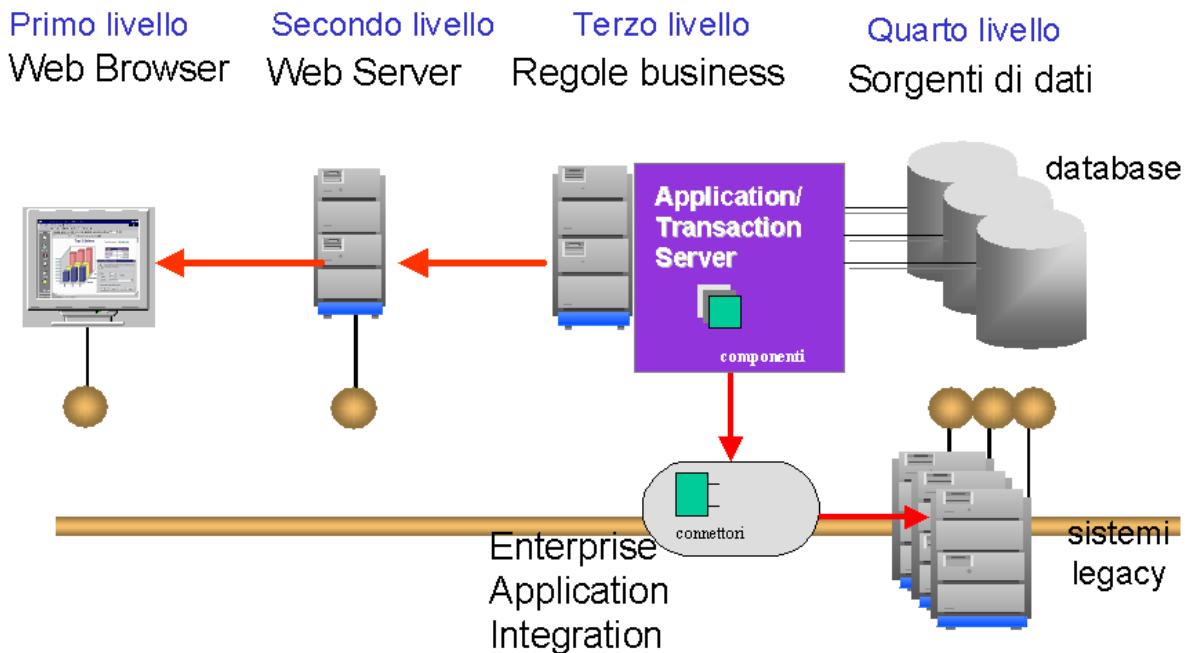


Figura 5.7: Un'applicazione Web che attraverso strumenti di system integration consente l'accesso via Web a diverse sorgenti di dati presenti nel sistema informatico dell'azienda.

Il problema delle compatibilità fra componenti ed interi applicativi

Spesso nuovi applicativi, realizzati per rispondere alle mutate esigenze del business aziendale, devono integrarsi con applicazioni ancora efficienti la cui architettura è però ormai datata.

Abbiamo visto inoltre che i componenti degli applicativi hanno dei grossi vincoli di compatibilità con le architetture hardware e software sottostanti.

In Figura 5.8 è rappresentata una stratificazione di applicativi software multitier, che esprime le dipendenze che i componenti di interfaccia utente, quindi ciò che l'utente vede e con cui l'utente interagisce, hanno in relazione agli strati sottostanti hardware e software.

Nel suo strato di presentazione utente, una determinata applicazione possiede un look-and-feel dipendente dalle librerie grafiche usate e opera entro un window manager (es. Explorer di MS Windows o Gnome/Kde di X-Windows/Unix). I componenti grafici usati nella realizzazione dell'applicativo spesso dipendono dal linguaggio/ambiente di sviluppo. Le librerie relative possono non essere presenti in tutti i sistemi operativi o nelle loro versioni. Se l'applicazione è Web, ossia la sua interfaccia utente sono pagine HTML, usa uno standard di HTML e JavaScript che può dipendere dal browser utilizzato, ovvero i componenti web possono non essere compatibili con una certa versione di browser/Java/sistema ospite.

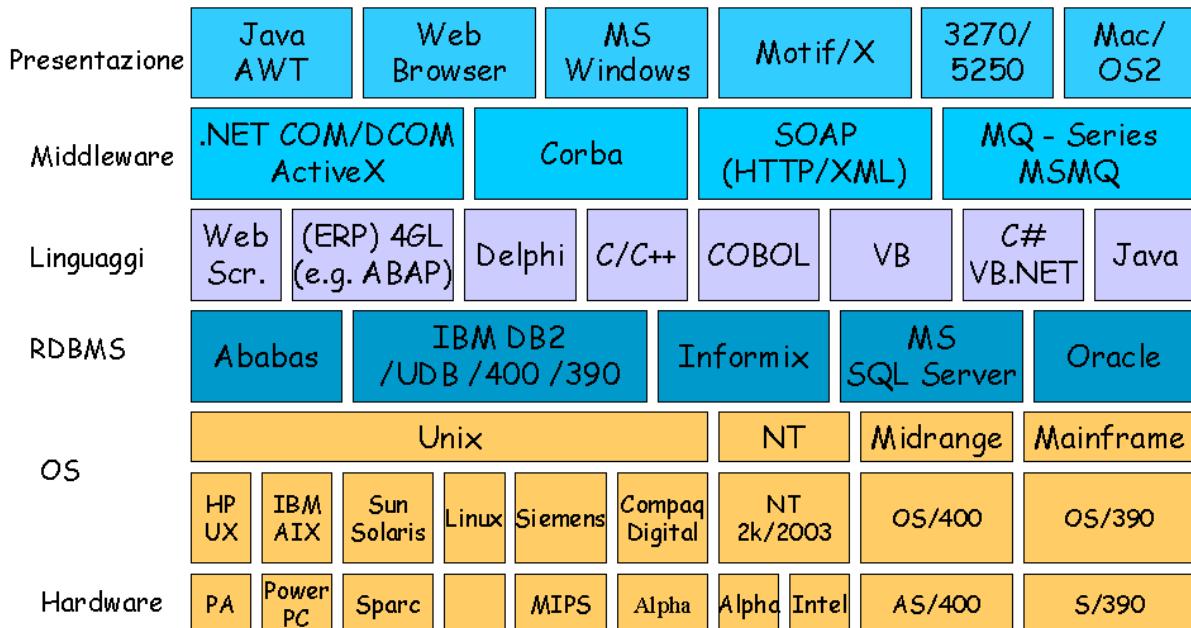


Figura 5.8: Dipendenze degli strati di presentazione rispetto ad hardware e software sottostante, con alcuni esempi per ogni strato.

Il protocollo di comunicazione può richiedere librerie esterne all'applicativo e il middleware richiesto può non essere compatibile col sistema operativo. Oppure le comunicazioni fra i processi client e server che formano l'applicativo nel suo insieme sfruttano meccanismi proprietari. Il contesto di applicazione potrebbe infine avere schermature (es. presenza di firewall).

Per il funzionamento delle componenti server con la logica business potrebbero essere richieste librerie run-time con una versione precisa, e magari le librerie richieste non sono supportate dal sistema operativo o dalla configurazione presente. Ciò può valere anche solo per una parte dell'applicativo (es. componenti Java, singole finestre VB). Oppure vengono sfruttate librerie appartenenti ad altri pacchetti software.

Per quanto riguarda la connessione al database l'applicativo potrebbe sfruttare caratteristiche specifiche di un database (dialetti SQL, tipi di dati...), oppure usare stored procedures (procedure memorizzate direttamente nel database e quindi fortemente dipendenti dal tipo di DBMS utilizzato). L'applicativo richiede le librerie di connessione al database, che potrebbero non essere disponibili in altri ambienti.

In pratica quindi la scelta del software viene compiuta ad un livello funzionale, cioè **in base alle esigenze di business cui l'applicativo software deve rispondere e in base alle caratteristiche della sua interfaccia utente, soprattutto in relazione alla produttività individuale che essa garantisce**. Questo però, a causa dei vincoli cui i componenti di interfaccia utente (e anche i corrispondenti componenti lato server) devono sottostare, può influenzare anche la scelta dei livelli sottostanti di software infrastrutturale (come i DBMS), dei sistemi operativi e dell'hardware (ossia i computer) che li ospitano.

Inoltre, la necessità di fare parlare tra di loro applicativi diversi, o addirittura di integrare sistemi informatici diversi, che spesso avviene in seguito a fusioni di aziende o inserimento massivo di nuovi sistemi, conduce alla creazione di interfacce di

comunicazione ad hoc tra due elementi (la cosiddetta **architettura accidentale**) il moltiplicarsi delle quali, se non tenuto sotto controllo, può portare ad una situazione di complessità elevata e scarsamente controllabile, nota in gergo come **spaghetti-integration** (si veda anche [Chappell 2004]).

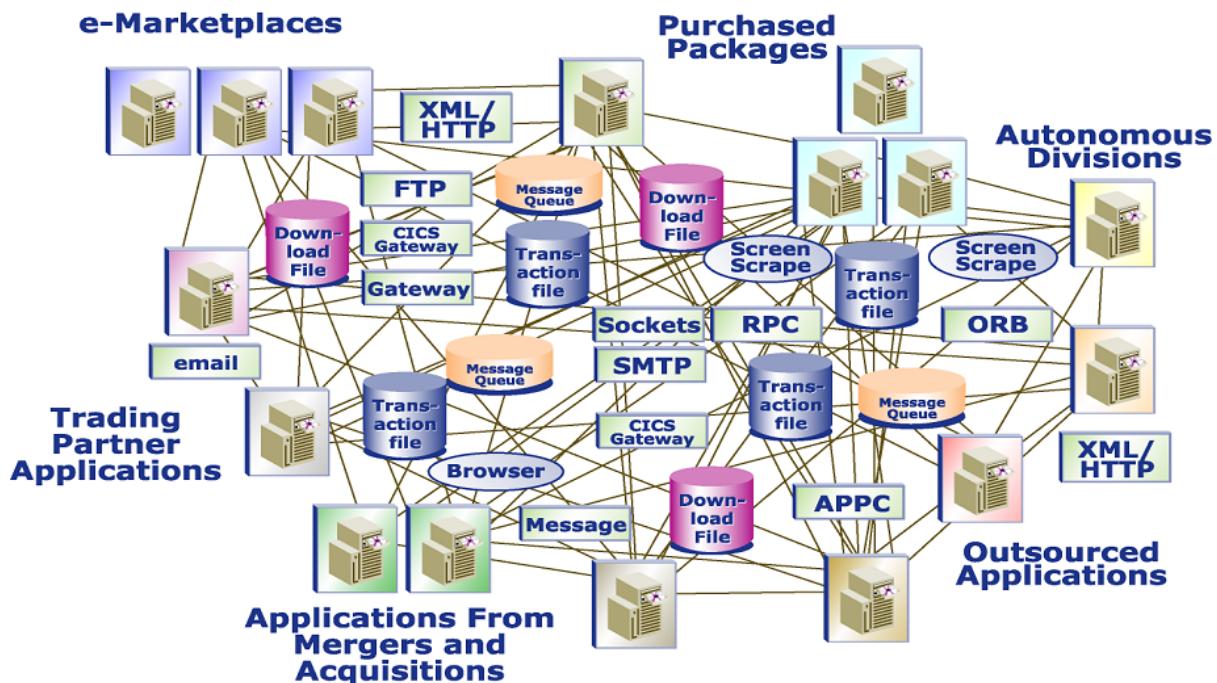


Figura 5.9: Esempio molto complesso di spaghetti-integration conseguente alla fusione di più aziende ed alla integrazione dei diversi sistemi informatici.

Panoramica su sistemi operativi maggiormente diffusi

Nei paragrafi precedenti sono stati presentati i modelli delle applicazioni maggiormente diffusi nei sistemi informativi attuali e le tipologie di servizi che le applicazioni offrono. Ora è necessario esaminare le piattaforme di base, ovvero i sistemi operativi maggiormente diffusi presso le aziende del panorama italiano.

IBM zOS (noto anche coi nomi precedenti di OS/390 e MVS)

E' il sistema operativo dei Mainframe IBM (i server zSeries, secondo la terminologia commerciale attuale di IBM, precedentemente noti come S/390). I mainframe sono computer server di robustezza eccezionale, sia hardware che software, ma sono anche estremamente costosi e tutti i loro componenti sono proprietari e prodotti solo da IBM, che ne cura anche, direttamente o tramite partner, l'assistenza e manutenzione. Per questo motivo il loro uso è limitato a grandi organizzazioni, come centri di calcolo di regioni, ministeri, la FIAT, la Barilla e a tutto il mondo bancario e finanziario. Infatti in Italia esiste solo una banca il cui sistema informatico non comprenda almeno un mainframe. Lo zOS ha una interfaccia utente proprietaria (il cosiddetto TSO), a riga comandi e menu, ma normalmente gli utenti hanno accesso solo agli applicativi cui sono abilitati e non vedono direttamente il sistema operativo. Gli sforzi di integrazione compiuti da IBM hanno portato sia il mainframe ad integrarsi con sistemi client-

server, sia strumenti come il Web e gli application server direttamente sul mainframe. L'accesso diretto al mainframe avviene mediante terminali dumb o, molto più frequentemente, mediante emulatori di terminali da PC, attraverso il protocollo tn3270. Il database relazionale presente di default in un sistema zOS è IBM DB2.

IBM OS/400

E' il sistema operativo dei sistemi IBM AS/400 (noti anche come server iSeries, secondo la terminologia commerciale attuale di IBM). Gli AS/400 sono "i fratelli minori" dei mainframe e l'interfaccia utente al sistema è simile. Anche i sistemi AS/400 sono server molto robusti ma completamente proprietari e pertanto più costosi di architetture analoghe basate su PC server, ma molto meno costosi dei mainframe. I sistemi AS/400 sono molto diffusi presso medie imprese, soprattutto del settore manifatturiero, presso assicurazioni e presso alcuni enti pubblici (grandi comuni e alcune province). Anche nel caso degli AS/400 l'IBM ha introdotto sistemi di integrazione con il mondo client-server come Rumba (file server) e iSeries Access, noto anche come Client Access (file server, amministrazione, accesso in modalità grafica, conversione di dati).

L'accesso diretto ai sistemi AS/400 avviene mediante terminali dumb o, molto più frequentemente, mediante emulatori di terminali da PC, attraverso il protocollo tn5250. Il database relazionale presente di default in un sistema AS/400 è IBM DB2/400.

HP VMS e OpenVMS

Il sistema operativo VMS fu introdotto da Digital (azienda storica di informatica, poi assorbita da Compaq, a sua volta parte di HP dal 2003) negli anni '70. E' un sistema server di ottima robustezza, che opera su hardware proprietario. Ha avuto una grande diffusione sino agli anni'90, ma oggi è in declino in quanto HP, come già in precedenza Compaq, pur curando la manutenzione dell'esistente, non sviluppa ulteriormente il sistema. Pertanto si trovano ancora presso aziende server basati su VMS, ma come eredità del passato. In generale i server VMS vengono sostituiti progressivamente con altre architetture. L'interfaccia utente è a riga comandi, come quella di UNIX, ma è possibile usare l'ambiente grafico X-Windows. Il database relazionale usato è normalmente Oracle, ma si trovano anche Informix e Sybase.

UNIX

Il sistema UNIX nacque all'inizio degli anni'70 e si diffuse enormemente nel settore del calcolo scientifico. In Italia oggi UNIX è usato in molte aziende, soprattutto come server, ma anche come client (quasi esclusivamente come postazione CAD, ossia per il disegno tecnico computerizzato, anche se questo settore subisce ormai una fortissima concorrenza da parte dei PC con sistema operativo Windows). In realtà occorre distinguere UNIX nei vari "dialetti", ossia implementazioni proprietarie di vari costruttori:

- Sun Solaris: opera sia su hardware dedicato, basato sui processori Sun Sparc, sia su piattaforma Intel (anche se quest'ultima versione è poco diffusa); è uno dei più

diffusi, soprattutto nel settore delle telecomunicazioni, dove l'intera infrastruttura di tutti gli operatori di telefonia mobile, dei grandi provider di Internet e di telefonia fissa è quasi completamente basata su server Sun, sia per la parte amministrativa, sia per la parte di gestione tecnica degli apparati; il database normalmente usato è Oracle;

- IBM AIX: opera su hardware dedicato, basato sui processori IBM PowerRisc (gli stessi dei sistemi AS/400); è molto diffuso nel settore delle grandi banche, dove i server AIX operano insieme ai mainframe, in alcuni enti pubblici e presso alcune aziende, spesso ancora come postazione CAD; i database usati sono Oracle e DB2;
- HP-UX: opera su hardware dedicato, basato sui processori HP PA-RISC, ma anche su piattaforme basate su Intel Itanium; è usato come server in vari settori e molto ancora come postazione CAD; il database usato è normalmente Oracle;
- Altri UNIX: esistono altri sistemi UNIX, come ad esempio SGI IRIX, HP Tru-64 (ex Digital Unix), SCO Unix per piattaforma Intel, ma ormai il loro uso è sempre meno diffuso nel panorama italiano.

Linux

Linux, il famoso clone di UNIX, nato negli anni'90 principalmente ad opera del finlandese Linus Torvalds, è un sistema operativo il cui uso è in continua ascesa, sia a livello mondiale, sia nel mercato italiano. Opera su quasi tutte le piattaforme hardware presenti sul mercato ed, essendo disponibile il suo codice sorgente, può essere adattato alle proprie esigenze.

Nel mercato italiano viene usato spesso come file server, print server, server di posta elettronica, Web server, database server o server di applicativi (spesso in sostituzione di vecchie piattaforme basate su SCO UNIX). Soprattutto negli enti pubblici, negli ultimi anni, si è avuta una larga diffusione di sistemi basati su Linux, per la stragrande maggioranza dei casi basati su PC Intel o AMD. Ma anche presso aziende, banche, enti accademici Linux viene usato. Solo in pochissimi casi Linux viene usato anche per le postazioni utente.

Inoltre Linux viene usato spesso nei settori delle appliances (ossia computer dedicati a particolari scopi), come, ad esempio, centralini telefonici, centraline di controllo per sistemi di videosorveglianza, videostation (console da attaccare direttamente alla televisione per usufruire di servizi interattivi via rete), banchi per la regolazione delle centraline delle automobili ecc... Anche nel settore dell'automazione sono state fatte positive esperienze con sistemi basati su Linux.

Microsoft Windows

E' sicuramente il sistema operativo più diffuso a livello mondiale e anche in Italia. A livello aziendale o di enti pubblici, la stragrande maggioranza delle postazioni client sono basate su una versione di questo sistema operativo. Nei dettagli occorre distinguere tra varie categorie.

Famiglia Windows 3.x/DOS: esistono ancora sistemi basati su Windows 3.x o addirittura su DOS in produzione, soprattutto nei PC industriali che governano il

funzionamento di apparati di automazione; nei sistemi informativi invece ormai essi sono pressoché scomparsi.

Famiglia Windows Home (95, 98, ME, XP Home): non è robustissima e ormai sta venendo progressivamente abbandonata dalle aziende. XP home viene usato specialmente nei portatili, mentre Windows98 sta uscendo dall'uso, anche perché non garantisce le protezioni minime per l'accesso che il D.L. 196/2003 impone per la sicurezza dei dati (si veda il capitolo 8 per approfondimenti).

Famiglia Windows Enterprise (NT, 2000, XP e i futuri Vista Business e Enterprise): sufficientemente robusta una volta configurata, viene usata sui client della maggior parte delle aziende ed enti pubblici. In particolare NT, da tempo non più supportato da Microsoft, è quasi scomparso, mentre Windows2000 e WindowsXP si dividono il mercato, con una prevalenza a favore di quest'ultimo.

Famiglia Windows Server (NT, 2000, 2003): è la piattaforma server più usata nelle piccole imprese, ma molto diffusa, anche come server dipartimentale, ad ogni livello. NT è ormai scomparso, la migrazione da 2000 a 2003 sta avvenendo progressivamente, ma lentamente, anche se il supporto di Microsoft a 2000 è cessato nel corso del 2006. Il database più usato su questo tipo di server è Microsoft SQL Server, ma anche Oracle è spesso presente.

Apple MacOS

Il sistema operativo MacOS viene usato sui computer Apple Macintosh. Sono esclusivamente postazioni utente e sono la piattaforma di riferimento nel mondo della grafica pubblicitaria e, in parte, anche dell'editoria. Vengono usati come postazioni utente in alcune aziende, specialmente nel mondo della moda. Il sistema oggi in uso è MacOS X, di struttura derivata dallo UNIX.

Panoramica sulle tecnologie correnti per lo sviluppo di applicazioni

Nelle software house, ma molto spesso anche entro i sistemi informativi di un'azienda non informatica, esiste la necessità di sviluppare nuove applicazioni software o di effettuare manutenzioni evolutive o nuove versioni di applicazioni esistenti.

Quali sono le tecnologie maggiormente usate? Quali soprattutto quelle su cui investire in autoformazione o su cui un dirigente può decidere di fondare un progetto?

La situazione ovviamente si evolve molto rapidamente nel tempo ed è fortemente dipendente dai contesti. Nonostante ciò, è possibile tracciare una panoramica generale delle tecnologie di sviluppo maggiormente usate nel settore informatico in Italia.

Nel corso dei decenni successivi all'introduzione dei calcolatori, centinaia di linguaggi ed ambienti di sviluppo sono stati rilasciati sul mercato, ma di questi la maggior parte è andata in disuso o è confinata a nicchie veramente ristrette. Nel seguito vedremo quindi alcune tecnologie di sviluppo, la loro diffusione in termini assoluti e i settori in cui esse sono principalmente presenti.

Nell'ambito di organizzazioni medio-grandi, principalmente di tipo bancario, è molto presente una tendenza conservatrice, non necessariamente dovuta a problemi di legacy, ma semplicemente di tipo economico: se un modulo software svolge il suo dovere da

decenni ed è ormai esente da errori, in quanto nel corso dell'uso sono stati individuati tutti gli errori esistenti, non è economicamente conveniente sostituirlo.

La famiglia del COBOL

Il COBOL, acronimo di COnmon Business-Oriented Language, è, storicamente, uno dei primi linguaggi di alto livello sviluppati. Il primo standard Cobol risale infatti all'inizio degli anni sessanta. Nel corso del tempo ovviamente il linguaggio si è molto evoluto, acquisendo nuove caratteristiche e potenzialità.

Oggi esistono tre tipi di Cobol molto usati, non compatibili tra loro:

- IBM Cobol, tipicamente usato su piattaforme Mainframe e AS/400, è molto diffuso nel settore bancario, ove la maggior parte degli applicativi sono realizzati con esso;
- AcuCobol, diffuso su piattaforma Windows, Linux e Unix, è stato usato per realizzare molti ERP Lite come TeamSystem o SiFides;
- MicroFocus Cobol, è un concorrente di AcuCobol, copre essenzialmente le stesse piattaforme e viene usato per lo stesso scopo.

Soprattutto in ambito bancario la manutenzione ordinaria ed evolutiva delle procedure Cobol, oltre che la realizzazione di nuovi moduli copre grandi mercati. A livello globale italiano è stimabile che la famiglia del Cobol sia usata in una percentuale variabile tra 10 ed il 20% del mercato complessivo dello sviluppo e manutenzione software.

La famiglia del Delphi

Il linguaggio a oggetti Borland Delphi, derivato dal Pascal, ha incontrato un grande successo per le sue caratteristiche di robustezza accompagnata alla facilità d'uso. Usato essenzialmente su piattaforma Windows, in quanto Kylix, la versione per Linux, non è mai stata sostenuta adeguatamente da Borland, copre circa il 10% del mercato. Il suo uso è però incerto nel futuro in quanto Borland stessa ha appena lanciato sul mercato anche un ambiente di programmazione per il linguaggio C# e la piattaforma .NET.

Il linguaggio Visual Basic

E' stato il prodotto di sviluppo di gran lunga più riuscito di Microsoft. La sua rapidità di sviluppo lo ha portato ad occupare una larga frazione del mercato. Viene molto usato tutt'ora entro i sistemi informativi di aziende medio-grandi per sviluppo interno di strumenti di reportistica e aggregazione dati. Anche in ambito automazione viene molto usato. L'avvento del mondo .NET sta però riducendo progressivamente il suo uso. Rimane molto diffuso il VBA (Visual Basic for Application), come linguaggio per la personalizzazione di applicazioni office e la realizzazione dei micro applicativi, anche da parte di personale non tecnico.

Il mondo Microsoft .NET

E' l'erede del Visual Basic cui ha aggiunto caratteristiche di eleganza e robustezza mutuate da altri ambienti. E' in rapida espansione, sia nell'uso del linguaggio VB.NET, sia in quello del linguaggio C#. Quest'ultimo è usabile anche in ambiente

Linux e Unix, grazie al framework MONO di Novell. L'uso di .NET copre già una buona fetta del mercato globale italiano, tuttavia ci vorranno anni prima che la massa degli sviluppatori Visual Basic (e anche dei pochi VisualC/C++) completi la transizione al mondo .NET.

Il mondo Java

Linguaggio effettivamente multipiattaforma, Java ha trovato il suo mercato ideale nei due “estremi” delle applicazioni software. Da un lato, la maggior parte delle grandi applicazioni Web e dei portali, soprattutto nell’ambito delle banche, delle telecomunicazioni e degli enti pubblici, è realizzata in Java. Dall’altro Java ha trovato un ottimo mercato nei telefoni cellulari e, in parte, anche nei palmari. Java copre una buona fetta del mercato italiano ed è tutt’ora in espansione, anche se molto meno di .NET.

Il mondo C/C++

I linguaggi C e C++, da cui Java e C# derivano, hanno visto via via ridursi la loro diffusione sul mercato. Oggi C e C++, sia su piattaforma Windows, sia su Unix/Linux o su sistemi come i palmari sono usati solo per applicazioni particolari come i device driver o per sistemi ad elevata efficienza come nell’ambito delle telecomunicazioni.

Il mondo dei 4GL

I linguaggi di quarta generazione (4GL) sono linguaggi di alto livello, tipicamente usati entro gli ERP per produrne personalizzazioni ed estensioni. Ogni ERP ha il suo e il più diffuso di tutti è ABAP (Advanced Business Application Programming) di SAP. Almeno il 10% del mercato italiano è occupato da questo tipo di sviluppo, ed è un mercato in cui le tariffe e i compensi per i professionisti che vi lavorano sono abbastanza elevati.

Il mondo assembler

L’assembler o linguaggio assemblatore è il linguaggio in cui ogni istruzione corrisponde ad una singola istruzione del linguaggio macchina del processore destinazione del linguaggio stesso. Per quanto riguarda la piattaforma PC, oggi solo per applicazioni molto particolari viene usato l’assembler, mentre un discorso diverso vale nel mondo dei PLC usati per l’automazione industriale, ove ogni PLC ha il suo assembler con cui viene programmato. In verità parlare di assembler è limitativo, in quanto la maggior parte di questi linguaggi dispone anche di strutture di controllo tipiche di linguaggi a più alto livello. Praticamente tutta l’automazione in Italia viene realizzata usando gli assembler dei PLC.

Il mondo Web

I linguaggi di scripting per il Web, sia lato client (JavaScript, ActionScript di Flash per esempio), sia lato server (ASP, PHP, JSP) hanno raggiunto una buona diffusione negli ultimi anni. Le applicazioni Web sono molto diffuse e spesso vengono realizzate estensioni Web di gestionali o ERP lite; nel mercato di portali e siti Web, essendo

molti i singoli o le società operanti rispetto alla richiesta, si è avuto negli ultimi anni un abbassamento delle tariffe.

Administration Scripting

I linguaggi di scripting per i sistemi operativi (UNIX Shell script, JCL del mainframe, REXX di AS/400 e altre piattaforme IBM, WSH e MSH di Windows) sono uno dei tool che ogni buon amministratore di sistema dovrebbe conoscere. La diffusione dei linguaggi va di pari passo con quella delle piattaforme su cui operano.

Il linguaggio RPG

Specifico della piattaforma AS/400, il linguaggio RPG è stato molto usato in passato per realizzare applicativi gestionali operanti su tale piattaforma. Oggi viene usato poco per lo sviluppo di nuovi applicativi, ma la manutenzione della grande mole di programmi esistenti copre una fascia interessante del mercato.

Altri linguaggi

Nel mercato italiano gli altri linguaggi sono ridotti praticamente ad applicazioni di nicchia, o alla manutenzione di software molto vecchio.

Le nuove soluzioni di integrazione: Service Oriented Architecture

Per quanto visto una buona integrazione delle applicazioni aziendali diventa una necessità per l'impresa. Si parla quindi di Enterprise Application Integration (EAI), ossia integrazione di applicazioni su scala di tutta l'azienda. L'EAI può avvenire a quattro livelli [Chappell 2004]:

1. **Integrazione orientata ai dati:** Avviene a livello di database o archivi dati, può essere in real-time o no e di solito è composta di trasferimenti batch, unioni di dati, repliche di dati o soluzioni complete ETL (Extract, Transform, Load);
2. **Integrazione orientata a funzioni e metodi:** è di solito integrazione di applicazioni (A2A), e può essere diretta, con paradigma request/response o basata su strumenti di middleware o su codice custom, sviluppato ad hoc (tenendo presente il pericolo dell'esplosione delle architetture accidentali, visto sopra);
3. **Integrazione di interfacce utente:** è la standardizzazione delle interfacce utente entro un unico modello, di solito basata sul browser (interfacce web), si parla infatti di enterprise business portal o enterprise application portal;
4. **Integrazione dei processi business:** agisce direttamente al livello dei processi business ed è la più efficiente da un punto di vista funzionale, ma non è facile da applicare quando vi sono prodotti software con logica business rigida; per essere flessibile, conduce implicitamente alla SOA.

La Service Oriented Architecture o SOA è una topologia di applicazioni software formata da servizi (ossia applicazioni che offrono servizi) e clienti dei servizi (service consumer, ovvero applicazioni che richiedono dei servizi), in relazione 1-a-1 tra di loro, ma “debolmente accoppiati” (loosely coupled), in modo tale che la variazione di

una interfaccia non forzi al cambiamento dell'applicativo che implementa i servizi stando dietro a tale interfaccia. In figura 5.10 viene rappresentata una coppia service consumer/service provider in una implementazione di SOA basata sugli standard dei Web Services o servizi Web.



Figura 5.10: Implementazione di SOA basata sui Web Services.

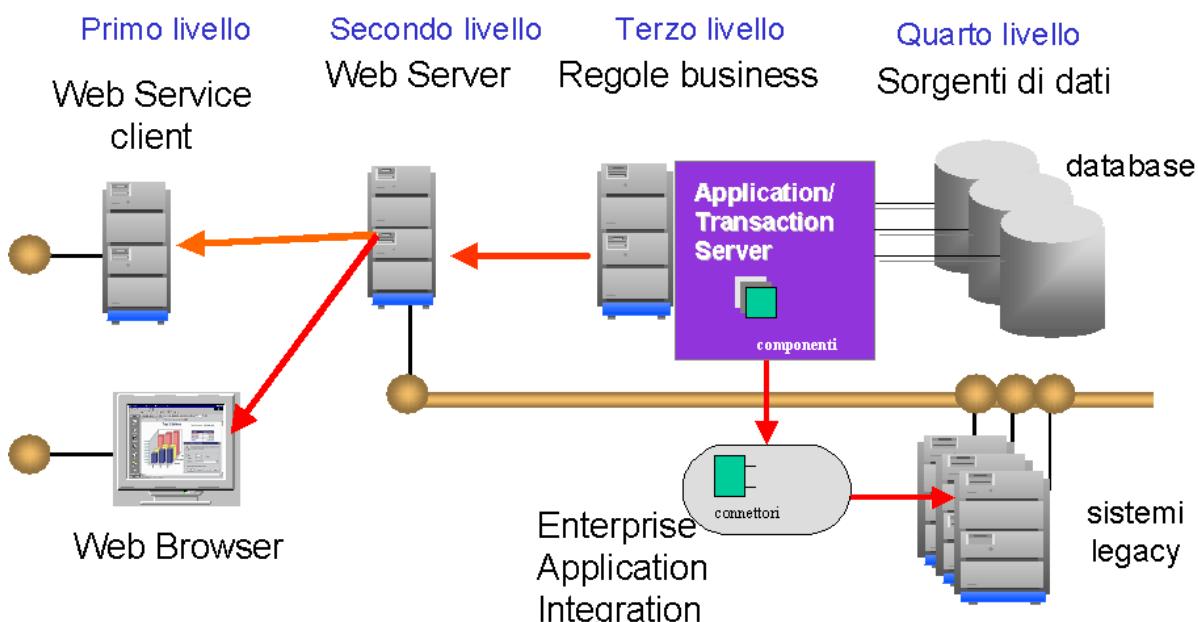


Figura 5.11: Uso combinato di applicazioni Web e Web Service entro un'azienda. Il sistema è l'evoluzione di quello presentato in figura 5.7.

I componenti fondamentali di una SOA sono i seguenti:

- **Service Provider** (fornitore di un servizio) è il componente responsabile di creare il servizio, pubblicare l'interfaccia del servizio (ossia l'insieme di strutture di comunicazione con cui si interagisce con il servizio dall'esterno) e provvedere l'implementazione effettiva che realizza il servizio, rispondere alle richieste in arrivo (realizzare effettivamente il servizio);
- **Service Requestor o Consumer** (cliente richiedente un servizio), è il componente utente del servizio, che deve trovare il servizio, per conoscenza diretta (ossia sapendo su che macchina entro una rete si trova effettivamente il servizio) o interrogando un repository (ossia un archivio che funge da "rubrica telefonica" di servizi, traducendo il loro nome nell'indirizzo di rete della loro interfaccia di

comunicazione); deve inviare i dati previsti dall’interfaccia del servizio e ottenere indietro i risultati;

- **Service Broker** (intermediario), è il componente intermediario che registra e categorizza i servizi per cui sono possibili interrogazioni con varie chiavi; crea e gestisce un repository dei servizi.

Questi elementi ed i compiti da essi svolti sono rappresentati in figura 5.12, dove sono rappresentati la pubblicazione (publishing) dei servizi, il reperimento (finding) e l’interfacciamento (binding).

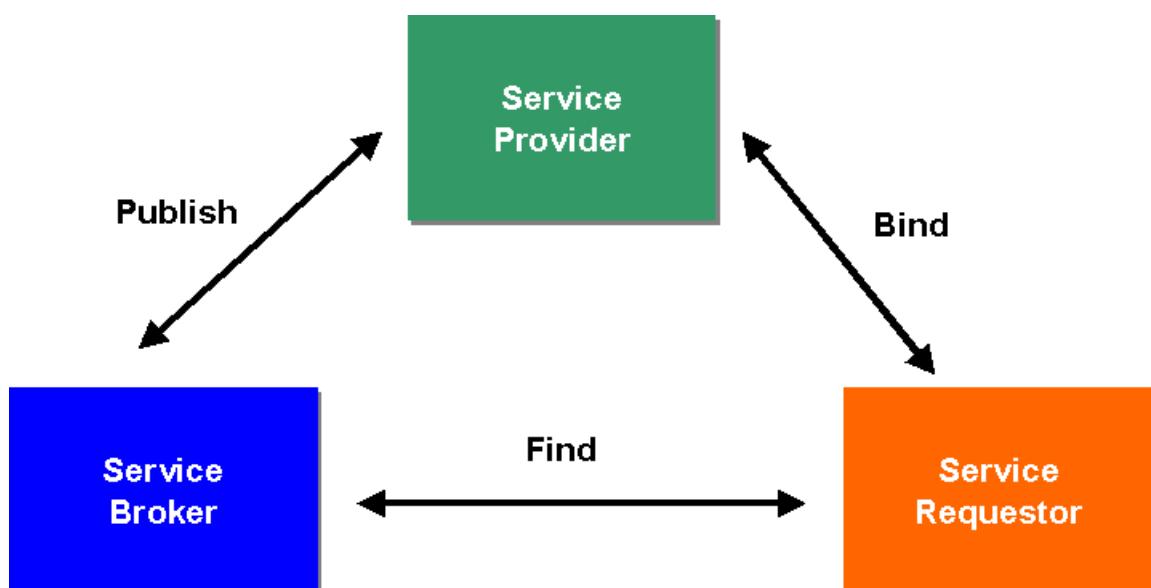


Figura 5.12: Le azioni compiute dagli elementi che formano la SOA.

Le implementazioni oggi esistenti della SOA sono realizzate tramite i Web Service, ove attraverso messaggi XML standardizzati trasmessi su protocolli internet come HTTP è possibile e operando con protocolli su di essi basati è possibile:

- Descrivere un servizio con WDSL
- Pubblicarlo con UDDI
- Trovarlo con UDDI
- Interfacciarsi con SOAP
- Invocarlo con SOAP
- Coordinare i flussi con strumenti come WSFL

Componente fondamentale delle ultime versioni di SOA è l’Enterprise Service Bus (ESB).

L’ESB aggiunge una coreografia, ovvero un ordinamento ed un coordinamento dei vari flussi di informazioni, prevedendo anche, ove necessario, la traduzione dei dati dai formati interni ad ogni componente verso un formato comune. Quindi grazie all’ESB, considerando gli elementi costitutivi, ossia i singoli Web Service, come centri di servizio funzionale, in grado di erogare ognuno uno specifico compito di elaborazione, un singolo processo business con le attività che lo formano ed i flussi

informativi che lo accompagnano viene “mappato” direttamente sull’ESB, disegnando una successione di trasferimenti di informazione, ossia di flussi informatici, fra i vari Web Service. Ciò significa che, se la granularità iniziale delle componenti funzionali in grado di compiere le attività è stata progettata bene, individuando componenti né troppo piccole né troppo grandi e di uso generale, una ridefinizione del processo non richiede interventi diretti sugli applicativi che stanno dietro i singoli Web Service, ma soltanto un ridisegno dei flussi informatici che li connettono. Chiaramente, per giungere a questo ambizioso obiettivo è richiesto non solo di disporre di una robusta infrastruttura tecnologica, ma anche di strumenti di analisi business molto precisi, che consentano di identificare con precisione le funzionalità necessarie per compiere le attività componenti i processi business, per implementarle come gruppi di Web Service. Le metodologie più moderne, come, ad esempio, gli ebXML [ebXML 2005] tendono ad usare l’analisi con UML for Business descritta nel capitolo 4. Un ulteriore evoluzione la visione del processo come un workflow, introducendo quindi l’ordine di esecuzione e di passaggio delle informazioni tra i vari Web Service, ciascuno dei quali diviene una unità elaborativa. In strutture di questo tipo l’**orchestration** prevede la gestione automatica di tutte le fasi di coordinazione attraverso l’inserimento di strutture di controllo, che combinino tra loro i Web Service seguendo le direttive di appositi programmi, come avviene nel modello su cui è basato il linguaggio BPEL (Business Process Execution Language), supportato da vari produttori (si vedano [BPEL 2006] e [BPM 2005] per approfondimenti).

Ormai molti dei grandi produttori di software come Microsoft, Oracle e IBM stanno trasformando i loro strumenti infrastrutturali e di sviluppo per rendere sempre più facile realizzare architetture basate sul modello SOA, che sembra quindi affermarsi come il modello di riferimento per il software di grandi aziende ed organizzazioni (ovvero il cosiddetto software di classe Enterprise) per i prossimi anni. Per approfondimenti si consigliano [Chappell 2004], [SOA 2005].

Il Grid Computing

L’approccio alla base del Grid Computing è quello di offrire servizi di calcolo in modo “impersonale” e mascherando i dettagli di tali servizi in analogia con quanto accade nella rete elettrica (in inglese “Power Grid” da cui il termine Grid Computing), in cui basta inserire una spina nella presa, senza conoscere l’origine dell’elettricità così ottenuta.

In realtà il Grid Computing, spesso basato nella sua infrastruttura sugli standard dei Web service visti nel paragrafo precedente, offre tre categorie di servizi.

1. **Storage Grid**, che offre una grande quantità di spazio di memorizzazione permanente (essenzialmente quindi spazio disco), senza che gli utenti o i sistemi di calcolo client conoscano i dettagli dell’infrastruttura che offre tale spazio. Questo comparto tecnologico ha avuto un enorme sviluppo nel corso degli ultimi anni con la nascita dei Network Attached Storage (Device) o NAS, ossia praticamente dei computer dedicati ad offrire solo il servizio di file server e, soprattutto delle Storage Area Network o SAN, ossia computer dedicati a gestire enormi file service, collegati con il resto della rete attraverso supporti di

comunicazione dedicati ad altissima velocità (tipicamente fibre ottiche con protocolli di comunicazione specifici). Ulteriori approfondimenti su questi tipi di dispositivi saranno presentati nel capitolo 8.

2. **CPU Grid**, che offre la capacità di usufruire per le elaborazioni della CPU di computer connessi alla rete, non usata per altri applicativi, garantendo quindi un uso ottimale delle risorse di calcolo presenti e diminuendo i tempi necessari per elaborazioni complesse, quali aggregazioni di dati, simulazioni ecc... Questo comparto è un'evoluzione del calcolo distribuito nato negli anni '90 nel mondo accademico. Sono presenti molti software che garantiscono questo tipo di servizi, sia commerciali sia freeware. Una evoluzione della CPU Grid è il virtual computing, che sarà trattato in dettaglio nel capitolo 8.
3. **Application Grid**, che offre la capacità di usare applicativi complessi quali sistemi CAD o di simulazione, accedendo ad essi in modo efficace attraverso la rete, senza essere direttamente presenti sulla macchina dove sono installati. Attraverso questo servizio, agendo con la struttura della SOA e dell'ESB visti nel paragrafo precedente, si possono creare sistemi software complessi come simulatori distribuiti di prototipi di auto, in cui la carrozzeria è simulata sui computer del produttore del telaio, il motore sui computer del produttore del motore e gli pneumatici su quelli del produttore di pneumatici, senza che i dettagli interni siano noti.

Per approfondimenti sul Grid Computing si consiglia [GRID 2005] come base di partenza.

Domande

1. Quali sono le componenti principali di un'applicazione?
2. Qual'è il ruolo delle reti entro i sistemi informatici odierni?
3. Cos'è il modello client-server? Che importanza ha entro i sistemi informatici moderni?
4. In base a quali criteri si procede alla scelta di un'applicazione entro un sistema informatico?
5. Cosa si intende per interfaccia utente di un programma applicativo? Quali problematiche sono legate all'interfaccia utente? Come essa interviene nel lavoro quotidiano?
6. Cosa sono le matrici di compatibilità del software? Come condizionano le scelte del software stesso?
7. Quali sono i vincoli tecnologici a cui l'inserimento di un nuovo componente informatico è soggetto?
8. Cos'è un sistema informatico distribuito? In che situazioni esso si trova? Che problematiche di gestione vi sono associate?
9. Cosa si intende per EAI? Che problematiche comporta?
10. Cosa è la SOA? Che vantaggi produce nei sistemi informativi?

Bibliografia

[ACPT 2002] P. Atzeni, S. Ceri, S. Paraboschi, R. Torlone - *Basi di Dati: Modelli e Linguaggi di Interrogazione* - McGraw-Hill Italia, Milano, 2002

[ACFPT 2003] P. Atzeni, S. Ceri, P. Fraternali, S. Paraboschi, R. Torlone - *Basi di Dati: Architetture e Linee di Evoluzione* - McGraw-Hill Italia, Milano, 2003

[AJAX 2006], pagina Web su AJAX di Wikipedia Italia, con molti collegamenti ad altri siti
<http://it.wikipedia.org/wiki/AJAX>

[BFM 2001] G. Bracchi, C. Francalanci, G. Motta - *Sistemi Informativi e aziende in rete* - McGraw-Hill Italia, Milano, 2001

[BPEL 2006] Sito web che raccoglie link utili su BPEL, <http://www.bpelsource.com/>

[BPM 2005] Business Process Management Initiative, su Web <http://www.bpmi.org/>

[Chappell 2004] D.A. Chappell – *Enterprise Service Bus* – O'Reilly, 2004

[De Marco 2000] M. De Marco - *Sistemi Informativi Aziendali* - Franco Angeli Edizioni, Milano 2000

[ebXML 2005] sito Web dello standard ebXML del consorzio OASIS, <http://www.ebxml.com>

[GMN 1995] S. Gai, PL Montessoro, P. Nicoletti - *Reti Locali: Dal Cablaggio All'Internetworking* - Scuola Sup. G. Reiss Romoli, 1995

[GRID 2005] sito web principale sul Grid Computing, <http://www.gridcomputing.com/>

[LL 2004] K Laudon, J Laudon – *Management dei sistemi informativi* – Ed. Pearson Education Italia, Milano, 2004

[Malhotra 1998] Y. Malhotra - *Business Process Redesign: An Overview* - IEEE Engineering Management Review, vol. 26, no. 3, Fall 1998, su Web <http://www.kmbook.com/bpr.htm>

[OASIS 2005] sito Web del consorzio OASIS, <http://www.oasis-open.org/home/index.php>

[OHE 1999] R. Orfali, D. Harkey, J. Edwards - *Client/Server Survival Guide, 3rd Edition* – Ed. Wiley, 1999

[OpenDoc 2005] Il formato OpenDocument del consorzio OASIS, su Web
http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=office

[OpenXML 2005] Il formato OpenXML di Microsoft, su Web
<http://www.microsoft.com/office/preview/developers/fileoverview.mspx>

[OTP 2002] D. O'Mahony, H. Tewari e M. Peirce - *Electronic Payment Systems for E-Commerce (2nd Edition)* - Ed. Artech House, 2002

[P2P 2006] Documento sui protocolli Peer-to-Peer, su Web <http://en.wikipedia.org/wiki/Peer-to-peer>

[SOA 2005] Sito Web che raccoglie articoli dedicati alla Service-Oriented architecture
<http://www.service-architecture.com/>

[Skype 2006] Sito della società Skype, su Web <http://www.skype.com>

[Tanenbaum 2003] A. S. Tanenbaum - *Reti di calcolatori, 4a edizione* – Ed. Pearson Education Italia, 2003

[VoIP INFO 2006] Sito di documentazione sul protocollo VoIP, su Web <http://www.voip-info.org>

[XML.com 2005] sito Web con catalogo dei tool per l'uso di XML, [http://www.xml.com/](http://www.xml.com)

[XML.org 2005] sito Web degli standard XML, [http://www.xml.org/](http://www.xml.org)

Giulio Destri

Soluzioni informatiche per l'impresa

I sistemi integrati di gestione: gli ERP

Con l'acronimo ERP (Enterprise Resource Planning) si intendono identificare i sistemi integrati di gestione, cioè insiemi di applicazioni software integrate, che gestiscono tutte le informazioni rilevanti dell'azienda in un'unica base dati centralizzata e che consentono di gestire in modo coordinato una molteplicità di attività dell'azienda, od al limite tutte le attività aziendali.

L'acronimo ERP è stato coniato agli inizi degli anni '90 per indicare i nuovi sistemi gestionali integrati per aziende manifatturiere, sistemi che cominciavano ad essere proposti come evoluzione dei precedenti prodotti costruiti sui modelli, noti come MRP (Material Requirement Planning) e MRP II (Manufacturing Resource Planning).

In pratica quindi un ERP è un sistema IT integrato per la gestione, che copre tutti i processi più importanti in un'azienda, come, ad esempio:

- Ciclo attivo e passivo (vendite e acquisti)
- Contabilità
- Logistica
- Human resource management
- Produzione

Il successo sul mercato dei sistemi ERP è dovuto ad un incontro tra nuove caratteristiche offerte e nuove richieste delle aziende clienti, che possono essere riassunte come segue.

Dal lato dell'offerta:

- la standardizzazione dei sistemi operativi (soprattutto UNIX e Microsoft Windows);
- la scalabilità delle nuove famiglie di server;
- il potenziamento delle reti;
- il successo dell'architettura client-server.

Dal lato della domanda:

- maggiore attenzione delle aziende per le proprie attività portanti;
- il cambiamento continuo dei mercati;
- la globalizzazione dei mercati;
- la pervasività dei sistemi informativi;
- il problema dell'anno 2000.

Poiché l'acronimo ERP è diventato di moda e non è un marchio attribuibile ad un singolo prodotto, oggi praticamente tutti i produttori di software gestionale definiscono di tipo ERP la loro offerta, ma i prodotti ERP propriamente detti dovrebbero offrire determinate caratteristiche:

- architettura flessibile e scalabile;

- efficace ambiente di personalizzazione;
- indipendenza da una base dati fisica;
- strumenti di modellazione dei processi;
- significativo volume di referenze;
- disponibilità in lingue e localizzazioni diverse;
- possibilità di modellare strutture aziendali complesse;
- vasta scelta di processi già coperti dalle funzionalità del software, attivabili attraverso una semplice parametrizzazione del sistema;
- ampia gamma di modelli di controllo di gestione di facile costruzione e totalmente integrati.

Tutti i grandi produttori di ERP hanno dotato i loro applicativi di un ambiente proprietario di sviluppo di nuovi moduli/funzionalità, che viene di solito messo a disposizione di partner e clienti, come, per esempio, avviene per l'ABAP (Advanced Business Application Programming) di SAP (si veda [ABAP 2006]).

La versatilità dell'ambiente di sviluppo è strategica sotto molti punti di vista, un ambiente di sviluppo avanzato comprende adeguati servizi di middleware e deve rappresentare uno snodo completo, tra il livello funzionale di un ERP e la sua infrastruttura tecnologica, che il cliente non dovrebbe vedere direttamente.

Si sta delineando la tendenza ad inserire un modellatore di processi integrato, che può opzionalmente sostituire il menù tradizionale, e tale passo facilita la creazione di una vera e propria "biblioteca" di processi aziendali per vari settori di mercato e diverse aree funzionali, che aiuta ed accelera l'installazione di nuovi sistemi ERP.

Partendo dal mondo manifatturiero, i produttori di sistemi ERP, forti di mezzi finanziari significativi, di componenti applicative già pronte, di esperienza di produzione industriale di software applicativo, si muovono in nuovi mercati mirando alle loro specifiche attività portanti. Tutti i grandi produttori di sistemi ERP hanno fatto annunci aggressivi di penetrazione del mercato delle medie aziende, potenzialmente enorme, anche se difficile e molto polverizzato.

Il concetto di verticalizzazione va analizzato a fianco e a complemento di quello di ricerca di nuovi mercati. Per **verticalizzazione** si intende lo sforzo di caratterizzare un sistema ERP per venire in conto alle specifiche esigenze di particolari segmenti del tradizionale mercato manifatturiero (es. aziende farmaceutiche, alimentari...).

Le nuove aree di maggior attenzione sono:

- commercio elettronico (B2C);
- supply chain (B2B), che verrà definito meglio in seguito;
- customer connection;
- business intelligence;
- sviluppo nuovo prodotto;
- erogazione servizi;
- manutenzione e assistenza;
- risorse umane.

Sotto il nome di ERP oggi andrebbero inclusi tutti i sistemi informatici che permettono di gestire in modo integrato tutte le informazioni scaturite dai processi aziendali e che includono anche i sistemi di gestione, quindi quei sistemi che permettono di cogliere i flussi di informazioni che possono essere utilizzati da tutte le funzioni aziendali.

Per un'azienda che vuole essere competitiva sul mercato globale è sempre più necessario darsi un'organizzazione interfunzionale, nella quale la gestione per processi stimoli un coinvolgimento nel business di tutte le funzioni aziendali attraversate trasversalmente dai processi.

Perché ciò possa avvenire con successo, è indispensabile che nell'azienda i dati e le informazioni siano condivisibili da tutti coloro che ne possano aver bisogno: cioè, è indispensabile una completa integrazione gestionale.

Le case produttrici di ERP hanno, nella progettazione e produzione dei loro prodotti, un'ottica rivolta all'azienda nel suo complesso ed ai suoi processi, non alle singole funzioni, come succedeva nella stragrande maggioranza dei software creati ad hoc da imprese minori.

Partendo da questi presupposti è chiaro perché si rileva che le grandi aziende vanno decisamente verso i sistemi ERP industrializzati standard, nonostante i grandi costi che la transizione comporta, soprattutto in termini di adattamenti e reingegnerizzazione dei processi aziendali interni.

Non è così, invece, per le piccole aziende, che preferiscono acquistare sistemi non ERP, o farsi costruire sistemi su misura da software houses minori, a costi decisamente più leggeri rispetto al "grande" ERP.

Va detto però che anche molti software gestionali tradizionali si stanno evolvendo rapidamente, dando origine ad un ERP in scala ridotta, il cosiddetto ERP lite, organizzato in modo modulare simile ad un ERP grande, ma di solito meno programmabile e disegnato specificatamente per i processi business più semplici, tipici di una media impresa.

Il mercato degli ERP internazionali è dominato da SAP R/3 e dal suo successore MySAP ERP della tedesca SAP AG, con circa il 55% del mercato mondiale, cui seguono i tre prodotti E-Business Suite, PeopleSoft Enterprise e JD Edwards EnterpriseOne della statunitense Oracle, leader anche del mercato dei DMBS (gli ultimi due sono frutto di assorbimenti societari), probabilmente destinati a fondersi nell'unico prodotto Oracle Fusion nei prossimi anni e Baan di SSA Global (USA). Anche Microsoft, attraverso l'acquisizione di due aziende, è entrata nel settore, con i prodotti Great Plains e Navision.

Il mercato italiano degli ERP Lite è invece più suddiviso: accanto alle suite AdHoc del Gruppo Zucchetti, seguono Team Systems con il prodotto omonimo, Esatto ed E di EsaSoftware, SiFides di SinfoPragma, ProJ di GruppoPro, xErp Diapason di GruppoFormula e molti altri.

Si vedano [O'Leary 2000], [Sheikh 2002], [BFM 2001] e [De Marco 2000] per approfondimenti.

Il Customer Relationship Management (CRM)

Il CRM è un sistema di interazione con i clienti che integra i dati provenienti dai diversi canali di contatto in un'unica base dati, condivisa da ogni area dell'azienda preposta al contatto con il cliente:

- marketing,
- vendite,
- customer service.

Il CRM è l'insieme di strategia, processi, cultura e tecnologia, che consente alle organizzazioni di incrementare le performance ed aumentare il valore attraverso una migliore comprensione dei bisogni dei clienti. Il CRM non è solo tecnologia, benché essa rappresenti un importante fattore abilitante; non è solo marketing, poiché deve coinvolgere tutta l'organizzazione in un cambiamento che è innanzitutto di tipo culturale.

Alla base di un approccio di marketing supportato da un sistema di Customer Relationship Management, vi è un circolo virtuoso che si sviluppa attraverso quattro fasi e si divide in due macro-componenti:

- Il **CRM operativo**, che è preposto alla mappatura e integrazione di tutti i canali di contatto con il cliente, ma anche all'esecuzione materiale di tutte le campagne ed azioni di marketing rivolte al cliente;
- Il **CRM analitico**, che è preposto all'analisi dei dati provenienti dalla componente operativa e dai sistemi gestionali (ERP) per profilazione, segmentazione valutazione dei clienti, al fine di ideare le offerte e le campagne di marketing più adeguate; presuppone l'esistenza di un Customer Data Warehouse che integra i dati dalle numerose fonti e ne facilita l'accesso (il concetto di data warehouse verrà spiegato insieme alla business intelligence).

In figura 6.1 viene riportato uno schema delle relazioni tra gli elementi del CRM.

Le fasi per lo svolgimento del “circolo virtuoso del CRM” possono essere schematizzate come segue:

- Identificare e segmentare i propri clienti;
- Valutarne il valore attuale e prospettico;
- Differenziare le offerte e le campagne/azioni di marketing in funzione
 - del profilo dei clienti target
 - della redditività attesa
 - del costo delle azioni ideate;
- Interagire: in questa fase si entra nella dimensione operativa del CRM, che comporta l'esecuzione materiale delle campagne/azioni di marketing prescelte; può trattarsi dell'invio di un messaggio informativo, promozionale, di offerta di un nuovo prodotto o servizio, di augurio, di invito ad un evento o seminario e così via, attraverso il canale di contatto preferito dal cliente;
- Apprendere e personalizzare: si tratta della fase che qualifica e distingue un qualsiasi sistema di analisi dei clienti da un vero e proprio processo in ottica CRM; è importante, infatti, che il sistema tracci la risposta positiva o negativa dei clienti contattati al fine di apprendere e adattare progressivamente la propria offerta alle esigenze personalizzate di ciascun segmento di clienti.

Naturalmente un ciclo di apprendimento in ottica di CRM non può trascurare di tracciare anche le nuove e impreviste azioni da parte di clienti che non erano stati compresi nella campagna di marketing in oggetto o che rappresentano nuovi contatti per l'azienda. Un cliente, infatti, può porsi in relazione con l'azienda in qualunque momento e per qualsiasi motivo e la sua interazione deve essere fatta confluire nel sistema con le informazioni relative. Nello schema proposto, tuttavia, si intende mettere in luce il ruolo proattivo dell'azienda nel cominciare il circolo virtuoso di conoscenza e interazione con i clienti.

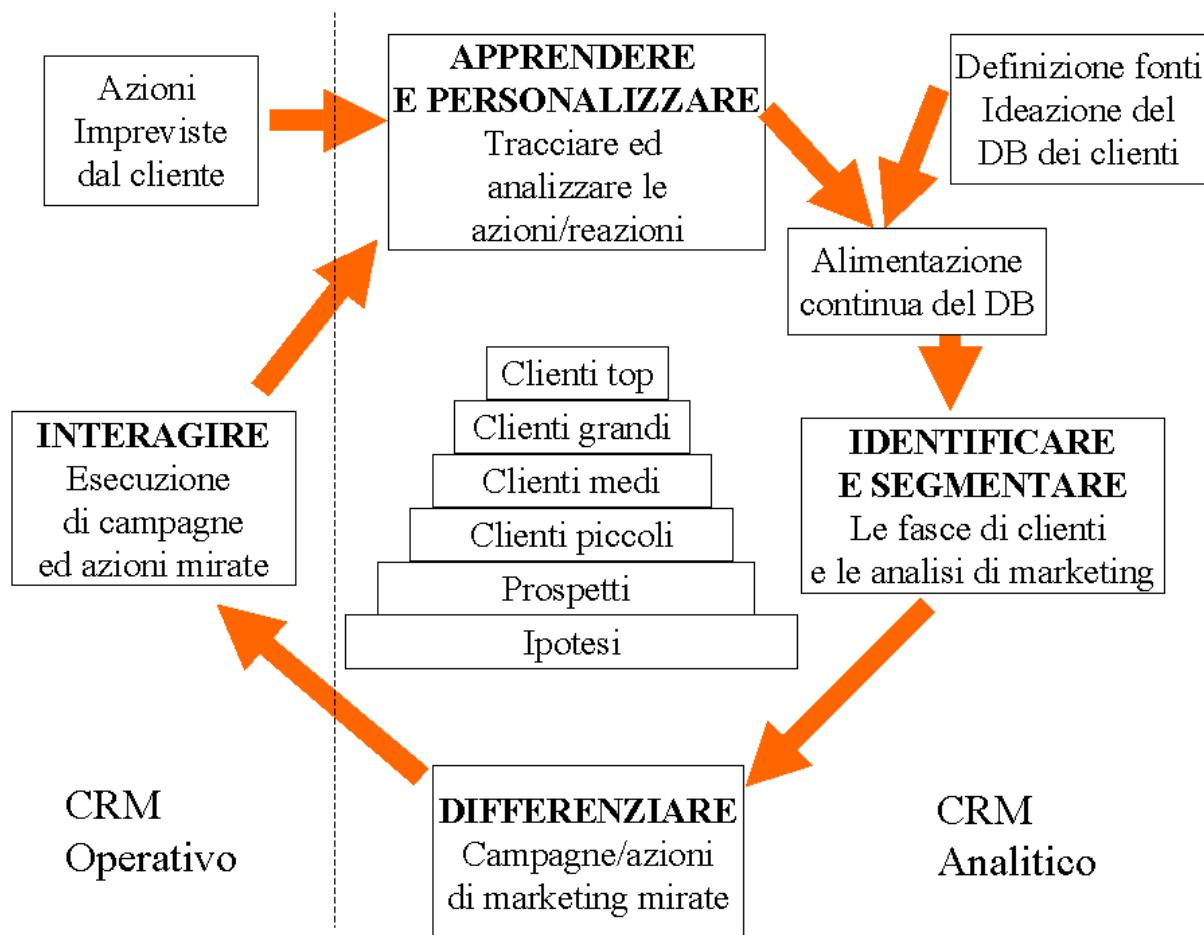


Figura 6.1: Uno schema delle relazioni tra gli elementi del CRM.

Uno dei risultati più importanti del CRM è quello di ottimizzare l'interazione con i clienti. Indirizzando il giusto messaggio, al giusto cliente, al tempo opportuno, attraverso il giusto canale. Il punto di partenza nella definizione della strategia di marketing è, come sempre, l'*analisi dei bisogni del cliente*, la differenza è la *disponibilità di strumenti informatici* che ampliano le possibilità di azione.

L'estremizzazione del processo è il cosiddetto Marketing one to one, ossia il marketing personalizzato per ogni cliente. Per giungere a tale traguardo è necessario:

- identificare i clienti dell'impresa

- classificare i clienti in gruppi omogenei
- sviluppare sistemi di interattività con i clienti
- personalizzare l'offerta di prodotti e servizi

Un software con funzioni di CRM è, ad esempio, Prometeo di AreaSP, che consente:

- La Gestione delle Anagrafiche complete dei clienti, siano essi persone fisiche od aziende o la combinazione delle due (persone che rappresentano aziende);
- La Gestione dei loro Recapiti (indirizzo fisico, telefono, cellulare, fax, anche da un punto di vista storico)
- La Gestione di Eventi come incontri, riunioni, trattative...
- La Gestione di Documenti esterni alla base dati del programma, come documenti Word o fogli Excel o file Autocad (in generale, qualunque tipo di file)
- La Gestione di ogni tipo di Associazioni fra queste entità (es. il verbale di una riunione compiuta con un certo cliente, l'offerta inviata ad un altro).

Il mercato dei grandi software per CRM è dominato dal prodotto Siebel di Siebel Systems (USA), acquisito da Oracle alla fine del 2005, cui seguono SAP CRM di SAP AG (Germania) ed altri.

La Supply Chain Management (SCM)

La SCM, traducibile letteralmente come “gestione della catena di approvvigionamento” è una metodologia, fondata sui principi della logistica, che include tutti quei processi di gestione aziendale che consentono di ottimizzare la consegna di prodotti, servizi ed informazioni dal fornitore al cliente, dove fornitore e cliente possono essere entrambi interni all’azienda, come già visto nel capitolo 2.

La logistica è essenzialmente una pianificazione di processi, organizzazione e gestione di attività, mirante ad ottimizzare il flusso di materiale e le relative informazioni all'interno e all'esterno dell'azienda. La gestione logistica, in una visione prettamente tradizionale, si occupa principalmente dell'ottimizzazione dei flussi materiali (beni) e di quelli immateriali (informazioni), ossia i flussi informativi, all'interno dell'impresa.

L'approccio basato sulla SCM invece riconosce che l'integrazione limitata all'interno della azienda non è più sufficiente. Oggi è diventato necessario ed indispensabile il coinvolgimento anche della rete di imprese che si trovano a monte e a valle nei processi e nelle attività che producono valore in termini di prodotti e servizi al consumatore finale. Viene definito in questo modo il concetto di **filiera**. Per esempio, per potere garantire la buona qualità del latte, un'azienda che vende latte confezionato deve controllare tutto ciò che va a formare il latte e quindi sia la razza delle mucche, sia il tipo di alimento che esse ricevono.

Pertanto, seguendo la visione a processi, le aziende non possono più essere viste come unità singole ma come configurazioni-costellazioni di imprese. La struttura assume una forma reticolare con nodi interrelati ad altri (rapporti tra imprese e clienti). I flussi informativi associati possono viaggiare via extranet (rete che collega l'impresa con entità esterne), intranet (rete interna collaboratori azienda) e Internet (rete telematica mondiale e computer connessi). In sostanza, la SCM si fonda sulla logistica e mira a costruire ed ottimizzare i legami ed il coordinamento tra fornitori, clienti e distribuzione e in questo senso è un complemento indispensabile del CRM nel

massimizzare il livello di servizio al cliente finale, ottimizzando contestualmente i costi operativi ed il capitale impegnato.

È importante sottolineare che, nei progetti di SCM, la collaborazione fra chi svolge il ruolo di fornitore e chi svolge il ruolo di cliente è indispensabile. È proprio attraverso questa collaborazione che si arriva a migliorare alcune funzioni come:

1. lo sviluppo della domanda, necessario al fine di comprendere più approfonditamente le esigenze dei consumatori;
2. la pianificazione della domanda, per realizzare piani di azione più attendibili e precisi;
3. il trattamento degli ordini;
4. la pianificazione della capacità produttiva e quindi il conseguente utilizzo ottimale degli impianti;
5. la pianificazione dell'utilizzo dei materiali.

In un recente corso di logistica, organizzato dal quotidiano economico "Il Sole 24 Ore" [Sole24ore 2006], autorevoli esperti hanno definito i sette punti fondamentali della SCM come segue:

1. **segmentazione della clientela**, ovvero è necessario per l'azienda intraprendere una segmentazione di mercato al fine di offrire un maggiore servizio solo a quei clienti capaci di valorizzarlo;
2. **adattamento del processo logistico-distributivo ai diversi segmenti di clientela**; si tratta di adattare le modalità di trasporto, la struttura distributiva e i canali di vendita alle esigenze del singolo cliente, o meglio della categoria di cliente;
3. ascolto dei "segnali del mercato" e **pianificazione collaborativa** al fine di evitare la distorsione delle informazioni riguardanti gli ordini di approvvigionamento (il cosiddetto effetto bull-whip, lett. "frusta da toro");
4. **differenziazione dei prodotti al più tardi possibile**; tale soluzione comporta una maggiore flessibilità e lo stoccaggio del minor numero di prodotti finiti;
5. **approvvigionamento orientato all'efficienza globale**, ovvero massima collaborazione con i maggiori fornitori;
6. **gestione delle informazioni attraverso l'ICT**; disporre di un adeguato sistema informativo è fondamentale per poter prendere le decisioni in azienda in modo non improvvisato;
7. **misurazione del livello di servizio ottenuto e del costo relativo**, necessario al fine di dirigere l'azienda nella giusta direzione.

Il mercato del software per SCM non ha un vero dominatore, tutti i grandi produttori di ERP hanno inserito le funzionalità di SCM nei propri prodotti o sviluppato moduli ad hoc. Accanto ad essi sono presenti altri produttori. Si veda [PE 2003] per approfondimenti.

La business intelligence

Gli elementi sinora visti sono orientati ai livelli funzionali più operativi, mentre la business intelligence è fortemente orientata verso i livelli manageriale e strategico. Le moli di dati che si accumulano con le normali operazioni transazionali aziendali possono essere sfruttate per vari tipi di operazioni, ma occorre strutturare i dati in modo diverso rispetto al database di produzione, per motivi sia di prestazioni sia di organizzazione logica dei dati stessi. Per business intelligence si intende un insieme di applicazioni e tecnologie per l'analisi dei dati, che comprende

- DSS (Decision Support System), già descritti nel capitolo 5
- Sistemi di interrogazione e reportistica (Query e Report)
- Strumenti OLAP (Online Analytical Processing), che saranno spiegati nel dettaglio in seguito
- Analisi statistiche
- Modelli previsionali
- Strumenti di Data Mining, che saranno spiegati nel dettaglio in seguito.

Gli scopi operativi della business intelligence sono:

- Reportistica direzionale
- Cruscotti aziendali
- Strumenti avanzati di navigazione nei dati
- Proiezioni territoriali dinamiche
- Sistemi predittivi.

In pratica quindi è compito della business intelligence di fornire il supporto per estrarre dai dati relativi l'attività quotidiana dell'impresa la informazione e, successivamente, la conoscenza per guidare l'impresa a livello strategico-decisionale. Il CRM analitico è basato anche sull'uso di metodologie di business intelligence.

Le basi di dati per l'attività quotidiana di OLTP normalmente sono caratterizzate da:

- Normalizzazione completa delle tabelle
- Alto numero di tabelle e di associazioni
- Dati memorizzati al minimo livello di granularità
- Interrogazioni che richiedono join di molte tabelle
- La struttura dei dati non varia di frequente
- Ottimizzato per inserimento dei dati e lettura piccolo numero di record alla volta.

Mentre, passando da un sistema transazionale ad un sistema di analisi, cambiano le caratteristiche di:

- normalizzazione
- prestazioni su query e modifica dei dati
- profondità storica
- complessità delle query
- dettaglio degli eventi rilevati.

La fase di Extract Transform and Load (**ETL**) rappresenta la fase di estrazione di dati dalla base dati di produzione, di trasformazione dei dati nella rappresentazione più adatta all'analisi da effettuare e di caricamento dei dati nel programma di analisi.

L'On Line Analytical Processing (OLAP) è il processo di analisi completa dei dati, il cui database ha le seguenti caratteristiche:

- Entità denormalizzate
- Disegno del database più semplice (meno tabelle e meno associazioni) per una comprensione più facile da parte dell'utente
- I dati memorizzati possono essere aggregati (riassuntivi)
- Le interrogazioni richiedono poche join
- Ottimizzato per la consultazione di grandi moli di dati, per l'utente finale è normalmente in sola lettura.

Esistono dei modelli di database generici pensati per queste esigenze come lo Star Schema e il Snowflake Schema (si vedano [AM 1997] e [Kimball 1996] per approfondimenti). Un database OLAP può essere realizzato sfruttando un generico database relazionale, ma esistono anche soluzioni specifiche diverse (OLAP DB Server), per quanto poco usate.

Il database per i sistemi OLAP può essere di due tipi:

- Il **data warehouse** è il “magazzino di dati” a livello di impresa, ossia un insieme di strumenti per convertire un vasto insieme di dati in informazioni utilizzabili dall'utente, con possibilità di accedere a tutti i dati dell'impresa, centralizzati in un solo database, coerenza e consolidamento dei dati, velocità nell'accesso alle informazioni e supporto per l'analisi dei dati;
- Il **data mart** è “magazzino di dati” a livello dipartimentale o di divisione, che rappresenta solo un segmento di un data warehouse; il data mart è fisicamente realizzato come un data warehouse, ma con una finalità più ristretta, in quanto i dati coprono solo alcune aree aziendali (ad es. vendite), e di conseguenza ha minori costi di realizzazione e garantisce l'ottenimento più rapido di risultati.

Il data warehouse è un progetto più vasto, complesso e costoso, ma garantisce maggiore coerenza dei dati a livello di tutta l'azienda (è cioè ottenibile “più” informazione). Da un data warehouse si possono ricavare velocemente dei data mart (top down) o, viceversa, si può costruire un data warehouse unendo più data mart realizzati nel tempo (bottom-up). Una relazione possibile fra data warehouse e data mart ed i flussi informativi ad essi associati vengono presentati in figura 6.2.

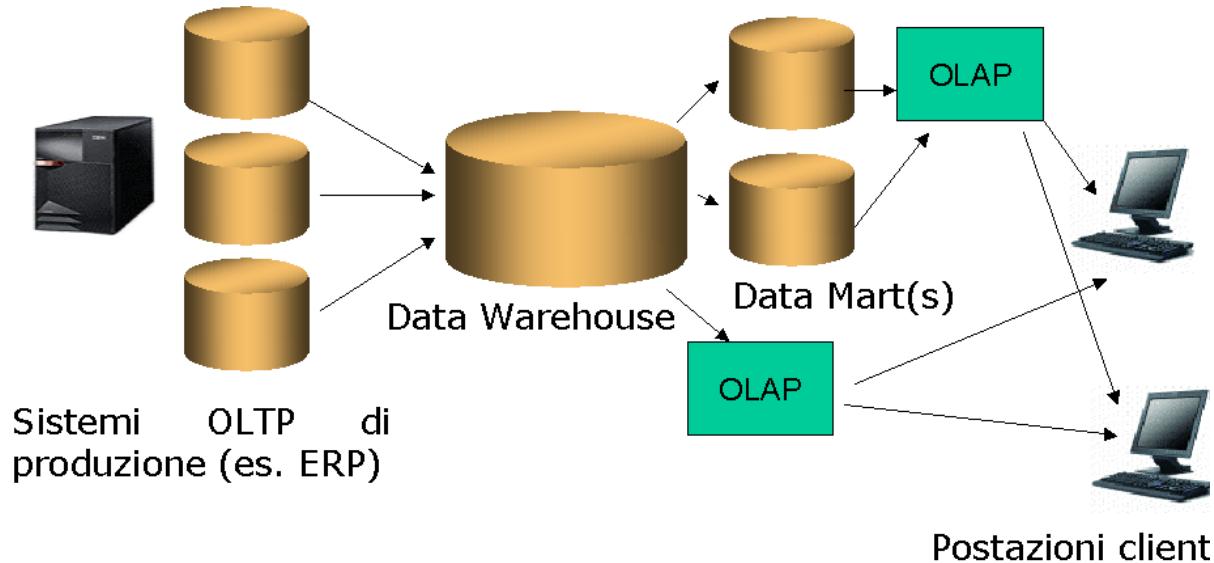


Figura 6.2: Un esempio di un data warehouse e vari data mart e le comunicazioni che fra essi intercorrono. A partire dalle basi dati aggregate nei data mart o nel data warehouse operano i vari strumenti OLAP.

L’obiettivo dell’OLAP è estrarre dai “dati grezzi” raccolti informazioni del tipo:

- **Associazioni:** situazioni connesse ad un unico evento
- **Sequenze:** eventi connessi da relazioni temporali
- **Classificazioni:** suddivisioni in gruppi ove valgono regole
- **Raggruppamenti:** definizioni di gruppi non noti a priori
- **Previsioni:** uso dei dati esistenti per scoprire dati futuri.

Il mercato degli strumenti di business intelligence è suddiviso fra:

- Strumenti di navigazione nei dati
 - SAS (grandi)
 - Business Objects (medi);
- Strumenti di reportistica
 - Crystal Report;
- Strumenti di **data mining**, ossia di “scoperta ed estrazione” automatica delle informazioni dai dati, spesso basati su sistemi di intelligenza artificiale (reti neurali, algoritmi genetici) o su correlatori statistici
 - CA-Neugents
 - Oracle Discoverer;
- Sistemi di supporto alle decisioni, nella maggior parte dei casi progettati ad hoc per gli scopi desiderati.

Un esempio di DSS è costituito dal sistema StrategyOne di CRIF (si veda [Destri 2002] per approfondimenti), il cui schema è presentato in figura 6.3. StrategyOne è un tipico DSS classificatore: una serie di vettori di input vengono presentati in ingresso e devono essere classificati, ossia associati a classi con determinate caratteristiche applicando le regole contenute nel DB ed ivi immesse da un operatore umano esperto del settore.

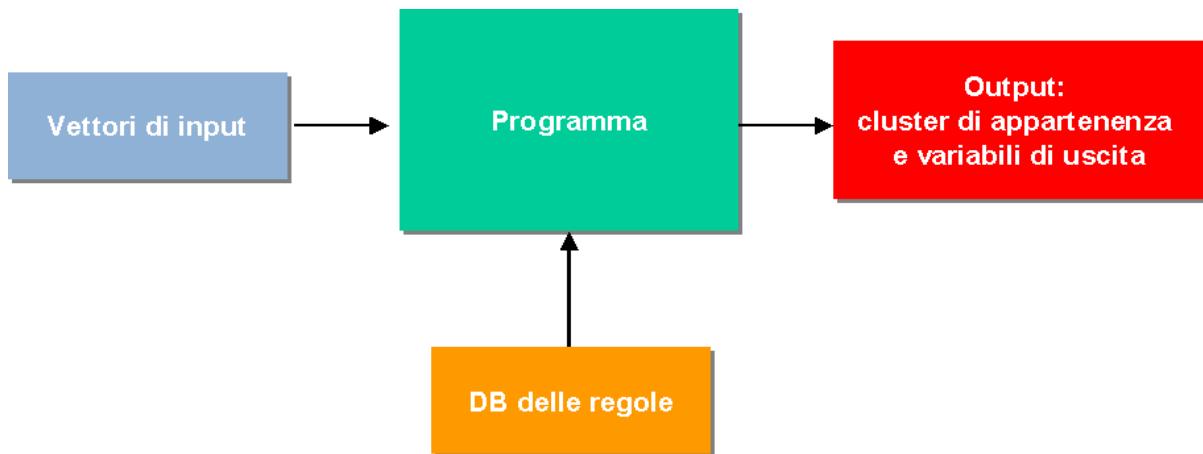


Figura 6.3: Un esempio di DSS classificatore (detto anche clusterizzatore, dal termine cluster usato come sinonimo di classe). Il sistema, sulla base delle regole memorizzate al suo interno, decide a quale classe o cluster dovrebbe essere associato il vettore di input.

Domande

1. Cos'è un ERP? Che ruolo svolge entro l'azienda?
2. Cos'è un sistema di CRM? Che importanza ha entro l'azienda? Che caratteristiche organizzative impone all'azienda per potere massimizzare la sua utilità?
3. Cos'è un sistema di SCM?
4. Cosa si intende per Business Intelligence?
5. Cos'è un Data Warehouse? Come viene normalmente usato entro l'azienda?
6. Cos'è un Decision Support System (DSS)?

Bibliografia

[ABAP 2006] Sito web del linguaggio ABAP entro il portale di SAP, <https://www.sdn.sap.com/irj/sdn/developerareas/abap>

[AM 1997] S. Anahory e D. Murray - Data Warehousing in the Real World: A Practical Guide for Building Decision Support Systems - Ed. Addison Wesley Professional, 1997

[BFM 2001] G. Bracchi, C. Francalanci, G. Motta - *Sistemi Informativi e aziende in rete* - McGraw-Hill Italia, Milano, 2001

[De Marco 2000] M. De Marco - *Sistemi Informativi Aziendali* - Franco Angeli Edizioni, Milano 2000

[Destri 2002] G. Destri - *Pattern e template nei progetti C++: un case study* – Computer Programming n. 110 – Edizioni Infomedia, Febbraio 2002, disponibile anche su Web all'indirizzo: <http://online.infimedia.it/riviste/cp/110/articolo16/articolo.htm>

[Kimball 1996] R. Kimball - The Data Warehousing Toolkit - Ed. Wiley, 1996

[O'Leary 2000] D. E. O'Leary - Enterprise Resource Planning Systems - Systems, Life Cycle, Electronic Commerce, and Risk - Ed. Cambridge Press, 2000

[OTP 2002] D. O'Mahony, H. Tewari e M. Peirce - *Electronic Payment Systems for E-Commerce (2nd Edition)* - Ed. Artech House, 2002

[PE 2003] C. A. Ptak, E. Schragenheim - *ERP - Tools, Techniques and Applications for Integrating the Supply Chain* – CRC Press, 2003

[Sheikh 2002] K. Sheikh - Manufacturing Resource Planning (MRP II) with Introduction to ERP, SCM, and CRM - Ed. McGraw-Hill, 2002

[Sole24ore 2006] Sito Web dei corsi de “Il sole 24 ore”
<http://www.formazione.ilsole24ore.com/>

Le professionalità nei sistemi informativi

Le risorse umane ed il loro ruolo

Come già più volte detto, il ruolo della componente umana entro i sistemi informativi è fondamentale, in quanto sono le risorse umane ad utilizzare le risorse tecnologiche secondo le regole espresse dalle procedure aziendali che formano i processi business. Pertanto la efficienza e l'efficacia di un sistema informativo dipenderanno principalmente dalle risorse umane e dalla loro abilità nell'usufruire delle risorse tecnologiche e nell'applicare le procedure aziendali.

I ruoli espressi dalle risorse umane possono essere molteplici e fortemente dipendenti dal contesto: spesso in un piccolo gruppo la stessa persona si troverà a ricoprire più ruoli. Per questo prima verranno presentati i ruoli “canonici” e poi esempi tipici del mondo delle imprese e degli enti pubblici italiani, analizzando il ruolo reale.

In generale occorrerà distinguere fra ambienti di sviluppo, ove viene sviluppato o personalizzato del software (perché tale azione è il core business, come avviene nelle aziende di informatica, o perché esiste uno sviluppo interno entro il sistema informativo di un'azienda non informatica) e ambienti di esercizio, in cui lo scopo del personale informatico è garantire il buon funzionamento delle risorse tecnologiche inserite nel sistema informativo, oltre che di provvedere agli acquisti ed ai ricambi di tali risorse nel tempo. Nel primo caso vi saranno più ruoli, con maggiori specializzazioni.

E’ importante ricordare che lo sviluppo può essere interno od esterno all’azienda destinataria finale del sistema software, più o meno complesso, che viene sviluppato. Un esempio di come si procede di solito è riportato in figura 7.1.

In seguito sono presentati i ruoli “canonici”, che prima qui di seguito elenchiamo. Per un ambiente di sviluppo (o di personalizzazione, come nel caso degli ERP) i ruoli possono essere:

- cliente
- utente (finale) del sistema
- acquirente
- venditore
- analista funzionale
- analista tecnico
- analista di progetto
- progettista di alto livello
- progettista di dettaglio
- specialista di ERP (o di altro software complesso)
- programmatore
- sviluppatore interfacce web
- tester
- capo area
- capo progetto
- amministratore di sistema (sistematista)

- amministratore di rete
- amministratore di DB (DBA)
- responsabile qualità
- responsabile sicurezza
- consulente agli acquisti (software/solution selector)
- solution provider

In un ambiente di esercizio i ruoli si riducono a:

- utente (finale) del sistema
- amministratore di rete
- amministratore di sistema (sistemista)
- amministratore di DB (DBA)
- responsabile sistemi informatici (anche chiamato EDP manager)

Nel resto del capitolo, prima verranno presentati i dettagli delle singole figure professionali sopra elencate, poi verranno presentati esempi reali, tipici del panorama di aziende informatiche e sistemi informativi italiani, di organizzazione di risorse umane. In realtà spesso le aziende di informatica hanno al proprio interno sia sezioni dedicate allo sviluppo, sia sezioni dedicate all'erogazione di servizi.

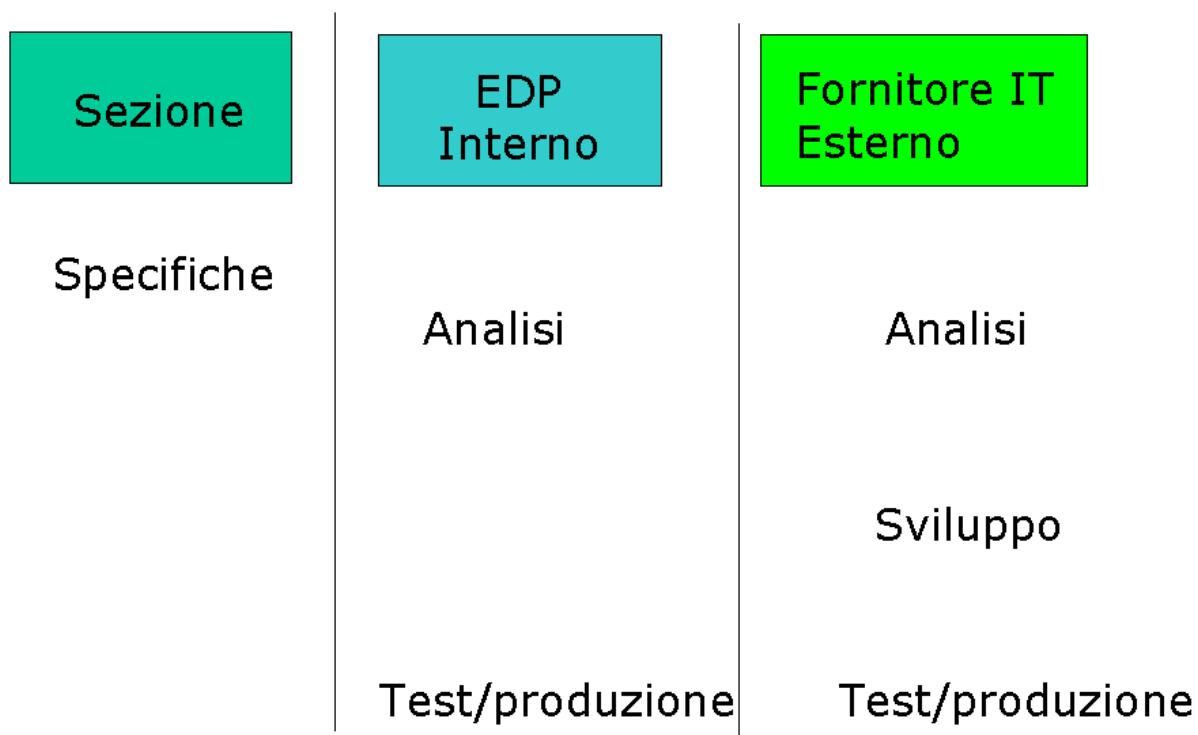


Figura 7.1: suddivisione tipica dei compiti fra reparti esterni ed interni ad un'azienda che richiede lo sviluppo o personalizzazione di un programma. Lo sviluppo potrebbe anche essere completamente interno.

I dettagli dei ruoli “canonici”

Il cliente

Il cliente è colui che è disposto a pagare per il valore associato ad un sistema software che viene sviluppato. Può essere esterno od interno all’organizzazione entro cui avviene lo sviluppo. In ogni caso, essendo il processo di sviluppo costoso, per via del costo delle risorse umane e tecniche impiegate al suo interno, deve essere pagato da un cliente.

In pratica quindi il cliente è il rappresentante dell’azienda cliente finale o dell’unità funzionale (dipartimento) cliente del servizio di sviluppo entro l’organizzazione. E’ importante osservare che non necessariamente il cliente è anche un utilizzatore finale del sistema che viene sviluppato.

L’acquirente

E’ il funzionario, tipicamente un responsabile dell’ufficio acquisti, che tratta commercialmente l’acquisto del programma/sistema software. Quasi mai è un utilizzatore finale del sistema ed è quasi sempre maggiormente attento agli aspetti del costo immediato rispetto all’efficienza che avrà il sistema una volta inserito entro l’organizzazione.

L’utente (finale) dei sistemi informatici

L’utente finale è l’operatore che usa materialmente il sistema software sviluppato. Può essere un semplice impiegato così come un dirigente. Quasi sempre non ha competenza e nemmeno cultura informatica e spesso, specialmente nelle piccole organizzazioni, ha dovuto imparare da autodidatta la maggior parte dell’uso degli strumenti informatici, ossia difficilmente ha partecipato a corsi di formazione al riguardo. Per questo motivo spesso gli errori dell’utente possono essere molto strani e imprevedibili da chi ha progettato il sistema software.

Negli ultimi anni comunque le aziende hanno cominciato a capire che inefficienze nell’uso dei sistemi informatici, dovute a mancanza di conoscenze, sono un grave handicap e una fonte di perdite economiche, e perciò il livello di conoscenze informatiche degli utenti è stato fatto crescere.

Il venditore

E’ la controparte dell’acquirente, ossia colui che deve portare a termine la vendita del sistema informatico, convincendo acquirente ed altri rappresentanti del cliente a portare a termine l’acquisto pagando la cifra più alta possibile. I migliori venditori hanno anche competenza tecnica e sanno suggerire soluzioni al cliente stabilendo un rapporto di fiducia con lui.

L’analista funzionale

Ha esperienza di analisi di processo ed ha compreso il processo specifico del cliente, lavorando con personale del medesimo. Scrive le specifiche funzionali di dettaglio, formalizzate, che il programma o sistema software deve seguire.

L'analista di processo

E' analogo all'analista funzionale, ma opera in contesti più grandi e complessi (grande impresa, grande ente pubblico, come un ministero). E' specializzato nel seguire uno o più processi (es. magazzino, logistica) entro un contesto specifico (es. azienda metalmeccanica). Conosce anche il funzionamento di un ERP e suo è il compito di stabilire quanto il processo reale è distante dal processo "mappato" nell'ERP e quindi, in sostanza, che tipo di Business Process Reengineering deve essere svolto.

Il progettista di alto livello (chiamato anche architetto o progettista architetturale)

Opera entro progetti di sistemi complessi. Il suo lavoro parte dall'analisi compiuta da analisti funzionali e di processo. Deve trasformare i flussi informativi in flussi informatici veri e propri e definire l'architettura del sistema, decidendo che componenti infrastrutturali usare (es. quale DB server).

Il progettista di dettaglio (chiamato anche analista tecnico)

Nel caso di progetti complessi il suo lavoro copre una parte del sistema disegnato dal progettista di alto livello, in casi più semplici può coprire tutto. Suo compito è trasformare le specifiche funzionali in specifiche tecniche ed istruzioni realizzative per i programmati. Esistono varie metodologie codificate per questa azione di trasformazione, si veda [ZBGK 2004] per approfondimenti.

Lo specialista di ERP (o di altro software complesso)

Collabora con l'analista di processo per definire le personalizzazioni e le modifiche (anche tecniche) che il software deve subire per potere essere inserito proficuamente entro i processi dell'azienda destinataria.

Il programmatore

Spesso è considerato il "muratore" dell'informatica, ma suo è il compito più importante ovvero scrivere e verificare il codice nel linguaggio di sviluppo scelto. Entro progetti complessi i programmati possono anche essere varie decine, a ciascuno dei quali viene affidato lo sviluppo di uno o più moduli software (in ambiente object-oriented di gruppi di classi). Deve conoscere al meglio il linguaggio di sviluppo e le metodologie utilizzate.

Il disegnatore di interfacce o grafico

E' essenzialmente un grafico, spesso privo di una profonda preparazione informatica, specializzato nella progettazione e nel disegno delle interfacce e delle pagine Web. Deve ovviamente possedere competenze relative ai vincoli che la realizzazione di una pagina Web pone rispetto ad altri strumenti IT (es. immagini non troppo grandi, colori di sfondo di un certo tipo ecc...).

Lo sviluppatore di interfacce utente Web

Svolge un compito simile al programmatore, ma specializzato nella realizzazione di interfacce, ovvero pagine, Web. Il suo lavoro è maggiormente concentrato sulla realizzazione grafica e molto meno su quella programmatica, anche se normalmente è suo compito anche la realizzazione delle miniprocedure in linguaggio Javascript presenti entro una pagina Web. Spesso non ha grandi competenze di programmazione.

Il tester

Deve svolgere i test dei moduli software realizzati, sia singoli, sia dopo la loro integrazione a costruire il sistema (test di integrazione). Deve stilare appropriati rapporti sul comportamento delle parti verificate, sia in caso di funzionamento corretto, sia in caso di errori. Spesso il test viene compiuto da programmatore che però dovrebbero lavorare sul codice scritto da altri. Per garantire una ricerca efficace degli errori è opportuno che almeno una parte dei test sia compiuta da utilizzatori finali o comunque da personale con il loro modus operandi.

L'amministratore di sistema (sistemista)

E' lo specialista del sistema operativo della macchina o delle macchine che ospitano il/i sistemi software. Nel caso di progetti di realizzazione deve definire le specifiche tecniche delle macchine affinché il sistema software funzioni al meglio, mentre nelle fasi di esercizio deve controllare e gestire i sistemi per garantirne il funzionamento ottimale. In quest'ultimo caso, specialmente in organizzazioni grandi, i sistemisti sono suddivisi tra coloro che si occupano di gestire i server, normalmente aventi maggiore esperienza, e coloro che gestiscono invece le postazioni utente.

L'amministratore di rete

Si occupa delle comunicazioni e suo compito è gestire l'infrastruttura di rete, dal cablaggio agli apparati trasmissivi (es. router, access point WiFi, collegamenti ADSL), tenendo eventualmente anche i rapporti con i fornitori di connettività (es. Telecom Italia, FastWeb...). Nei progetti di sviluppo viene chiamato a disegnare i canali di comunicazione per i flussi informatici associati al progetto stesso.

L'amministratore di DB (DBA)

E' una figura tipica di organizzazioni medio-grandi. Il suo compito è manutenere in modo ottimale il DBMS ed il suo contenuto (ossia le tabelle e le altre strutture ausiliarie che contengono i dati), per garantire il buon funzionamento degli applicativi che accedono a tali dati. In contesti come le banche il ruolo di amministratore del database si divide in due: l'amministratore della base dati, il cui compito è agire sulle strutture dati, e l'amministratore del DBMS, il cui compito è invece quello di garantire il funzionamento migliore possibile degli applicativi software che formano il DBMS, oltre che il salvataggio periodico (ossia il backup) dei dati stessi. Nei progetti di sviluppo contribuisce al design della base dati e/o al dimensionamento del database server.

Il capo area

Figura dirigenziale che copre una o più aree funzionali entro il contesto di un sistema informatico (ad esempio il coordinatore dei sistemisti, il direttore degli sviluppatori, il responsabile dell'informatica utente ecc...). Un caso particolare di capo area è il responsabile generale dei sistemi informatici, il cosiddetto EDP manager, figura che di solito non ha grande competenza tecnica, ma maggiormente competenza funzionale.

Il capo progetto

E' il responsabile della realizzazione di un progetto informatico. Può essere considerato come il responsabile del processo che deve condurre a termine il progetto in questione. Può essere il superiore dei capi area o dovere collaborare con loro, non di rado contrattando la disponibilità delle risorse umane da loro controllate.

Il responsabile qualità

E' il responsabile delle procedure di qualità nella gestione del progetto o dell'esercizio. Deve sovraintendere, verificando il rispetto delle procedure relative alla qualità e suggerire le correzioni metodologiche da attuare qualora riscontri variazioni rispetto al desiderato. Deve collaborare con il capo progetto, fornendogli anche una vista sulla situazione effettiva del progetto stesso. E' una figura trasversale, che interagisce con tutti e deve essere dotata di buona diplomazia per svolgere al meglio il proprio ruolo.

Il responsabile della sicurezza

Questo ruolo molto di rado è assegnato da solo ad una risorsa umana: nella maggior parte dei casi è svolto dal sistemista di rete o da un altro sistemista. Compito del responsabile sicurezza è quello di garantire il rispetto delle regole di sicurezza entro l'azienda od organizzazione. Le problematiche associate alla sicurezza informatica saranno trattate nel prossimo capitolo.

Il consulente acquisti (software selector)

Figura relativamente nuova per il panorama italiano. Tipicamente esterno all'organizzazione bisognosa del sistema software, il consulente acquisti deve possedere competenze funzionali come un analista di processo e conoscere le varie soluzioni software per quel processo particolare presenti sul mercato (e l'hardware su cui esse operano). Il suo ruolo è quello di consigliare la soluzione migliore per l'azienda in relazione a quella particolare esigenza con un compromesso costi-benefici.

Il solution provider

Figura relativamente nuova per il panorama italiano. E' un fornitore dell'azienda che, producendola in proprio o reperendola sul mercato, fornisce all'azienda cliente la soluzione rispetto alla esigenza. Svolge un compito simile a quello del consulente acquisti, ma vende anche la soluzione.

Alcuni esempi di organizzazioni “reali”

Piccola Software House

Normalmente un’azienda di questo tipo è concentrata sullo sviluppo di pochi prodotti sia di uso generale (come gestionali o ERP lite), sia fortemente specifici, a bassa complessità. Talvolta vengono sviluppati progetti ad hoc, ossia programmi o sistemi specifici per un cliente, da cui poi eventualmente saranno tratti prodotti di uso più generale, processo noto in gergo come “pacchettizzazione”. Entro un contesto di questo tipo sono presenti poche figure con esperienza che fungono da analisti, progettisti e capi progetto (e talvolta anche da vendori) e un numero superiore di sviluppatori, cui spesso è affidata tutta la realizzazione di un progetto, testing compreso. I programmatore non sono fortemente specializzati, nel senso che devono essere in grado di realizzare tutte le parti di un sistema. Le tecnologie usate per lo sviluppo sono poche o addirittura una sola.

Grande Software House

Un’azienda di questo tipo è molto più strutturata e normalmente ciascuna risorsa umana è dedicata ad uno solo dei compiti descritti in precedenza. L’azienda può comprendere molte centinaia di persone ed è divisa in sezioni, ciascuna dedicata ad uno scopo specifico, come ad esempio lo sviluppo di un solo programma o alla consulenza relativa ad un singolo prodotto.

Azienda di servizi IT

Può avere dimensioni piccole o grandi ed al suo interno può anche essere presente una sezione di sviluppo che rientra nei modelli visti sopra. Ma l’attività prevalente è quella di erogare servizi di vendita di hardware e software ed assistenza su quanto installato presso i clienti. Pertanto molte delle risorse umane presenti sono sistemisti di vario tipo. L’attività di assistenza può essere erogata “in remoto”, sia attraverso la tradizionale assistenza telefonica, sia attraverso l’accesso remoto tramite collegamenti punto-punto o VPN (reti private virtuali, che saranno trattate in dettaglio nel capitolo 8), grazie alle quali si interviene direttamente sulle infrastrutture informatiche presso il cliente. Ma in molti casi è tutt’ora necessario che i tecnici si rechino presso il cliente, sia per attività di installazione, assistenza e manutenzione, sia per l’erogazione di corsi di formazione al personale del cliente.

Grande gruppo di sviluppo IT

Un gruppo di lavoro di questo tipo, che può essere formato di molte decine di persone, può essere interno ad una grande software house, ma anche trovarsi presso un cliente finale. Questo avviene molto spesso in settori come le aziende di telecomunicazioni e le banche. Un gruppo di lavoro di questo tipo ha un compito specifico di sviluppo o manutenzione di un sistema, è affidato alla guida di un capo progetto che delega il coordinamento di sezioni specifiche del progetto (ad esempio, la gestione sistemistica) a capi area. Le risorse umane presenti nel progetto sono dei tipi definiti in precedenza e possono anche essere appartenenti a diverse aziende o essere direttamente dei liberi professionisti.

Piccola azienda

Il ruolo primario di addetti IT entro un'azienda di questo tipo è garantire il buon funzionamento dei sistemi informatici presenti nell'azienda stessa. La maggior parte dell'attività è quindi di tipo gestione di sistema ed assistenza agli utenti. Le persone addette ai sistemi IT sono poche e impegnate nella gestione di tutte le parti del sistema, oltre che, spesso, anche addette alla gestione di altri impianti come quello elettrico o telefonico.

Media azienda

Il ruolo primario è lo stesso del caso precedente, ma, essendo il sistema da gestire più ampio, sono presenti anche strutturazioni e specializzazioni. Il numero di addetti IT può essere anche di una decina, coordinati da un dirigente. Talvolta vengono realizzati internamente all'azienda anche programmi o personalizzazioni complesse di programmi esistenti. Spesso collaboratori esterni sono presenti entro i centri, per periodi più o meno lunghi.

Grande azienda

Le infrastrutture IT sono molto grandi ed il numero di addetti IT può essere molto grande. Sono presenti strutturazioni e quindi suddivisioni in aree (ad esempio, gestione reti, gestione server, gestione informatica utente, gestione database, gestione ERP ecc...). I ruoli delle varie risorse umane sono chiaramente definiti da un organigramma preciso. Nel caso di aziende aventi varie sedi sul territorio sono spesso presenti vari centri di elaborazione dati, anche se negli ultimi anni l'applicazione della amministrazione remota ha prodotto la concentrazione delle attività informatiche in un solo centro. Nel caso di multinazionali inoltre spesso sono presenti solo pochi grandi centri su scala continentale, operando riduzioni del personale e/o impiantando i centri ove il personale costa meno (ad esempio, Europa orientale o Asia). Entro i centri elaborazione dati sono quasi sempre presenti anche consulenti esterni.

Piccolo comune

La situazione è simile a quella della piccola azienda. Lo scopo è garantire il funzionamento migliore possibile dei sistemi. Gli addetti ai servizi IT normalmente sono molto pochi e spesso condivisi fra più piccoli comuni per ridurre le spese. Talvolta il servizio è svolto da consulenti esterni.

Grande comune

Un comune di una città grande è paragonabile ad una media o grande azienda. Non soltanto devono essere garantiti i servizi, ma esistono sviluppi interni, personalizzazioni. Per grandi città, come Milano, devono essere anche garantiti servizi distribuiti, accessi via rete al cittadino e simili. Perciò sono presenti figure professionali diverse, organizzate in modo analogo alla grande azienda.

Provincia

I centri di calcolo delle province, oltre ad erogare servizi in modo analogo alla grande azienda, devono gestire comunicazioni e rapporti con i comuni da un lato e con le regioni e lo stato dall'altro. Negli ultimi anni sono state inserite quindi anche molte professionalità legate alle comunicazioni, come gli amministratori di rete.

Regione

Le regioni hanno centri di elaborazione dati molto grandi, con centinaia di addetti. Gli organigrammi sono tipicamente suddivisi per funzioni, ovvero per servizi erogati: per esempio troviamo il catasto, i servizi sociali, le anagrafi regionali ecc... La situazione è quindi analoga a quella della grande impresa.

Piccola Banca

I sistemi informatici sono suddivisi nettamente tra la parte server (mainframe e altri server) e informatica utente. Lo sviluppo interno non è molto presente, per lo più limitato a personalizzazioni. Gli analisti funzionali sono di solito persone di estrazione economica. Per il resto la strutturazione è simile a quella della media azienda.

Grande Banca

Situazione simile a quella della piccola banca. Sono presenti diversi grandi centri sul territorio, spesso con capacità di sostituzione, per garantire la continuità di servizio anche in caso di catastrofe. Entro questi centri viene compiuta la teleassistenza ad opera degli addetti all'informatica periferica. Esiste molto sviluppo interno e sono presenti molti consulenti esterni, spesso distaccati da parte di aziende fornitrice.

Domande

1. Chi sono gli attori e quali sono le relazioni tra loro durante lo sviluppo di una applicazione informatica?
2. Che ruolo deve svolgere il responsabile acquisti del cliente?
3. Che ruolo svolge l'analista funzionale?
4. Che ruolo svolge il capo progetto entro un progetto informatico?
5. Che ruolo svolge il programmatore entro lo sviluppo informatico?
6. Che ruolo svolgono gli amministratori di sistema?

Bibliografia

[BFM 2001] G. Bracchi, C. Francalanci, G. Motta - *Sistemi Informativi e aziende in rete* - McGraw-Hill Italia, Milano, 2001

[GMN 1995] S. Gai, PL Montessoro, P. Nicoletti - *Reti Locali: Dal Cablaggio All'Internetworking* - Scuola Sup. G. Reiss Romoli, 1995

[LL 2004] K Laudon, J Laudon – *Management dei sistemi informativi* – Ed. Pearson Education Italia, Milano, 2004

[OHE 1999] R. Orfali, D. Harkey, J. Edwards - *Client/Server Survival Guide, 3rd Edition* – Ed. Wiley, 1999

[ZBGK 2004] W. Zuser, S. Biffl, T. Grechenig, M. Kohle - *Ingegneria del Software con UML e Unified Process* - Ed McGraw-Hill, 2004

La sicurezza informatica

Le problematiche della sicurezza informatica

Il problema della sicurezza informatica è oggi uno dei più critici tra quelli che affliggono i sistemi informativi moderni. Basta sfogliare cronache anche non del settore per leggere resoconti, più o meno precisi di problemi inerenti la sicurezza. Se fino agli anni '90 i sistemi erano più isolati e si potevano mettere in atto contromisure per la protezione, oggi l'uso della rete Internet è una necessità fondamentale per qualsiasi azienda e quindi i sistemi non possono più essere isolati e sono esposti a nuove minacce. L'avvento di nuove tecnologie, come, ad esempio, le reti wireless, non ha fatto altro che peggiorare il problema.

Si può affermare che oggi la situazione della sicurezza informatica è sempre più critica e ogni settimana le cronache registrano nuovi attacchi, in gran parte dei casi non denunciati.

Le cause di questi problemi sono molteplici. Innanzitutto la rapidissima evoluzione dei sistemi e la necessità per i produttori di software di rilasciare con sempre maggiore rapidità nuove versioni di programmi applicativi conduce a software non completamente testato in produzione. Inoltre, tra il 1998 e il 2002, il problema dell'anno 2000 e l'avvento dell'Euro hanno assorbito la maggior parte delle risorse dei sistemi informatici e questo, unito al calo dei budget dopo la crisi del 2001, ha fatto calare drasticamente le possibilità di spesa per la sicurezza. L'inserimento in produzione di nuove piattaforme tecnologiche come Windows2000 e XP ha prodotto ulteriori problemi. Inoltre, dal 1990 ad oggi la sofisticatezza delle tipologie di attacco non ha fatto che crescere, ma nello stesso tempo, la disponibilità di programmi applicativi utilizzabili anche come strumenti di attacco, legali e non, diffusi in rete, rende possibile anche a persone con minore competenza tecnica di compiere attacchi di estrema pericolosità. Infine la diffusione di virus informatici in grado di propagarsi attraverso la rete ha assunto ormai i contorni di una vera e propria epidemia.

I rischi diretti per un'azienda possono essere riassunti in:

- Furti di denaro, informazioni, dati sui propri clienti
- Perdita di produttività dovute a corruzione dei dati o danneggiamento dei sistemi, con in più spese e tempo perso per il ripristino delle normali condizioni di operatività

Ma accanto ad essi sono presenti numerosi rischi indiretti, come, ad esempio:

- Perdita di potenziali clienti
- Svantaggi sui propri prodotti
- Impatto negativo sul proprio brand name
- Esposizioni legali dovute al non rispetto delle clausole di riservatezza e al non rispetto delle leggi sulla privacy (si ricordi che nel testo del "decreto privacy" D.L. 196/2003 si definisce il principio di responsabilità sui dati, si veda [DL196 2003])

Oggi la sicurezza informatica è una necessità per qualsiasi azienda od organizzazione, ma nel contempo lo sfruttamento delle potenzialità delle reti è anch'esso una necessità.

Pertanto le due esigenze devono convivere attraverso una consapevole politica di gestione. Se sino a poco tempo fa il personale EDP si doveva concentrare solo sui livelli applicativi, oggi esso è impegnato a fornire all'azienda tutta una serie di nuove funzionalità e la conoscenza delle tecnologie di reti e delle basi di sistemi operativi diventa una necessità per garantire consapevolmente le appropriate misure di sicurezza.

Nel resto del capitolo saranno in primo luogo presentate le varie categorie di minacce alla sicurezza, e poi affrontati i rimedi per la protezione dei dati memorizzati, delle trasmissioni e dei sistemi, nonché per la gestione dell'identità elettronica.

In figura 8.1 è mostrato una visione schematica del problema “sicurezza informatica” con suoi componenti principali e le loro caratteristiche. Come si vede la informatica basata sulle reti, detta anche spesso “network IT” o “networked IT” crea la possibilità di nuovi attacchi e quindi crea nuovi rischi. I rimedi tecnici permettono di proteggere i dati, i canali di comunicazione e l'identità di sistemi ed operatori che accedono alla rete, e anche di proteggere i sistemi stessi. Ma la tecnica non è sufficiente, controlli per la sicurezza devono coesistere con il business attraverso una oculata politica di gestione.

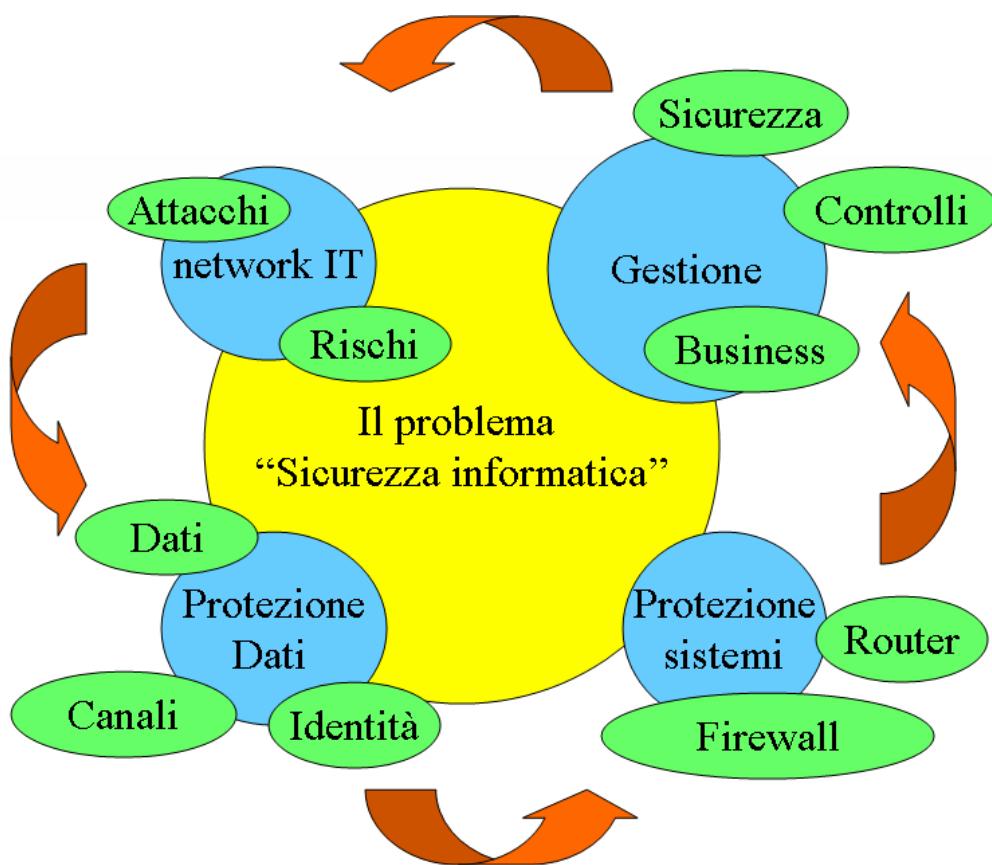


Figura 8.1: Un sunto delle relazioni fra le componenti del problema “sicurezza informatica”.

Capire nei dettagli le problematiche di sicurezza è indispensabile per impostare qualsiasi politica di gestione. Un primo punto da definire è la differenza fra le

problematiche di Safety e di Security. Ambedue i termini inglesi sono tradotti in italiano con sicurezza, ma il loro significato è molto diverso.

Con **Safety** si indicano le problematiche legate alla prevenzione dei danni da guasti accidentali dei sistemi informatici.

Con **Security** invece si indicano le problematiche legate a problemi di attacco o uso fraudolento dei sistemi.

La prevenzione di problemi di Safety

Parlare di **Safety** significa descrivere le tecniche di salvaguardia dei dati che li difendono da guasti tecnici accidentali o danneggiamenti fisici subiti dai sistemi. In un contesto di sicurezza informatica è opportuno trattare anche le problematiche di safety sia perché esistono leggi che regolamentano la protezione dei dati dalle perdite accidentali, sia perché spesso le tecniche di safety sono un'ottima risposta anche a problemi di security.

Per prima cosa è importante individuare i punti critici dei sistemi informatici rispetto a guasti e problemi accidentali. Alcuni di tali punti sono:

- I sistemi operativi non sempre sono sufficientemente robusti rispetto a condizioni operative non infrequenti
- Le macchine hanno parti meccaniche soggette ad usura (ventole, dischi etc...)
- La componentistica elettronica che forma i sistemi informatici può presentare dei problemi
- Gli utenti non esperti possono commettere errori nell'uso dei sistemi e/o dei programmi
- Il software non è mai completamente esente da errori e altri problemi possono essere creati dalla installazione di nuovi programmi o di nuove versioni di programmi e/o librerie esistenti (si ricordi la matrice delle compatibilità e dipendenze presentata nel capitolo 5)
- L'alimentazione elettrica può essere soggetta a variazioni e/o interruzioni e non sempre filtri e gruppi di continuità rappresentano protezioni sufficienti
- I dispositivi di storage e i computer possono essere danneggiati da catastrofi ambientali (fulmini, altri fenomeni atmosferici, inondazioni, incendi, terremoti...)

In relazione a questi problemi occorre tenere presente due aspetti distinti di protezione:

1. La **protezione dei dati**, il “tesoro” contenuto entro i sistemi, che viene realizzata attraverso apposite politiche di salvataggio dei dati stessi;
2. La garanzia di **continuità di servizio** (la cosiddetta **business continuity**), ovvero la capacità del sistema di continuare ad erogare il servizio informatico per cui esso è stato installato anche in condizioni problematiche; in alternativa la minimizzazione del tempo di fermo macchina (il cosiddetto downtime) e quindi la capacità di riprendere ad erogare il servizio nel minor tempo possibile.

E’ infatti importante osservare che la salvaguardia dei dati è sicuramente una parte importante, ma non è tutto: disporre solo dei dati su CD-ROM senza i sistemi informatici e gli applicativi che li rendono usufruibili da parte degli utenti non è molto utile per il business.

Prima di descrivere le tecniche tipiche di salvaguardia dei dati è necessario definire con precisione cosa intendiamo per “dati”: nell’ambito di un sistema informatico, tipicamente quali sono i dati da salvaguardare? Un elenco parziale è qui riportato:

- Contenuto di DBMS relazionali, sia sotto forma dei file di dati interni al DBMS, sia sotto forma di file di esportazione (i cosiddetti export o dump);
- Archivi documentali/multimediali, sia contenuti direttamente entro file system, sia entro programmi di gestione file, come Lotus Notes o Microsoft SourceSafe;
- Micro applicativi (es. generatori report), tipicamente realizzati con linguaggi di macro di programmi, come Microsoft Office;
- DB personali (es. elenco indirizzi della rubrica di Windows o di altro programma di gestione posta elettronica);
- Archivi di Directory Service, sia contenuti nell’interno dei programmi, sia esportati in vari formati;
- Configurazioni dei programmi e delle postazioni di lavoro.

I dati dell’elenco sopra riportato hanno importanza diretta o indiretta per il business. Il contenuto di database o i file possono essere direttamente importanti. Ma le configurazioni dei programmi e degli ambienti di lavoro, o i micro applicativi, possono essere egualmente importanti, in quanto garantiscono all’utente l’ambiente abituale e quindi la produttività normale per il proprio lavoro.

Fisicamente questi dati si trovano sia sui dischi di file server e database server, sia sui dischi delle singole postazioni utente. Occorre pertanto provvedere alle tecniche migliori possibili per la salvaguardia di questi dati.

La prima soluzione per la salvaguardia dei dati è il salvataggio dei file che li contengono, su supporti opportuni di memorizzazione, procedura che prende il nome di **backup**. In un sistema distribuito di dimensioni medio-grandi, essendo pressoché impossibile procedere al backup di tutte le postazioni client, il primo passo è quello di procedere al salvataggio dei dati rilevanti dalle postazioni client su apposite porzioni dei dischi dei server, che saranno poi salvate. Questo processo di copiatura sarà compiuto dagli operatori o potrà essere organizzato in modo automatico, ad esempio, configurando i sistemi affinché le cartelle personali degli utenti non siano locali alle singole postazioni client ma posti sul server.

Alcuni supporti tipicamente usati per il backup sono:

- Nastro DAT (da pochi ad alcune decine di GB)
- Nastro DSS (decine di GB)
- Nastro VHS (24 GB)
- Altri nastri, di capacità ancora più elevate
- CD, DVD ROM
- Dischi estraibili
- Flash memory Pen
- Dischi distribuiti.

Il backup può essere centralizzato, quindi associato ai server e compiuto a livello di tutta la rete, o personale, quindi compiuto dai singoli utenti sui propri dati. In ogni caso, il backup dovrebbe essere gestito da un’apposita politica. Per definire una politica di backup occorre partire dalle seguenti domande:

- Cosa si deve salvare?
- Quanto è grande la mole di dati?

Con che frequenza avviene il salvataggio?

Sulla base delle risposte si stabilisce una appropriata politica di salvataggio, che definisce con che periodicità i dati, in modo parziale o totale, devono essere salvati (es. giorno/settimana/mese)

Sulla base delle risposte si definiscono quindi le caratteristiche del backup, che può essere:

- Incrementale o alle differenze, dove si compie un salvataggio totale ogni tanto, mentre periodicamente si salvano tutti e soli i file modificati rispetto alla data dell'ultimo salvataggio, risparmiando spazio sui supporti di backup e tempo di trasferimento dei dati;
- Totale, ove si salva tutto l'insieme dei dati.

In alcuni casi, specialmente quando sono in esercizio grandi strumenti di storage (NAS, SAN ecc...), come già accenato nel capitolo 5, le moli di dati che devono essere salvaguardate cominciano a diventare piuttosto grandi, nell'ordine dei terabyte o addirittura delle decine di terabyte.

Avendo a che fare con volumi di questo tipo le soluzioni più robuste, ma anche più lente, come i semplici nastri non sono più utilizzabili e si deve ricorrere ad altre strade. La prima è quella di duplicare le strutture di storage, realizzandole in luoghi geografici diversi, garantendo la sincronizzazione dei dati attraverso reti di comunicazione ad alta velocità. Tale sincronizzazione può anche avere luogo in tempo reale, ma normalmente viene compiuta solo periodicamente. Questa soluzione, oltre che salvaguardare i dati, può garantire anche una business continuity, se la seconda struttura di storage può anche prendere il posto della prima in caso di guasti a quest'ultima. Chiaramente questa soluzione presenta un costo notevole e può essere affrontata solo da organizzazioni di dimensioni medio-grandi.

La seconda soluzione è invece data dagli array di nastri, chiamati anche juke-box di nastri. Una categoria particolare di essi è il gestore robotizzato di nastri, che tiene traccia della collocazione fisica di ogni dato salvato al suo interno, ossia della cassetta dove il dato è stato registrato e offre quindi lo stesso tipo di servizio di un filesystem, ma ovviamente molto più lento, visto che ogni accesso ai dati coinvolge movimenti meccanici e il successivo scorrimento del nastro selezionato sino al punto desiderato.

Oltre al backup, comunque necessario per tutti i motivi sopra citati, si stanno affermando anche tecnologie per rendere più robusti i singoli dispositivi di storage, al fine di massimizzare la business continuity. La più importante di tali tecnologie è quella degli array di dischi, meglio nota come Redundant Array of Inexpensive Disks o Redundant Array of Independent Disks (RAID), basata sulla duplicazione dei dati (mirroring) e sulla suddivisione dei dati stessi tra più dischi. Esistono diversi livelli di RAID:

- RAID 0: suddivisione senza ridondanza
- RAID 1: mirroring dei dischi
- RAID 2: mirroring + data check
- RAID 3: sudd. dati (byte)+ parity check su un disco

- RAID 4: sudd. dati (blocchi)+ parity check su un disco blocco
- RAID 5: sudd. dati fra dischi e parity check distribuito

A livello pratico sono normalmente in uso sistemi RAID 1, 2 e 5. In particolare nei sistemi di qualità migliore il RAID è accompagnato dalla caratteristica di hot swap, ossia di potere cambiare il singolo disco guasto dell'array, senza dovere fermare il sistema od alcuno dei servizi in esso contenuti, garantendo un ottimo livello di business continuity. Il servizio RAID può essere realizzato tramite appositi componenti hardware, più efficienti, o via software.

Parallelamente ai dispositivi di storage è avvenuta un'evoluzione dei collegamenti tra storage e computer, così che oggi troviamo diversi tipi di soluzioni:

- Sistemi RAID interni al server, che però in caso di problemi al server diventano inutilizzabili;
- Storage device con “bus” dedicato, come rappresentato in figura 8.2; questi dischi condivisi fra due (o più) server sono alla base dei **cluster** di server, dove anche in caso di guasto ad uno dei server il sistema è in grado di continuare ad erogare i servizi;
- Network Attached Storage (NAS), sistema di storage autonomo, con un sistema operativo ridotto, e connesso direttamente alla rete locale, come rappresentato in figura 8.3;
- Storage Area Network (SAN), batteria di sistemi di storage e di backup autonomi, connessi ai vari server attraverso reti dedicate ad altissima velocità, solitamente in fibra ottica, come rappresentato in figura 8.4; più precisamente, con SAN si intendono sia la rete di connessione fra dispositivi di storage e server, sia, più frequentemente, tutto il sistema.

Per raggiungere livelli più alti di sicurezza (safety) si usano le architetture ridondate, evoluzione dei cluster, in cui ogni componente del sistema (server, storage, rete ecc...) è almeno duplicato, in modo tale da minimizzare le probabilità di guasti. Esempi di architetture ridondate sono riportati nelle figure 8.5 e 8.6. Un altro vantaggio delle architetture ridondate (ma anche dei cluster) è il bilanciamento di carico o load balancing, ossia la capacità di suddividere il carico tra le varie componenti del sistema. Un'ulteriore evoluzione per la business continuity e l'ottimizzazione del carico è un'applicazione basata su un approccio simile al grid computing, il cosiddetto **virtual computing**. Il virtual computing consiste nell'interporre fra l'hardware ed i livelli software soprastanti uno strato ulteriore, chiamato macchina virtuale. La macchina virtuale presenta una emulazione completa di tutto l'hardware di base che forma una piattaforma e viene “percepita” dal sistema operativo installato al suo interno come se fosse una piattaforma hardware reale in tutto e per tutto. L'installare entro un server, sopra il suo sistema operativo di base, un livello di macchine virtuali consente di separare le risorse in modo completo. I sistemi contenuti dentro le macchine virtuali diventano completamente separati fra loro e un eventuale guasto molto grave in una applicazione può, nel caso più estremo, condurre solo al blocco della macchina virtuale in cui essa opera, senza interferire in alcun modo con le applicazioni presenti entro le altre macchine virtuali. In questo modo viene realizzata la completa separazione dei servizi presenti in una stessa piattaforma hardware. Nei sistemi virtuali

di classe enterprise, le macchine virtuali possono essere “mappate” dinamicamente su architetture hardware distribuite e a run-time, senza dovere arrestare in alcun modo i servizi contenuti nelle macchine virtuali, le risorse ad esse dedicate possono essere modificate, ad esempio, attribuendo più RAM così da permettere un buon bilanciamento rispetto al carico. Uno schema di un sistema basato su macchine virtuali è presentato in figura 8.7. Storicamente le macchine virtuali derivano, o meglio, sono ispirate dalle architetture a sottosistemi IBM presenti nei mainframe sin dalla fine degli anni '80.

I software per macchine virtuali più diffusi nella classe base, in cui la riconfigurazione di una macchina virtuale ne richiede il riavvio, sono VMWare Server e VMWare player, entrambi prodotti commerciali ma con licenza gratuita e XEN, prodotto open source. Nella classe enterprise troviamo invece VMWare Enterprise, che non richiede un sistema operativo sottostante in quanto “contiene” un Linux e quindi si installa direttamente sull’hardware e le funzionalità di virtualizzazione dei sistemi UNIX commerciali di classe enterprise come Sun Solaris 10 e IBM AIX 5.x. In particolare questi sistemi si installano solitamente su architetture hardware particolari, chiamate **server blade**, in cui sono presenti molte schede con le CPU e la RAM e lo storage è un NAS o una SAN. In modo analogo alle caratteristiche di hot swap dei dischi, nei server blade è possibile sostituire “a caldo”, senza fermare e riavviare il sistema, le schede delle CPU e della RAM in caso di guasti. E’ il software di virtualizzazione a riconfigurare autonomamente il numero di CPU e la quantità di RAM a disposizione di ogni singola macchina virtuale, permettendo quindi un ottimo livello sia di business continuity sia di load balancing. Lo schema di un server blade è presentato in figura 8.8. Si vedano [BM 2005] e [Morris 2004] per approfondimenti.

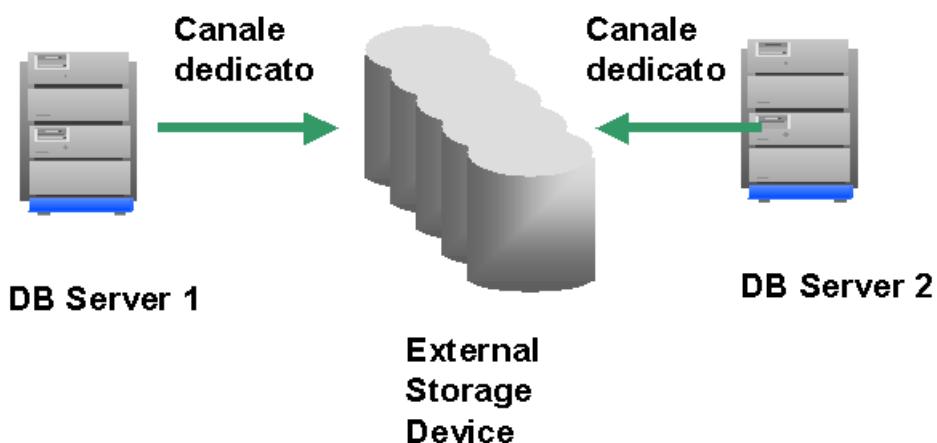


Figura 8.2: Schema di un sistema RAID condiviso fra due server, configurazione tipicamente usata nei piccoli cluster.

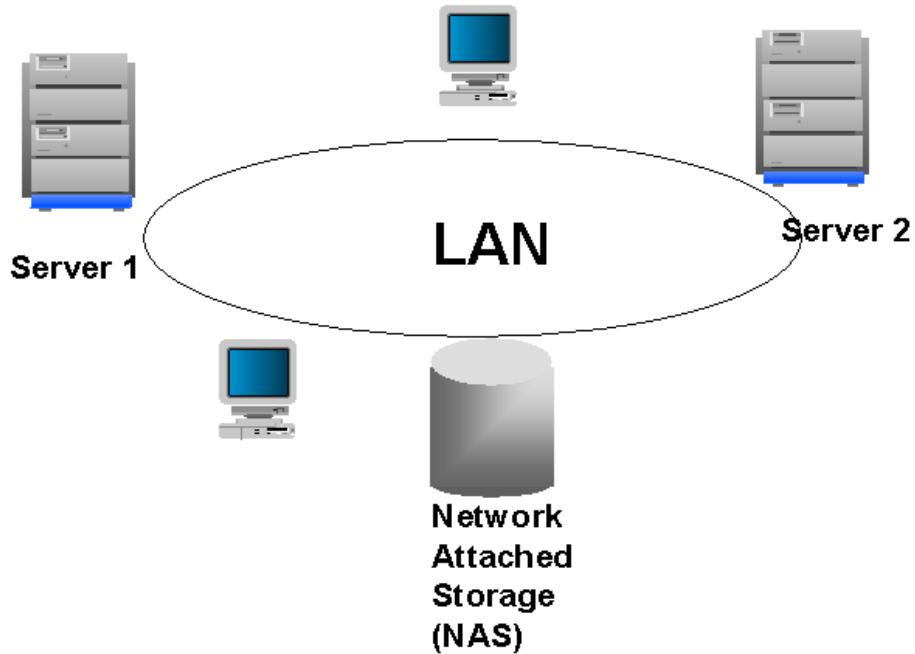


Figura 8.3: Schema di un sistema NAS (Network Attached Storage).

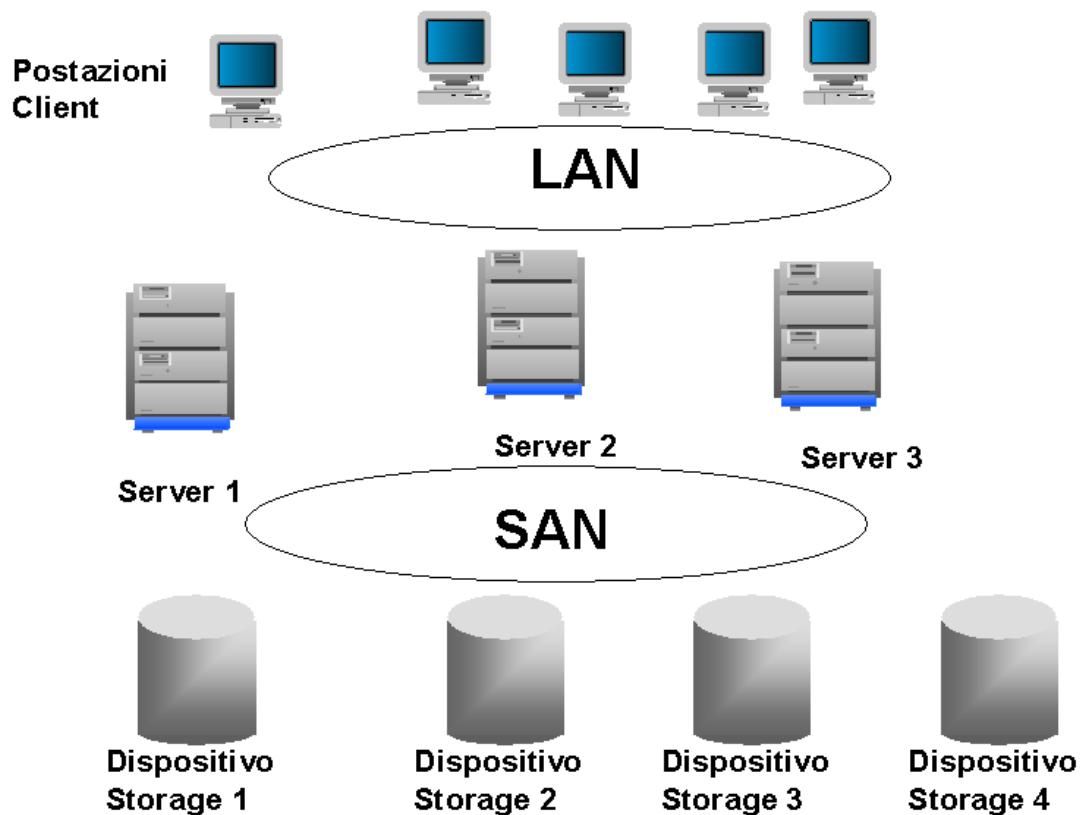


Figura 8.4: Schema di un sistema SAN (Storage Area Network); si ricordi che col termine SAN può essere indicata sia tutta la struttura sia la sola rete dedicata che connette i server ai dispositivi di storage.

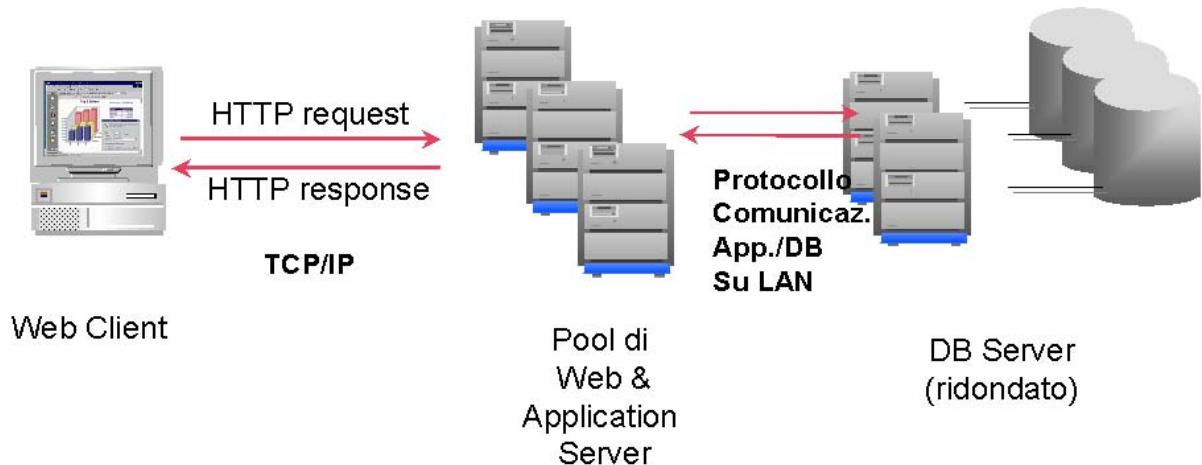


Figura 8.5: Esempio di architettura ridondata, sia nello strato dei server applicativi, sia in quello dei database server; questa struttura consente sia il bilanciamento di carico sia la tolleranza ai guasti.

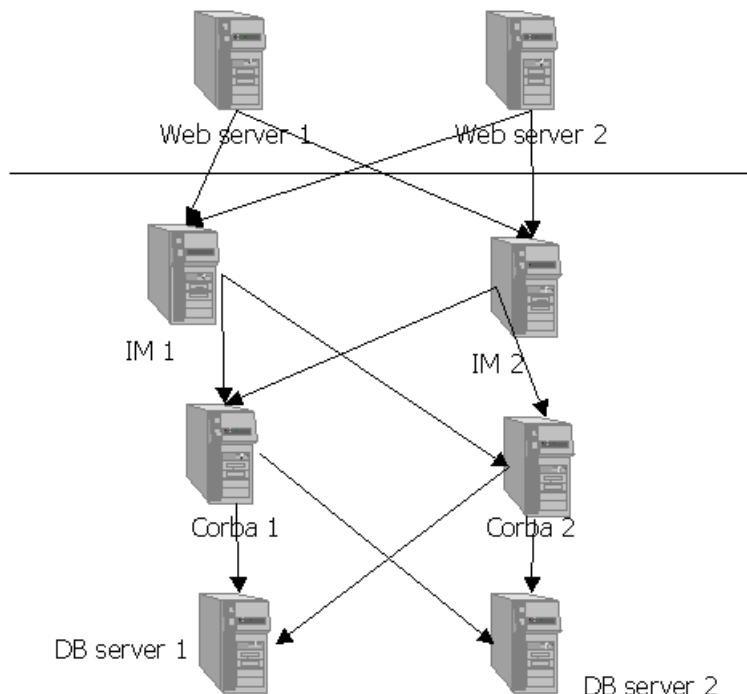


Figura 8.6: Esempio di sistema multi-tier in architettura ridondata a duplicazione totale, in cui ogni strato applicativo viene implementato su due server, ciascuno dei quali può usufruire dei servizi garantiti da ogni server degli altri strati. La linea orizzontale rappresenta la protezione di un firewall posto tra diversi tier, che protegge gli strati inferiori.

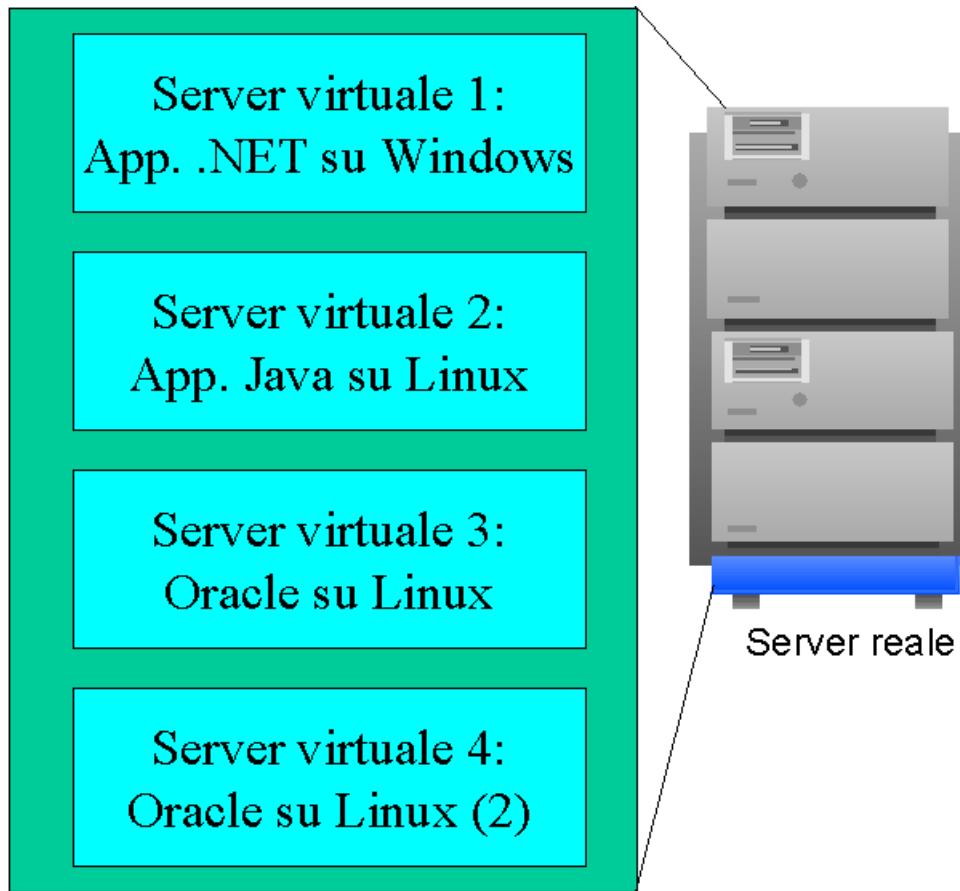


Figura 8.7: Esempio di macchina virtuale sopra un sistema operativo con 4 sottosistemi, ognuno dei quali è dedicato ad una singola applicazione. In particolare troviamo una macchina virtuale con Windows su cui gira un applicativo .NET, una con Linux su cui gira Java e due copie dello stesso Oracle che girano in “cluster virtuale”. Dal punto di vista del resto del sistema le 4 macchine sono indipendenti l’una dall’altra e problemi su una non si ripercuotono sulle altre. Solo problemi fisici del server reale possono avere influenza su tutte le macchine.



Figura 8.8: Alcuni server blade di IBM. Ogni “fetta” rappresenta una macchina fisica completa, su cui, quando è presente, il sistema di virtualizzazione alloca una o più macchine virtuali.

I problemi di security: le minacce umane alla sicurezza

Parlare di **Security** significa descrivere le tecniche di salvaguardia dei dati e dei sistemi che li difendono da attacchi o usi fraudolenti.

Le tipologie di attacco o comunque di uso fraudolento sono molteplici, gli attacchi possono essere diretti, indiretti o addirittura “impersonali”, come avviene, ad esempio, nel caso dei virus informatici. Pertanto, per potere capire meglio le protezioni applicabili, occorre prima definire nei dettagli le possibili categorie di attacco.

Per prima cosa occorre definire cosa si intende per attacco e quale può essere il suo scopo, poi chi è l'attaccante, ovvero quale può essere il suo profilo. Poi occorre classificare gli attacchi per tipologia di attacco e base di partenza, che definisce se un attacco avviene attraverso la rete o direttamente dall'interno di un sistema bersaglio. Infine occorre comprendere il ruolo dei virus informatici, forse il più grave fra tutti i problemi legati alla security.

I singoli tipi di attacco possono essere suddivisi come di seguito presentato:

- **Intrusione in un sistema:** è un accesso non autorizzato ad un servizio informatico (come, per esempio, un database server, un sito Web riservato) presente su un server, o anche su un client, come per esempio nel caso della condivisione dischi di Windows; un'intrusione tra le più frequenti nel mondo Windows è, appunto, l'accesso non autorizzato ad una condivisione dischi, mentre nel mondo UNIX è l'ingresso non autorizzato nel sistema da terminale remoto (telnet o SSH);
- **Impersonificazione:** è l'assunzione non autorizzata di una identità elettronica entro una rete od un sistema, con la possibilità di svolgere le azioni abilitate per tale identità elettronica; un caso particolare è l'impersonificazione di un intero servizio, come quello di una banca, che viene perpetrata spesso in alcuni casi di truffe informatiche;
- **Intercettazione:** è la cattura di dati in transito lungo una rete che vengono copiati ed usati fraudolentemente, intercettando dati sensibili come password e numeri di carte di credito; in taluni casi l'attaccante può anche interferire sui dati stessi, modificandoli; la tecnica di intercettazione più usata nelle reti locali viene chiamata sniffing; la protezione migliore contro questo rischio è la protezione dei dati con tecniche di crittografia, che sarà presentata nei prossimi paragrafi;
- **Abuso:** è l'uso oltre le proprie possibilità autorizzate di risorse messe a disposizione dai sistemi; esempi possono essere l'uso di reti aziendali per scaricare illegalmente programmi, musica o film; un caso molto frequente di abuso è lo spamming, ossia l'invio massiccio di messaggi di posta non richiesti a molti destinatari, che sta diventando una vera e propria piaga di Internet;
- **Denial-of-service:** spesso indicato come DoS, è un tipo di attacco che ha come obiettivo il blocco, anche solo temporaneo, di un servizio o di un intero computer server; sono già stati denunciati casi di attacchi di questo tipo legati a fenomeni di ricatto (cyber-racket) o di veri e propri atti di terrorismo (cyber-terrorismo); il metodo più semplice per ottenere un DoS è il sovraccarico del servizio oggetto dell'attacco;

- **Azione di Virus informatico:** un virus informatico è un programma che compie azioni fraudolente “mimetizzandosi” entro altri programmi e riproducendo autonomamente copie di se stesso, propagandole poi anche attraverso le reti informatiche; ormai esistono molte migliaia di virus, le cui azioni portano a perdite di dati e a fermi macchina con danni per molti milioni di euro ogni anno.

Gli obiettivi per cui un pirata informatico compie un attacco possono essere molteplici e magari coinvolgere più tipologie tra quelle descritte sopra.

- **Accedere a dati riservati**, ossia rubare tali dati, per gli scopi più diversi; da un punto di vista pratico quest’azione si può distinguere in:
 - Entrare in un RDBMS, interrogandolo direttamente per ottenere dati, andando oltre le regole previste per l’accesso ai dati stessi;
 - Entrare in un archivio di posta, leggendo e/o modificando messaggi non propri;
 - Entrare in un archivio documentale;
 - Entrare in un filesystem, accedendo a file senza autorizzazione;
 - Entrare in un directory service od un domain service, catturando dati sensibili, come numeri di telefono o codici personali;
 - Impadronirsi di indirizzi di posta, crimine quasi sempre connesso con il mercato nero degli indirizzi per lo spamming;
 - Impadronirsi di un archivio di chiavi crittografiche, ovvero di certificati digitali, usati per il riconoscimento dell’identità elettronica e/o di username e password;

questi tipi di attacco possono essere svolti in vari modi, dal punto di vista tecnico; le tipologie più frequenti saranno esaminate in seguito.

- **Assumere un’identità**

Gli obiettivi di assumere un’identità elettronica non propria sono molteplici:

- Entrare in un sistema con privilegi non propri, facendo sì che eventuali azioni fraudolente possano essere attribuite a qualcun altro;
- Potere usare a scrocco servizi non propri, casistica estremamente ampia;
- Vedere dati non propri (si ricade nel caso precedente);
- Accedere a risorse finanziarie non proprie (si ricade nel caso seguente);
- Fare uno scherzo, ad esempio inviando una mail a nome di qualcun altro.

- **Effettuare transazioni finanziarie fraudolente**

Il crimine informatico legato a servizi finanziari è sicuramente uno dei più frequenti. Per giungere al furto di denaro elettronico si può volere:

- Rubare il codice carta di credito;
- Rubare codici per l’accesso Home Banking;
- Attaccare direttamente sistemi bancari;
- Usare codici validi per accedere a un sistema e poi assumere privilegi non propri;
- Effettuare azioni interne ai sistemi, si calcola a tal proposito che la maggior parte degli attacchi non sono compiuti dall’esterno di una rete, ma dal suo interno, per esempio, ad opera di dipendenti disonesti;

- Impersonificare interi sistemi, per esempio, costruendo una pagina Web analoga a quella di accesso al sistema di Home Banking che inganna gli utenti; un fenomeno che rientra in questa categoria è il **phishing**, in cui il pirata informatico invia e-mail in apparenza provenienti dalla banca; queste e-mail contengono form Web in cui il cliente è invitato ad inserire i propri dati personali, che vengono in tal modo catturati dal ladro.
- **Usare risorse senza averne diritto**
 - Accesso alla rete esterna entro un'azienda od organizzazione, superando eventuali regole restrittive presenti;
 - Uso di servizi in modo contrario alle regole, per esempio, scaricando film in modo illegale;
 - Spamming
 - Uso dei server come ponte per attacchi a terzi, caso particolare ma sempre più frequente, in cui la macchina oggetto dell'attacco non è in realtà l'obiettivo finale del pirata informatico, ma semplicemente viene usata come ponte per l'attacco verso il vero obiettivo; un caso particolare è il Distributed Denial of Service (DDoS), che sarà spiegato nel paragrafo dedicato ai virus.
- **Mandare fuori servizio un sistema (DoS)**
 - Bloccare un servizio (programma server)
 - Bloccare un server (computer server)
 - Bloccare la rete
 - Bloccare/alterare il DNS
- **Mostrare al mondo quanto si è bravi entrando nei sistemi altrui**

L'obiettivo dell'ingresso può essere un ricatto (cyber-ricatto) od un'azione distruttiva (es. cyber-terrorismo), ma può anche essere solo dimostrativo, sia, semplicemente, per dimostrare la propria competenza, sia per, un domani, potersi proporre sul mercato come consulente di sicurezza; una evoluzione di questo è il servizio di **ethical hacking**, in cui esperti informatici, in accordo con i responsabili di un sistema informatico, tentano di violarne la sicurezza e poi stendono una precisa relazione sulle vulnerabilità riscontrate, in modo da poterle poi correggere.

I pirati informatici possono appartenere a varie categorie, come elencato nel seguito.

- **Hacker:** nonostante la connotazione negativa data negli ultimi tempi a questo termine, hacker indica un esperto di informatica che per motivi non fraudolenti, primo fra tutti quello di acquisire maggiore conoscenza, effettua prove, fra cui i tentativi di accesso ai sistemi, magari con l'ultimo obiettivo, definito sopra, di proporsi poi come consulente all'azienda od organizzazione che possiede i sistemi oggetto del suo attacco.
- **Cracker:** è sostanzialmente sinonimo di pirata informatico, che cerca di entrare nei sistemi o di intercettare dati per scopi fraudolenti, spesso per puro obiettivo distruttivo, senza secondi fini.
- **Spia:** il suo obiettivo è il furto di dati, per i motivi più vari.

- **Sabotatore:** il suo obiettivo è creare danni in un sistema, dal rallentamento sino al denial of service, per i motivi più vari.

L'attacco non sempre è diretto ad una struttura precisa, come, ad esempio, un'azienda o una banca, ma spesso può essere “impersonale”. Esistono infatti hacker e cracker che ricercano sistemi vulnerabili entro le reti informatiche pubbliche come Internet, per gli scopi sopra detti. In pratica quindi il loro obiettivo è scoprire le vulnerabilità di un sistema ed entrarvi prima ancora di sapere se effettivamente dentro il sistema ci saranno dati di interesse o la possibilità di fare un ricatto. In particolare poi impersonale è quasi sempre il ruolo dei pirati informatici che creano i virus, la cui azione attraverso le reti potrà poi estendersi a livello planetario, attaccando sistemi di cui il creatore del virus non conosce nemmeno l'esistenza.

Ma come può avvenire un ingresso non autorizzato in un sistema informatico, oppure una intercettazione dei dati? I sistemi informatici sono strumenti complessi ed hanno molti punti vulnerabili, come parzialmente elencato nel seguito

- **Vulnerabilità dei dati**

I dati, per quanto rappresentati con formati proprietari, possono comunque essere ricostruiti da parte di chi vi accede illegalmente. Pertanto, da un file catturato entrando in un disco od intercettato in una comunicazione via rete possono essere estratte informazioni preziose.

- **Vulnerabilità dei programmi applicativi**

I programmi applicativi sono ormai estremamente complessi e spesso vengono rilasciati sul mercato dopo essere sottoposti a test non esaustivi, col rischio concreto di errori e vulnerabilità ancora presenti in essi. Queste problematiche valgono anche per i programmi in grado di connettersi in rete come client (per esempio, il browser) e possono essere sfruttate dall'esterno, tipicamente da siti a cui ci si connette, per usare in modo fraudolento il PC che ospita tali programmi.

- **Vulnerabilità dei programmi server**

In questa sede si intende per programmi server l'insieme dei programmi in grado di accettare connessioni per richieste di servizio attraverso la rete. Anche per essi vale lo stesso problema di tutti gli altri programmi applicativi. Eventuali errori o vulnerabilità presenti possono essere sfruttati da attaccanti che, una volta ottenuta la connessione, li sfruttano per compiere azioni fraudolente sui server.

- **Vulnerabilità dei sistemi operativi**

Anche i sistemi operativi sono composti di programmi e vale quindi lo stesso principio per le loro componenti server, o per i programmi di supporto in essi presenti. In particolare, entro i sistemi operativi sono presenti librerie condivise comunemente usate da molti programmi. Un'azione spesso perpetrata dai pirati informatici che riescono ad avere accesso ad un sistema consiste proprio nel manomettere tali librerie.

- **Vulnerabilità dei sistemi fisici**

Sino a tempi recenti, la vulnerabilità dei sistemi fisici non costituiva un problema importante, in quanto era sufficiente erigere protezioni fisiche intorno

ai sistemi, come, per esempio, chiudendo con una porta blindata la stanza server o usando cavi corazzati, per eliminare in modo pressoché completo i pericoli di tali vulnerabilità. In tempi recenti però, con il diffondersi delle reti wireless, il problema della vulnerabilità fisica è diventato molto più grave. Un canale radio può sempre essere intercettato. Inoltre, disponendo di strumenti molto costosi, e trovandosi molto vicino ad un computer o ad un cavo di trasmissione, è anche possibile intercettare direttamente la radiazione elettromagnetica a bassissima potenza emessa direttamente dagli apparati elettronici che lo costituiscono e ricostruire i bit associati a tale radiazione, ricostruendo le informazioni presenti all'interno.

- **Vulnerabilità delle trasmissioni**

Le trasmissioni che passano lungo reti wireless o anche che passano attraverso canali pubblici sono per definizione insicure e soggette al rischio di intercettazione e/o alterazione dei dati in transito.

Ed ora si descriveranno dettagliatamente alcuni tra i più diffusi di questi tipi di attacchi. L'elenco presentato non è assolutamente esaustivo e si rimanda a [Klander 1998] e [MSK 2005] per approfondimenti.

Attacchi alle reti: intercettazione

La maggior parte delle reti locali, sia basate su cavo, sia wireless, usano un meccanismo interno detto broadcast (trasmissione uno-a-tutti) che rende possibile per i pirati l'intercettare tutto il traffico in transito, attraverso la riconfigurazione di una scheda di rete effettuata con opportuni software detti sniffer. L'azione fraudolenta è rappresentata schematicamente in figura 8.9.

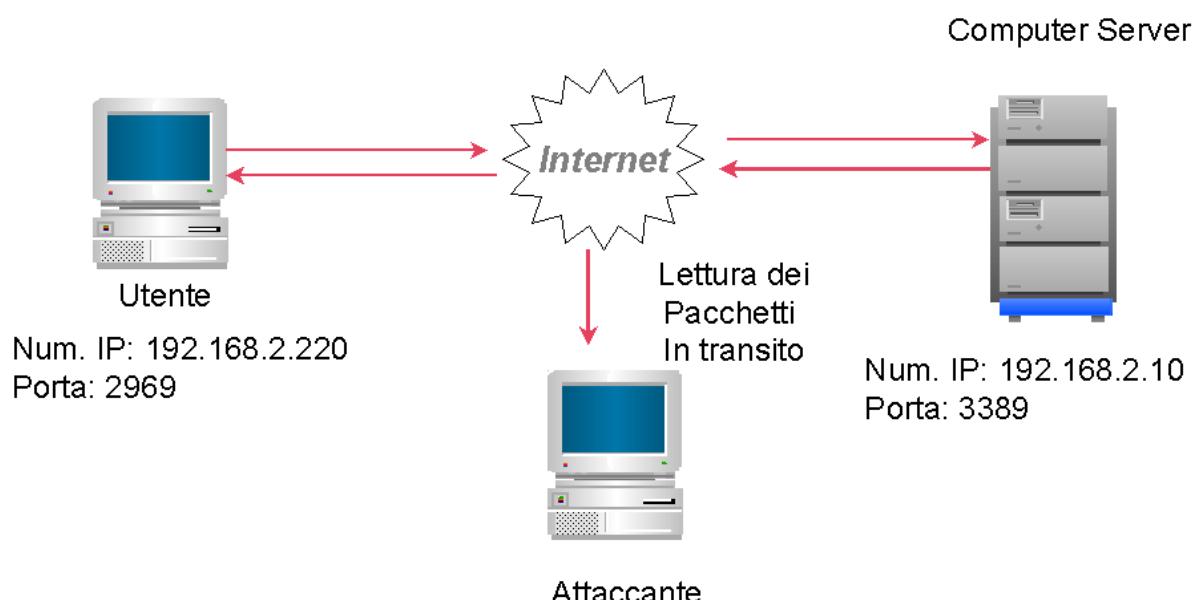


Figura 8.9: Intercettazione dei dati in transito (in gergo sniffing)

Attacchi alle reti: dirottamento di sessioni

Sfruttando alcune proprietà interne del protocollo IP su cui si basa Internet, è possibile un tipo di attacco in cui il campo indirizzo mittente dei pacchetti trasmessi viene falsificato il valore (IP Spoofing). L'attacco potrebbe servire per superare protezioni, ma un uso particolare della combinazione di esso con la intercettazione è il dirottamento di sessione, schematizzato in figura 8.10. Individuando i dati di una connessione legittima e falsificando l'IP si “subentra” alla sessione valida, estromettendo l'utente legittimo.

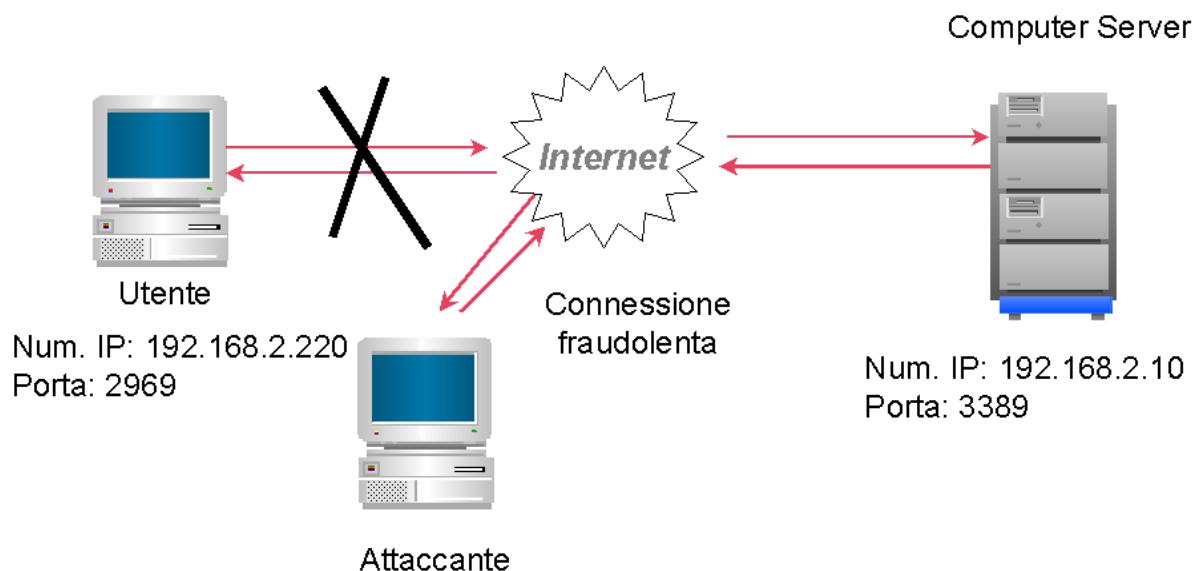


Figura 8.10: Dirottamento di sessioni

Attacchi diretti ai server TCP/IP

I servizi TCP/IP sono oggi il cuore di tutti i sistemi client-server e peer-to-peer che operano sulle reti. Un programma server TCP/IP è un applicativo che apre un server socket su una data porta IP e si pone in attesa di connessioni IP (ascolto), attendendo le comunicazioni sotto forma di flussi di byte, come spiegato nel capitolo 5. Una volta stabilita la connessione il client invia un certo numero di byte al server (richiesta), che risponderà con una stringa di byte e così via.

Sia i protocolli tipici di Internet (HTTP, FTP, SMTP ecc...), sia quelli proprietari si basano per il loro funzionamento su scambio di stringhe di byte fra client e server. In alcuni casi sullo stesso socket passano sia comandi sia dati, in altri casi (es. FTP) i dati e i comandi viaggiano su canali separati.

Ma cosa accade se vengono aperte 10000 o 100000 connessioni contemporanee con un servizio? Il computer server che lo ospita spesso non è in grado di sopportarle. Questo ultimo caso è l'attacco di sovraccarico, il cui obiettivo è di solito il mandare fuori servizio il sistema oggetto dell'attacco. Questo è un esempio dell'attacco Denial-of-Service o DoS, definito all'inizio del paragrafo.

Un'altro attacco più subdolo è il **buffer overflow**, costituito, in pratica, dall'invio ad un server TCP/IP di un numero di byte superiore a quanto previsto dal protocollo. Qualora, come purtroppo avviene in moltissimi casi, non siano previsti controlli sulla

lunghezza delle stringhe ricevute, i byte ricevuti in più “sfondano” i confini dell’area di memoria riservata e “passano” in altre parti del programma o addirittura oltre il programma stesso. Nella parte “in più” vengono inseriti comandi nella rappresentazione binaria del sistema del server, che il programma server esegue direttamente o “passa” al sistema operativo, causando le azioni fraudolente in essi prescritte.

Il programma SendMail (il primo server di posta elettronica basato sul protocollo SMTP) è stato nominato “il più bacato della storia” per i buffer overflow presenti al suo interno. Ad esempio, inviando comandi opportuni ai primi server di questo tipo era possibile ottenere una sessione di root su macchine UNIX, prendendo il controllo totale di tali macchine.

La prima versione del modulo TCPBios delle macchine Windows95, che implementava la condivisione dischi e stampanti fra macchine Windows attraverso TCP/IP, era molto vulnerabile al buffer overflow. Inviando un pacchetto di dimensioni superiori al previsto e contenente le istruzioni opportune sulla porta 139 usata da TCPBios di un PC Windows95, era possibile provocarne il blocco immediato (attacco noto anche come WinNuke). Tale problema è stato corretto rapidamente in uno dei primi Service Pack (ossia aggiornamento di sistema) rilasciato da Microsoft.

Un altro esempio è il ping of death, in cui si produceva il blocco del sistema oggetto di attacco con l’invio di un messaggio ping, normalmente usato semplicemente per verificare l’esistenza di una connessione valida TCP/IP fra sorgente e destinatario, più lungo del previsto.

Una variante del buffer overflow è l’invio dei comandi non previsti, tipico, ad esempio, del mondo Web, che consiste in pratica nell’inviare insiemi di byte di lunghezza lecita rispetto ai limiti prescritti dal protocollo, ma contenenti comandi/stringhe non previste da quest’ultimo. In tal modo si può arrivare anche al blocco del servizio o del sistema.

Attacchi tipici della posta elettronica (SMTP)

La posta elettronica di Internet (o delle Intranet) usa il protocollo SMTP o il suo successore ESMTP. I server di posta elettronica, basati su TCP/IP, sono, come già detto, soggetti a tutti i problemi comuni ai server TCP/IP, ma hanno in più alcuni problemi particolari:

- **Mail Overflow:** la capacità delle caselle di posta è, di solito, limitata ed inviando alcuni messaggi particolarmente voluminosi si può arrivare a saturare tale capacità, per cui il sistema risulta incapace di accettare nuove mail sinchè non viene vuotata la casella;
- **Spamming:** l’invio di posta non desiderata, vera piaga di Internet, può comportare due pericoli distinti:
 - innanzitutto il sovraccarico dovuto alla ricezione di tanti messaggi indesiderati, con conseguente rallentamento del funzionamento e pericolo di saturazione delle caselle di posta;
 - inoltre ogni server di posta può comportarsi da relay, ossia da instradatore per posta non a lui direttamente destinata, proprio come un ufficio postale

“fisico”, inviandola ad altri server di posta; se non vengono inseriti adeguati controlli un server di posta può venire usato per invio di posta non autorizzata a tutti i server cui egli può inviare messaggi; normalmente i server di posta sono abilitati a spedire posta solo se questa viene ricevuta da indirizzi e-mail e/o IP fidati;

- **Impersonificazione e-mail:** nella versione base il protocollo dell'e-mail non verifica l'indirizzo del mittente, per cui, connettendosi a un server e-mail e dando i comandi opportuni, si possono inviare mail anonime o a nome del mittente voluto; esistono programmi in grado di rendere facilissima questa operazione anche per pirati privi di nozioni tecniche.

Attacchi tipici del mondo Web (HTTP)

Essendo basato su TCP/IP, anche il servizio Web è soggetto a tutti i pericoli tipici dei server TCP/IP sopra descritti. Accanto ad essi esistono alcuni attacchi tipici.

Il servizio Web mette a disposizione di chi accede un'area del disco del server, ove stanno i documenti e le pagine HTML che vengono distribuite. Un attacco è quello che ha come obiettivo il costringere il server a visualizzare anche file appartenenti ad aree esterne, come, per esempio, il file delle password di sistema. Nel Web odierno inoltre la maggior parte dei server sono dinamici, ossia generano pagine HTML a partire da dati reperiti in un database, come nello schema presentato in figura 5.6 o attraverso integrazioni con il resto del sistema informatico aziendale, come visualizzato in figura 5.7. L'obiettivo dell'attacco divene allora il “superamento” del Web server per accedere direttamente alle fonti di informazione presenti a monte di esso, agendo sulle componenti del sistema che stanno in mezzo, soprattutto attraverso l'invio di comandi non previsti dal protocollo.

Si possono quindi definire tre tipologie di attacco:

- diretto al web server, per esempio per provocare un DoS;
- diretto alle componenti dinamiche applicative, per DoS o per fare compiere ad esse azioni fraudolente;
- diretto alle componenti a monte (dette talvolta di back-end), come, per esempio, il database o direttamente l'Application Server.

Un esempio di attacco, basato su comandi non previsti, è il “Translate:f” del server MS IIS, in cui, prima che l'errore nel programma venisse corretto, attraverso l'invio della stringa “translate:f” come parametro al termine dell'indirizzo Web, era possibile ottenere la visualizzazione del file di configurazione Global.asa, e con esso conoscere la struttura presente oltre il Web server e in alcuni casi anche username e password per l'accesso diretto al database.

Attacchi diretti ai sistemi operativi

Esistono essenzialmente due famiglie di sistemi operativi: quelli per i server e quelli per i client. Ciò non significa necessariamente che gli attacchi siano tutti specifici per gli uni o per gli altri, ma, essendo normalmente l'obiettivo dell'attacco diverso, tale sarà anche il tipo di attacco.

Per le macchine client infatti gli attacchi hanno di solito l'obiettivo di trarre vantaggio da informazioni in esse presenti o di prendere il controllo della macchina per usarla come testa di ponte per un attacco verso altre infrastrutture informatiche. Inoltre gli attacchi vanno distinti fra locali (ossia perpetrati agendo direttamente sulla console della macchina) e remoti (perpetrati attraverso la rete).

Gli attacchi locali si possono raggruppare in alcune categorie:

- Ingresso diretto non autorizzato, attraverso il furto o la scoperta della password; in tale categoria rientra anche la disattivazione dello screen-saver, normalmente protetto da password; poiché nella stragrande maggioranza dei casi, l'utente principale della macchina ha anche privilegi di amministratore, una volta superata la barriera la macchina è alla mercé dell'attaccante;
- Azioni dirette sul sistema, per esempio installazione di programmi fraudolenti, quale il temibile keystroke logger o intercettatore tasti, che cattura in un file nascosto tutte le battiture di tasti e quindi i caratteri immessi nella tastiera, compresi anche tutti gli username e password, mettendoli a disposizione del pirata che lo ha installato;
- Lettura password dalla memoria, in Windows e in Unix le password sono presenti in opportuni file, che possono poi essere copiati e usati per un attacco a forza bruta, che prova a caso o seguendo l'ordine di un dizionario innumerevoli password, confrontandole con le versioni crittografate e memorizzate nei file stessi.

Accanto agli attacchi locali, anche le macchine client sono soggette agli attacchi perpetrati attraverso la rete, suddivisibili in queste categorie:

- Connessione diretta a risorsa condivisa, quale disco, stampante, periferica esterna, ottenuta attraverso il furto di una password o perché la risorsa non è protetta da password;
- Vulnerabilità di applicazioni server eventualmente presenti (ossia di applicativi che accettano connessioni attraverso la rete); come visto prima, ogni applicazione server, e quindi anche quelle di condivisione periferiche, possono avere le loro debolezze, sfruttabili dagli attaccanti;
- Installazione di un trojan (backdoor), ossia di un programma che consente il controllo remoto di una macchina, appartenente ad una categoria particolare dei virus informatici, trattati nel prossimo paragrafo;
- DoS, attraverso buchi di sicurezza come il sovraccitato ping of death.

Inoltre le piattaforme client sono le più soggette alla nefasta azione dei virus informatici, che saranno trattati nel prossimo paragrafo.

Gli attacchi verso i server hanno, oltre a quelli in comune con i client, anche ulteriori possibilità. Occorre distinguere fra il mondo Windows ed il mondo Unix. Altre piattaforme sono molto meno soggette ad attacchi.

Nel mondo Windows l'attaccante può tentare varie azioni, come riportato di seguito.

- Il crack (ossia la individuazione fraudolenta) delle password, azione che si può effettuare intercettando la rete e usando un programma come L0phtCrack; oppure alternativamente è possibile con comandi **net** di Windows, che

governano le condivisioni, cercare un utente valido di risorsa condivisa e tentare di trovarne la password:

- La ricerca dell’utente Administrator o di altri utenti con i privilegi di amministratore, per poi agire sui file (es. sostituendo una utility di sistema) e/o installare applicativi fraudolenti nel sistema o comunque prenderne il controllo.
- L’attacco al registry, ossia l’archivio fondamentale di configurazione di Windows, per cambiare qualcosa nelle configurazioni della macchina.
- L’attacco a file di sistema al SAM (Security Account Manager, il file crittato delle password).
- L’intercettazione dei tasti, attraverso l’installazione di un keystroke logger nell’avvio automatico, come già visto anche per i client.
- L’accesso al sistema via terminal server o telnet, per potere avere una interfaccia utente completa da remoto e/o eseguire comandi da remoto (attraverso le Remote Procedure Call o RPC), qualora questa possibilità sia attivata nel sistema oggetto dell’attacco. In particolare si deve considerare l’azione del comando runas, che rende possibile eseguire un comando a nome di qualcun altro solo con l’immissione della password; pertanto, se si entra in un sistema e ci si impadronisce della password di Administrator, diviene possibile fare tutto senza nemmeno collegarsi direttamente come Administrator, con meno possibilità di destare sospetti qualora i collegamenti degli amministratori vengano registrati.

Inoltre occorre considerare anche il servizio di domain server del mondo Windows2000/2003 server, ossia le Active Directory. Nelle Active Directory è il deposito centrale di informazioni di un dominio. Un’analisi delle Active Directory è quindi in grado di trarre una mole enorme di informazioni. Un programma, ad esempio Visual Basic, che agisce da una macchina facente “legalmente” parte di un dominio, può facilmente aprire una connessione con le Active Directories e leggerne il contenuto. Inoltre, se si accede alle Active Directories in scrittura, è possibile danneggiare in modo irreparabile un dominio, a meno che non esistano backup del contenuto dell’archivio delle Active Directories stesse. In ogni caso l’operazione di ripristino delle Active Directories è sempre lunga e rischiosa e per tutto il tempo necessario il controllo di accesso centralizzato al dominio è fuori servizio.

Nel mondo UNIX occorre distinguere fra due categorie di attacco.

Esistono attacchi “interni” al sistema (perpetrati attraverso la shell, ossia l’interfaccia principale dei comandi, che però non richiede necessariamente all’utente di essere sulla console della macchina, in quanto accessibile normalmente da remoto); tra questo tipo di attacchi troviamo alcuni presentati in seguito (molti dei quali presuppongono però l’aver ottenuto privilegi di amministratore di sistema):

- Lettura del file delle password (ed eventuale suo cracking)
- Cambio delle configurazioni (di ambienti, programmi ecc...)
- Intercettazione dei tasti (ossia l’installazione di keystroke logger)
- SUID dei file locali (ossia superamento delle barriere di accesso legate ai privilegi impostati nel sistema);
- Attacchi alla shell stessa;

- Uso improprio dei segnali, comandi trasmessi ai processi in esecuzione;
- Attacchi al kernel, che possono arrivare alla sua modifica fraudolenta.

La maggior parte degli attacchi è invece “esterna” al sistema ed avviene direttamente attraverso la rete, sfruttando i servizi aperti su TCP/IP presenti nel sistema stesso:

- Forza bruta per indovinare una password (nelle configurazioni di default non è di solito previsto un numero massimo di tentativi di accesso al sistema quando si sbaglia la password);
- Buffer overflow per attaccare un programma server;
- Server X, il server dell’interfaccia grafica X-Window, se non opportunamente configurato, offre notevoli vulnerabilità, che vanno dalla capacità di eseguire programmi in remoto alla possibilità di attivare un keystroke logging;
- Servizi TCP/IP con particolari vulnerabilità, come, ad esempio, i servizi FTP e TFTP per il trasferimento di file tra computer;
- RPC (come nel caso di Windows), l’esecuzione diretta di comandi da remoto, se tale servizio non viene disattivato;
- NFS, il sistema di condivisione cartelle di UNIX;
- Samba, il sistema di condivisione cartelle tra UNIX e Windows.

Le tipologie di attacco viste in questi brevi elenchi non possono essere esaustive, si raccomanda la consultazione di [Klander 1998] e [MSK 2005] per approfondimenti.

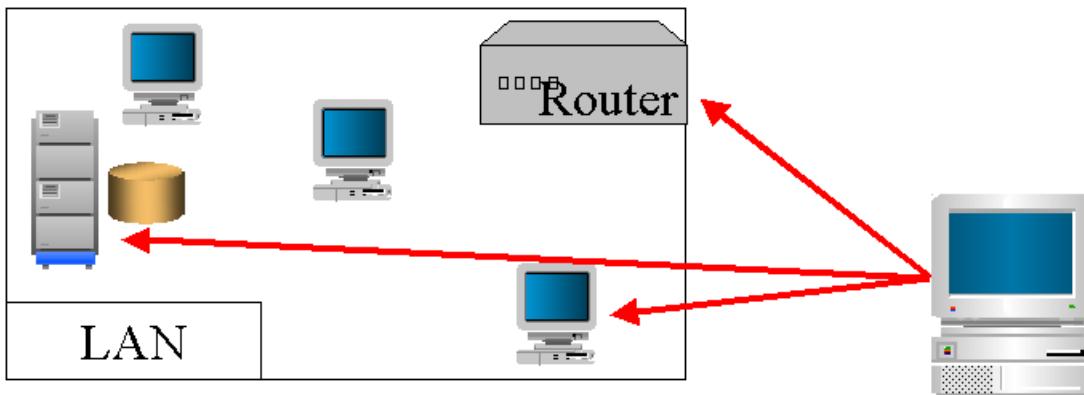
Attacchi alle reti

L’attacco globale ad una rete di computer può essere ottenuto sfruttando un nodo client o server della rete, ovvero un servizio su di esso operante, come visto in precedenza, oppure agendo direttamente sulle infrastrutture di rete, in primo luogo i router, gli apparati che instradano i pacchetti tra le reti.

I router ospitano un sistema operativo, per esempio nel caso di CISCO, l’azienda leader di mercato nel settore, tale sistema prende il nome di IOS. Per quanto molto meno vulnerabile di un server, anche un router, specialmente in occasione di nuove versioni del suo sistema operativo o di errori di configurazione, può venire attaccato e consentire l’ingresso ai pirati. Si sono in passato verificati anche casi clamorosi di configurazioni ingenue (ad esempio scelta di username e password facili da indovinare su molti router), che hanno portato a grandi problemi. Uno schema dell’attacco ad una rete è presentato in figura 8.11.

L’attacco normalmente non è “occasionale” ma viene compiuto da pirati informatici molto abili ed accuratamente pianificato, attraverso varie fasi. Un esempio di attacco in più fasi è:

- Footprinting e ricostruzione, fasi tese a scoprire la struttura generale della rete;
- Ricerca di punti critici
- Portscan di router e server, in cerca dei servizi su di essi installati;
- Attacco a un nodo, per ottenere in esso un ingresso ed installare uno sniffer;
- Sniffing, intercettazione di tutto quanto passa nella rete.



Attaccante

Figura 8.11: L’attaccante di una rete può tentare l’attacco al router, ai server interni alla rete o ai client per usarli come testa di ponte.

Attacchi molto gravi

La manomissione del Domain Name Server può condurre al dirottamento dei collegamenti che agiscono in base al nome logico dei server (tipicamente i collegamenti Web), come rappresentato in figura 8.12.

La manomissione del server di dominio può abilitare il pirata informatico a fare qualsiasi cosa entro il dominio, come rappresentato in figura 8.13.

La manomissione di un router può condurre al dirottamento di ogni collegamento, oltre che essere la testa di ponte per l’invasione di una rete, come schematizzato in figura 8.14.

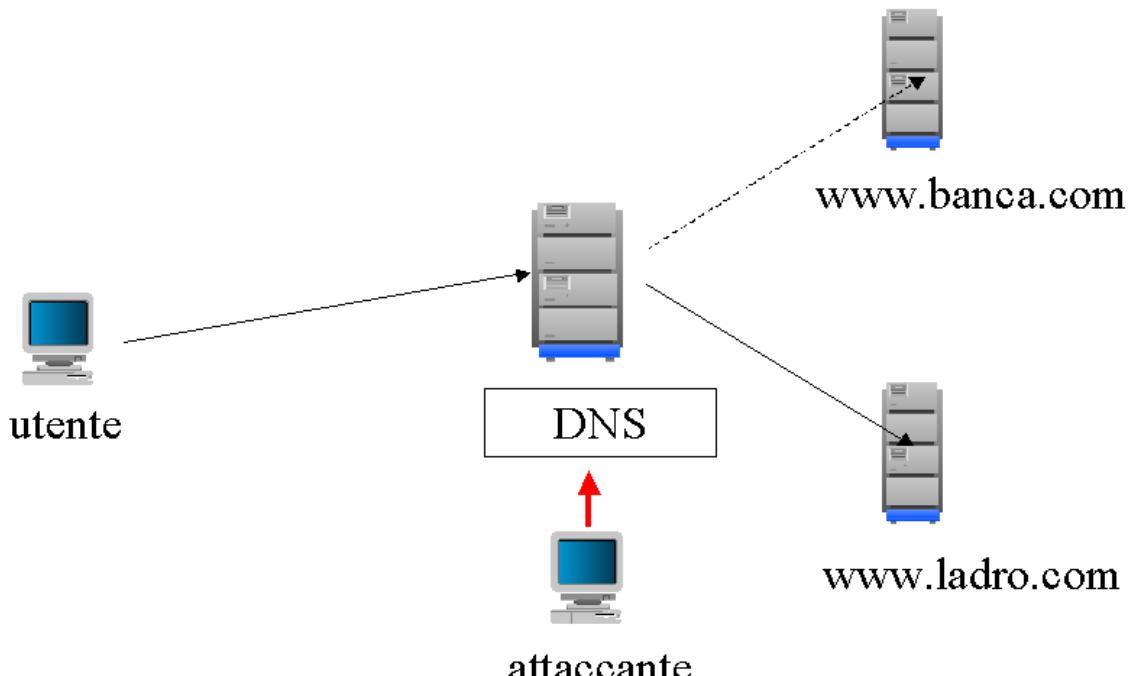


Figura 8.12: Manomissione o hacking del DNS, in seguito alla quale l’indirizzo logico www.banca.com viene tradotto nel numero IP del server www.ladro.com.

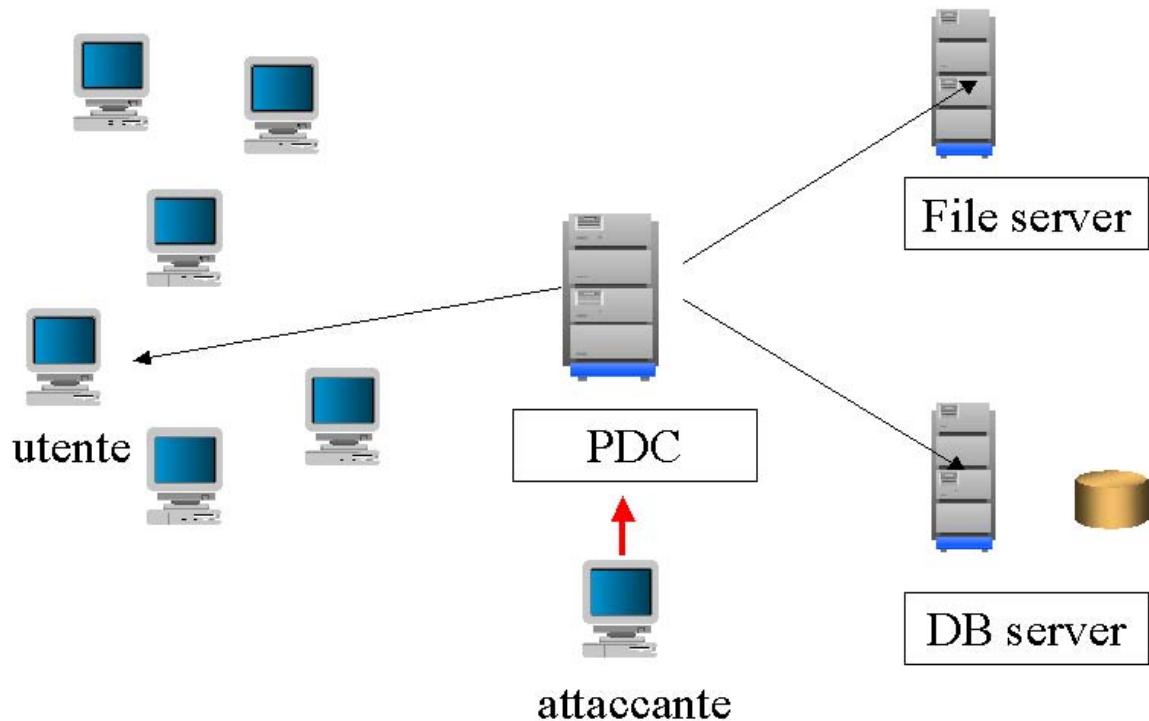


Figura 8.13: Manomissione o hacking del domain server, in questo caso un Primary Domain Controller Windows, in seguito alla quale, per esempio, l’attaccante può assumere l’identità di un utente legittimo e con essa accedere al file server e/o al database server.

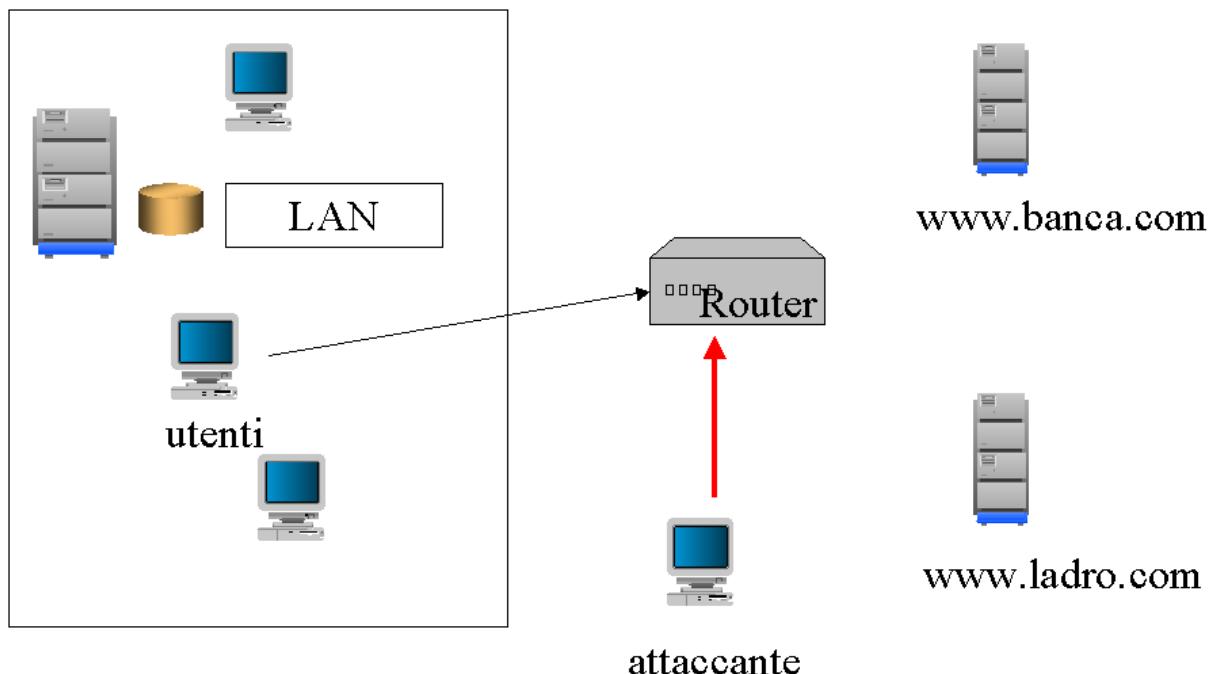


Figura 8.14: Manomissione o hacking di un router, in seguito alla quale le sessioni sono deviate direttamente a livello di rete.

Attacchi all'organizzazione aziendale

Accanto a tutti questi attacchi, diretti alle infrastrutture tecnologiche, esiste una categoria di attacchi classificabili come “organizzativi” in quanto sfruttano vulnerabilità della organizzazione aziendale. Questa categoria di attacchi viene spesso definita “social engineering” e presuppone lo studio dell’organizzazione aziendale per individuarne i punti deboli, di solito costituiti da risorse umane. Ad esempio, una volta noto il nome dell’amministratore di sistema e il numero telefonico diretto di un impiegato, si può telefonare a lui spacciandosi per l’amministratore di sistema e cercare di ottenere così dati che consentano l’accesso alla rete come username e password. Una variante di questo attacco è il “phising”, in cui viene spedita una mail, completa di intestazione imitata in modo perfetto, ai clienti di una banca con l’apparente indirizzo di un ufficio della banca come mittente e la richiesta di inviare i propri dati, comprensivi di username e password, ad un sedicente indirizzo appartenente alla banca stessa o di inserirli in una form web già presente nel messaggio, inviandoli in realtà in mano al pirata informatico vero mittente del messaggio stesso.

Un altro problema legato all’organizzazione più che a motivazioni tecniche è la sicurezza della password. Una definizione di password presente nel Dizionario dell’Hacker (si veda [TNHD 2006]) è “combinazione del nome della fidanzata, dell’amante o dei figli o degli animali domestici e di date di nascita”. Tale definizione chiarisce la situazione: la password viene troppo spesso scelta facile da indovinare per chi conosce l’utente che la usa e la presenza di strumenti come i generatori automatici di password basati sui dizionari rende facile l’indovinarla anche per pirati che non conoscono assolutamente l’ignara vittima. Per questo i nuovi regolamenti e lo stesso Decreto Privacy del 2003 (si veda [DL196 2003]) stabiliscono alcuni criteri di sicurezza minima che la password deve avere, come, ad esempio, lunghezza minima, alternanza di caratteri maiuscoli e minuscoli, numeri ed altri caratteri presenti, non essere direttamente una parola del dizionario ecc...

I problemi di security: il ruolo dei virus

Un **virus informatico** (di solito indicato semplicemente come virus) è un insieme di istruzioni comprensibili dal computer che svolgono un’attività dannosa e/o fraudolenta.

Il virus solitamente non compare direttamente sotto forma di file, ma si “mimetizza” entro programmi e/o documenti che ne risultano “infettati”.

Nel panorama della sicurezza informatica i virus costituiscono senz’altro la minaccia più grave, che statisticamente provoca i peggiori danni e le maggiori perdite economiche sotto forma di fermi macchina e/o perdite di dati. Inoltre i virus delle ultime generazioni sono divenuti temibili come agenti di attacco, che i pirati sfruttano per condurre attacchi del tipo descritto prima. Per questi motivi è importante conoscere nei dettagli le tipologie di virus per capire meglio e potere valutare la minaccia da essi rappresentata. Occorre ricordare che in un PC odierno, avente normalmente molti

programmi in esecuzione ad un dato istante, una volta lanciato un programma, è praticamente impossibile verificarne l'esecuzione e le azioni da essa compiute.

Una prima classificazione dei virus è fatta dal punto di vista realizzativo, ossia in base a come i virus vengono realizzati dai pirati informatici.

Virus degli eseguibili

Sono i virus più antichi (sono infatti apparsi nel mondo DOS negli anni '80). Infettano i file eseguibili (EXE, COM) o le loro librerie (OVL, DLL). Sostituiscono il proprio codice a parte del codice del programma e quasi sempre non aumentano la dimensione o modificano la data del programma infettato. Il contagio può interessare solo i file dei programmi mandati in esecuzione durante l'attività del virus o, indiscriminatamente, tutti i file dei programmi presenti sul disco fisso o sui dischetti non protetti in scrittura inseriti nel computer

Virus del Boot Sector

Storicamente derivano dai precedenti, ma infettano (anche) il boot sector, la parte di disco dove stanno le istruzioni che provvedono al caricamento del sistema operativo, perciò il virus viene sicuramente caricato in memoria insieme al sistema operativo. Per eliminarli serve effettuare un avviamento del computer infettato con una copia non infetta del sistema operativo.

Virus del BIOS

Attaccano il BIOS, ossia la memoria permanente programmabile (EPROM) dove sono scritte le istruzioni di avviamento e la configurazione del computer. Possono provocare il blocco hardware e per riparare i danni occorre sostituire la EPROM (cosa non facile nel caso di computer più vecchi di un paio di anni).

Virus scripting

Infestano i file .BAT o i file di scripting del sistema operativo (.WSH, .JS, .VB). Sono comandi scritti in chiaro o crittografati, contenuti come testo entro tali file. Possono essere scoperti (anche se magari non decifrati) ispezionando i file.

MacroVirus

Sono la famiglia più in espansione di tutte. Sono costituiti da istruzioni fraudolente scritte nei linguaggi di macro di programmi applicativi, e poste entro documenti. Si attivano all'apertura del documento infetto con la applicazione ad esso collegata. Infestano tutti i documenti di MS-Office, ma anche file PDF, PostScript e file di dati di altri programmi dotati di macro (es. StarOffice/OpenOffice). Non sempre sono tipici di un'applicazione ma possono estendersi anche ad altre (ad esempio, a tutti i programmi di Office).

Possono agire anche all'esterno dei documenti infestati (ad esempio, con chiamate a sistema operativo o ad altre applicazioni) e la loro azione può essere qualsiasi. In particolare temibile è la loro combinazione con posta elettronica e Web o reti in genere.

Nel caso più frequente sono attaccati i file .DOC di Word. I MacroVirus infettano quasi sempre il modello Normal.DOT, che contiene anche le macro globali “legali” di Word, per poi potere infettare qualunque documento che viene aperto.

Virus del terminale

Sono tipici del mondo delle Shell UNIX e sono costituiti da sequenze di comandi (caratteri di escape) per il terminale. Sono contenuti entro file di testo (soprattutto entro messaggi e-mail) e per attivarli basta visualizzare il file. Sono ormai estremamente rari.

Web Virus

Sono concettualmente analoghi ai precedenti, in quanto si attivano con la semplice visualizzazione della pagina entro un browser. Sono contenuti entro pagine Web o HTML in genere (trasmissibili anche via posta elettronica).

Si dividono in gruppi:

- **Web Scripting Virus:** sono costituiti da pagine HTML contenenti istruzioni fraudolente (tipicamente VBScript o JavaScript), si attivano alla visualizzazione della pagina, grazie all’interprete contenuto nel browser, e sono fra i più pericolosi;
- **Virus Java:** sono contenuti entro gli Applet Java; nonostante la macchina virtuale Java del browser sia “isolata” esistono delle falle, che tali virus sfruttano e possono come minimo causare il crash del browser;
- **Virus ActiveX:** sono contenuti entro i controlli ActiveX lato client, ossia vere e proprie librerie dinamiche scaricate insieme alla pagina; interagendo con il sistema operativo locale, possono svolgere qualsiasi azione; un esempio molto famoso è il virus “spegnimento remoto” del ComputerCaos Club di Amburgo, che provocava lo spegnimento normale di ogni macchina Windows95 che aprisse la pagina che lo conteneva;
- **Cookie ostile:** i cookie sono file di testo inviati insieme ai messaggi HTTP per trasmettere informazioni di vario tipo; sono normalmente conservati dal browser fino alla scadenza in essi contenuta; un cookie ostile può leggere gli altri cookie presenti e le informazioni del browser (es. identità dell’utente che lo sta usando).

Virus misti

E’ purtroppo ormai comune vedere virus appartenenti a più di una categoria di quelle qui presentate. Ad esempio, possiamo avere un MacroVirus che apre il browser, dirigendolo verso una pagina Web contenente un WebVirus.

Una seconda classificazione è in base all’obiettivo che i virus si pongono, ossia per il quale essi vengono programmati.

Ficcanaso

Ha lo scopo di rubare informazioni entro il sistema infettato. Le informazioni trovate vengono ritrasmesse indietro di solito via rete (ad esempio, attraverso una connessione diretta al server del pirata informatico o attraverso la posta elettronica), ma esistono anche Virus che propagano “al mondo” quanto hanno “trovato”, per esempio spedendo file riservati a tutti gli indirizzi e-mail che trovano nella rubrica del computer infettato.

Devastator

Ha lo scopo di procurare più danni possibili ad un sistema, viene costruito quindi per provocare un DoS. Tutti i danni sono possibili, dalla distruzione selettiva di tutti i file .doc sino alla distruzione dei BIOS delle macchine.

Propagator

Si deve riprodurre nel massimo numero di esemplari possibili, a tal fine sfrutta tutti i meccanismi disponibili (ad esempio, e-mail, condivisioni di rete, Web...). Non necessariamente produce danni, o almeno non immediatamente quando infetta un sistema. Spesso i propagatori sono vettori per altre azioni, tipicamente contengono un client destinato a connettersi ad un server vero oggetto dell’attacco. I tantissimi client così ottenuti possono provocare il DoS per sovraccarico del server attaccato. Questo attacco è il Distributed Denial-of-Service (DDoS), già citato in precedenza.

Subdolo

Si introduce nei sistemi in modo difficile da scoprire e anche i suoi effetti sono difficili da scoprire. Per esempio può provocare lo scambio casuale di righe e parole nei file .DOC, il che comunque produce un danno molto grave ai dati.

Worm

Si propaga solo attraverso la rete e spesso prende il controllo dei sistemi infettati. I suoi effetti possono essere i più vari.

Batterio

Ha lo scopo di provocare un temporaneo DoS del sistema attaccato e a tal scopo ne satura le risorse (CPU, RAM, spazio disco) sino a provocarne il crash.

Trojan (horse)

E’ un cavallo di Troia, ha lo scopo di fare entrare qualcos’altro nel sistema attaccato o di aprire la strada a qualcuno. Sfrutta la rete per favorire gli accessi.

TimeBomb e LogicBomb

Sono Virus “silenti” ossia infettano un sistema senza compiere alcuna altra azione immediatamente. I Time Bomb si attivano in occasione di dati particolari (ad esempio, il virus Michelangelo si attiva il 6 marzo di ogni anno, data di nascita del pittore), mentre le Logic Bomb si attivano in occasione di eventi particolari (ad esempio, il

licenziamento del loro autore con la conseguente disattivazione del suo accesso ad un sistema).

Adware malevolo

Il software pubblicitario, chiamato anche adware, ha lo scopo di propagandare prodotti o l'acquisto di versioni a pagamento di programmi freeware o di raccomandare il pagamento di versioni shareware di software. Per questo viene inserito entro pagine web o direttamente entro tali prodotti. Una categoria di adware malevoli viene usata dai pirati informatici come trojan o per rubare informazioni o per pubblicizzare siti pornografici. In molti casi l'adware non viene rilevato dai normali software anti-virus, ma richiede speciali disinfectatori, chiamati appunto anti-spyware.

Virus Mutante

Categoria quasi esclusivamente tipica dei macro virus. Durante la sua azione, il codice del virus cambia. In taluni casi si è verificato anche il “crossover”, ossia la fusione di virus diversi in documenti a infezione multipla, con la produzione di nuovi virus che combinano caratteristiche di entrambi i predecessori.

Virus Combinato

Comprende in sè più di una delle caratteristiche suddette e dei corrispondenti modi di agire e quindi è estremamente pericoloso. Purtroppo tutti i nuovi virus stanno evolvendo in questa direzione.

Una ulteriore classificazione suddivide i virus in base ai meccanismi tecnici usati per perpetrare la loro azione.

Virus Polimorfici

Il codice fraudolento che forma le loro istruzioni viene crittato e la criptazione cambia infezione dopo infezione. In pratica la “firma” del Virus muta, rendendo molto più difficile la sua identificazione.

Virus Stealth

Nascondono le modifiche arredate ai file o al sistema con la loro presenza. Per esempio, non modificano data e ora o dimensioni dei file infettati. Oggi la maggior parte dei virus operano in questo modo.

Virus Lenti

Infettano un solo file per volta. Attaccano una utility di sistema e infettano i file che vengono aperti da quella utility.

Virus Retro

Attaccano direttamente i software antivirus, che saranno trattati in seguito. Ad esempio, distruggono gli archivi delle impronte virali. Esistono anche virus anti-virus che cioè disinfezionano da altri virus, rimuovendoli.

Virus Multipartite

Sfrutta attacchi contemporanei con più tecniche (es. Boot, exe). Tende a infettare progressivamente tutte le utility di sistema una volta installatosi in memoria

Virus Armored (corazzato)

Si nascondono (“corazzano”) entro codice “esterno” fuorviante. Sono difficili da riconoscere e analizzare.

Virus Companion

Tipici del mondo DOS, creano un nuovo file accanto all'eseguibile che intendono infettare (tipicamente il file .COM rispetto a .EXE). Il file .COM viene eseguito prima e infetta il sistema.

Virus Phage

Sostituiscono il proprio codice a quello del programma infettato invece che attaccarsi semplicemente ad esso. Provocano danni non riparabili ai programmi infettati.

Un problema secondario generato dai virus è quello degli **hoak**. Con il termine hoak si indica un messaggio di avvertimento per un nuovo virus inesistente, diffuso via e-mail con l'invito a spedirlo a quante più persone possibili

Ciò genera panico ingiustificato e, in definitiva, perdita di tempo e risorse.

Ma può accadere anche l'imprevisto: per esempio alcuni anni or sono, si diffuse un hoak denominato aol4free.com che venne presto riconosciuto come tale. Poco dopo un pirata informatico creò un virus reale molto dannoso con questo nome, che si propagò e produsse danni.

In generale è bene diffidare degli hoak e verificare le informazioni presso siti certi.

I **software anti-virus** sono la migliore protezione contro i virus. Il metodo più semplice con cui essi agiscono è la ricerca delle impronte virali, ossia l'insieme di byte che definisce univocamente ogni virus, entro la quale vengono fatte valere le regole note contro virus Stealth e simili. Come già detto, esistono anche software particolari, i cosiddetti anti-adware o anti-spyware che si preoccupano di cercare anche virus appartenenti alla categoria degli adware malevoli, normalmente non rintracciati dai comuni anti-virus.

Esistono anche motori “euristici” che analizzano ed interpretano i codici “insoliti” trovati, ma sono lenti in quanto devono comprendere le azioni malevole eventualmente presenti entro tali software.

Durante la fase di disinfezione l'Anti-virus cerca di ripristinare la forma originale del file infettato, ma non sempre ciò è possibile, se i danni inferti dal virus non sono recuperabili.

La migliore protezione contro le perdite è il salvataggio dei dati, analogamente a quanto già detto per le protezioni dei dati dai guasti accidentali.

Un esempio di virus molto potente è Nimda, apparso nel 2001. Questo temibile virus era formato da una combinazione di MacroVirus e Virus degli eseguibili. Usava diversi meccanismi di propagazione:

- Web Upload
- E-Mail
- Condivisione dischi (creandole quando possibile)
- Infezione di file di Office
- Creazione file eseguibili infetti in varie cartelle di sistema di Windows.

E' stato sperimentato che un singolo virus Nimda, pervenuto attraverso un messaggio di posta elettronica infetto, è stato in grado di infettare una intera rete di circa 200 PC nel giro di poco più di un'ora. La rimozione del virus e la riparazione dei danni hanno poi richiesto diversi giorni di lavoro.

La protezione delle informazioni

Riassumendo quanto visto nei paragrafi precedenti, è possibile affermare che, in una postazione client, i pericoli maggiori sono essenzialmente l'infezione da virus attraverso la posta elettronica o la rete locale, in particolare di trojan e backdoor (trojan particolari, predisposti per consentire un ingresso esterno ad un pirata, per esempio per fargli prendere il controllo remoto del PC invaso), l'infezione attraverso pagine Web con virus e l'intrusione nei dischi.

I pericoli corsi dai server sono gli stessi, con in più tutte le caratteristiche dei server.

A ciò si deve aggiungere la trattazione dei problemi legati alle reti.

Focalizzando l'attenzione sulla necessità di proteggere le informazioni, allora sono individuabili tre scenari distinti:

- Sicurezza sui dati e dei programmi che li contengono;
- Sicurezza del canale trasmissivo, per porre rimedio al pericolo delle intercettazioni;
- Sicurezza dei server.

Per potere in questi tre scenari capire i metodi per la protezione dei dati occorre rispondere alle seguenti domande:

- Quali sono i dati?
- Dove sono i dati?
- Come si accede ai dati?
- Come si interpretano i dati?

Come già visto nel capitolo 3, i dati, dal punto di vista informatico, possono essere espressi digitalmente in varie forme, con file di testo, file di documenti in formato proprietario, file con contenuto multimediale, file di database server ecc.... In generale comunque i dati entro i file sono rappresentati o con formati pubblici, la cui rappresentazione dei dati è pubblicamente documentata, o proprietari, che quindi possono essere interpretati solo dal programma che li ha generati. Ma anche nel caso di formati proprietari la sicurezza non è garantita: un abile hacker, se accede ai file, è in grado di ricostruirne i tracciati e di estrarne le informazioni utili.

L'unica garanzia per una protezione sicura dei dati è la **cifratura** o **crittografia** dei dati stessi, ovvero la loro alterazione in una forma non leggibile a tutti, ma soltanto ai possessori degli strumenti adeguati per la decifrazione.

In generale si può definire **cifratura** (o anche, in modo meno preciso, codifica) un'operazione che consente di trasformare un dato messaggio in modo illeggibile a tutti, tranne a chi possiede la chiave di decodifica opportuna. Esistono molti tipi di cifratura. Prima di procedere occorre definire con precisione i termini relativi alle entità in gioco:

- Testo in chiaro (in inglese plaintext) è il testo originale, o l'insieme dei dati originali;
- Testo cifrato o codificato (in inglese ciphertext) è il testo crittografato;
- Cifratura o codifica o criptazione (in inglese encryption) è l'operazione di "traduzione" del testo da "chiaro" a "cifrato";
- Decifratura o decodifica o decriptazione (in inglese decryption) è l'operazione inversa;
- Chiave (in inglese key) è la o le entità usata/e per la codifica;
- Steganografia (in inglese steganography) è la tecnica di nascondere dati dentro altri, per esempio un messaggio può essere composto con le lettere iniziali di tutte le parole dispari di un altro messaggio, oppure un dato può essere "nascosto" dentro una immagine bitmap a toni di grigio sfruttando il bit meno significativo in ogni pixel; se tale bit rimane uguale alla versione originaria, il valore del bit "nascosto" è zero, altrimenti è 1; l'estrazione del dato "nascosto" si ottiene semplicemente confrontando l'immagine originale con quella trasformata;
- Crittoanalisi (in inglese cryptanalysis) è la scienza che studia il modo per decifrare un messaggio crittato, ricostruendo la sua forma originaria o comunque estraendo l'informazione in esso contenuta, senza conoscere la chiave o addirittura il metodo con cui il messaggio è stato cifrato; esistono varie tipologie di crittoanalisi:

- attacco al solo testo cifrato (in inglese ciphertext-only attack) che si attua quando si è in possesso del solo messaggio in forma cifrata;
- attacco al testo noto (in inglese known-plaintext attack) che si conduce quando si è in possesso del testo in chiaro e della sua forma cifrata e si può quindi applicare il confronto;
- attacco a testo scelto (in inglese chosen-plaintext attack) che si attua quando si è possesso del decifratore ovvero si può cifrare un messaggio qualsiasi e poi applicare anche il confronto;

invece alcuni dei metodi applicabili per la crittoanalisi sono:

- analisi di frequenza (in inglese frequency analysis), in cui si osserva il ripetersi di certe configurazioni nel messaggio cifrato, facendo ipotesi ragionevoli sulla struttura del messaggio originale (per esempio, la lingua in cui è scritto);
- indici di coincidenza (in inglese index of coincidence), in cui si confrontano il testo in chiaro e quello cifrato, osservando il ripetersi di

- simboli in posizioni corrispondenti, per ricostruire il metodo di cifratura con cui l'uno viene trasformato nell'altro;
- approccio forza-bruta (in inglese brute-force approach), in cui si provano esaustivamente tutti i valori e/o i metodi possibili; tale metodo è entrato in uso effettivo solo con l'avvento dell'uso dei computer.

Per capire il funzionamento dei sistemi crittografici occorre vedere degli esempi concreti. Pertanto viene ora analizzato uno dei primi sistemi crittografici in uso nella storia, il cosiddetto algoritmo di Cesare (così chiamato in quanto il suo uso era attribuito a Giulio Cesare), che agisce cifrando i testi. L'algoritmo di Cesare si basa sullo scorrimento (rotatorio) dell'alfabeto dei caratteri. Si dice ordine dell'algoritmo il numero di posizioni di cui viene ruotato l'alfabeto durante l'operazione. Per esempio, si supponga che l'ordine sia 4 e che lo spazio nei messaggi sia un carattere avente indice posizionale pari a zero. Allora, come è facilmente verificabile aggiungendo 4 all'indice delle lettere, una frase come "se magna" diventa "widqekre". Applicando le suddette tecniche di crittoanalisi all'algoritmo di Cesare però se ne evidenziano i limiti. Poiché ogni lingua ha statistiche precise sulla frequenza di comparizione delle lettere, studiando la frequenza dei caratteri del testo cifrato con l'algoritmo di Cesare, si può intuire come funziona e diviene quindi facile indovinare che si tratta di uno scorrimento. Inoltre esso è sensibile ad un attacco a "forza bruta", ossia per prove esaustive delle combinazioni possibili dello scorrimento.

Per rafforzare l'algoritmo di Cesare si può introdurre la variante detta "con chiave", in cui lo scorrimento dei singoli caratteri del messaggio avviene in base ai valori dei caratteri di una "parola chiave", per cui ogni carattere del messaggio risulta traslato nell'indice alfabetico in modo diverso rispetto agli altri. E' evidente la maggiore robustezza di questa evoluzione. Vediamo un esempio di applicazione:

- Testo in chiaro: "fido"
- Parola chiave: "abcd"
- Testo cifrato: "gkgs"

Appare evidente che, più lunga è la chiave, più sicura è la protezione rispetto ad attacchi a "forza bruta", ma anche ad attacchi basati sulla analisi di frequenza. Questa regola è fondamentale in ogni algoritmo crittografico ed è per questo che, come si vedrà meglio in seguito, nei sistemi moderni vengono specificate lunghezze minime delle chiavi per ottenere livelli considerati sufficienti di sicurezza.

Un'altro metodo per la cifratura dei testi è la sostituzione semplice completa, in cui, attraverso una mappatura, ad ogni carattere dell'alfabeto se ne sostituisce un altro.

Per esempio, supponiamo di avere la mappatura sotto rappresentata, in cui il carattere sottolineatura (in inglese underscore) rappresenta lo spazio.

Mappatura chiave:

ABCDEF	GHIJKL	MNOPQR	STUVW	XYZ
QWERTYU	IOPASD	FGHJKL	ZXCVBNM	

Testo in chiaro: "IO SONO"

Testo cifrato: "OFMKF F"

I passi di un attacco con crittoanalisi condotti verso questa crittografia del testo potrebbero essere:

- Individuare la lingua del testo originale;
- Determinare il sistema di cifratura usato (ad esempio Cesare con chiave o sostituzione semplice);
- Ricostruire la chiave (ad esempio l'ordine di Cesare, la chiave di Cesare o la tabella di sostituzione);
- Ricostruire completamente il testo in chiaro originale.

Per esempio, nel caso che la lingua del testo originale sia l'inglese, l'analisi di frequenza delle lettere stabilisce che la lettera "e", la più frequente, compare mediamente nel 13% dei casi. Se il metodo usato è la sostituzione semplice, la lettera o il simbolo che compare nel 13% dei casi è un buon candidato per essere il corrispondente della lettera "e".

L'evoluzione verso una maggiore sicurezza dei metodi basati sulla sostituzione semplice conduce ai **sistemi polialfabetici**. Se in un sistema monoalfabetico un dato testo cifrato corrisponde sempre ad uno ed un solo testo in chiaro, in un sistema polialfabetico un dato testo cifrato può cambiare il suo corrispondente testo in chiaro.

Un tipico sistema polialfabetico userà da 2 sino a 26 differenti alfabeti, uno per ogni lettera dell'alfabeto latino internazionale. In particolare, i sistemi polialfabetici **periodici** ripetono lo stesso insieme di alfabeti ciclicamente nel tempo, mentre i sistemi polialfabetici **aperiodici** non si ripetono mai nello stesso ordine.

I sistemi periodici sono generalmente meno sicuri degli aperiodici per via della ripetizione ciclica regolare delle chiavi. D'altra parte i sistemi aperiodici sono più difficili da usare, a meno che l'operazione di cifratura e decifratura non venga compiuta automaticamente da una macchina cifratrice o da un computer.

Un esempio classico di sistema aperiodico è costituito dal **quadrato o tabella di Vigenère**, rappresentato in figura 8.15.

Il quadrato di Vigenère include tutti i possibili allineamenti di un alfabeto diretto standard. Anche alfabeti misti possono essere usati in questo quadrato. Se tutti e 26 gli alfabeti sono usati, allora ogni lettera del messaggio originale può corrispondere a qualsiasi altra lettera.

Per usare il quadrato di Vigenère si procede nel seguente modo:

- le lettere del messaggio originale sono allineate nella parte superiore del quadrato in orizzontale;
- la chiave (ovvero le lettere della chiave), poste in verticale, individuano quale delle 26 sequenze deve essere usata e all'incrocio, per ogni lettera del messaggio si ottiene la corrispondente lettera cifrata.

		Testo in chiaro																									
		a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Chiave	A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
	B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
	C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
	D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
	E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
	F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
	G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
	H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
	I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
	J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
	K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
	L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
	M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
	N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
	O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
	P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
	Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
	R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
	S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
	T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
	U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
	V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
	W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
	X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
	Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
	Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Figura 8.15: Il quadrato o tabella di Vigenére

Il quadrato di Vigenére può essere usato sia per sistemi periodici, sia per aperiodici. Si raccomanda [LZ 2004] per approfondimenti sulle basi teoriche della crittografia.

I moderni sistemi di crittografia impiegano sofisticate tecniche logico-matematiche per cifrare i dati. Entro il computer tutte le informazioni sono rappresentate come numeri binari e, quindi, non necessariamente la cifratura dei testi deve avvenire carattere per carattere, anzi, solitamente le unità cifrate sono blocchi di 64 o 128 bit (8 o 16 byte).

Nella maggior parte dei casi i computer eseguono le operazioni di crittografia moltiplicando e/o dividendo i valori numerici, che rappresentano i dati, per numeri molto estesi e decodificano i dati applicando gli stessi numeri.

In generale la sicurezza di un metodo crittografico si basa sulla segretezza del metodo e/o sulla segretezza della chiave. L'esperienza pratica ha però portato in evidenza le difficoltà di tenere nascosto un metodo crittografico, per cui oggi la maggior parte degli algoritmi sono pubblici e la sicurezza è basata sulla protezione della chiave.

Le regole viste in precedenza per la crittoanalisi continuano a valere. Se, in particolare, è possibile un confronto fra dati in chiaro e dati crittografati, il compito dell'attaccante è enormemente facilitato.

Si definiscono **codici perfetti** (in inglese perfect ciphers) quelli impossibili da decifrare attraverso la crittoanalisi non basata sulla forza bruta.

Alcuni esempi possono essere un elenco senza limiti di chiavi non ripetitive o una serie “infinita” di numeri casuali generati da computer. Quasi sempre però questi sistemi sono troppo complessi e/o lenti per essere applicabili praticamente nel mondo reale.

L’evoluzione delle tecnologie ha infine portato alle due grandi famiglie di algoritmi crittografici moderni: quelli a **chiave simmetrica** e quelli a **chiave asimmetrica**; questi ultimi sono noti anche come algoritmi a chiave pubblica e chiave privata.

Negli algoritmi a chiave simmetrica, il cui funzionamento è schematizzato in figura 8.16, la stessa chiave viene usata sia per la criptazione sia per la decriptazione. Lo studio statistico ha dimostrato che, con la potenza dei calcolatori di oggi, con la lunghezza delle chiavi maggiore o uguale a 128 bit il sistema è praticamente al sicuro da attacchi a forza bruta. Infatti il tempo necessario per condurre tali attacchi, attraverso prove esaustive dei codici possibili, è troppo elevato per renderli praticamente utili. Va detto però che solo 15 anni fa questo limite era di 56 bit e la potenza dei calcolatori continua a crescere. I limiti dei sistemi a chiave simmetrica sono legati alla sicurezza della chiave. Il sistema è sicuro solo finché la chiave è solo nelle mani delle persone effettivamente abilitate ad averne accesso. Inoltre un algoritmo a chiave simmetrica può proteggere efficacemente dall’accesso non voluto ai dati o dall’intercettazione dei medesimi quando sono in transito lungo una rete, ma non è in alcun modo di aiuto quando si tratta di definire l’identità del mittente, ovvero di garantire l’autenticazione del mittente di un messaggio. I più diffusi algoritmi di crittografia simmetrica sono il DES (DES sta per Data Encryption Standard, algoritmo molto usato in passato negli enti governativi statunitensi), il 3-DES (è una sua evoluzione che ne sta progressivamente prendendo il posto), il Blowfish e l’IDEA.

Negli algoritmi a chiave asimmetrica, il cui funzionamento è schematizzato in figura 8.17, due chiavi distinte vengono usate nelle fasi di criptazione e decriptazione. Le due chiavi sono legate fra di loro e devono essere generate insieme, anche se la conoscenza di una sola non è in alcun modo di aiuto nel trovare l’altra. Formalizzando il tutto, si può dire che, avendo definito

- E = azione di crittografia
- D = azione di decrittografia
- M = messaggio originale

si ottiene:

$$D(E(M)) = M$$

$$E(D(M)) = M$$

Ovvero il ruolo delle chiavi è duale: ciò che viene crittografato con una può essere decrittato solo con l’altra, ma entrambe possono svolgere lo stesso ruolo, ovvero essere

usate per crittografare ciò che sarà decifrato con l'altra o per decifrare ciò che è stato crittografato con l'altra. Nell'applicazione più frequente della crittografica asimmetrica, una delle due chiavi (definita **chiave privata**) viene mantenuta segreta e l'altra (definita **chiave pubblica**) viene distribuita, associata con l'identità del possessore della chiave privata. In tal modo divengono possibili alcuni scenari di uso:

- un mittente esterno crittografa il messaggio servendosi della chiave pubblica del destinatario, che, finché la sicurezza della sua chiave privata è mantenuta, rimane l'unico in grado di decifrare il messaggio (sicurezza rispetto al destinatario);
- il possessore della chiave privata crittografa il messaggio con la sua chiave; tutti coloro che possiedono la chiave pubblica sono in grado di leggerlo, ma, finché la sicurezza della sua chiave privata è mantenuta, rimane dimostrata l'identità del mittente del messaggio (sicurezza rispetto al mittente);
- combinando insieme le due tecniche diviene possibile garantire sia la sicurezza del destinatario, sia quella del mittente: come mostrato in figura 8.18, Bob (il mittente) applica prima la sua chiave privata (garanzia del mittente) e poi la chiave pubblica di Alice (garanzia del destinatario); Alice, quando riceve il messaggio applica nell'ordine inverso le chiavi corrispondenti e riottiene il messaggio originale, con la garanzia della protezione durante la trasmissione e la certezza che il mittente sia Bob.

La lunghezza minima delle chiavi asimmetriche per garantire la protezione da un attacco a forza bruta è superiore a quella delle chiavi simmetriche e, con la potenza dei calcolatori attuali, deve essere maggiore o uguale di 1024 bit.

Per ulteriori approfondimenti sulla crittografia si raccomanda [LZ 2004]. Il testo di riferimento per applicazioni molto avanzate è invece [Scheiner 1995].

Dalle applicazioni della crittografia nascono i nuovi servizi di identità digitale o elettronica e di certificazione dei messaggi, che saranno trattati nel successivo paragrafo.

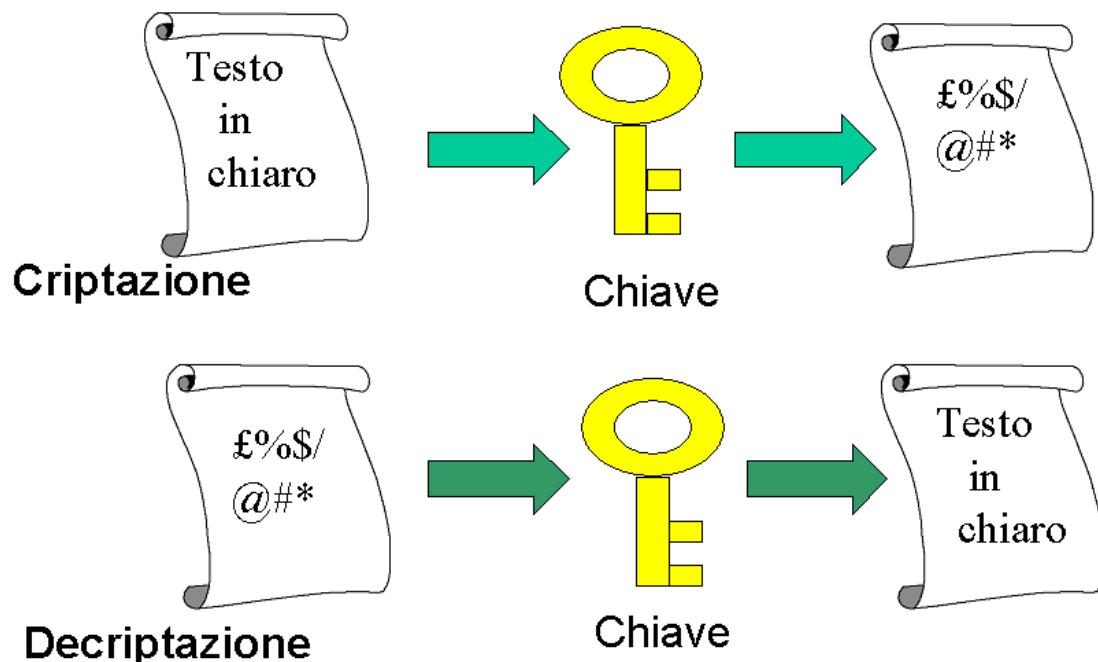


Figura 8.16: Schema di funzionamento della crittografia simmetrica.

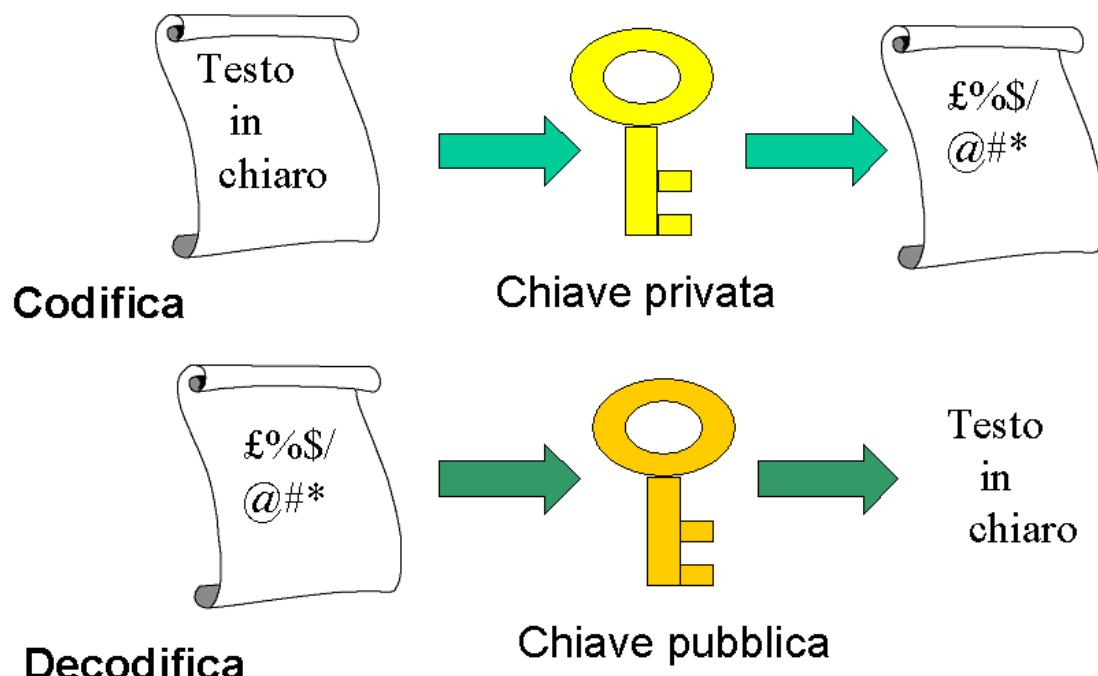


Figura 8.17: Schema di funzionamento della crittografia asimmetrica.

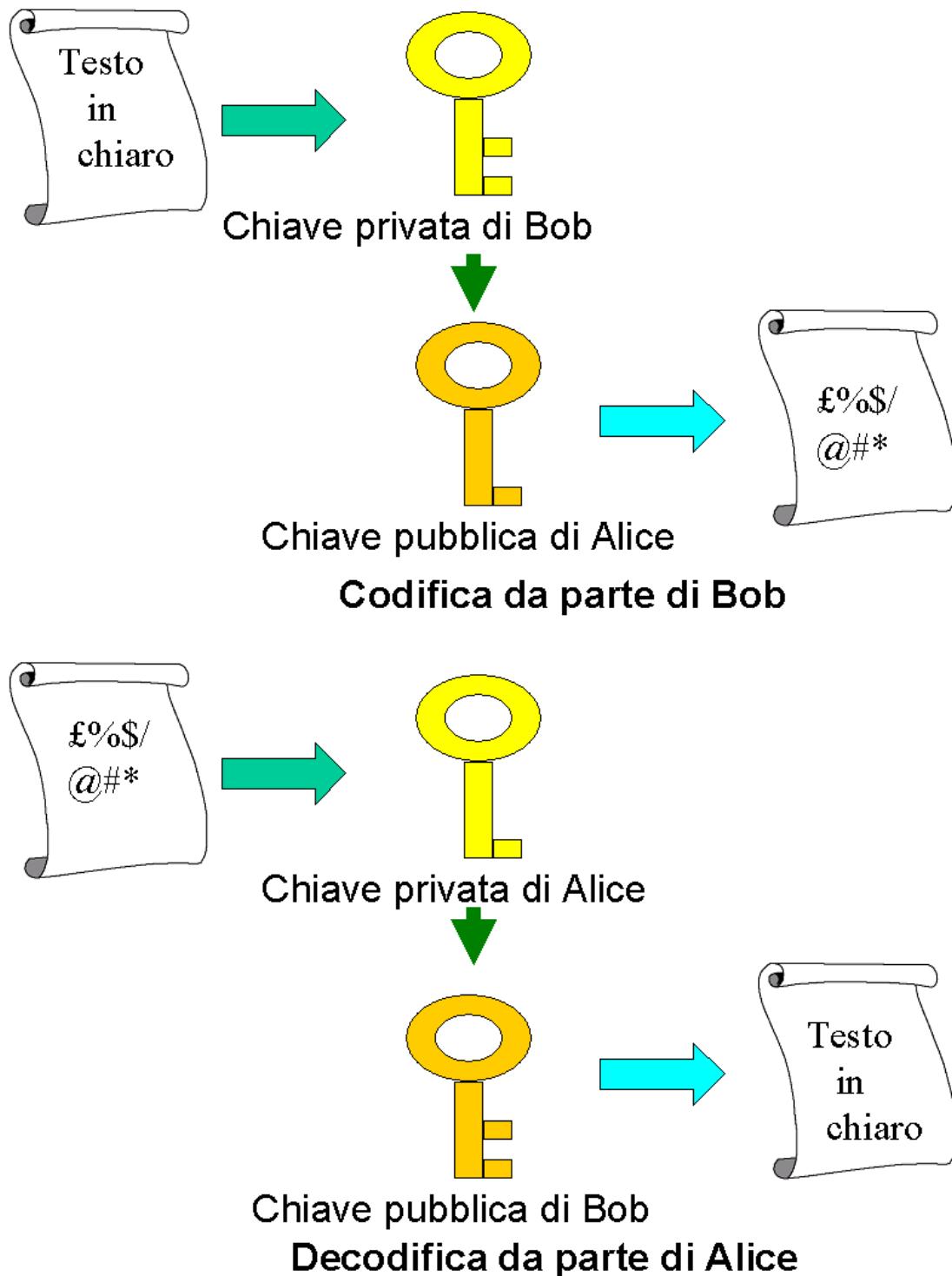


Figura 8.18: Uso della doppia chiave, in alto durante la trasmissione e in basso durante la ricezione.

L'identità elettronica

L'avvento delle tecnologie legate alla crittografia ha reso disponibili una serie di nuovi servizi, non soltanto legati alla protezione dei dati, ma anche alla protezione dell'identità del mittente e alla certezza di non alterazione del messaggio, che potrebbe comunque avere luogo anche sulla sua forma cifrata.

Per raggiungere questo scopo, accanto alla crittografia propriamente detta, sopra definita, viene usata anche un'altra tecnologia, il cosiddetto **hashing** dei messaggi, originalmente derivata dalle tecniche di rilevamento degli errori di trasmissione delle prime comunicazioni digitali. In pratica, le tecniche di hashing consentono, dato un qualsiasi documento digitale, di ricavare da esso un codice numerico ad esso univocamente associato chiamato anche impronta digitale del documento o, usando la terminologia inglese, message digest o digital fingerprint, come rappresentato in figura 8.19. Il codice è associato all'insieme dei byte che forma il documento e qualsiasi alterazione nel documento stesso produce una variazione del codice hash da esso ottenuto. Il codice è monodirezionale, vale a dire che esso non può essere in alcun modo usato per riottenere il messaggio stesso. In figura 8.20 viene mostrata una delle applicazioni: insieme al messaggio viene trasmesso il codice hash cifrato e una volta riottenuto il messaggio in chiaro, ad esso viene applicato l'algoritmo di hash ed i due codici vengono così confrontati. Se non coincidono allora il messaggio è stato in qualche modo alterato (in inglese tampered).

Un altro tipo di pericolo si verifica qualora il pirata informatico che intercetta una rete, pur senza riuscire a decodificare i messaggi, osservando nel tempo il loro contenuto e le azioni ad esso associate, riesce a intuire il ruolo dei messaggi stessi e, tentando di duplicarli, può riuscire a provocare fraudolentemente le azioni ad essi associate; questo è il cosiddetto attacco a replica (in inglese reply attack). Una delle protezioni più efficaci è il cosiddetto cryptographic nonce o, semplicemente, **nonce**, ossia in pratica l'aggiunta di un numero casuale destinato ad essere usato una sola volta ad ogni messaggio, in modo tale che l'intercettazione e l'eventuale duplicazione possa essere scoperta. In generale vengono usati per i nonce tre tipologie di dati: interi sempre crescenti, indicazione dell'istante di invio e quest'ultimo combinato con un numero casuale.

Combinando insieme le varie tecniche sinora viste, si giunge alla **firma digitale** di un messaggio, come rappresentato in figura 8.21. Il messaggio viene sottoposto all'hashing e al digest così ottenuto, che permette di verificare la forma originale del messaggio, viene applicata la chiave privata del mittente, che dà la garanzia del mittente. La firma così ottenuta viene spedita insieme al messaggio.

Un'altra tecnica di protezione dei messaggi è il message enveloping o “imbustamento del messaggio”, in cui, dopo avere crittografato il messaggio con una chiave simmetrica, si procede alla crittografia di tale chiave con la chiave pubblica del destinatario, come rappresentato in figura 8.22. La chiave può essere spedita insieme al messaggio originale o in un messaggio successivo, ma in ogni caso la sicurezza rimane garantita. Rispetto alla tecnica della doppia chiave questa ha il vantaggio di essere meno pesante da un punto di vista computazionale e, come si vedrà nel

paragrafo successivo, viene usata per questo nei sistemi di comunicazione cifrata in tempo reale.

Un ulteriore tecnica è quella della “firma cieca” o blind signature, usata in alcuni sistemi di pagamento elettronico (si veda [OTP 2002]). In questa tecnica la firma serve a dare validità ad un documento, senza però poterne vedere il contenuto, come nel mondo reale potrebbe avvenire quando un ufficio postale appone il timbro su una busta senza visualizzarne il contenuto. Per questo gli algoritmi devono essere commutativi rispetto alla applicazione della tecnica crittografica usata per creare la busta digitale. In tal modo si ottiene lo schema rappresentato in figura 8.23: il messaggio originale viene cifrato ottenendo la busta digitale, su cui viene poi apposta la firma che, come se vi fosse della “carta carbone” entro la busta, rimane anche successivamente alla rimozione della busta.

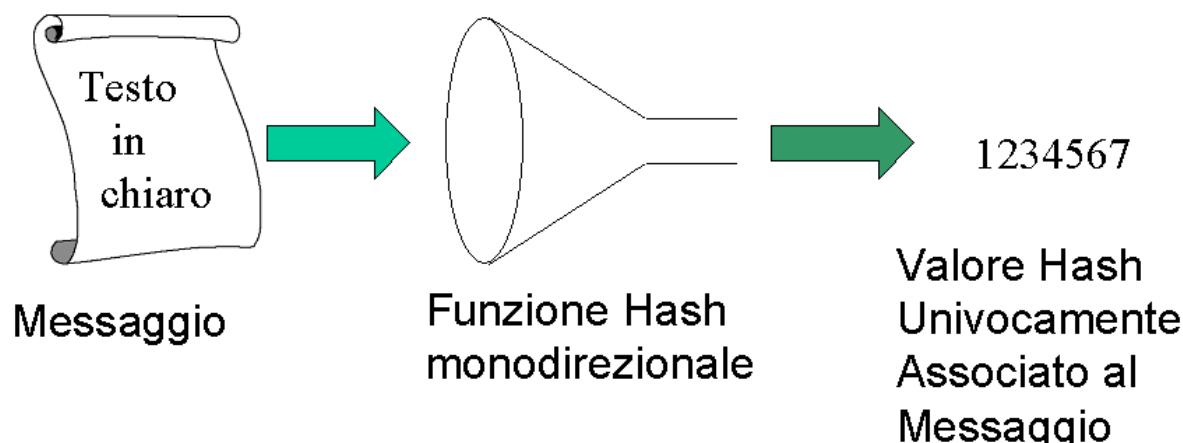


Figura 8.19: Il funzionamento dell'hashing

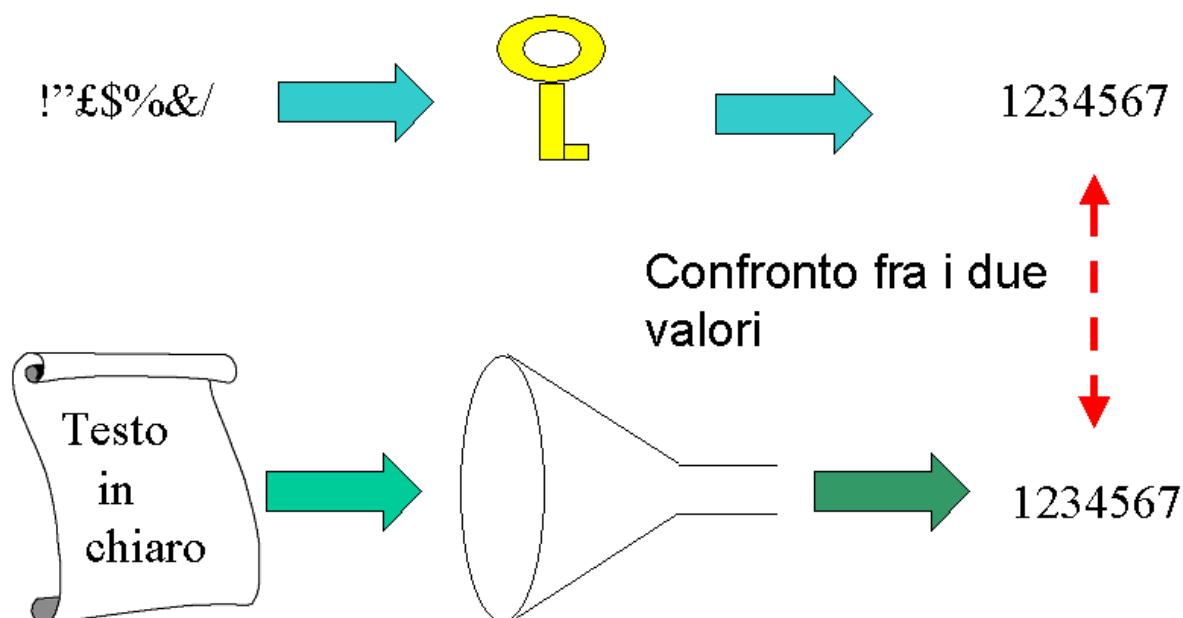


Figura 8.20: Applicazione dell'hashing alla verifica di integrità di un messaggio.

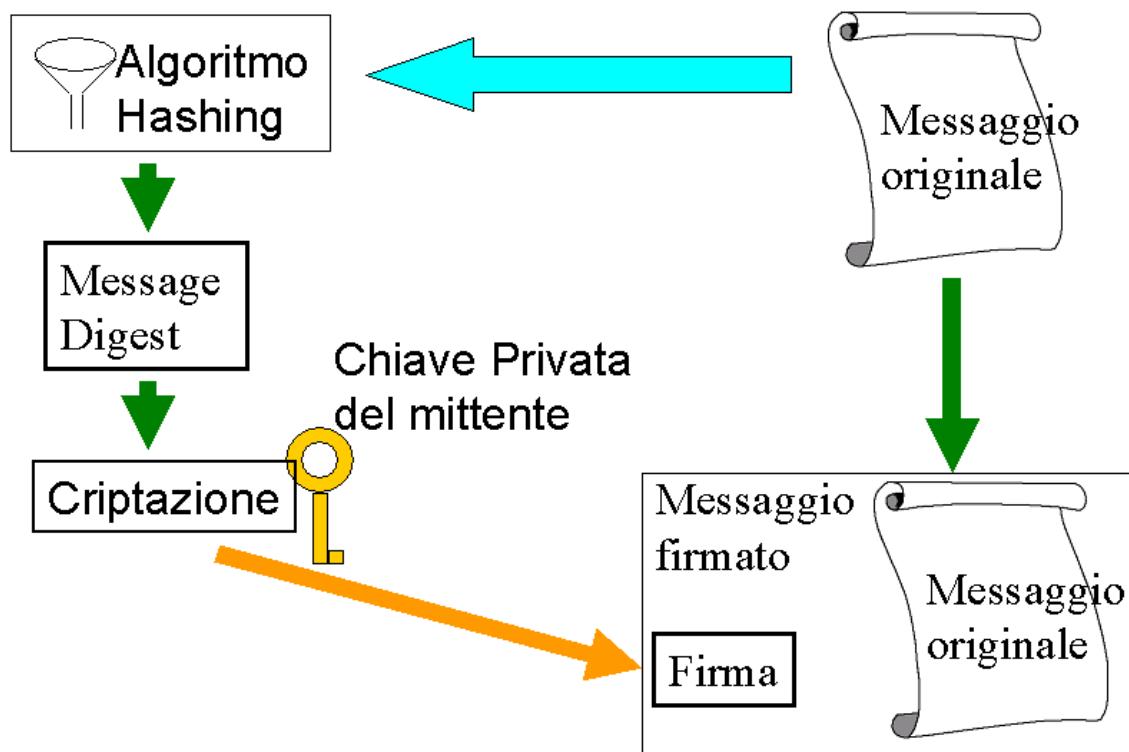


Figura 8.21: Schema della costruzione della firma digitale di un messaggio.

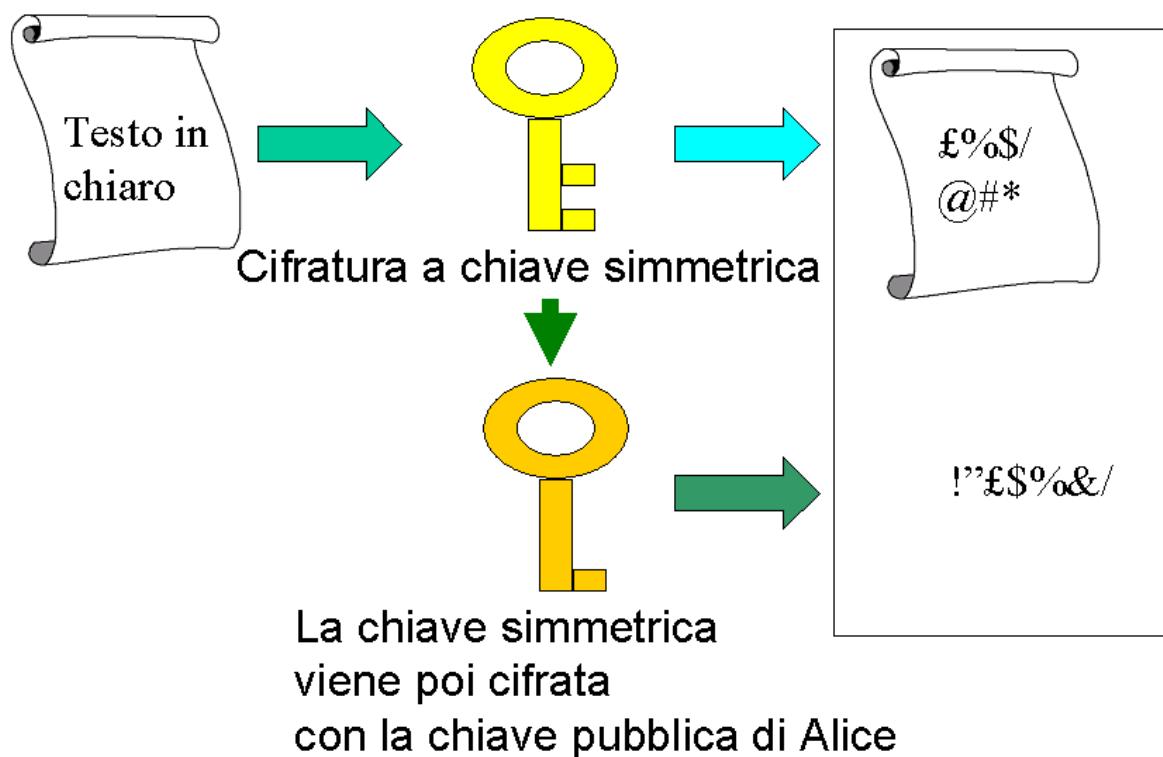


Figura 8.22: La tecnica del message enveloping.

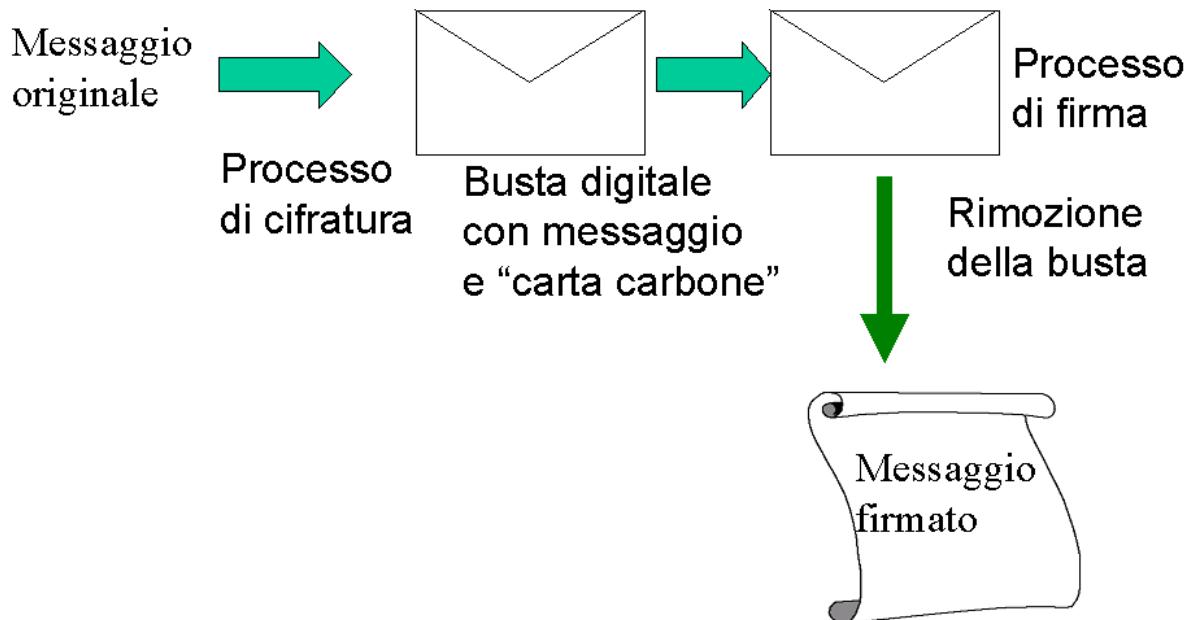


Figura 8.23: Schema del funzionamento della blind signature.

Le applicazioni nelle transazioni elettroniche di queste tecnologie sono ormai definite a termini di legge. In particolare una firma digitale è un valore univoco che un particolare software crea applicando una funzione matematica e una chiave di codifica a un messaggio o un file. La firma digitale, come già accennato in precedenza, deve confermare sia l'identità dell'autore sia la non manipolazione del messaggio durante la trasmissione.

Le firme digitali usate negli USA e le funzioni matematiche associate sono codificate nello standard DSS, un altro standard molto importante è l'MD-5 (per approfondimenti si veda [RFC 1321]).

A livello europeo, come stabilito nella Dir. 1999/93/Ce (G.U.C.E. 13/12/1999), si definisce come **“Firma Elettronica Avanzata”** un insieme di dati in forma elettronica, allegati ad altri dati per garantire autenticazione, che soddisfa i seguenti requisiti:

- essere connesso in maniera unica al firmatario;
- essere idoneo ad identificare il firmatario;
- essere creato con mezzi sui quali il firmatario può conservare il proprio controllo esclusivo;
- essere collegato ai dati cui si riferisce in modo da consentire la identificazione di ogni successiva modifica di detti dati.

Un'applicazione della firma digitale sono i certificati digitali, ossia una rappresentazione visuale di un valore digitale univoco che verifica il contenuto di un file ed il suo produttore sulla base di un sistema di verifica e di firma, controllato da organizzazioni terze, considerate fidate sia dal produttore, sia dal destinatario (indicate anche dall'acronimo TTP, dalle iniziali delle parole inglesi Trusted Third Party).

In particolare, i certificati a chiave pubblica sono strutture dati elettroniche che associano in modo biunivoco una chiave pubblica ad un dato (ad esempio, a un file) o ad una macchina (per esempio, il server che ospita un sito Web). Normalmente tali certificati hanno anche una scadenza (ad esempio, un anno). Quando si trasmette una firma digitale nel Web, il documento è verificato da un'autorità di certificazione digitale (come, per esempio, l'azienda statunitense Verisign, leader di questo mercato; si veda anche [Verisign 2006]).

La Public Key Infrastructure (PKI) è una organizzazione tecnico-amministrativa che ha l'incarico di fornire, gestire, revocare i certificati a chiave pubblica.

E' gerarchica e si compone di

- Autorità di certificazione
- Autorità di registrazione
- Autorità di revoca

L'originalmente prevista struttura basata su autorità centrali nazionali e sovranazionali (ad esempio, in Europa, EuroPKI) non si è sviluppata completamente e a tutt'oggi troviamo aziende private, come la già citata Verisign, insieme ad autorità governative, come, ad esempio l'US Government External Certificate Authority (ECA), a svolgere il compito di autorità di certificazione. Per approfondimenti sulla PKI si consiglia [Entrust 2006].

Un'applicazione importante della firma elettronica avanzata è il contratto informatico, ossia un contratto, stipulato a distanza attraverso la rete, in cui le informazioni sono trasmesse in formato digitale attraverso la rete in tempo reale. In Italia ha validità giuridica con il D.P.R. 10 Nov. 1997 n.513, uno dei, purtroppo, rari casi in cui la legislazione italiana ha preceduto quella comune europea. Presuppone sistemi di autenticazione, sicurezza e non ripudiabilità per potere essere valido. In base ad esso quindi una transazione è valida quando sono garantiti:

- la riservatezza dell'informazione digitale
- la paternità dell'informazione digitale
- l'integrità dell'informazione digitale
- la non ripudiabilità dell'informazione digitale
- la certezza del momento in cui è avvenuto l'accordo digitale
- la conferma della ricezione dell'informazione digitale

Le applicazioni pratiche di un meccanismo di questo tipo sono molteplici; solo a titolo di esempio possiamo citare:

- commercio elettronico reale, estensibile ad ogni categoria di beni e/o servizi;
- voto elettronico sicuro via Internet;
- in generale ogni tipo di transazione finanziaria su rete pubblica.

Per proteggere meglio le chiavi private su cui la verifica dell'identità elettronica si basa, nuovi supporti tecnologici stanno apparendo. Fra questi molto usata è la **smartcard**, piccola tessera di plastica simile alle carte di credito, ma che, accanto ad una eventuale banda magnetica presente per retrocompatibilità, contiene anche un chip di memoria e/o un microprocessore, con possibilità di conservare con maggiore sicurezza le informazioni memorizzate ed una molto maggiore capacità di immagazzinarne. Molti sistemi di autenticazione sono basati sulla chiave contenuta in

una smartcard e richiedono per il loro funzionamento che la smartcard sia inserita entro un apposito lettore.

Anche la nuova carta di identità elettronica è una tessera simile a una smartcard, dotata sia di chip, sia di banda ottica e magnetica, con i dati personali principali visibili. Essa è in grado di memorizzare tutti i dati della carta di identità, insieme a strumenti (come i certificati digitali) per l'autenticazione via rete.

Per il futuro è prevedibile una estensione dei meccanismi a chiave asimmetrica. In generale, al di là delle tecniche di base, per potere usufruire di tutti i nuovi servizi possibili, è necessaria la messa in opera di nuove grandi infrastrutture tecnologiche e dell'organizzazione per gestirle.

Per approfondimenti sulla identità elettronica si consiglia [Windley 2005].

Applicazioni operative di crittografia ed autenticazione

In questo paragrafo saranno esaminate alcune applicazioni operative di metodi di crittografia ed autenticazione, rivolte ai problemi della protezione di dati esistenti sotto forma di file o di dati in transito lungo canali di comunicazione, come mostrato nella figura 8.24.

Per ciò che riguarda i file, attraverso la crittografia è possibile proteggere un singolo file, sostituendolo con la sua copia crittografata oppure un intero file system, rendendolo totalmente incomprensibile a chi non possieda la chiave per la decifrazione.

Per quanto riguarda i canali è possibile cifrare soltanto il contenuto di messaggi (ad esempio, spedendoli come allegati cifrati attraverso la posta elettronica), oppure proteggere completamente il contenuto dell'intero canale (ossia cifrando l'intero campo dati di ogni pacchetto), realizzando la protezione completa. La protezione di tutto il canale può avvenire a livello applicativo, come, ad esempio, fanno i protocolli SSL e SSH e le loro applicazioni, oppure addirittura a livello del trasporto e rete del TCP/IP, come, ad esempio fa il protocollo IPSec.

Esempio di applicazione ai singoli file: PGP

Uno degli algoritmi più diffusi per la protezione dei file è il Pretty Good Privacy, meglio noto con l'acronimo PGP. PGP è stato originariamente sviluppato da Phil Zimmermann nel 1991 ed è stato così influente che il suo progetto è stato trasformato dalla IETF in uno standard Internet chiamato OpenPGP (si veda anche [OpenPGP 2004]). Una lunga serie di trasformazioni commerciali ed acquisizioni di società ha trasformato PGP nel tempo e, al momento attuale, esistono varie versioni:

- una versione commerciale “ufficiale” sviluppata e promossa dalla PGP Corporation [PGP 2004];
- una versione freeware gestita e sviluppata dalla Free Software Foundation [GnuPG 2005], che si rifà strettamente allo standard IETF;
- altre versioni commerciali.

Le versioni sono sostanzialmente compatibili tra loro, anche se non sempre possono interoperate con le versioni più vecchie precedenti al 2002.

PGP è disponibile su varie piattaforme sotto forma di programmi nativi, librerie e plug-in per altri programmi e rende possibili essenzialmente le seguenti funzioni:

- usare crittografia asimmetrica e simmetrica e, se necessario, generare le chiavi;
- crittografare file presenti sul disco;
- crittografare il contenuto di cartelle condivise presenti su file server, proteggendo l'accesso ai dati in esse contenuti;
- crittografare in automatico messaggi di e-mail in spedizione e, se in possesso della chiave, decifrarli in ricezione;
- verificare, grazie all'hashing, eventuali alterazioni di file e messaggi;
- interfacciarsi con archivi pubblici di chiavi.

E' possibile ovviamente usare PGP in modo non trasparente con tutti gli altri protocolli di scambio dati, cifrando il file prima di inviarlo sul canale non sicuro.

Ulteriori informazioni sul PGP possono essere reperite in [PGPI 2004], [Zimmermann 2004] e [Veridis 2005].

Esempio di applicazione al file system: EFS

A partire da Windows2000, Microsoft ha introdotto nei propri sistemi operativi la possibilità di crittografare automaticamente intere parti del filesystem attraverso la funzionalità Encrypted file system (EFS). In modo totalmente trasparente per l'utente, tutti i file e le cartelle su cui lui ha diritto di proprietà possono essere protetti con la cifratura. In tal modo, anche in seguito al furto di un computer o di un hard disk, risulta impossibile accedere ai dati. La crittografia utilizzata per proteggere i file si basa sull'uso di una chiave simmetrica seguendo lo standard DESX, versione potenziata del DES, che viene memorizzata entro il profilo dell'utente. Però questo fatto, qualora la chiave non venga copiata anche esternamente, rende il tutto vulnerabile al danneggiamento del profilo. Nel caso, ad esempio, in cui a causa di un attacco al domain server occorra rifare i profili di tutti gli utenti anche sulle postazioni client, la chiave potrebbe non essere recuperabile, con conseguente perdita dei dati cifrati.

Ulteriori informazioni su EFS possono essere reperite in [EFS 2000] e [NTFS-EFS 2000].

Esempio di protocollo semi-cifrato: SCMP

Simple Commerce Messaging Protocol (SCMP) è un protocollo che usa gli standard MIME per inserire un messaggio cifrato (il cosiddetto carico utile o payload) con le informazioni importanti al suo interno entro un canale di comunicazione non protetto. L'implementazione più diffusa di SCMP usa normalmente HTTP come protocollo di comunicazione e tutte le indicazioni relative alla gestione tecnica della sessione di trasferimento dati compaiono in chiaro, mentre il messaggio è cifrato ed un eventuale pirata che intercetti la sessione lo può solo vedere come un allegato cifrato al messaggio. SCMP non è ulteriormente in sviluppo, in quanto le sue funzionalità stanno confluendo negli standard dei Web Service. Ulteriori informazioni possono essere reperite in [SCMP 2000].

Una libreria per la protezione dei canali: SSL

Secure Socket Layer (SSL) è un protocollo di sicurezza general-purpose, che opera al di sopra del livello trasporto nello stack TCP/IP. Fu creato originalmente da Netscape Corporation, ma è stato poi esteso e standardizzato dalla IETF con la creazione dello standard Transport Layer Security (TLS) definito nella [RFC 2246].

Scopo primario di SSL è fornire sistemi di crittografia per comunicazioni affidabili e riservate sulla rete sfruttabili in applicazioni quali, ad esempio, posta elettronica e sistemi di autenticazione. SSL non ha implementazioni proprietarie di crittografia, ma usa internamente gli standard più diffusi. Il protocollo SSL provvede alla sicurezza del collegamento garantendo:

- Autenticazione fra le parti: l'identità nelle connessioni può essere autenticata usando la crittografia asimmetrica, ovvero a chiave pubblica (RSA, DSS, EL-Gamal). Così ogni client comunica in sicurezza con il corretto server, prevenendo ogni interposizione. È prevista la certificazione del server e, optionalmente, quella del client, attraverso l'inserimento di certificati forniti da un'autorità di certificazione come, ad esempio, Verisign;
- Confidenzialità nella trasmissione dei dati: la crittografia è usata dopo un handshake (accordo) iniziale per definire una chiave segreta di sessione, che viene usata per la successiva crittografia simmetrica (AES, 3DES, RC4, ecc.);
- Integrità dei messaggi: il livello di trasporto include un controllo dell'integrità del messaggio basato su un apposito MAC (Message Authentication Code), che utilizza funzioni hash sicure (SHA, MD5, ecc.). In tal modo si verifica che i dati spediti tra client e server non siano stati alterati durante la trasmissione.

Come mostrato in figura 8.25, SSL opera inserendo uno strato software tra i livelli applicativi ed il TCP/IP. Tale strato ha il compito di cifrare il contenuto dei dati prima che essi passino nel canale di trasmissione e di decifrarli a destinazione. In tal modo tutto il contenuto dei pacchetti TCP/IP in transito lungo la rete è crittografato e, nel contempo, gli strati superiori applicativi non vedono nemmeno l'operazione di cifratura/decifratura. Grazie a questa caratteristica SSL è entrato come strato di protezione crittografica in moltissimi protocolli esistenti, creando, ad esempio, SSL-Telnet e HTTPS, l'implementazione sicura basata su SSL di HTTP, sicuramente la forma più usata di SSL.

Esistono molte implementazioni commerciali e non di SSL, ma la più diffusa è sicuramente quella open source e freeware reperibile presso [OpenSSL 2006].

Ulteriori informazioni su SSL sono disponibili in [SSL 1996].

Un protocollo applicativo per la protezione dei canali: SSH

Secure Shell (SSH), shell sicura è un protocollo che permette di stabilire una sessione di collegamento remoto cifrata ad interfaccia a linea di comando con un altro host. Inizialmente nasce negli anni '90 per rimpiazzare i protocolli di terminale remoto non crittografati telnet e rlogin, assumendo tutte le proprietà di rlogin, comprese quelle di esecuzione di comandi remoti e di copia di singoli file, come schematizzato in figura 8.26. L'evoluzione progressiva ha poi arricchito SSH di nuove funzionalità, fra cui le più importanti, presenti nella attuale versione 2, sono il trasferimento remoto completo

di file e cartelle (attraverso il servizio SFTP, praticamente un analogo cifrato di FTP) e il port forwarding, attraverso cui SSH permette di realizzare dei tunnel criptati, che permettono di trasportare sessioni TCP arbitrarie all'interno della connessione criptata, permettendo di proteggere da intercettazione protocolli non sicuri, o di aggirare limitazioni di routing. Questa ultima funzionalità consiste nell'aprire una socket TCP sul client SSH (local port forwarding) o sul server (remote port forwarding). Le connessioni ricevute su questa porta vengono inoltrate dall'altro capo della connessione SSH, verso un host e una porta specificata, potenzialmente anche diversi da quelli sui cui agiscono client e server SSH, come schematizzato in figura 8.27.

Rispetto a SSL, protocollo normalmente disponibile solo in forma di librerie, SSH è un applicativo vero e proprio, disponibile su varie piattaforme come server (che offre la possibilità di connettersi) e come client (che crea la connessione verso la macchina remota). In particolare è possibile fare passare attraverso il tunnel cifrato anche le connessioni di terminal server di Windows (o protocolli equivalenti come VNC) o le connessioni X-Window del mondo Unix, realizzando quindi facilmente un terminale grafico remoto protetto da crittografia.

Le implementazioni più diffuse di SSH sono le seguenti:

- SSH server e client (implementazione commerciale) di SSH Corporation, disponibile su tutti gli Unix, su Linux, Windows e IBM z/OS (si veda [SSH Corp 2006] per approfondimenti);
- SSH server e clienti di WRQ e F-Secure corporation (implementazione commerciale, derivata dalla precedente), disponibile su Unix e Windows (si veda [WRQ 2006] per approfondimenti);
- OpenSSH, implementazione freeware, disponibile su tutti gli Unix e, solo per il client, anche su Windows, perfettamente interoperabile con le precedenti (si veda [OpenSSH 2006] per approfondimenti); internamente questa implementazione usa direttamente la versione freeware di SSL definita sopra come libreria;
- Inoltre SSH server viene usato direttamente entro apparati di rete come i router CISCO per garantire la possibilità di amministrazione remota sicura.

Come SSL, SSH si basa sullo scambio di chiavi asimmetriche per l'autenticazione ed usa una chiave di sessione per la cifratura in tempo reale dei dati in transito. Anche SSH può usare internamente vari standard di crittografia, come, per esempio, Blowfish o Triple-DES.

Reti private Virtuali

Una rete privata virtuale, molto spesso indicata con il termine VPN (acronimo di Virtual Private Network) è un insieme di computer collegati fra loro attraverso una rete pubblica di comunicazione come Internet, in cui i collegamenti sono protetti da sistemi di autenticazione e crittografia, in modo da garantire lo stesso grado di sicurezza che si avrebbe con una rete privata di interconnessione. Il vantaggio principale delle VPN, rispetto alle reti dedicate, sta nel costo: in luogo del noleggio di costosi canali di comunicazione attraverso aree geografiche, è sufficiente disporre di un accesso permanente ad Internet, presso le sedi da collegare fra di loro, e di un

sistema di crittografia ed autenticazione robusto per stabilire la protezione dei dati e la autenticazione degli accessi.

Le VPN possono operare tipicamente in due modalità:

- nella configurazione da un singolo computer verso una rete aziendale (la cosiddetta configurazione del “road warrior”), rappresentata in figura 8.28;
- nella configurazione tra due reti aziendali, rappresentata in figura 8.29.

E' possibile realizzare VPN attraverso le tecniche sinora viste, per esempio usando SSH (si veda a tal proposito [Destri 1998]), o attraverso altri protocolli come OpenVPN, che usa internamente SSL (si veda [OpenVPN 2006] per approfondimenti). Ma esiste una soluzione che opera a livello di rete, usata spesso dalle aziende di telecomunicazioni per offrire il servizio di VPN ai propri clienti, basata sulla tecnologia IP Security (IPsec).

IPsec è uno standard per ottenere connessioni basate su reti IP sicure, sviluppato per la versione 6 del protocollo IP e poi implementato come servizio aggiuntivo anche nell'attuale IP versione 4. La sicurezza viene raggiunta attraverso la cifratura e l'autenticazione dei singoli pacchetti Ip e, quindi, direttamente a livello di rete. La capacità di fornire protezione a livello di rete rende questo protocollo trasparente al livello delle applicazioni che non devono essere modificate. Internamente anche Ipsec richiede una fase di scambio chiavi e poi l'uso di una chiave simmetrica per la cifratura, che però avviene a livello di pacchetti IP. Per approfondimenti su Ipsec si consiglia [Ipsec doc 2006].

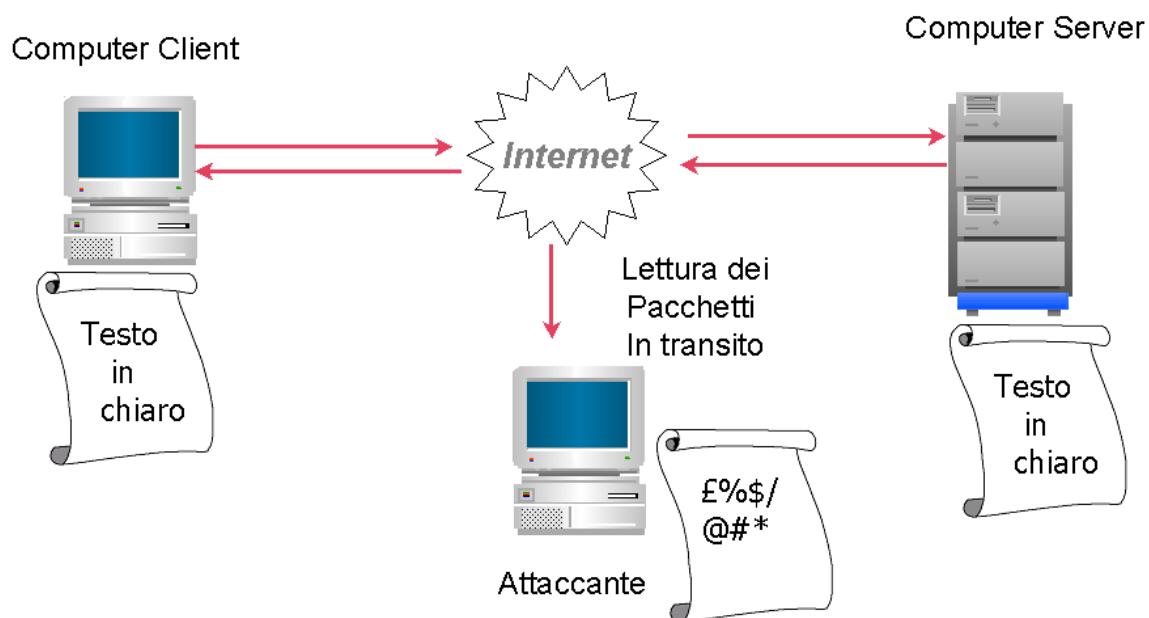


Figura 8.24: Schema della protezione critografica dalle intercettazioni lungo un canale di comunicazione. L'attaccante, se non in possesso della chiave, non ricava dati utili dalla propria intercettazione.

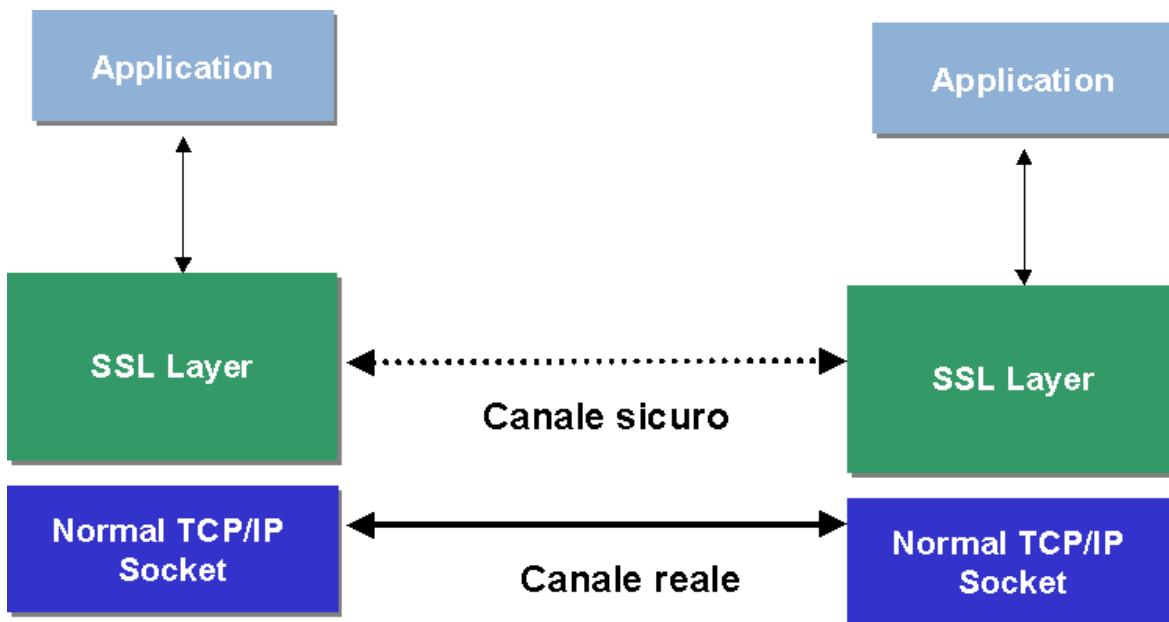


Figura 8.25: Azione di SSL entro lo stack TCP/IP.

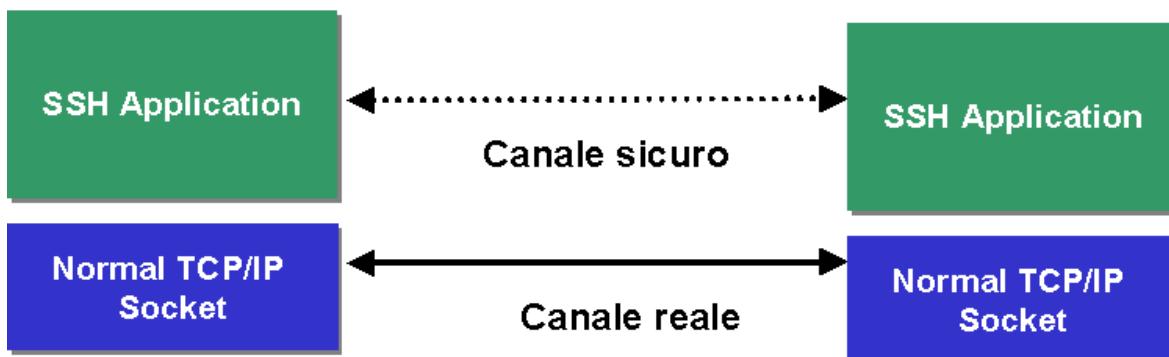
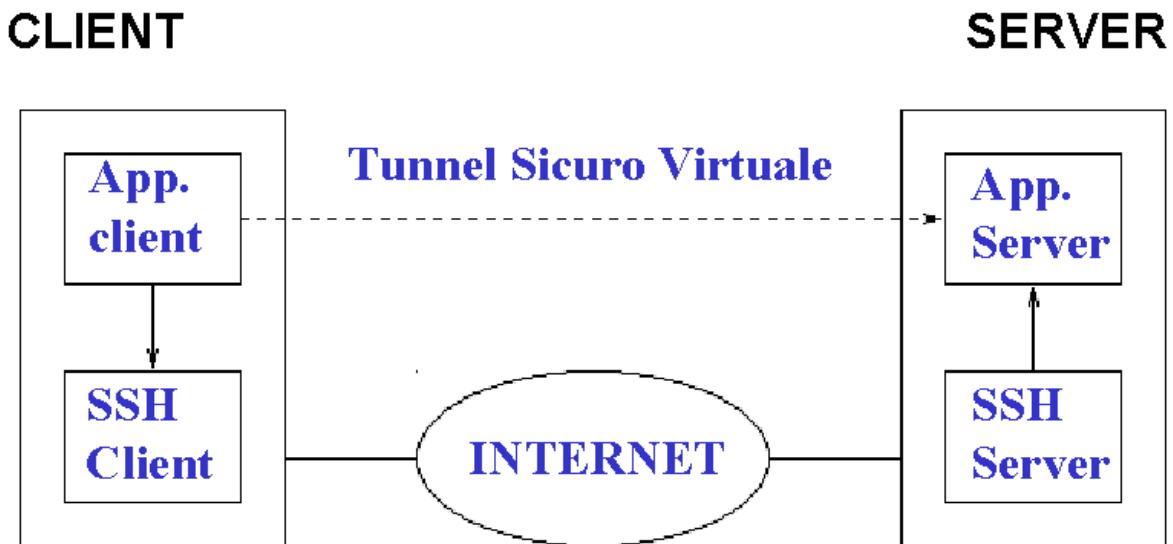


Figura 8.26: Schema del funzionamento di SSH. Viene garantita una comunicazione sicura tra le applicazioni SSH client e server.



PORT FORWARDING

Figura 8.27: Schema del funzionamento del tunneling di SSH. L'applicazione client vede un proxy locale della vera applicazione server attraverso il client SSH. I pacchetti inviati al client SSH sono da questo cifrati, incapsulati entro pacchetti SSH e spediti al server SSH, che provvede ad estrarli e ad inviarli alla vera applicazione destinazione. Le risposte percorrono il cammino inverso.

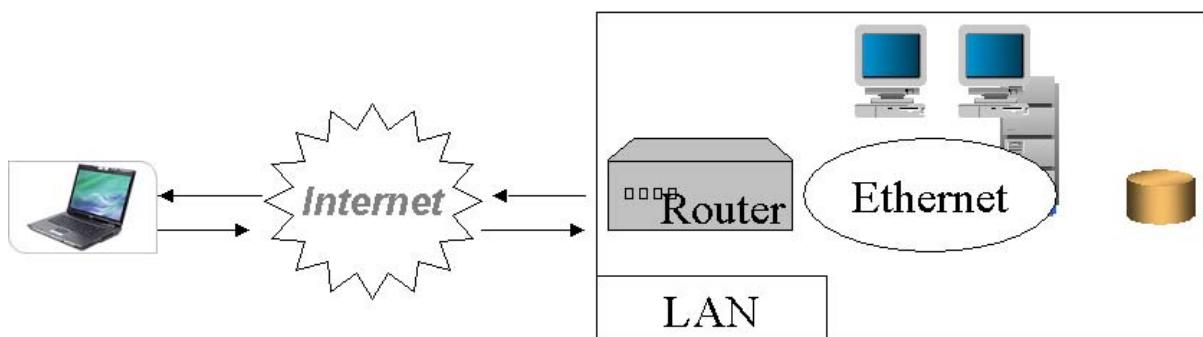


Figura 8.28: VPN nel collegamento “road warrior” da un singolo computer verso una rete aziendale. Il PC entra a fare parte della rete locale vedendola attraverso il tunnel cifrato della VPN. Tutte le porte e quindi anche tutti i servizi vengono fatti passare attraverso il tunnel.

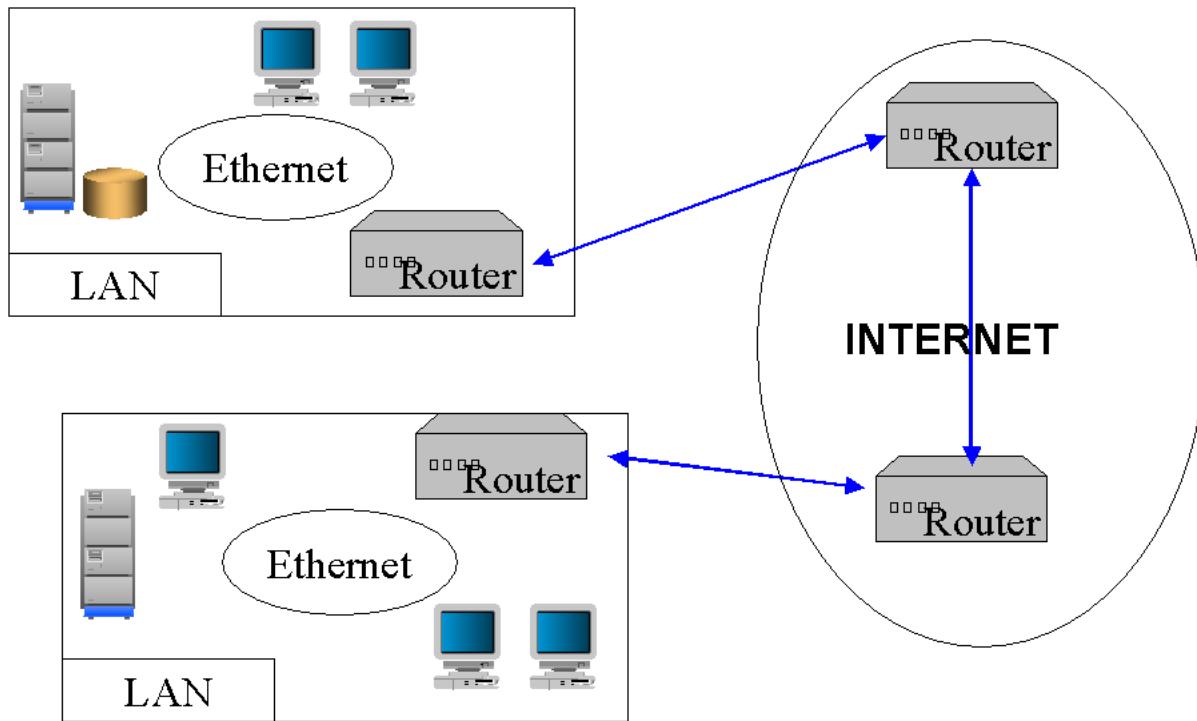


Figura 8.29: VPN nel collegamento tra due reti aziendali. Il passaggio può anche essere molto complesso, ma per i computer interni alle due reti, a parte la velocità di collegamento, il passaggio è del tutto trasparente.

La protezione dei sistemi

Fino ad ora sono state trattate le tecniche per la protezione dei dati sia entro i sistemi sia in transito lungo le reti. Ma, come già visto nella parte sulla safety, proteggere i dati non è sufficiente. Un sistema non in grado di erogare la sua funzione entro l'azienda o l'organizzazione rappresenta per questa un danno economico, potenzialmente notevole. Pertanto il paragrafo corrente è dedicato alle tecniche per la protezione dei sistemi dagli attacchi.

Ricordiamo che per intrusione in un sistema si intende normalmente una delle tre azioni seguenti:

- Ingresso non autorizzato in un sistema (login)
- Accesso non autorizzato a servizi e/o dati di un sistema
- DoS di un sistema

Lo strumento maggiormente usato per la protezione dei sistemi dalle intrusioni è il **firewall**. Secondo la definizione della National Computer Security Agency, l'ente governativo statunitense che si occupa di sicurezza informatica, un firewall è un “sistema o gruppo di sistemi che impongono una politica di controllo degli accessi fra due reti che comunicano fra loro”.

Esistono tre categorie principali di firewall:

- firewall personale (in inglese personal firewall), operante come applicazione su un singolo computer, che, interagendo con le librerie di rete TCP/IP presenti

entro il computer stesso, verifica quali programmi accedono alla rete in trasmissione o ricevono dati su determinate porte e permette le comunicazioni solo ai programmi abilitati entro il suo insieme di regole;

- firewall interno ad un'organizzazione, che separa fra di loro le sezioni, non consentendo, ad esempio, accessi diretti ad un database server a chi sta dall'altra parte del firewall;
- firewall di confine (detto anche di rete), che separa una rete aziendale dal mondo esterno, ovvero dal collegamento verso Internet e lascia passare, in un senso o nell'altro, soltanto il traffico abilitato in base alle regole in esso contenute; questa configurazione è mostrata in figura 8.30.

Le ACL (ACcess List) sono le regole che stabiliscono il funzionamento dei filtri del firewall: ogni pacchetto che deve attraversare il firewall viene valutato in base alle ACL e respinto o fatto passare

Sono possibili due politiche generali per le ACL:

- Tutto ciò che non è esplicitamente negato è permesso (default allow);
- Tutto ciò che non è esplicitamente permesso è negato (default deny), oggi usata normalmente.

Vengono considerate varie informazioni per trasmettere/bloccare un pacchetto:

- Indirizzo IP di origine dei dati
- Indirizzo IP di destinazione
- Tipo di protocollo trasporto: TCP, UDP, ICMP
- Porta di origine e di destinazione (servizio di destinazione)
- Se il pacchetto indica una richiesta di connessione
- Quale applicativo ha trasmesso o deve ricevere il pacchetto (informazione disponibile solo se il firewall opera sulla stessa macchina dell'applicativo o fa parte di un sistema di network monitoring)
- Tipo di protocollo applicativo, per esempio HTTP, SSH... (disponibile solo nei firewall più avanzati)
- Se il pacchetto fa parte della sessione di un utente con privilegi o no (blocco selettivo degli indirizzi sulla base di livelli di accesso).

Per definire un piano di sicurezza che contempi l'uso di un firewall, è importante anche ricordare quali funzioni non sono svolte da un firewall “puro”, ovvero non facente parte di una suite integrata di sicurezza comprendente, per esempio, anche un anti virus:

- Garantire l'integrità dei dati
- Proteggere dai virus
- Proteggere da disastri
- Autenticare le fonti dei dati
- Garantire la riservatezza dei dati

Inoltre a livello aziendale per potere definire le ACL che governeranno il funzionamento del firewall, ed anche scegliere il tipo di architettura e/o il prodotto, occorre conoscere il traffico associato ai flussi informativi dei processi business che lo dovrà attraversare. Per questo può essere utile la seguente lista di interrogativi:

- Gli utenti Internet devono poter prelevare file dal/dai server della rete?
- Gli utenti Internet devono poter inviare file al/ai server della rete?
- Devono esistere sbarramenti selettivi per determinati utenti e/o per determinati host?
- Esiste un sito Web interno alla rete accessibile da Internet?
- Devono essere fatte sessioni di terminale (SSH, telnet o terminal server) attraverso il firewall (in un senso o nell'altro)?
- Che protocolli devono passare attraverso il firewall?
- Che livello di accesso a Internet e al Web devono avere i dipendenti, eventualmente suddivisi in livelli di privilegio diversi?
- Che risorse umane si possono/devono impiegare per la gestione/verifica del firewall?
- Cosa può succedere alla rete, se un hacker riesce comunque ad entrare?

Per quanto riguarda le azioni tecniche di un firewall, esistono diverse tipologie, da cui dipendono le possibilità del firewall stesso:

- Firewall **packet filter**, che si limitano a valutare gli header di ciascun pacchetto, decidendo quali far passare e quali no sulla base delle ACL configurate; alcuni di essi valutano anche se esiste il flag di richiesta di connessione negli header e ammettono o non ammettono questo tipo di pacchetti per alcune porte TCP in base a quanto scritto nelle ACL;
- Firewall **stateful inspection**, che tengono traccia di alcune relazioni tra i pacchetti che li attraversano, per esempio, ricostruendo lo stato delle connessioni TCP, o i protocolli che aprono più connessioni; ciò permette, ad esempio, di riconoscere pacchetti TCP malevoli che non fanno parte di alcuna connessione, o di garantire comunque il funzionamento di protocolli complessi;
- Firewall **Application Layer Gateway**, che effettuano controlli fino al livello applicazione dello stack ISO/OSI; tra di essi sono anche i proxy, che si usano quando la configurazione della rete privata non consente connessioni dirette verso l'esterno, ma tutto avviene attraverso il proxy, connesso sia alla rete privata sia alla rete pubblica, che permette alcune connessioni in modo selettivo, e solo per i protocolli che supporta; i proxy possono essere trasparenti, nel qual caso le applicazioni che accedono alla rete “pensano” di essere connesse direttamente all'esterno e non devono essere riconfigurate, o non trasparenti, nel qual caso le applicazioni devono essere riconfigurate per poter usare i proxy.

Spesso connesse con il firewall sono le funzionalità di Number Address Translator (NAT) e Port Address Translator (PAT) che rimappano, rispettivamente, un numero IP della interfaccia esterna, come mostrato in figura 8.31, e una porta dell'esterno, come mostrato in figura 8.32, su una macchina interna alla rete. Attraverso queste funzionalità operano i firewall **a livello circuito**, entro i quali vengono realizzati i cosiddetti “tunnel bloccati”, ossia regole che ruotano tutti i pacchetti inviati su una porta di una delle interfacce verso una porta su un indirizzo IP nella rete dall'altra parte.

Per la realizzazione di un firewall si può avere un dispositivo hardware, essenzialmente costituito da un computer dedicato, che ospita solo il software del

firewall, o una implementazione software, in cui il computer ospita il software che svolge le funzioni di firewall di rete entro un sistema operativo, e nello stesso tempo il computer svolge anche altri compiti (ad esempio il server di posta).

I firewall di rete più usati sono nella configurazione dual-home host, mostrata in figura 8.32: il computer che ospita il firewall ha due interfacce di rete, una esterna e l'altra interna, il routing diretto fra le due schede è disabilitato ed i pacchetti passano attraverso lo strato software del firewall e sono soggetti alle regole delle ACL. Questo computer è talvolta chiamato anche bastion host, termine con cui si indica un computer della rete “particolarmente preparato a respingere attacchi contro la rete stessa” (bastione).

Altre due configurazioni possibili coinvolgono l'uso degli screening router, ossia router che esplicano anche l'azione di filtraggio selettivo dei pacchetti in transito in base alle ACL in essi programmate e che, quindi, si comportano essenzialmente da firewall a livello pacchetto, anche se raramente hanno regole che coinvolgono anche le porte TCP.

Nella prima, chiamata anche firewall screened-host, mostrata in figura 8.33, il firewall non è connesso direttamente ad una rete esterna, ma si trova dietro a uno screening router.

Nella seconda, chiamata anche firewall screened-subnet, mostrata in figura 8.34, il firewall si trova nella sottorete fra due screening router. Questa configurazione è la più sicura.

I firewall possono operare anche a protezione e separazione di più reti, come mostrato in figura 8.35. Nel caso poi vi siano server Web e/o di posta, o comunque che ospitano vari servizi e devono essere acceduti dall'esterno, normalmente il firewall ha più di due interfacce di rete e questi server non sono nella LAN, ma in una rete da essa disgiunta, chiamata anche zona demilitarizzata, come mostrato in figura 8.36. Nella stessa figura è mostrata anche la presenza di una ulteriore zona separata per gli accessi wireless, per garantire la migliore protezione contro i vari tipi di attacchi.

Oltre che operare nei livelli alti attraverso i firewall, la protezione può agire a vari livelli dello stack TCP/IP, come mostrato sotto:

- Applicazione: firewall applicativo / proxy
- Trasporto e Rete: firewall packet filter e screening router
- Collegamento dati: smart hub e switch
- Livello fisico: interruttori per cavi e schermature radio nel caso di wireless.

La protezione garantita dal firewall e/o da sistemi ai livelli inferiori di rete può non essere sufficiente. Negli ultimi anni sono apparsi sul mercato anche alcuni dispositivi che al firewall aggiungono nuove funzionalità di monitoraggio del traffico, chiamate anche difese proattive. In pratica, in caso di attacco o di riconoscimento del passaggio dei dati che definiscono la firma del passaggio di un worm o altro attacco catalogato (il confronto viene fatto in modo analogo alla ricerca di un virus), non viene bloccato tutto il traffico relativo alla porta attaccata o tutto il traffico IP della rete di partenza, ma, automaticamente, solo il traffico proveniente dall'IP sorgente e destinato a quella porta. Viene inoltre riconosciuto l'IP dell'attaccante, e qualora l'attacco continui, tutto il traffico da esso proveniente viene bloccato. In pratica la difesa viene

automaticamente commisurata al tipo di attacco e le stesse ACL si evolvono nel tempo adattandole dinamicamente alla situazione che si viene a creare. I dispositivi che analizzano il traffico possono anche non agire attivamente. In entrambi i casi tali dispositivi vengono chiamati anche Intrusion Detection System (IDS). Diverse aziende producono IDS, una delle più famose è la statunitense Internet Security Systems [ISS 2006].

Esistono comunque casi in cui anche le difese proattive non bastano. Occorre aggiungere alla protezione dell'ingresso nella rete anche la protezione dei sistemi. Serve quindi inserire le funzioni di monitoraggio della rete o network monitoring. Questa azione può essere rivolta al:

- Monitoraggio del traffico e dei suoi flussi (il network monitoring propriamente detto)
- Monitoraggio del carico dei sistemi, per scoprire eventuali condizioni insolite sulle macchine:
 - Numero utenti collegati via servizi (Telnet, FTP, connessioni disco, connessioni al DBMS, ecc...)
 - Numero di connessioni socket TCP/IP aperte
 - Carico della macchina (CPU, RAM)
 - Spazio disco disponibile su varie partizioni
- Monitoraggio dei processi, per scoprire eventuali processi dovuti a una sessione non autorizzata o un trojan in esecuzione

In generale dovrebbero essere controllate tutte le operazioni più importanti che avvengono sia a livello di rete sia a livello dei singoli server, ciò presuppone un'accurata analisi dei log. Con analisi statistiche dei risultati dovrebbero risultare evidenti attività "anomale". Per avere un controllo ancora migliore, bisogna andare oltre, cercando di risalire a "chi ha fatto un'attività insolita e perché" (potrebbe essere perfettamente lecita). L'accounting, ossia l'attività di riconoscere, tracciare ed addebitare le azioni di calcolo, riprenderà spazio man mano che si diffonderà l'outsourcing dei sistemi informatici, in quanto strumento indispensabile per conteggiare i pagamenti dei servizi di calcolo.

Occorre osservare che un adeguato monitoraggio dei sistemi non protegge solo da problemi di security, ma anche di safety, come possono essere problemi fisici ad un disco (che normalmente non appaiono di colpo, ma preceduti da sintomi riconoscibili), o problemi "organizzativi", quali la saturazione dello spazio sul disco o l'insufficiente memoria per l'esecuzione delle applicazioni presenti in un server.

Normalmente le operazioni di verifica e monitoraggio non possono essere fatte "manualmente" nelle reti complesse, come, ad esempio, quelle delle compagnie di telecomunicazioni o delle banche, che possono comprendere centinaia di computer server ospitanti migliaia di applicazioni. Esistono software specifici per questo compito, i cosiddetti network & system monitors. Il mercato di questo tipo di applicazioni è dominato da Tivoli di IBM e Unicenter di Computer Associates (CA), ma anche altri produttori, come Oracle, stanno progressivamente entrando in questa fascia. Si vedano anche [Tivoli 2006] e [Unicenter 2006] per approfondimenti.

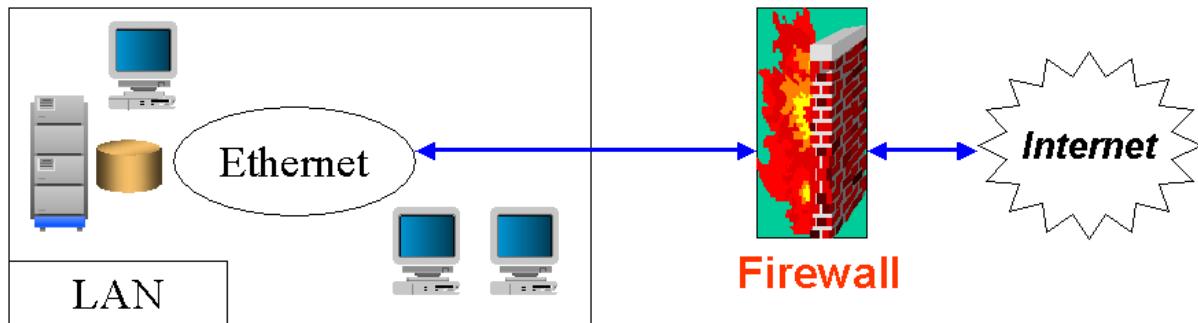


Figura 8.30: Un firewall operante per la protezione di tutta una rete aziendale. Tipicamente la rete è isolata e tutto il traffico passa attraverso il firewall.

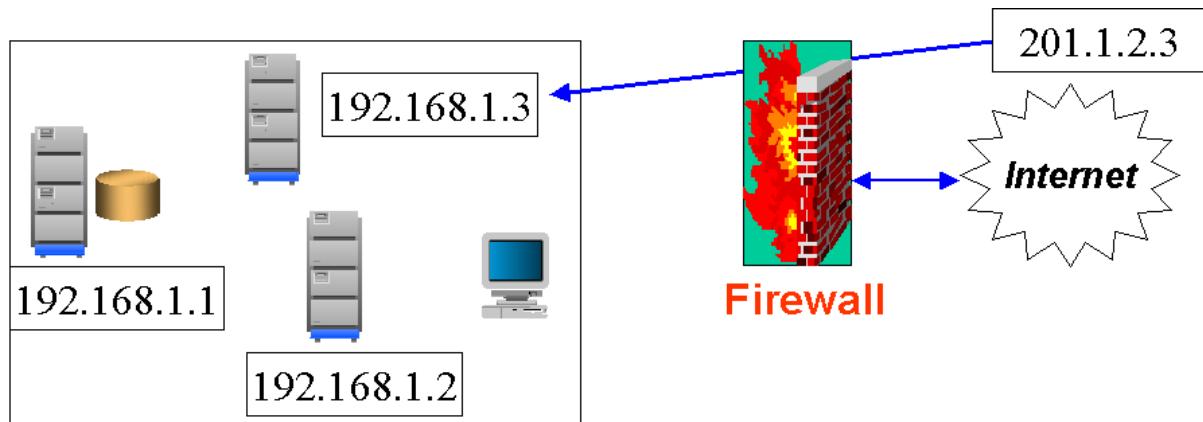


Figura 8.31: Il servizio Number Address Translator (NAT). Un tunnel fisso attraverso il firewall porta un IP mappato sulla interfaccia esterna del firewall verso un IP della rete interna. In tal modo gli host della rete interna vengono visti dall'esterno con un IP diverso dai loro.

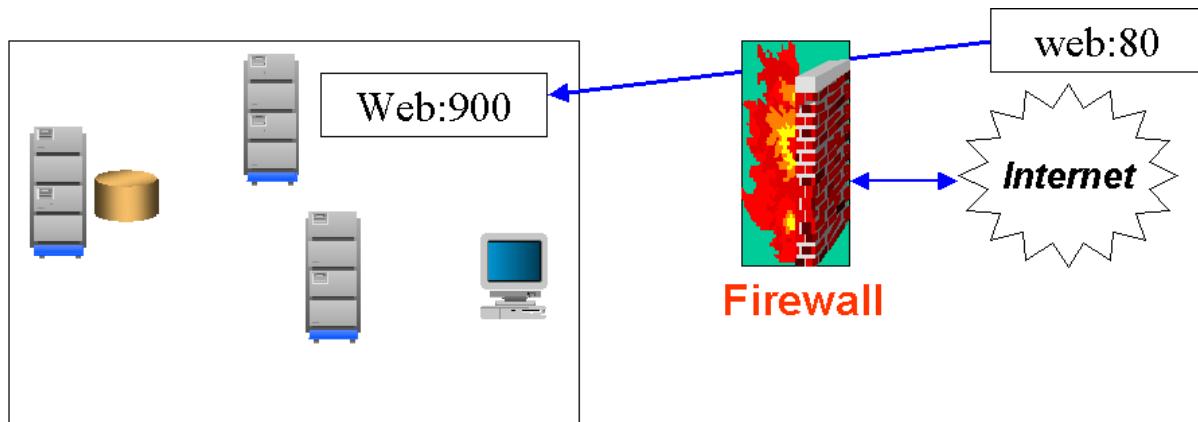


Figura 8.32: Il servizio Port Address Translator (PAT). Un tunnel fisso attraverso il firewall porta una porta TCP mappata sulla interfaccia esterna del firewall verso una porta di un IP della rete interna. Nell'esempio il server Web opera internamente sulla porta 900. La porta vista "esternamente" al firewall è diversa da quella reale interna alla rete.

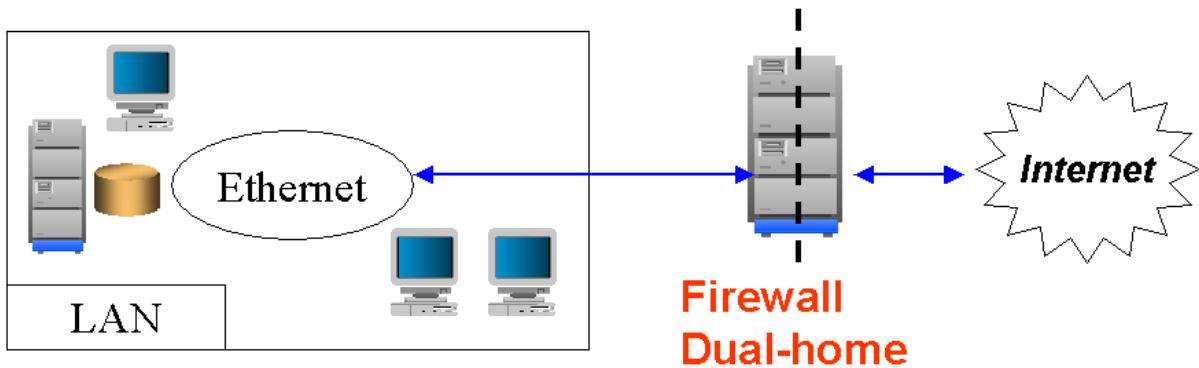


Figura 8.33: Un firewall in configurazione dual-home host. Una interfaccia di rete (interfaccia esterna o rossa) è connessa alla rete esterna, l'altra (interna o verde) alla rete interna. Se esistesse una terza interfaccia per la rete DMZ sarebbe anche chiamata arancione.

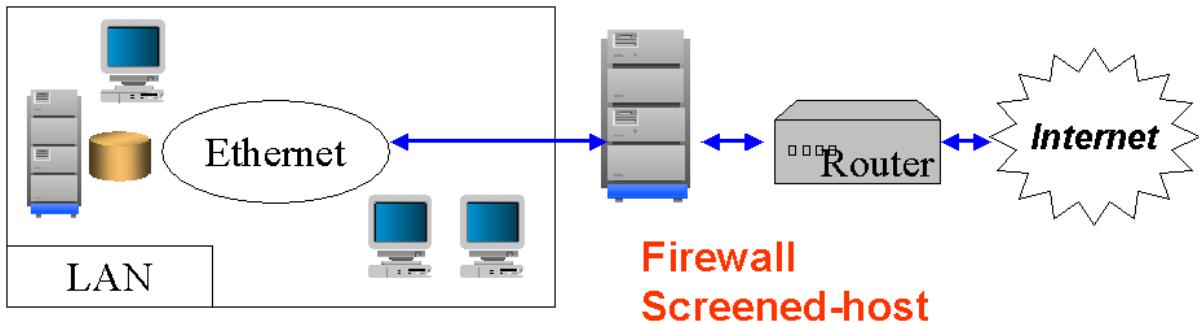


Figura 8.34: Un firewall in configurazione screened-host. Il firewall è separato dalla rete esterna da uno screening router.

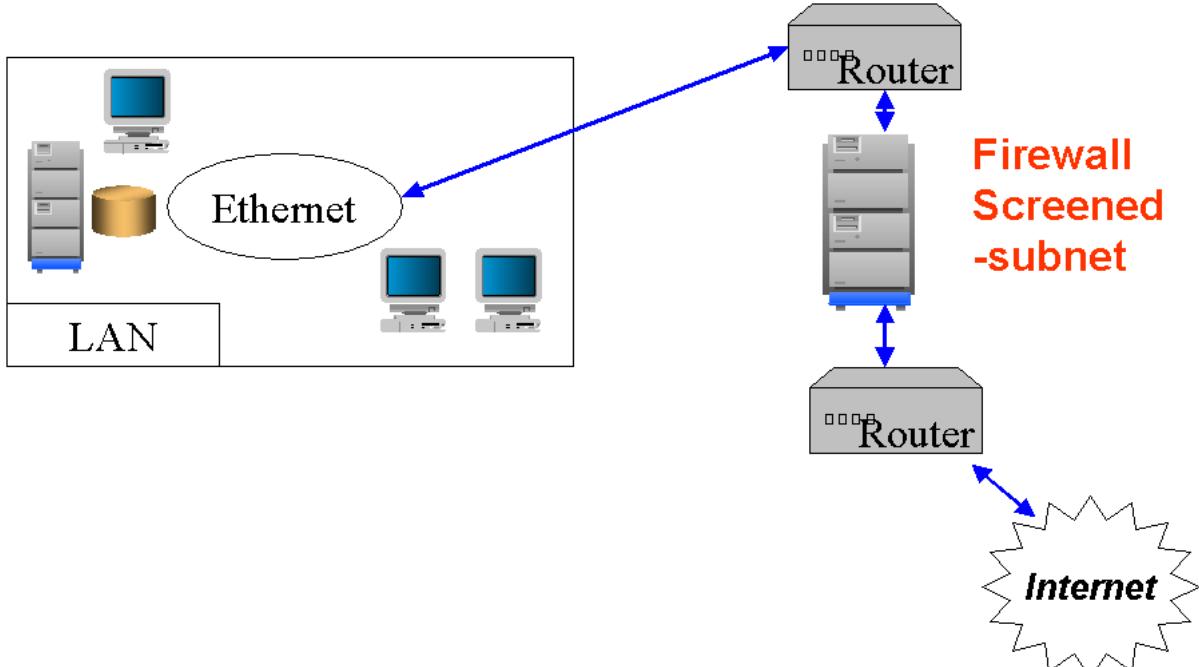


Figura 8.35: Un firewall in configurazione screened-subnet. Il firewall è separato, sia dalla rete esterna, sia da quella interna, da screening router; la rete dovrebbe avere differente classe IP rispetto alle altre due.

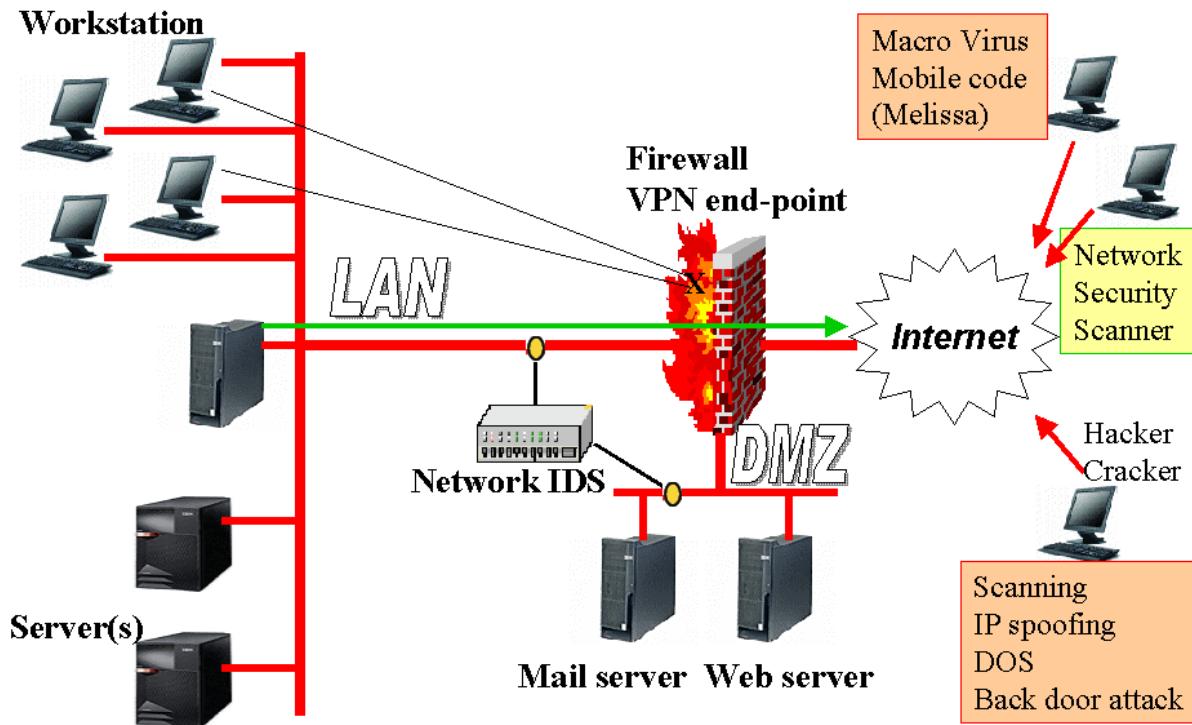


Figura 8.36: Struttura di una rete con vari sistemi di protezione applicati. Si noti la zona demilitarizzata (DMZ) contenente i server che sono liberi per l'accesso dalla rete esterna.

Gestire la sicurezza

Fino a questo momento sono state esaminate soltanto soluzioni tecnologiche per la protezione dei sistemi. Ma tali soluzioni, da sole, non sono mai sufficienti. Anche per la sicurezza occorre ricordare che il sistema informativo è composto non solo dalle risorse tecnologiche, ma anche dalle risorse organizzative, che nel caso della sicurezza devono prendere la forma di regolamenti e procedure per prevenire gli attacchi e i problemi, e, soprattutto, delle risorse umane, che devono applicare tali procedure.

Si parla pertanto di politiche di gestione dei sistemi in relazione alla sicurezza, che però devono essere inserite entro le politiche generali di uso, stabilendo il giusto compromesso fra sicurezza ed uso dei sistemi. Non si deve mai dimenticare che lo scopo dei sistemi informativi e della loro componente informatica è di essere di ausilio al business, ovvero di produrre valore per l'azienda attraverso il supporto dei processi aziendali.

Non esistono politiche di gestione buone per tutti i casi, ma al più linee guida generali da rispettare per stabilire le proprie politiche. La politica di gestione infatti deve essere decisa caso per caso, in funzione di tanti fattori:

- Scopo della rete e dei sistemi
- Tipologie di applicativi in uso
- Esperienza tecnica e pratica (in inglese skill) degli amministratori e degli utenti
- Politica del rischio stabilita in azienda

- Rapporto costi/benefici, tra le misure di sicurezza adottate e il loro costo, diretto e indiretto (costituito normalmente dall'impatto che esse hanno sulle esecuzioni dei processi e sulla qualità del lavoro) ed i benefici (sotto forma di minori rischi o di costi di problemi evitati) ricavati dalla loro adozione.

Esistono due modalità contrapposte fra loro nell'affrontare il problema sicurezza entro un'azienda o un'organizzazione.

La prima ha un approccio “militare”, avente per obiettivo la sicurezza “assoluta” entro l'azienda. La direttiva primaria è scoprire in anticipo tutti i tipi di attacco e prevenirli. Si ritiene che la tecnologia delle protezioni può risolvere i problemi e in generale i prodotti nuovi per la sicurezza sono sempre considerati migliori dei precedenti.

Ma questo tipo di approccio genera spesso seri problemi: i responsabili della sicurezza tendono a dire sempre di no alle richieste degli utenti, anche quando esse sono effettivamente motivate da necessità di business. La sicurezza, anche a causa del continuo aggiornamento dei sistemi, assorbe troppe risorse e, nel tempo, può divenire un problema per il business dell'azienda, interferendo con le attività.

Un esempio specifico di questo approccio può essere il caso di una grande azienda in cui operino molti gruppi di lavoro diversi. Se uno di questi ha bisogno di realizzare una VPN con un'altro gruppo operante presso un'altra azienda per scambio sicuro di dati e, senza valutare l'impatto effettivo sulla sicurezza e quindi le modalità con cui inserire tale VPN entro i sistemi per minimizzare il rischio, i responsabili di sicurezza vietano la cosa, ecco che si è prodotto un ostacolo per il business. Oltre al problema diretto potranno generarsi anche problemi indiretti, in quanto il gruppo, vistosi negata una esigenza legittima, tenderà a trovare una soluzione alternativa, magari “girando intorno” alle protezioni e creando potenziali buchi nella sicurezza.

La seconda modalità, molto più aderente alla concezione della gestione oculata dell'azienda, stabilisce un approccio legato alla **gestione del rischio**. La sicurezza è “relativa”, in quanto esistono tantissimi rischi e bisogna, per quanto possibile, tenerli in considerazione. E' utile osservare, a tal proposito, che la stessa attività imprenditoriale è sempre soggetta a rischi, ovvero che il rischio è insito negli affari. Incidenti durante un trasporto, interruzioni di corrente, eventi metereologici, terremoti, furti, crisi economiche ecc... sono sempre esistiti, ben prima dell'avvento dei computer. In questa accezione i rischi informatici sono solo altri rischi da tenere in considerazione, ma da approcciare sempre attraverso una loro valutazione consapevole. Sono sempre esistiti gli incidenti: le aziende previdenti si riprendono e vanno avanti. Esistono anche molte soluzioni, ma la loro efficacia ed applicabilità dipendono fortemente dal contesto.

Per questo l'obiettivo di una gestione del rischio consiste nella sua riduzione, ovvero nel computo del rapporto ottimale costi/benefici sopra definito, come schematizzato in figura 8.37.

L'approccio può essere basato su tre punti:

- Ridurre il rischio con la tecnologia, inserendo nei sistemi informatici le protezioni tecnologiche più opportune rispetto alla situazione esistente;
- Ridurre il rischio con procedure opportune, garantendo l'uso migliore possibile degli strumenti inseriti da parte delle risorse umane;

- Ridurre il rischio trasferendolo ad entità esterne all'azienda, attraverso la stipula di assicurazioni, oppure demandando l'intera attività di gestione dei sistemi informatici o alcune sue parti ad aziende esterne (outsourcing), i cosiddetti **service provider**; il rischio viene allora quantificato attraverso il premio della polizza assicurativa o la quota del canone di servizio che lo copre.

In molti casi reali la situazione è intermedia fra i due: entro la stessa azienda od organizzazione possono esservi, magari in sezioni diverse o in situazioni operative diverse, entrambi gli approcci.

La gestione della sicurezza richiede allora di individuare, nei limiti del possibile, le caratteristiche di sicurezza di sistemi e reti. In alcuni casi, ad esempio nel caso di un collegamento radio diretto fra due utenti, il pericolo è facilmente individuabile e costituito dalla sola intercettazione. Un appropriato sistema di crittografia ed autenticazione copre in modo sufficiente rispetto a tale pericolo. Nel caso generale di una comunicazione in Internet invece la situazione è molto più complessa ed i pericoli aumentano di conseguenza. Un esempio è mostrato nella figura 8.38, dove i due utenti, Alice e Bob, che devono comunicare fra di loro attraverso le reti locali delle rispettive aziende e la rete Internet pubblica sono soggetti a tutte le tipologie di rischio definite nei paragrafi precedenti, e che possiamo riassumere come segue:

- I computer di Alice e Bob possono essere infettati da virus di ogni tipo;
- Le reti locali possono essere soggette ad attacchi all'organizzazione, che, ad esempio, possono ingannare gli utenti facendo loro rivelare password o altri dati personali;
- I file server delle reti locali possono essere attaccati ed i file compromessi;
- I server di autenticazione (nel mondo Windows costituito dal server di dominio con gli archivi delle Active Directory) possono essere attaccati e le identità elettroniche (certificati, chiavi....) in essi contenute rubate;
- I DNS possono essere attaccati ed alterati;
- Router e gateway possono essere attaccati ed alterati e/o posti fuori servizio (DoS);
- Tutte le reti locali sono soggette ad errori nelle configurazioni o addirittura nel software o nei sistemi operativi;
- La rete esterna è soggetta a vari pericoli (per esempio, intercettazione);
- Dalla rete esterna possono partire attacchi verso le strutture aziendali (per esempio, virus Web, DoS da sovraccarico ecc...).

Pertanto una valutazione accurata del rischio non può prescindere dalle seguenti conoscenze:

- Conoscenze dei processi business necessari per l'azienda e dei flussi informativi interni ed esterni all'azienda ad essi necessari;
- Conoscenza dei costi dei processi e del valore da essi generato;
- Conoscenza degli archivi dei dati necessari alle attività dei processi ed alla storizziazione dei dati per custodire il patrimonio informativo dell'azienda;
- Conoscenza delle procedure aziendali esistenti;
- Conoscenza delle risorse umane, dei loro skill e della loro attitudine al rispetto di procedure e regolamenti;

- Conoscenze dei programmi applicativi utilizzati nell'esecuzione delle attività
- Conoscenze dei sistemi informatici che ospitano tali programmi
- Conoscenze delle reti che connettono tra loro i sistemi informatici;
- Conoscenze (almeno a grandi linee) della struttura delle reti pubbliche ed altri elementi non sotto il controllo dell'azienda, che vengono attraversati dai flussi informativi.

Esistono parametri e livelli “precisi” per la sicurezza, stabiliti anche a livello internazionale, come, ad esempio, gli Orange Book del Dipartimento della Difesa (DoD) statunitense, o gli standard BS7799 inglese ed il derivato standard ISO 17799 (si vedano [Orange 2006], [BS7799 2006] e [ISO 17799 2006] per ulteriori dettagli), in parte recepiti anche dal già citato Decreto legislativo 196/2003, meglio noto come “Decreto privacy” [DL196 2003].

L'approccio per stabilire i livelli di pericolo e le contromisure può essere basato sul “buon senso” comune. Se in una rete informatica aziendale si è isolati dall'esterno in quanto il collegamento passa attraverso un firewall, è probabile che la maggior parte dei pericoli verrano dall'interno e saranno dovuti all'azione di virus o di dipendenti infedeli.

Quando l'azienda è grande, una sola protezione verso l'esterno può non bastare a garantire una buon livello di sicurezza. Conviene allora intervenire con il cosiddetto modello della sicurezza a cipolla o a strati, schematizzato in figura 8.39, dove sono mostrati i diversi livelli di protezione, dall'esterno passando per la rete locale, i server centrali ed i dati. In figura 8.40, viene mostrata una applicazione del modello a cipolla: una rete con due firewall, il primo a protezione di tutta la rete ed il secondo a protezione del solo nucleo di server interni.

La conoscenza del rapporto costi-benefici è una necessità: per potere quantificare in modo corretto il rischio servono metodologie di analisi che saranno definite nel prossimo capitolo, relativo alle metodologie di gestione di tutto il sistema informativo, al quale si rimanda per l'organizzazione delle politiche di gestione della sicurezza.

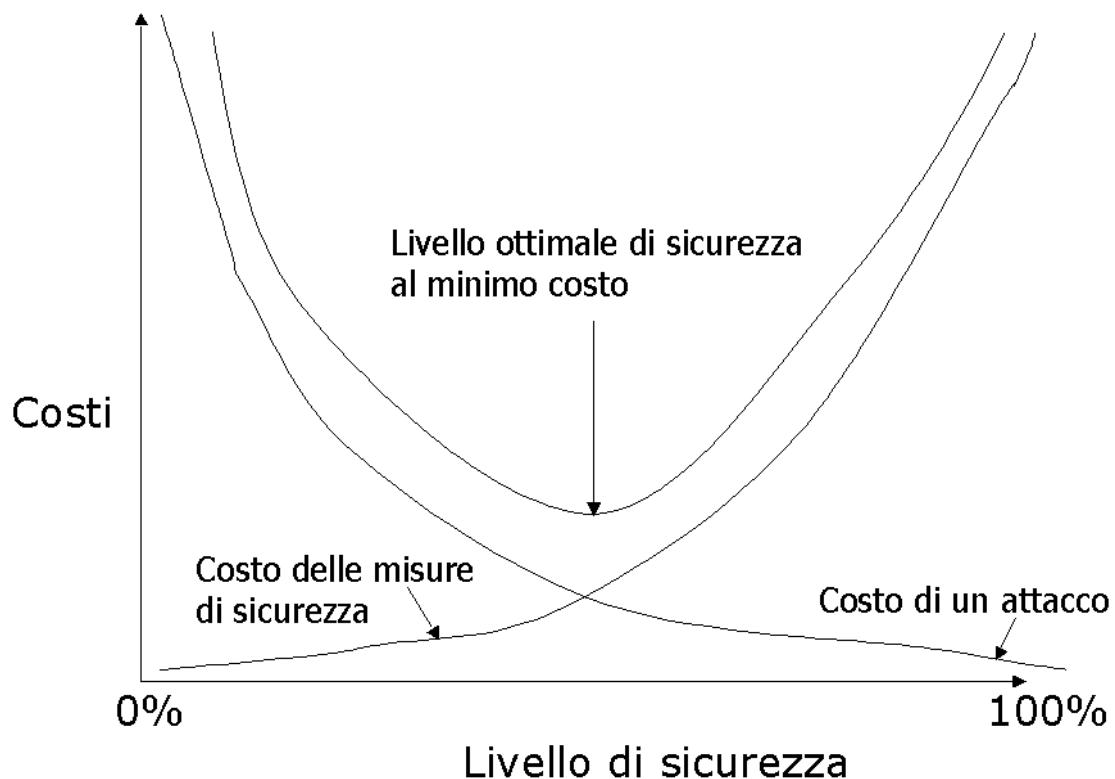


Figura 8.37: Ricerca del migliore compromesso costi-benefici nella gestione della sicurezza informatica.

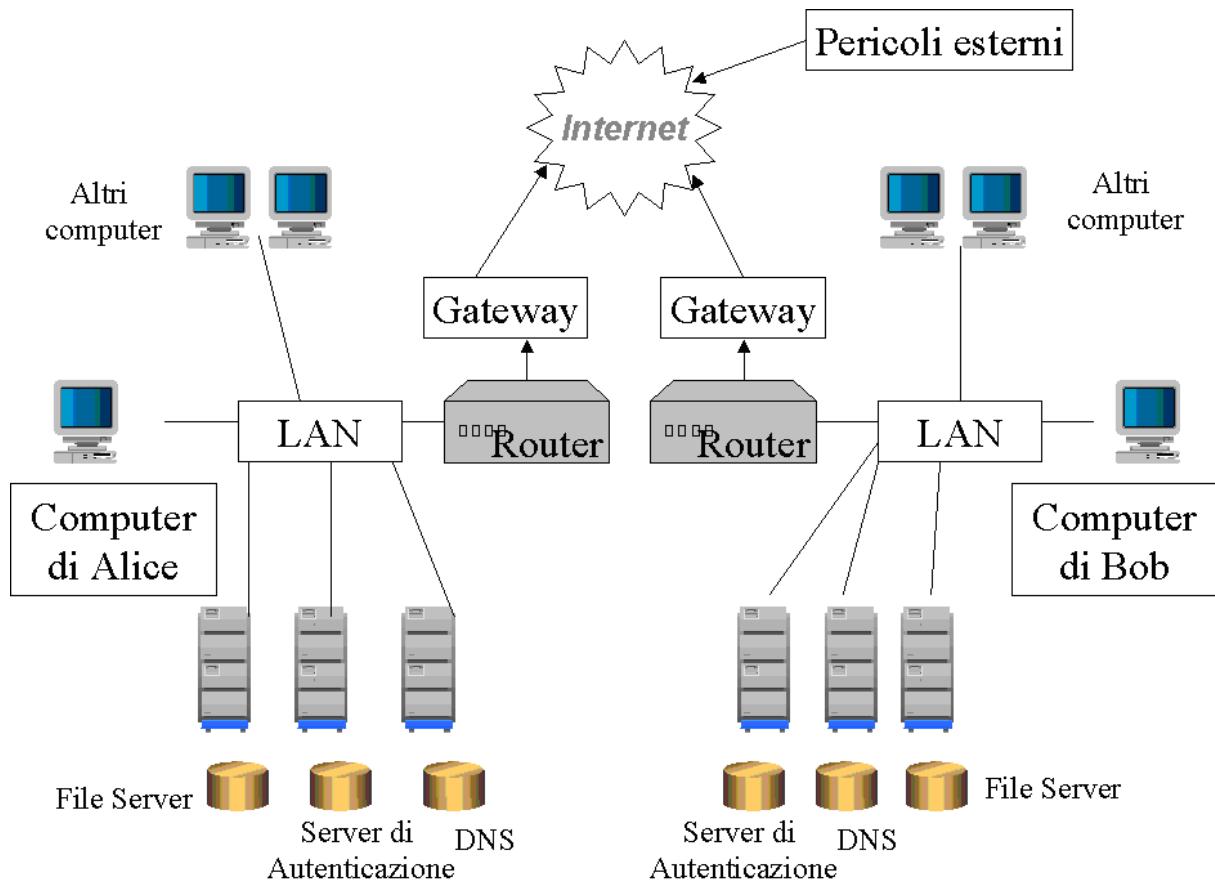


Figura 8.38: Schema riassuntivo degli elementi usati dagli utenti Alice e Bob nella loro comunicazione attraverso i propri PC, le proprie distinte reti aziendali e il collegamento Internet pubblico. Tali elementi sono soggetti alle vulnerabilità viste in precedenza e quindi Alice e Bob corrono i vari pericoli conseguenti a tali vulnerabilità.

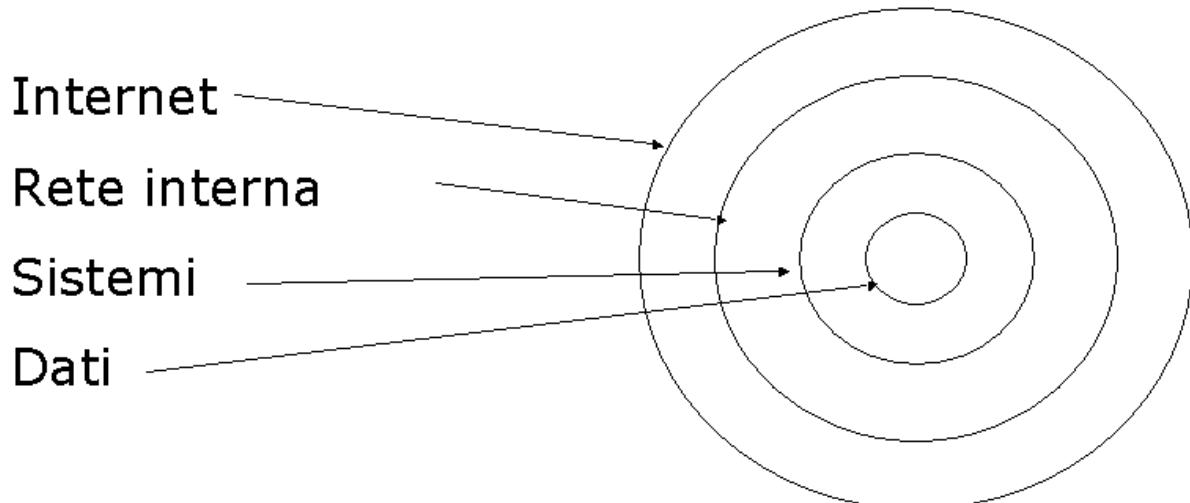


Figura 8.39: Il modello di sicurezza a cipolla, detto anche a strati. Con l'applicazione di questo modello il superamento della prima barriera (ad esempio, il firewall esterno) non conduce necessariamente al furto dei dati in quanto esistono ancora barriere da superare.

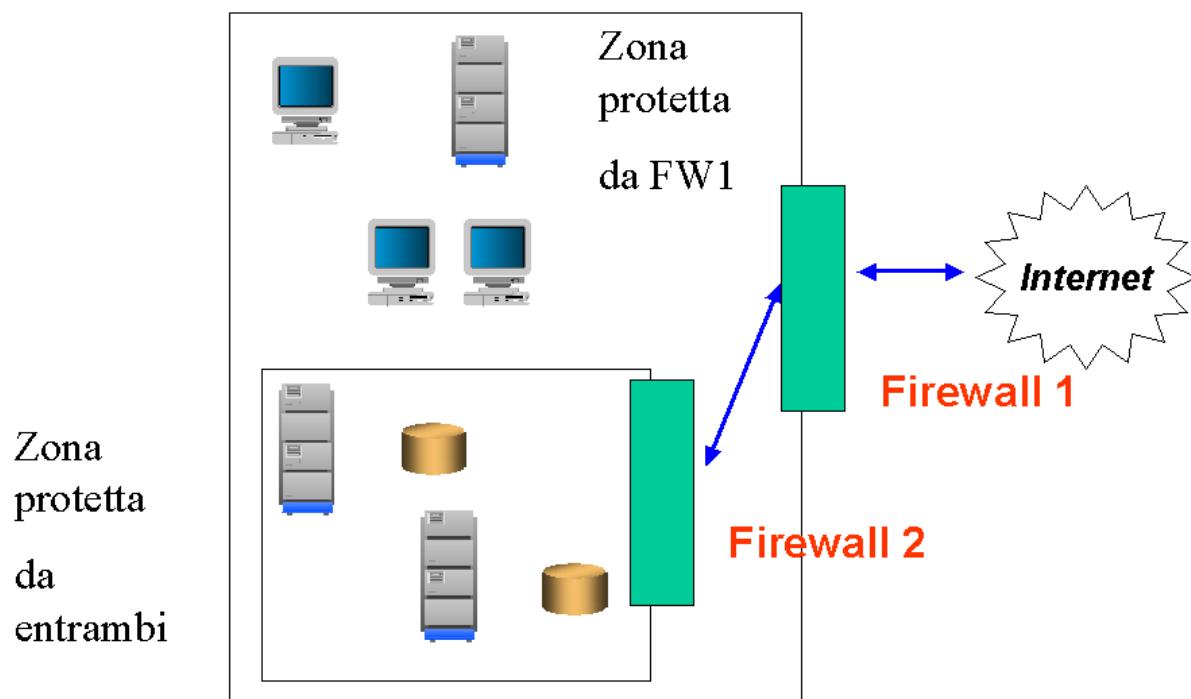


Figura 8.40: Applicazione del modello di sicurezza a strati: una rete con due firewall, di cui il secondo protegge i server.

Domande

1. Quali sono i problemi di safety? Come si può porvi rimedio?
2. Quali sono i problemi di security?
3. Quando la sicurezza può diventare un problema per il business?
4. Come si può proteggere una comunicazione su una rete pubblica?
5. Cosa si intende per intrusione in un sistema?
6. Cosa si intende per impersonificazione?
7. Cos'è il Denial-Of-Service?
8. Quali sono i pericoli che minacciano il servizio della posta elettronica?
9. In base a quali criteri si devono definire le regole di un firewall?
10. Cosa si intende per virus mutante? Che pericoli comporta per un sistema?
11. E' sufficiente l'uso della tecnologia per prevenire i problemi di sicurezza informatica?

Bibliografia

[BFM 2001] G. Bracchi, C. Francalanci, G. Motta - *Sistemi Informativi e aziende in rete* – Ed. McGraw-Hill Italia, Milano, 2001

[BM 2005] R. Barker e P. Massaglia - *Storage Area Networking Essentials: A Complete Guide to Understanding & Implementing SANs* - Ed. Wiley, 2005

[BS7799 2006] Siti Web dello standard BS7799

- <http://www.thewindow.to/bs7799/>

- <http://www.gammassl.co.uk/topics/hot1.html>

[Destri 1998] G. Destri. Una rete privata virtuale con F-Secure SSH, *LOGIN* n. 10, Edizioni Infimedia, Maggio/Giugno 1998.

[DL196 2003] Decreto Legislativo n. 196/2003 - "Codice in materia di protezione dei dati personali", su Web <http://www.camera.it/parlam/leggi/deleghe/Testi/03196dl.htm>

[EFS 2000] Encrypted File System: a survey, su Web http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/encrypt_overview.mspx

[Entrust 2006], documentazione su PKI presso la Entrust Corporation, su Web <http://www.entrust.com/pki.htm>

[GnuPG 2005] The GNU Privacy Guard. Un'implementazione della Free Software Fundation dello standard OpenPGP, sito Web <http://www.gnupg.org>

[GMN 1995] S. Gai, PL Montessoro, P. Nicoletti - *Reti Locali: Dal Cablaggio All'Internetworking* - Scuola Sup. G. Reiss Romoli, 1995

[Ipsec doc 2006] sito principale di Ipsec, su Web <http://www.ipsec-howto.org>

[ISO15408 2006] Siti Web dello standard ISO15408 sulla sicurezza dei prodotti:

- <http://www.iso15408.net/>
- <http://www.commoncriteriaportal.org/evaluation.html>
- <http://www.iso-standards-international.com/iso-5725-kit70.htm>

[ISO17799 2006] Siti Web dello standard ISO 17799

- <http://www.iso17799-web.com/>
- <http://www.iso17799.net/>
- <http://www.iso17799software.com/>
- <http://www.iso17799-made-easy.com/>

[ISS 2006] sito della società Internet Security Systems, su Web <http://www.internetsecuritysystems.com/>

[Klander 1998] L. Klander - *Hacker Proof, sicurezza in rete* - Ed. Jamsa Press/McGraw Hill, 1997-98

[LL 2004] K Laudon, J Laudon – *Management dei sistemi informativi* – Ed. Pearson Education Italia, Milano, 2004

[LZ 2004] A. Languasco, A. Zaccagnini - *Introduzione alla Crittografia* - Ulrico Hoepli Editore, Milano, 2004

[Morris 2004] S. Morris - *A Blade Server Primer* - Ed. Prentice Hall, 2004

[MSK 2005] S. McClure, J. Scambray e G. Kurtz - *Hacker! 5.0* - Ed. Apogeo, 2005

[NTFS-EFS 2000] La protezione di NTFS 5 con la crittografia: Encrypted File System, su Web <http://www.ntfs.com/ntfs-encrypted.htm>

[OHE 1999] R. Orfali, D. Harkey, J. Edwards - *Client/Server Survival Guide* - Wiley 3rd edition, 1999

[OpenPGP 2004] Gruppo standard per la versione 'IETF - RFC 2440' di PGP, sito Web <http://openpgp.org>

[OpenSSH 2006] implementazione open source e freeware di SSH, sito Web <http://www.openssh.org>

[OpenSSL 2006] implementazione open source e freeware di SSL, sito Web <http://www.openssl.com>

[OpenVPN 2006] sito del progetto OpenVPN, su Web <http://openvpn.net>

[OTP 2002] D. O'Mahony, H. Tewari e M. Peirce - Electronic Payment Systems for E-Commerce (2nd Edition) - Ed. Artech House, 2002

[PGP 2004] PGP Corporation, l'attuale custode, venditore e sostenitore della versione 'ufficiale' di PGP, sito Web <http://www.pgp.com>

[PGPI 2004] Informazioni sulle versioni open source attualmente disponibili di PGP, incluse le versioni 2.x, e informazioni generali su GPG e PGP, sito Web <http://www.pgpi.org>

[RBSWH 2001] R. Russell, T. Bidwell, O. Steudler, R. Walshaw e B. Huston - "Hack Proofing Your E-Commerce Site" - Ed. Syngress, 2001

[RFC 2246] Lo standard TLS come definito dall'IETF, sito Web <http://www.ietf.org/rfc/rfc2246.txt>

[Scheiner 1995] B. Schneier - "Applied Cryptography" - Ed. John Wiley and Sons, 1995

[SCMP 2000] proposta di specifica IETF su SCMP, sito Web <http://quimby.gnus.org/internet-drafts/draft-arnold-scmp-06.txt>

[SSH 2006] l'implementazione di SSH Corporation di SSH, sito Web <http://www.ssh.com/products/client-server/>

[SSL 1996] Lo standard Secure Socket Layer (SSL) 3.0 del 1996, sul Web <http://wp.netscape.com/eng/ssl3/>

[Stevens 1994] W. R. Stevens - "TCP/IP Illustrated – Vol. 1 & 2" - Ed. Addison-Wesley, 1994

[Tivoli 2006] pagina principale della suite Tivoli presso IBM, su Web <http://www-306.ibm.com/software/tivoli/>

[TNHD 2006], The New Hacker's Dictionary, sito Web http://www.outpost9.com/reference/jargon/jargon_toc.html

[Unicenter 2006] Pagina principale della suite Unicenter presso Computer Associates, su Web <http://www3.ca.com/it/Solutions/Solution.aspx?ID=315>

[Veridis 2005] Una versione PGP compatibile con OpenPGP, sito Web <http://www.veridis.com/openpgp/en/index.asp>

[Verisign 2006] Sito aziendale della Verisign Corporation, su Web <http://www.verisign.com>

[Windley 2005], P.J. Windley – *Digital Identity* – Ed. O'Reilly, 2005

[WRQ 2006] l'implementazione di F-Secure corporation e WRQ di SSH, sito Web <http://www.wrq.com/products/reflection/ssh/>

[Zimmermann 2004] Home Page del creatore di PGP, con numerose informazioni sul programma, sito Web <http://philzimmermann.com>

Altri siti Web importanti per la sicurezza:

Odysseus Project : <http://www.wastelands.gen.nz/odysseus/index.php>

Packet Storm: <http://packetstormsecurity.org/>

SecurTeam: <http://www.securiteam.com/>

ZoneH: <http://www.zone-h.com/en/defacements>

I problemi dei Cookie: http://wp.netscape.com/newsref/std/cookie_spec.html

New order: <http://neworder.box.sk/>

Vulnerability database: <http://nvd.nist.gov/>

Ass. Italiana per la Sicurezza informatica: <http://www.clusit.it/>

CNIPA sicurezza: http://www.cnipa.gov.it/site/it-it/Attivit%C3%A0/Sicurezza_informatica/

Siti di associazioni e raccolte documenti sulla sicurezza

- <http://www.defcon.org>
- <http://www.astalavista.box.sk>
- <http://www.antionline.com>
- <http://www.sans.org>
- <http://www.incidents.org>

- <http://www.mybookmarks.com/public/vanstrien>
- <http://www.freeonline.org/dir/c-346/Sicurezza%20informatica>
- <http://www.edipi.com/gsi/>
- http://www.governo.it/governoinforma/dossier/sicurezza_informatica/index.html
- <http://www.studiocataldi.it/questionario.asp>
- <http://www.sicurezzainformatica.it/>

Sicurezza di Windows

- <http://www.windowsitpro.com/WindowsSecurity/>
- <http://www.microsoft.com/security>
- <http://www.ntbugtraq.com>
- <http://www.ntobjectives.com>
- <http://www.ntshop.net>

Sicurezza rispetto ai virus informatici

<http://www.symantec.com/avcenter/reference/corpst.htm>
<http://www.f-secure.com>

Standard di Internet

- <http://www.ietf.org>
- <http://www.w3.org>
- <http://www.iana.org>

Firewall

- <http://www.free-firewall.org>
- <http://www.firewallguide.com>

La gestione dei sistemi informativi

Nei capitoli precedenti sono state esaminate le varie caratteristiche dei sistemi informativi, i processi business, la risorsa informazione, le risorse tecnologiche e le risorse umane. Nel presente capitolo verranno esaminati strumenti per la valutazione, pianificazione e gestione dei sistemi informativi e verranno introdotte le risorse organizzative.

Strumenti per la pianificazione delle attività

La pianificazione dei dettagli delle attività che compongono i processi, per essere efficace, richiede un elevato grado di precisione, e quindi la loro scomposizione in elementi sufficientemente piccoli da potere definire per ciascuno di essi le dipendenze da risorse umane e materiali della loro effettuazione. Molto spesso è necessario quindi arrivare alle attività scomposte nelle singole azioni od operazioni elementari che le formano. Si pensi, ad esempio, alla pianificazione delle ferie in un ufficio: per le attività, tranne che nel caso in cui l'intera azienda chiuda, deve essere sempre garantita la presenza di almeno una risorsa umana in grado di effettuarle e verso la quale quindi l'attività o la singola azione ha una relazione di dipendenza.

Questo processo di scomposizione, che può anche essere pensato come una forma più dettagliata di quanto visto nell'analisi di processi ed organizzazione, prende il nome di Work Breakdown Structure (WBS). In pratica quindi la WBS deve arrivare ad esprimere i dettagli minimi di quanto visto nella LRC nel capitolo 2.

I risultati della WBS possono essere espressi con documenti testuali o tabelle simili alla LRC, ma spesso vengono usati due strumenti grafici: il diagramma delle dipendenze e/o una sua evoluzione, il Grafo di Progetto o Program Evaluation and Review Technique, meglio noto con il suo acronimo PERT.

Il diagramma delle dipendenze, che si può pensare derivato dal class diagram, definisce con simboli simili a quelli usati per le classi le attività e le risorse impiegate in esse ed esprime le dipendenze fra le varie entità ed attività coinvolte nel processo. Talvolta vengono usati simboli grafici diversi (es. rettangoli arrotondati od ovali per le attività e rettangoli veri e propri per le classi, con la stessa simbologia usata in UML for business). Il diagramma può fotografare la situazione in un dato momento (es. iniziale e finale) e può esprimere lo stato effettivo (IS) e quello desiderato (SHOULD). Il PERT è molto più simile ad un diagramma di attività, in quanto oltre alle informazioni delle dipendenze fra attività ed azioni da un lato e risorse o regole dall'altro, conserva anche le informazioni di flusso logico presenti nel diagramma delle attività. In generale comunque possiamo dire che il PERT è la precisazione di un grafo i cui nodi rappresentano attività da svolgere ed i cui archi esprimono condizionamenti tra attività. Le tecniche basate sul PERT e il connesso metodo CPM (acronimo di Critical Path Method) sono tecniche ideate ed utilizzate per affrontare l'analisi di grandi progetti di ricerca, costruzione, programmazione ed organizzazione aziendale. Lo scopo principale è quello di pianificare e coordinare diverse attività che concorrono al raggiungimento dell'obiettivo in oggetto, assegnando nel tempo le risorse disponibili a ciascuna attività in modo "razionale".

Una volta definite tutte le relazioni di dipendenza e le durate delle singole attività, si può passare alla pianificazione temporale vera e propria, ossia alla mappatura dei tempi di esecuzione delle attività su un asse temporale corrispondente al calendario, e quindi comprensivo delle informazioni relative alle festività, orari di lavoro ecc...

In questa fase vengono usati i diagrammi di Gantt, detti anche diagrammi a barre. Due sono i tipi fondamentali di diagrammi Gantt:

1. diagramma di Gantt delle attività, in cui l'asse orizzontale rappresenta il tempo e le attività del progetto vengono indicate sull'asse verticale; le singole attività sono rappresentate come barre parallele all'asse temporale orizzontale; esso evidenzia tempi, dipendenze e criticità e permette di monitorare giorno per giorno l'andamento dei progetti, come mostrato in figura 9.2;
2. diagramma di allocazione delle risorse o diagramma degli incarichi in cui le attività sono già suddivise fra le varie risorse, ovvero sono state ad esse assegnate; le risorse sono rappresentate come punti sull'asse verticale, per cui la barra corrispondente rappresenta i periodi di occupazione delle singole risorse, come mostrato in figura 9.3.

Nelle figure seguenti è rappresentato un esempio generico di un semplice processo suddiviso in cinque attività, di cui le B e C sono alternative alla D. Viene presentato il PERT ed il conseguente Gantt.

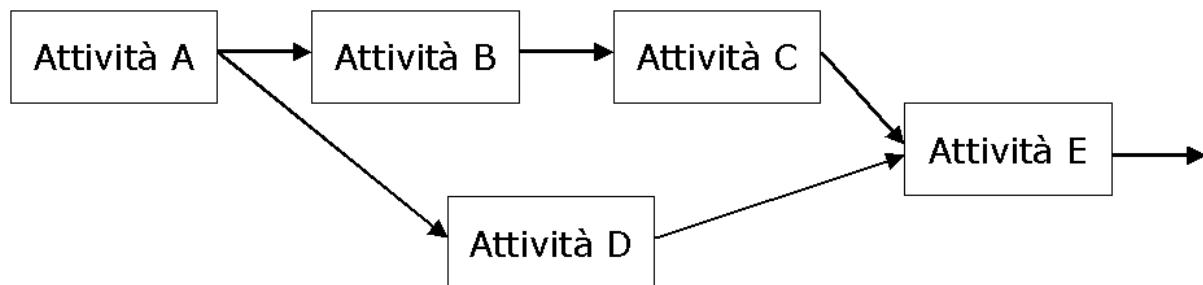


Figura 9.1: Il diagramma PERT di un processo suddiviso in 5 attività, di cui le B e C sono alternative alla D. Tutte le attività sono bloccanti, ossia non esistono parallelismi e un'attività conseguente può iniziare solo dopo che la precedente si è conclusa.

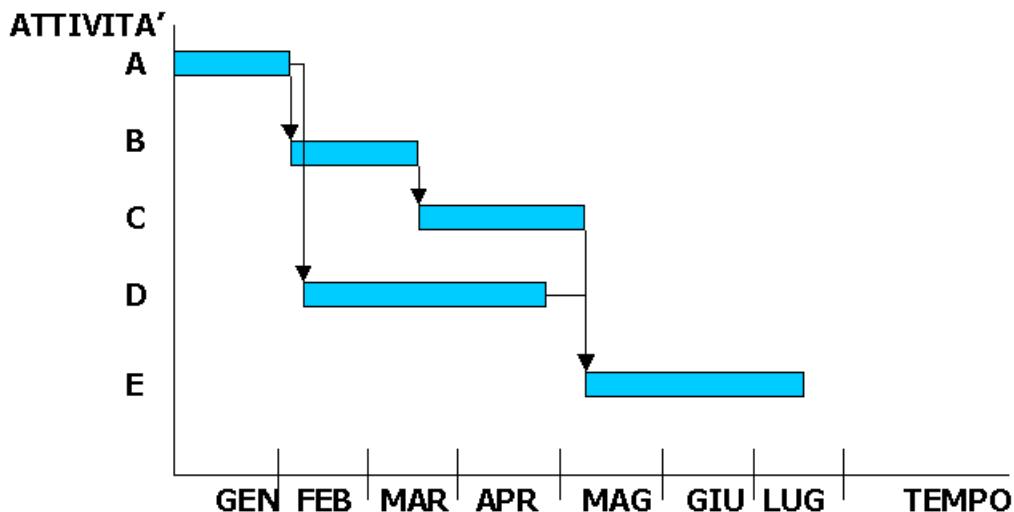


Figura 9.2: Il diagramma Gantt delle attività risultante dal PERT di figura 9.1; si ricordi che le attività B e C sono alternative alla D.

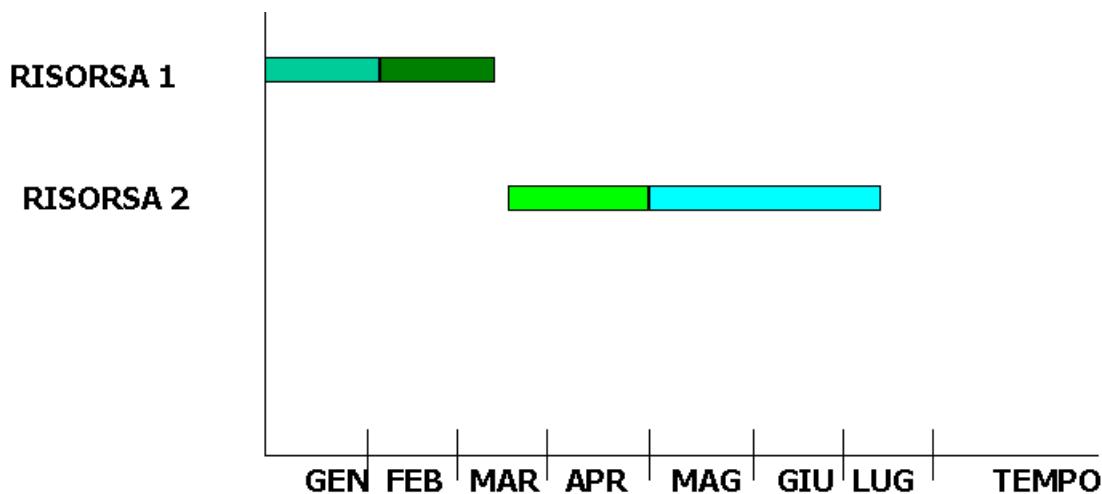


Figura 9.3: Supponendo di avere due risorse umane in grado di svolgere le attività del diagramma di figura 9.2, ecco una ipotesi di diagramma di allocazione di risorse risultante. Le quattro attività sono evidenziate dai diversi colori e si suppone che venga scelta la successione A, B, C, E, con la risorsa 1 allocata per A e B, mentre la risorsa 2 viene allocata per C ed E.

Uno strumento per valutare i ricavi: il Return Of Investment (ROI)

Nel comparto ICT il ritorno di investimento o ROI deve prevedere una valutazione quantitativa (ove possibile) e qualitativa (ove non sia possibile una quantificazione) dei benefici attesi da un certo investimento in risorse ICT.

Il calcolo del ROI comunque presenta dei limiti. Innanzitutto non sempre il costo del rischio è calcolabile, o si arriva alla esclusione di costi difficili da quantificare. Inoltre

gli investimenti a lungo termine possono essere penalizzati e in alcuni casi il calcolo è troppo semplificativo e non integrabile con la pianificazione.

Valutare i benefici dell'investimento significa sia compiere una valutazione qualitativa (azione talvolta indicata con il verbo qualificare) di tutti i potenziali benefici (tangibili e non), e quantificare i benefici tangibili e, per quanto possibile, anche gli intangibili. Il ROI deve essere definito sui valori monetizzati, ed integrato con valori quantificati ma non monetizzati, completando l'analisi con considerazioni strategiche e organizzative.

Per qualificare i benefici occorre tenere presente che il processo di lavoro è manuale (umano), comprende molti trasferimenti di informazione tra vari formati e anche attività parassite (es. correzioni orografiche) sono inserite nel processo. Vanno anche considerate le potenzialità, come l'automazione dei processi, l'eliminazione di alcuni trasferimenti di informazione e la riduzione delle attività parassite e, in generale, la conseguente velocizzazione del lavoro e le conseguenti economie di velocità.

I punti caratterizzanti la valutazione qualitativa possono essere in genere i seguenti:

- Riduzione dei costi
- Spostamento dei costi
- Costi evitati
- Miglioramento delle prestazioni
- Maggiori ricavi
- Riconfigurazioni delle relazioni
- Vantaggi competitivi
- Riduzione del rischio
- Sopravvivenza competitiva.

L'analisi quantitativa dei benefici presuppone una divisione fra benefici monetizzabili, quali:

- Riduzione del personale
- Riduzione del costo di struttura
- Riduzione di altri fattori produttivi
- Riduzioni del magazzino
- Eliminazione dei costi tecnologici di sistemi obsoleti

e benefici non monetizzabili, quali:

- Riduzione dei tempi di servizio
- Riduzione dei tempi di evasione ordini
- Maggiore rapidità di esecuzione di operazioni.

Uno strumento per valutare i costi: il Total Cost of Ownership (TCO)

Spesso, quando si procede all'acquisto di un sistema, ci si preoccupa solo del suo costo hardware e software iniziale, ma i costi di gestione ed aggiornamento sono elevati ed una politica di gestione oculata deve definire le procedure per ridurli. I componenti dei costi da valutare per il calcolo del TCO sono i costi di:

- Acquisizione hardware

- Acquisizione software (sistema operativo + applicativi)
- Installazione
- Addestramento
- Supporto
- Manutenzione (spesso uno dei maggiori)
- Infrastrutture
- Downtime (tempo di fermo macchina)
- Spazio, infrastrutture, energia necessari per il funzionamento continuativo dei sistemi.

Ad esempio, consideriamo il problema dell'accessibilità continua e della salvaguardia dei dati, ricordando che per dati informatici in senso ampio si intendono:

- Contenuto di DB relazionali
- Archivi documentali/multimediali
- Micro applicativi (es. generatori report)
- DB personali (es. elenco indirizzi)
- Archivi di Directory Service
- Configurazioni dei programmi e delle postazioni di lavoro.

In un sistema fortemente centralizzato tutti i dati risiedono o nel DB o, comunque, entro file sui dischi del server. In un sistema distribuito i dati sono ripartiti su più server e hanno una forma molto varia. Spesso poi ci sono dati importanti “sparsi in giro” per i client. Per la conservazione dei dati è necessario centralizzare la raccolta dei file almeno su server dipartimentali, in quanto un backup (ossia il salvataggio periodico del contenuto) automatico dei dischi dei client (es. sfruttando le condivisioni di dominio) diventa rapidamente ingestibile. Gli utenti devono procedere alla salvaguardia dei propri dati, copiandoli in cartelle condivise soggette a backup. In un sistema aziendale l'inaccessibilità di certi dati, dovuta a un fermo macchina o ad altre cause, provoca una perdita economica più o meno grave. Un certo tempo di fermo macchina può essere fisiologico (es. manutenzioni periodiche), ma in ogni caso si deve minimizzare tale valore o spostarlo su periodi non critici per il lavoro (es. notte, weekend).

Per conoscere il TCO si deve essere in grado di misurare anche il costo del fermo macchina, che può essere espresso dalla seguente formula

- O = Operatori (ossia persone coinvolte)
- T = Tempo di fermo macchina
- P = Percentuale di inattività
- C = Costo di una persona nell'unità di tempo
- F = Reddito prodotto da una persona nell'unità di tempo

Il costo diviene $\text{€} = \mathbf{O} * (\mathbf{T} * \mathbf{P} / 100) * (\mathbf{C} + \mathbf{F})$

Per esempio, dando i valori seguenti ai parametri, si ottiene

- $O = 5$ venditori
- $T = 4$ ore
- $P = 60\%$
- $C = 35$ (euro per ora)
- $F = 100$ (euro per ora)

$$5 * (4 * 60 / 100) * (35 + 100) = 1620 \text{ €}$$

Le cause di fermo macchina o malfunzionamenti sono molteplici: i sistemi operativi non sempre sono sufficientemente robusti rispetto a condizioni operative non infrequentate, le macchine hanno parti meccaniche soggette ad usura (ventole, dischi etc...) e la stessa componentistica elettronica può presentare dei problemi, così come gli utenti non esperti possono commettere errori nell'uso dei sistemi che portano anche al loro blocco. In generale, il problema della complessità dei sistemi è sicuramente elevato: vecchi e nuovi bug software, modularizzazione del software ed interconnessione dei componenti che formano gli applicativi, problemi di comprensione da parte dell'utente ed effetti dell'installazione di nuovi software/release possono anche essere notevoli.

Ritornando ad un discorso generale, il calcolo della spesa informatica può essere compiuto con gli strumenti abituali dell'analisi aziendale:

- Analisi aggregata (livello ed incidenza della spesa informatica)
 - Entità assoluta della spesa
 - Incrociata (dimensione spesa/dim. Impresa)
- Analisi disaggregata (struttura della spesa informatica)
 - Per tipo risorsa
 - Per funzione del reparto
 - Per prodotto (investimento)

Il valore assoluto della spesa si ottiene con dati puntuali e relative voci identificative, con le serie storiche (curve di spesa) e l'associazione con le acquisizioni di strumenti ICT, ma è spesso conveniente, ove possibile, il suo incrocio con volumi produttivi:

- A valore (es. fatturato)
 - Dati puntuali: incidenza percentuale
 - Serie storiche: curve di incidenza
- A quantità (es. unità prodotta)
 - Dati puntuali: costi per unità prodotta
 - Serie storiche: curve dei costi unitari

ed il suo incrocio con monte risorse:

- A valore (es. costo del personale)
 - Dati puntuali: incidenza percentuale
 - Serie storiche: curve di incidenza
- A quantità (es. numero dei dipendenti)
 - Dati puntuali: costi per addetto
 - Serie storiche: curve costi per addetto.

Altri strumenti importanti per la gestione e pianificazione

Livello di servizio o Service level

Definizione condivisa tra le parti relativa a una qualità (di solito misurabile quantitativamente) che un servizio erogato (anche non informatico) deve avere.

Esempi di livelli di servizio:

- Tempo massimo di risposta di un portale durante un collegamento Web
- Tempo massimo di risposta ad una chiamata di assistenza.

Garanzia o accordo di livello di servizio o Service Level Agreement (SLA)

Accordo tra le parti relativo ad un valore di un livello di servizio, per il quale il cliente paga

Esempi:

- Il sistema deve essere in esercizio per il 99% del tempo dell'anno
- La banda minima garantita deve essere di 10 Mbit/s.

Tolleranza ai guasti o fault tolerance

Capacità di un sistema informatico di continuare ad operare (magari non con lo stesso livello di servizio) anche in caso di alcuni tipi di guasto

Esempi:

- Doppio alimentatore in un server a ridurre il rischio di blocco in seguito a guasto in questo fondamentale componente
- Dischi in mirror.

Tempo di fermo macchina o Downtime

Tempo in cui un sistema informatico è fuori servizio, può anche essere relativo ad un singolo servizio (es. un processo). Può produrre un danno economico all'azienda cui il sistema appartiene, come visto nel paragrafo precedente.

Tempo di attività o Uptime

Tempo in cui un sistema informatico è attivo, può anche essere relativo ad un singolo servizio (es. un processo). Viene spesso usato nel computo degli SLA.

Bilanciamento di carico o load balancing

Capacità di un sistema informatico di reggere meglio il carico computazionale dei servizi erogati, suddividendolo tra le varie componenti che lo formano. Può anche essere relativo ad un singolo servizio.

Recupero dai guasti o disaster recovery

Capacità e/o insieme di regole relative al recupero di una condizione di funzionamento normale a partire da una condizione di guasto più o meno grave. Può essere relativo a singole componenti e/o servizi.

Continuità di servizio o Business continuity

Capacità e/o insieme di regole atte al garantire il funzionamento normale, o comunque l'erogazione di un insieme di servizi minimi, da parte di un sistema informatico in caso di malfunzionamenti. E' di solito supportata da un Business Continuity Plan, documento che stabilisce le procedure da seguire per garantire il funzionamento dei vari servizi.

Piano di sicurezza o Security Plan

Insieme completo delle procedure quotidiane volte a garantire sicurezza e continuità delle normali operazioni dei sistemi informativi.

Piano di recupero o Recovery Plan

Insieme delle procedure volte a ripristinare, correttamente e nel minor tempo possibile, l'operatività con l'erogazione di uno o tutti i servizi dopo il verificarsi di un danno accidentale e/o un attacco informatico. Il recovery plan deve indicare con precisione le azioni da compiere e le risorse umane cui esse sono assegnate, applicando i principi visti nella LRC e nella WBS.

Piano di recupero da disastri o Disaster Recovery Plan (DRP)

Insieme delle procedure relative al ripristino delle infrastrutture IT in seguito ad eventi catastrofici a bassa probabilità di accadimento, come, per esempio, incendi, inondazioni, attentati.

Analisi di impatto sul business o Business Impact Analysis (BIA)

Azione di analisi tesa a definire sia gli effetti dannosi che un fermo macchina o altro guasto può produrre sui processi business, sia il loro costo. Viene usata sia per le valutazioni del TCO, sia per costruire i Business Continuity Plan e/o i DRP.

L'obiettivo della BIA è correlare specifiche componenti del sistema ai processi critici che esse supportano ed il suo output servirà a definire le opportune strategie di recupero. L'insieme dei passi fondamentali per la BIA è schematizzabile come segue:

- Identificare i processi critici qualora questa conoscenza non sia ancora disponibile
- Identificare le risorse IT che supportano i processi critici
- Valutare gli impatti delle indisponibilità delle varie risorse
- Stabilire il tempo massimo di indisponibilità di una risorsa
- Definire le priorità di recupero.

Le politiche di gestione

La gestione dei sistemi informativi non è dissimile da quella di altri settori interni all'azienda. Possiamo suddividerla in diverse parti:

- Gestione Operativa
- Gestione delle Risorse (umane, tecnologiche, organizzative)
- Gestione della Configurazione dei sistemi
- Gestione dei Problemi (ove entrano anche le procedure di salvataggio dei dati e i piani di disaster recovery visti in precedenza).

Le regole organizzative attraverso cui la gestione viene compiuta sono anche chiamate politiche di gestione, ma occorre suddividerle nelle tre componenti fondamentali.

Politiche (general)

Una politica (in inglese policy) definisce una posizione di alto livello su un argomento, non definisce come fare qualcosa e nemmeno definisce i dettagli. Le politiche cambiano di rado e sono stabilite dalla direzione centrale di un'azienda o, quanto meno, con essa concordate.

Esempi di politiche sono:

- Information Security Policy
- Disaster Recovery Policy
- User Administration Policy
- Monitoring Policy.

Standard

Uno standard stabilisce come qualcosa dovrebbe essere o essere configurata. Gli standard non specificano come qualcosa viene svolto nei dettagli e dovrebbero cambiare seguendo processi e tecnologia.

Esempi di standard in ambito dei sistemi informativi sono:

- Standard di configurazione di macchine UNIX o Windows
- Standard di configurazione di un database e di un sito web
- Standard di classificazione dei dati
- Standard di scelta dei nomi degli identificatori dentro un programma.

Procedure

Le procedure provvedono istruzioni dettagliate su come implementare le politiche e definiscono anche chi è responsabile per ogni azione, passo dopo passo. Sono create anche attraverso l'uso di strumenti come LRC, WBS, PERT e Gantt.

Le procedure cambiano di frequente e dovrebbero essere aggiornate regolarmente attraverso un processo standardizzato, definito seguendo apposite politiche.

Un esempio di procedura, relativa alla gestione di sicurezza, è il seguente.

Definire esattamente cosa fare in caso di intrusione nel sistema, considerando

- Cosa si definisce come attacco, ovvero come attacco “sufficientemente grave”?
- Chi ha il potere decisionale?

- Chi deve essere contattato (ad esempio tecnici, amministratori di sistema, entità esterne all'azienda ecc...)?
- Come e quando avviene l'escalation, ossia il passaggio del comando operativo ad una persona più in alto nell'organigramma aziendale, o comunque il suo coinvolgimento nella risoluzione del problema?
- Che investigazioni devono essere svolte?
- Che aspetti legali devono essere curati per proteggere l'azienda (per esempio, per prevenire possibili azioni da parte di clienti i cui dati sono stati trafugati)?

Gestione corrente e gestione del cambiamento

Per gestire correttamente i sistemi ICT nell'azienda occorre conoscere le seguenti informazioni:

- I sistemi informativi e le risorse ICT in essi contenute sono rispondenti ai bisogni aziendali?
- L'uso che viene fatto delle risorse ICT è ottimale?
- Che livello di competenza per l'uso delle risorse ICT esiste entro l'azienda?
- I processi aziendali associati alle risorse ICT sono ben strutturati?

E, qualora la situazione corrente (AS-IS) non sia soddisfacente, occorre porsi obiettivi chiari a cui si vuole tendere (TO-BE), pianificare il cambiamento (eventualmente definendo anche un BPR se i processi stessi devono essere modificati) definire per esso un budget il più possibile preciso e valutare che impatto avrà il transitorio sul "normale" funzionamento aziendale.

Quando si valuta la necessità di una nuova soluzione ICT si deve risolvere anche il dilemma "make, buy o customize", tipico di tanti altri sistemi:

- **buy** (o acquisto): acquistare una soluzione IT presente sul mercato e provvedere eventualmente ad adattarla (con interventi poco rilevanti rispetto al costo della soluzione);
- **customize** (o adattamento): adattare, più o meno grandemente, una soluzione totale o parziale esistente; i costi dell'adattamento possono essere una frazione rilevante o addirittura maggioritaria del costo totale della soluzione;
- **make** (o realizzazione da zero): costruire da zero (o da "semilavorati", come, per esempio, librerie software) una soluzione ad hoc.

Purtroppo spesso vengono compiuti errori nelle valutazioni: in primo luogo c'è l'estremo di considerare l'informatica come un male necessario e non le sue potenzialità, così come esiste l'estremo opposto di considerare l'informatica come una panacea, che da sola risolve tutti i problemi. Le risorse umane sono sempre un componente fondamentale: scarso coinvolgimento e scarsa motivazione degli operatori sono sempre un punto critico in ogni fase di cambiamento, così come una pianificazione imperfetta della fase di transitorio e dell'addestramento del personale alle nuove strutture ICT.

Spesso infine non è noto come vengono usati effettivamente, in relazione alle loro potenzialità, strumenti ICT entro l'azienda, situazione che si verifica spesso per gli strumenti di produttività individuale come Office, del quale, in molti casi, gli operatori non conoscono sufficientemente tutte le potenzialità in relazione alle proprie mansioni.

Alla fine delle valutazioni il management deve giungere coscientemente ad una decisione, che normalmente è una delle seguenti:

- Conservare inalterato l'esistente
- Manutenzione evolutiva dell'esistente
- Sostituzione totale o parziale dell'esistente
 - Cambio sistema (trasporto applicazione)
 - Cambio applicazione
 - Cambio sistema e applicazione
- Outsourcing totale o parziale
- Riorganizzazione comparto ICT
 - Concentrazione dei CED
 - Delocalizzazione dei CED
 - Outsourcing dell'informatica periferica
 - Downsizing, ossia riduzione delle risorse impiegate
- Integrazione di sistemi e applicazioni
 - di dipartimenti diversi (quindi interni all'azienda e già esistenti)
 - In seguito ad acquisizioni di altre aziende.

Per approfondimenti si consigliano [BFM 2001] e [LL 2004].

La gestione della sicurezza

La sicurezza entro un'azienda non è data solo dall'installazione di un firewall o dall'adozione di un anti-virus su tutti i computer, ma deve essere "distribuita" in tutta l'organizzazione. "Il computer più sicuro è quello spento e chiuso in una cassaforte", frase attribuita ad un esperto di sicurezza del ministero della difesa statunitense, esprime chiaramente il concetto che la sicurezza assoluta non esiste. Per questo l'approccio migliore alla gestione della sicurezza è quello basato sull'analisi del rischio definito nel capitolo 8. Lo scopo primario dei sistemi informatici è essere di ausilio al business, più o meno direttamente. Si deve ricordare che non sempre i produttori di software tengono presenti le necessità di sicurezza ed integrazione e che una conoscenza d'insieme del sistema e dei processi business e dei flussi informativi ad esso legati è indispensabile per pianificare qualsiasi politica di sicurezza.

Le misure di sicurezza non devono mai essere di ostacolo reale al funzionamento dei programmi ma, allo stesso tempo, le richieste degli utenti devono avere un limite nelle esigenze di sicurezza.

Il punto debole della sicurezza sono molto spesso gli utenti. Per esempio, connessioni "non ufficiali" ad Internet, fatte via modem da PC connessi alla rete interna, consentono di bypassare qualsiasi firewall. E' poi vero che qualsiasi operazione di sicurezza che richieda un intervento esplicito dell'utente o che richieda uno sforzo di attenzione è statisticamente destinata prima o poi a fallire, come per esempio avverrebbe se l'utente, non disponendo di un antivirus che svolge tale compito in automatico, dovesse sottoporre manualmente al controllo antivirus tutti i dischetti. Un altro problema molto grave riguarda le password di accesso, che troppo spesso vengono scelte facilmente indovinabili. La gestione delle password dei sistemi è una delle attività più complesse e allo stesso tempo critiche. In una rete

Windows2000/XP/2003 è molto spesso necessario che l'utente di un PC sia in possesso della sua password di amministratore locale, il che però, come visto nel capitolo 8, espone la rete al rischio che la stazione di lavoro venga usata come cavallo di troia per attaccare l'intera rete. Inoltre, applicando le direttive del Decreto Privacy, occorre forzare gli utenti all'aggiornamento periodico delle password e sarebbe buona cosa anche costringerli ad una scelta minimamente sicura, applicando le regole basilari di alternanza maiuscola-minuscola nelle lettere, dell'inserimento di numeri e altri caratteri speciali e di scegliere una lunghezza minima per la password.

Ma per applicare efficacemente tutto questo, è necessario che gli utenti siano responsabilizzati rispetto ai rischi di sicurezza: infatti se un utente sente regolamenti/procedure unicamente come un peso, tenderà a non rispettarli. Per questo è necessario il coinvolgimento del management centrale dell'azienda: gli addetti ai sistemi informatici devono rendere il management consapevole dei rischi, e fornire ad esso l'elenco delle possibili soluzioni, con pro e contro ovvero rapporto costi-benefici. Pertanto le linee guida per stabilire una adeguata politica di gestione della sicurezza sono formate dai passi seguenti:

1. **Analisi del rischio**, in base alla quale si deve conoscere cosa deve essere protetto e quanto è importante, applicando le metodologie viste in precedenza ed evitando di cadere nella “trappola” del determinare tutto egualmente importante e soggetto a rischi gravi;
2. **Definire e scrivere delle politiche** che guidino correttamente nell'approccio alle varie problematiche di sicurezza;
3. Sulla base delle politiche, **stabilire degli standard** per la scelta dei sistemi, degli applicativi e delle loro configurazioni;
4. Seguendo politiche e standard, **definire le procedure di attuazione operativa delle protezioni e di controllo** e documentarle chiaramente, stabilendo quali risorse tecniche sono coinvolte e attribuendo gli assegnamenti dei compiti alle opportune risorse umane;
5. **Provare, verificare ed eventualmente rivedere**: i controlli devono essere provati e migliorati quando necessario, anche servendosi di test operativi.

Un metodo di analisi del rischio non deve essere necessariamente complesso, per esempio, il computo può essere fatto sulla base dei seguenti approcci:

- Valore di quanto protetto + indice del pericolo = rischio
- Valore + impatto = rischio
- Probabilità + impatto = rischio

Occorre ricordare che nessuna analisi è perfetta e spesso è necessario solo identificare i rischi maggiori e definire dove sono necessari i maggiori controlli. Una volta ottenute le liste di rischio, si procede ad identificare le aree critiche da coprire con politiche, standard e procedure, iniziando dalle più importanti, per poi tornare alle liste per verificare se le protezioni applicate sono adeguate al livello di rischio. Il processo deve essere iterativo e viene ripetuto al variare delle situazioni e con l'evoluzione dell'azienda e le trasformazioni che in essa devono essere applicate.

Uno dei componenti più importanti, troppo spesso sottovalutato, è il Disaster Recover Plan (DRP), definito nei paragrafi precedenti. Un buon DRP deve avere le seguenti caratteristiche:

- Deve godere dell'appoggio della direzione aziendale;
- Deve avere uno scopo ben definito;
- Deve introdurre una chiara catena di comando e delega dell'autorità;
- Deve essere privo di singoli punti di fallimento (single point of failure);
- Deve essere abbastanza flessibile da prevedere cambiamenti nelle condizioni operative in cui sarà applicato.

E, nei dettagli, deve contenere:

- la struttura organizzativa, con ruoli e responsabilità (fra cui i Disaster Planning Coordinator, cui sono delegati i poteri decisionali)
- l'ambito di applicazione, in particolare l'elenco dei processi critici per l'azienda (mission critical)
- le risorse coinvolte (umane e non)
- le strategie e le procedure operative per il recupero della funzionalità delle risorse IT malfunzionanti
- i controlli preventivi da applicare
- i requisiti di formazione del personale
- i criteri di testing e manutenzione del piano, che consentano la sua verifica e l'adattamento sempre migliore alle condizioni operative.

Il rapporto costi-benefici tra il costo della definizione ed attuazione del DRP e il costo degli impatti dei guasti deve essere valutato in funzione del tempo desiderato per il recovery, come schematizzato in figura 9.4, dove si evince che il costo del recupero è inversamente proporzionale al tempo richiesto per il recupero stesso, mentre il costo dell'impatto, come già visto in precedenza, è direttamente proporzionale al tempo.

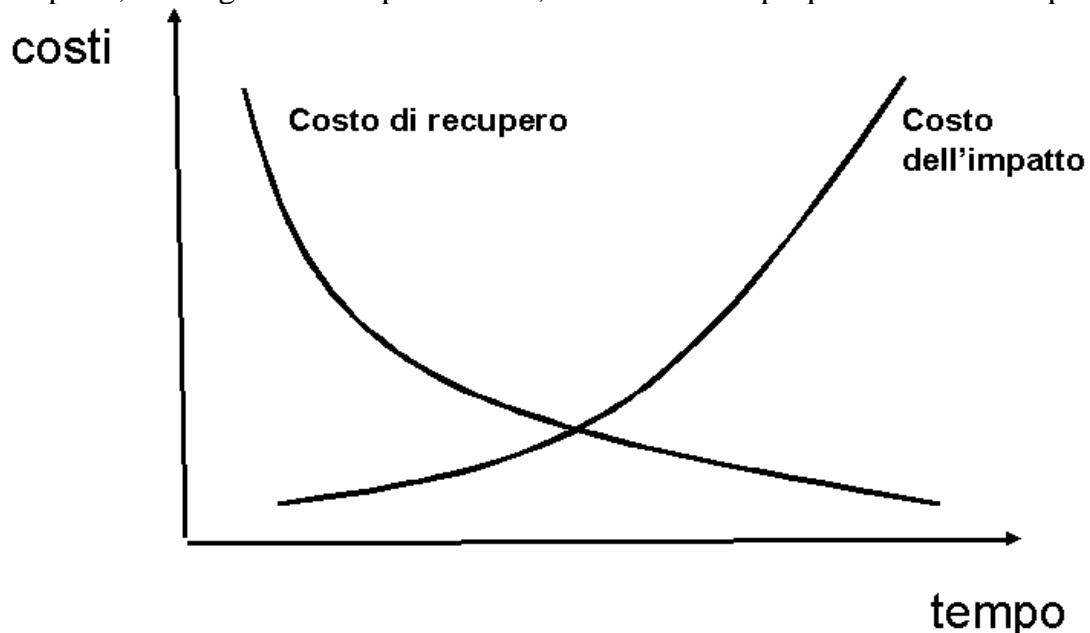


Figura 9.4: Rapporto costi-benefici tra costo del fermo macchina e costo delle procedure di riattivazione, in base al tempo.

La gestione del progetto informatico

Quando l’analisi dell’esistente evidenzia troppe distanze da quanto desiderato occorre mettere in opera un progetto per colmare il gap. Supponendo di avere risolto il dilemma “make, buy or customize” con la decisione di costruire o adattare, il progetto deve essere gestito, applicando i vari strumenti di pianificazione e gestione visti in precedenza. Negli anni’90 l’applicazione dei principi di ingegneria del software, associata all’avvento dei linguaggi ad oggetti, ha condotto alla stesura di modelli di progettazione e delle relative procedure codificate. In particolare uno dei processi di sviluppo più diffusi prende il nome di Unified Process (si vedano [ZBGM 2004] e [Maciaszek 2002]). Non sempre l’unified process o i suoi derivati sono applicabili in toto, ma in molti casi essi possono definire un insieme di linee guida per realizzare le varie fasi del progetto. Pertanto una versione semplificata dell’unified process, applicabile sempre, può seguire le fasi riportate nel seguito. Occorre ricordare sempre che i passi della analisi e della progettazione non devono essere vissuti come imposizioni, bensì come un ausilio al mantenimento della precisione e, conseguentemente, di un migliore controllo sull’andamento dei lavori e del progetto in toto. Nella conduzione del progetto vengono applicati come strumenti i diagrammi UML definiti nel capitolo 4.

1. Capire “cosa” si vuole ottenere

E’ la prima fase del lavoro congiunto con il cliente o l’utente finale e, a partire dalle sue intenzioni iniziali e dai suoi desiderata, ha lo scopo di produrre un documento informale, scritto in linguaggio naturale, che spieghi brevemente le intenzioni del cliente. In pratica questo primo documento serve a circoscrivere i limiti del lavoro successivo. Deve essere breve e il più possibile preciso e indicare con precisione l’argomento del progetto successivo.

Questo documento dovrebbe avere le seguenti caratteristiche:

1. Indicare chi è il cliente finale;
2. Indicare i requisiti che il cliente richiede;
3. Definire di che tipo di lavoro si tratta;
4. Indicare vincoli imposti da altri software o sistemi o ambienti esistenti con cui si debba interoperare o entro cui il risultato del progetto debba operare;
5. Descrivere “informalmente” e a grandi linee il lavoro da svolgere.

2. Definire i concetti e le entità del progetto

In questa fase si deve definire la terminologia del progetto, identificando con precisione le entità (persone, ruoli, luoghi, oggetti materiali, eventi, strutturazioni ecc...) coinvolte nel sistema del mondo reale (ovvero del dominio di business) che hanno importanza per il sistema informatico obiettivo del progetto. E’ importante identificare con precisione le entità, allo scopo sia di definire meglio i loro scenari d’uso (Passo 3, gli Use Case), sia di individuare le Classi Entità (Passo 4, il Class Diagram d’analisi).

Il risultato di questa fase è il documento Glossario, che definisce con precisione tutti i termini corrispondenti alle entità coinvolte, evitando ambiguità.

3. Definire esattamente le funzioni di quanto si vuole ottenere

In questa fase devono essere individuati con precisione gli scenari d'uso del sistema, ovvero, in modo più generale, gli scenari di interazione fra il sistema e gli attori, ovvero le entità esterne al sistema con cui esso interagisce e comunica. I passi necessari in questa fase possono essere così suddivisi:

1. Definizione esatta del *boundary* o confine del sistema (entro sistemi particolarmente complessi questa fase può anche essere applicata a sotto-sistemi);
2. Identificazione e definizione degli *attori*, ossia delle entità esterne con cui il sistema (o i sotto-sistemi) oggetto dell'analisi interagiscono e comunicano;
3. Individuazione dei vari scenari di uso/interazione fra sistema ed attori, che corrisponderanno ai singoli casi d'uso, identificati dalle singole ellissi nel diagramma; si ricordi che i casi d'uso descrivono cosa si vuole che il sistema faccia, non come questo comportamento deve essere implementato (modello della *scatola nera* o *black box*, definito nel capitolo 1);
4. Definizione delle interazioni entro i singoli casi d'uso; tali interazioni, strutturate nella forma della richiesta dell'attore cui corrisponde una risposta del sistema (tenendo conto anche di eventuali comunicazioni asincrone o autonome del sistema quali ad esempio allarmi), andranno a costituire le *descrizioni* dei singoli casi d'uso;
5. Esame dei diagrammi così ottenuti e delle loro descrizioni per potere procedere alla raccolta a fattore comune di parti fra i singoli use case entro diagrammi, facendo uso delle relazioni *extends* ed *include* definibili tra i vari casi d'uso;
6. Il passo 5 può essere iterato più volte; occorre tenere conto della granularità del problema e del grado di definizione e precisione che si vuole raggiungere; inoltre occorre tenere presente che un singolo caso d'uso spesso dà origine ad una singola maschera (sia essa maschera testuale, singola window in ambiente Windows Form o pagina Web); infine si tenga presente che spesso da un caso d'uso deriverà anche un caso di test durante la fase di test del sistema informatico realizzato.

Il prodotto di questo passo è l'insieme completo degli use case inseriti entro uno o più use case diagram, ognuno corredata di adeguata descrizione, strutturata chiaramente in forma di request-response e considerando sia il percorso principale di interazione (*basic course*) sia gli eventuali percorsi alternativi (*alternative courses*), quali quelli che si verificano in presenza di errori nei dati introdotti ecc... Il diagramma e le descrizioni devono essere ben strutturati, chiari ed esaurienti, in quanto tutti i passi successivi si baseranno su di essi.

Siccome gli use case diagram non esprimono direttamente relazioni di flusso logico/temporale fra i loro componenti, può essere utile esplicitare tali relazioni attraverso un activity diagram derivato, che definisce le attività associate ai singoli use case (potrebbero, in tal caso, essere necessarie ulteriori scomposizioni od aggregazioni) e le relazioni logiche e temporali che tra esse intercorrono. Da questo diagramma deriva, più o meno direttamente, anche il diagramma di navigazione fra le finestre o maschere che costituiscono l'interfaccia esterna utente dell'applicazione in progetto.

I diagrammi non sono normalmente sufficienti e si possono aggiungere ulteriori descrizioni, che rendano più preciso il tutto.

4. Defnire con precisione le entità e le relazioni che le legano

In questa fase, che parte dal Glossario realizzato in fase 2 e dallo/dagli Use Case Diagram (corredati anche degli Activity Diagram) realizzati in fase 3, deve essere realizzato il diagramma delle classi di analisi. Tale diagramma deve indicare chiaramente tutte le classi entità, ossia le classi definibili come “proiezioni”, nel dominio della applicazione software, delle entità del dominio del problema dove la applicazione software andrà ad operare, più eventuali altre classi individuate nel corso dell’analisi che siano di importanza per i concetti funzionali che definiscono i requisiti del progetto. In pratica nel diagramma, che è l’equivalente da un punto di vista del ruolo (e l’evoluzione da un punto di vista storico e metodologico) del diagramma Entità-Relazioni (ER) usato nelle metodologie di sviluppo più tradizionali, devono essere chiaramente indicate:

1. Tutte le classi entità che fanno parte del dominio del problema;
2. Gli attribuiti caratteristici di tali classi, eventualmente procedendo alla individuazione dei singoli attributi o dei gruppi che consentano una identificazione univoca delle istanze delle classi, ovvero dei singoli oggetti; tali attributi costituiscono le *chiavi*;
3. Le *associazioni* che tra tali classi intercorrono, ossia tutti i legami logici che tra esse intercorrono; queste associazioni (che corrispondono alle *relazioni* dei diagrammi ER) sono importanti perché in sede implementativa di codice indicheranno anche la visibilità necessaria tra le classi, cioè quali altre classi (eventualmente appartenenti ad altri package o namespace) una certa classe dovrà vedere, definendo quindi la loro *interdipendenza*;
4. I versi di tali associazioni (ad esempio, se la classe magazzino deve conoscere la classe prodotto, non è sempre vero il viceversa);
5. Le molteplicità di tali associazioni (es. uno-a-molti, molti-a-molti), l’eventuale necessità di definire *classi di associazione* (si ricordi, ad esempio, la proprietà dell’auto che svolge il ruolo di classe di associazione fra proprietario ed auto);
6. Eventuali *rapporti di inclusione* legati a tali associazioni, suddivisi fra aggregazione e composizione; si ricordi che l’eliminazione di una composizione, indicata con il diamante nero, elimina anche tutti i suoi elementi componenti, mentre l’eliminazione di una aggregazione, indicata con il diamante bianco, non elimina anche i componenti, che hanno anche una natura indipendente;
7. Eventuali rapporti di ereditarietà fra le classi, ottenuti applicando i principi di generalizzazione e specializzazione, ovvero “raccogliendo a fattor comune” attributi e metodi o aggiungendone di nuovi;
8. Si ricordi che da questo diagramma, eventualmente passando attraverso un diagramma EER, deriverà anche la base dati relazionale dell’applicazione: i rapporti di molteplicità devono essere chiari perché dalle associazioni derivano le relazioni tra le chiavi che collegano le tabelle entro la base dati;

9. I metodi delle classi possono ancora non essere completamente definiti in questa fase.

Il processo che conduce al diagramma finale è ovviamente iterativo e può dirsi stabilizzato quando tutte le relazioni (in senso ampio) fra le classi sono chiaramente individuate. Il Class Diagram di analisi è fondamentale per tutti i passi di progetto che seguono. Normalmente vanno incluse anche descrizioni, più o meno ampie, delle caratteristiche delle classi e delle relazioni che tra loro intercorrono.

5. Scegliere l'architettura del sistema che si vuole realizzare

La scelta architettonica è un passo fondamentale, in quanto i passi successivi sono da essa condizionati, come si era visto nel capitolo 5 (si riveda la figura 5.8). Esistono comunque regole generali importanti che aiutano nello svolgimento, quali il pattern Model-View-Controller (MVC) ed il conseguente approccio multicanale alla realizzazione delle interfacce utenti (si vedano [Desio 2006], [BMRSS 1996] e [TOGAF 2006] per approfondimenti). Seguendo tale metodo, si separa nettamente la interfaccia utente vera e propria (View), che ha lo scopo di presentare semplicemente dati all'utente ed è ovviamente soggetta ai vincoli dal tipo di mezzo o canale utilizzato (interfaccia a finestre grafiche, Web, PDA, cellulare, Set-Top Box TV...), dal reattore agli eventi trasmessi dall'utente (Controller), che usa i metodi forniti dagli strati interni dell'applicazione (Model e relativi Adapter) per garantire all'utente i servizi associati agli eventi inviati dall'utente stesso. Grazie all'approccio multicanale, eventualmente corredata dall'uso di altri strati di Adapter, diviene possibile riutilizzare (almeno in buona parte) il controller (ed ovviamente gli strati sottostanti) cambiando solo la view quando si cambia canale, passando, ad esempio, da una applicazione Window ad una Web sostituendo alla finestra il modulo web corrispondente (ad esempio, se si usa il linguaggio Java, il servlet).

La scelta dell'architettura deve anche segnalare limiti e criticità nel sistema che sarà realizzato.

L'output di questa fase sono documenti tecnici architettonici, che saranno poi corredati da eventuali Component Diagram e Deployment Diagram solo al termine della fase di progetto vera e propria.

6. Progettare nei dettagli il sistema

In questa fase occorre definire chiaramente tutte le classi che fanno parte dell'applicazione software da implementare. Il Class Diagram di Progetto è l'elenco completo delle classi, con tutte le loro relazioni e su di esso si basa anche il dimensionamento della fase di sviluppo (ovvero scrittura vera e propria del software).

Il processo che permette di giungere al diagramma delle classi di progetto è necessariamente iterativo. Si parte dal diagramma delle classi di analisi e devono essere inserite tutte le classi di servizio, ossia le classi infrastrutturali, non necessariamente derivate dalla fase di analisi, che permettono al programma nel suo insieme di operare correttamente ed in modo efficiente. Le classi di servizio sono ovviamente fortemente dipendenti nella loro struttura dall'architettura scelta e da eventuali framework utilizzati nel progetto. Se un diagramma di analisi ben fatto può

essere spesso utilizzato con diverse tecnologie ad oggetti, ovvero essere punto di partenza per progetti analoghi realizzati su piattaforme diverse, un diagramma di progetto è chiaramente molto più influenzato dalla tecnologia usata. Il processo usa anche altri diagrammi UML.

1. I diagrammi di interazione (sequence, che pone enfasi sulla sequenza temporale delle interazioni, e collaboration, che pone enfasi sulla dipendenza fra le classi), sono di importanza fondamentale sia per la definizione dei metodi che le classi offrono le une alle altre (e dei loro argomenti e valori di ritorno), sia per l'individuazione di eventuali “colli di bottiglia” che vengono risolti con l'inserimento di nuove classi. In teoria ad ogni use case corrisponde almeno un sequence o collaboration diagram: infatti ogni corso di eventi individuato nell'analisi con gli use case dovrebbe produrre una precisa sequenza temporale di invocazione di metodi all'interno dell'insieme delle classi constituenti il sistema software. Non sempre è però indispensabile realizzarli tutti, specie nei casi di corsi di eventi molto simili, nel qual caso bastano le opportune descrizioni di accompagnamento.
2. I diagrammi di attività, che derivano anch'essi dagli Use Case, dando ad essi una sequenza temporale e logica, possono aiutare molto nella definizione della Mappa di Navigazione fra le finestre, consentendo di definire completamente l'interfaccia utente di un applicativo ed eventualmente di realizzare i prototipi d'analisi (finestre vuote vere e proprie o gli schematics).
3. I diagrammi di stato sono anch'essi molto importanti per valutare l'evoluzione temporale delle singole classi (o meglio degli oggetti da esse istanziati) o di sotto-sistemi che esse vanno a costituire, aiutando ad individuare eventuali condizioni critiche o colli di bottiglia.

L'obiettivo finale è comunque la realizzazione del Class Diagram di Progetto, completo di tutte le classi. Spesso per motivi di chiarezza (specialmente in progetti grandi dove le classi sono molto numerose) il diagramma viene diviso in package, associazioni di classi corrispondenti ad unità funzionali, indicando esternamente ad essi solo i legami che fra i singoli package intercorrono. Ciascun package viene poi rappresentato completamente entro un diagramma di secondo livello. Quasi sempre questa suddivisione funzionale viene anche portata a livello implementativo servendosi delle aggregazioni tipiche dei linguaggi (es. package del Java, namespace di C#). L'obiettivo deve essere sempre quello di avere un diagramma leggibile, che serve come mappa per lo sviluppo. Da questo diagramma possono anche essere generati gli scheletri delle classi attraverso opportuni strumenti CASE, oppure essere ottenuti i *fogli di specifica*, ossia i documenti che descrivono ciascuna classe con attributi, metodi, vincoli e controlli da implementare.

7. Definire le strutture di contorno

Usando i diagrammi realizzati in precedenza, si arriva a definire le parti implementative di contorno del progetto, che devono essere opportunamente documentate come segue.

1. Definizione della base di dati, attraverso un Extended Entity-Relationship (EER), eventualmente corredata dagli script di creazione delle tabelle e vincoli che genera la base dati nello specifico DBMS scelto.
2. Definizione dell'insieme dei singoli componenti software (package, DLL, Jar ecc...) che devono essere prodotti, con l'indicazione delle loro interdipendenze, attraverso un opportuno Component Diagram o più di uno.
3. Definizione della distribuzione dei componenti sulla o sulle piattaforme di produzione prescelte, attraverso uno o più opportuni Deployment Diagram.
4. Stesura di opportuni documenti che corredano il progetto e la installazione; in particolare devono essere chiaramente indicati eventuali limiti della/e piattaforma software ed hardware utilizzata.
5. Stesura dell'opportuno manuale utente dell'applicazione.
6. Definizione delle scadenze e pianificazione dell'esecuzione temporale del progetto, in base ai dimensionamenti svolti e alle risorse a disposizione, sfruttando le metodologie viste all'inizio del capitolo.
7. Definizione dei test e dei singoli casi di test.
8. Pianificazione del collaudo e dell'entrata in produzione.
9. Definizione della successiva fase di manutenzione.

ICT e business: situazione corrente e possibili evoluzioni future

E' importante ricordare che lo scopo primario dei sistemi informatici è fare business, più o meno direttamente, o producendo direttamente reddito, o svolgendo nel modo più efficiente possibile compiti entro i sistemi informativi.

Dal 1995 nell'ICT si è assistito alla nascita di quattro nuovi comparti, avvenuti in rapida successione:

- **E-commerce**, il commercio elettronico basato su Internet
- **E-Business**, la facilitazione di funzioni, processi e specifiche strategie aziendali utilizzando tecnologie web/internet per la condivisione e l'integrazione di flussi informativi ed applicazioni
- **M-commerce**: commercio elettronico basato su tecnologie di telefonia mobile
- **M-business**: la facilitazione di funzioni, processi e specifiche strategie aziendali utilizzando tecnologie di telefonia mobile.

Ad essi è seguito anche un nuovo modo di definire le interazioni fra gli attori coinvolti nei processi business. In particolare il commercio elettronico ha condotto alle interazioni business-to-consumer (B2C) fra azienda ed utente o cliente finale e business-to-business (B2B) fra aziende clienti e fornitori. Tali interazioni si sono estese anche oltre la vendita di beni e/o servizi, si pensi, ad esempio, ai servizi di home banking e phone banking. Ma presto sono nate nuove interazioni, come la enterprise-to-enterprise (E2E) fra aziende che comprende l'E-business, la enterprise-to-administration (E2A), fra imprese e pubblica amministrazione e la citizen-to-administration (C2A) fra cittadino e pubblica amministrazione. Anche in Italia questi ultimi due tipi di interazione, costituenti parte di un contesto più ampio, chiamato E-

government, stanno prendendo piede (si pensi, ad esempio, alla dichiarazione delle tasse per via telematica o alla possibilità di presentare denunce alla polizia via Internet).

È interessante osservare che i cambiamenti strutturali indotti da questi fenomeni non sono confinati all'interno delle singole imprese, ma in qualche modo sono andati anche oltre i confini. L'E-commerce, soprattutto nel mercato nordamericano, ha profondamente modificato le modalità di interazione fra impresa e clienti, mentre l'E-business ha avuto un impatto simile su fornitori e dipendenti e, se anche è forse troppo presto per affermarlo con certezza, è possibile prevedere che l'M-business (mobile business) indurrà cambiamenti ancora più profondi, perché ancora più "onnipresente". L'integrazione in atto fra le tecnologie di Internet e la telefonia mobile condurranno nei prossimi anni alla integrazione totale. Gli effetti dell'M-Business, e questo lo rende un fenomeno del tutto nuovo, si avverteranno su tre piani diversi:

- infrastrutture e dispositivi di interazione
- applicazioni ed esperienze
- relazioni e supply chain.

In generale possiamo riassumere questa evoluzione con alcuni punti base.

Da un modello "tradizionale" precedentemente adottato, in cui vi era distanza fra ICT e il business e l'interazione tra il cliente e l'organizzazione avveniva quasi esclusivamente attraverso risorse umane (per esempio, sportelli, filiali, agenti di commercio, venditori, rappresentanti...), oggi si sta progressivamente passando ad un nuovo tipo di modello, caratterizzato dai seguenti punti:

- Convergenza fra IT e il business
- Aumento della globalizzazione e business fra imprese (B2B/B2C)
- Modelli di e-business sempre più frequenti
- il cliente/utente/cittadino interagisce con l'organizzazione attraverso l'ICT

Come esempi dei processi in atto si possono citare la ricerca dei luoghi delle vacanze, ormai compiuta in massima parte su Internet, la scelta del modello di un prodotto elettronico come uno stereo od un cellulare, anch'essa compiuta molto spesso su Internet o l'E-Government, ossia l'interazione cittadino-istituzione attraverso le tecnologie di Internet (e un domani gli SMS).

Inoltre stanno apparendo nuovi tipi di vendita di beni immateriali, come risultato diretto della ICT, come per esempio:

- Vendita di musica on-line
- Vendita di suonerie e servizi per cellulari
- Vendita di film on-line
- Vendita di software scaricato direttamente da Internet.

Nel contempo stanno avvenendo cambi drastici anche nell'industria e servizi ICT, quali, ad esempio:

- Delocalizzazione della produzione software
- Delocalizzazione dei centri di calcolo
- Delocalizzazione del call center.

In questo scenario, con una rapidità sempre crescente delle variazioni nei mercati, per l'azienda diviene una necessità vitale l'essere rapida nella reazione ai mutamenti e nuove esigenze imposte dal “mercato globale”, per le quali sono necessarie:

- Uso ottimale dell'ICT
- Organizzazione per processi, adattandoli alle dimensioni dell'azienda stessa
- Possibilità di riprogrammare rapidamente i flussi di informazioni associati al lavoro (workflow) presenti entro i sistemi informativi
- Politiche di gestione di qualità.

A livello globale, le tendenze relative alle tecnologie ICT in atto possono essere riassunte come segue:

- Spostamento ulteriore verso server con piattaforma x86 (Intel o AMD)
- Adozione di Linux come server in crescita
- Uso massiccio di accessi larga banda e VPN/intranet/extranet per collegare sedi geograficamente distanti fra loro
- Adozione del desktop remoto, o comunque di modelli client-server multitier in cui l'elaborazione viene concentrata sui server
- Adozione di metodologie di costruzione di interfacce utente basate su descrittori XML e facilmente adattabili a tipi diversi di piattaforma client (Windows, Web, palmare, telefono cellulare...)
- Collegamento fra sistemi basato su SOA, con progressiva adozione di ESB come strumenti di controlli del workflow
- Uso di macchine virtuali per separare le applicazioni fra loro
- Concentrazione dei data center (proprietari e in outsourcing)

Domande

1. Cos'è la WBS e come può essere usata per scoprire i legami fra gli elementi di un sistema informativo?
2. Cos'è il TCO? Come interviene nelle scelte relative agli acquisti?
3. Cos'è il ROI? Come deve essere usato per la pianificazione del sistema ICT?
4. Cosa si intende per Service Level Agreement o SLA?
5. Cosa si intende per Business Continuity?
6. Cos'è il Disaster Recovery Plan?
7. Cosa sono le politiche di gestione?
8. Cosa è uno standard?
9. Cos'è una procedura?
10. Come si può analizzare la necessità di cambiamento?
11. Quali sono i passi per gestire al meglio un progetto informatico?
12. Quali sono le tendenze in atto nel mercato odierno dei sistemi informativi?

Bibliografia

[ACPT 2002] P. Atzeni, S. Ceri, S. Paraboschi, R. Torlone - Basi di Dati: Modelli e Linguaggi di Interrogazione - McGraw-Hill Italia, Milano, 2002

[ACFPT 2003] P. Atzeni, S. Ceri, P. Fraternali, S. Paraboschi, R. Torlone - Basi di Dati: Architetture e Linee di Evoluzione - McGraw-Hill Italia, Milano, 2003

[BFM 2001] G. Bracchi, C. Francalanci, G. Motta - *Sistemi Informativi e aziende in rete* - McGraw-Hill Italia, Milano, 2001

[BMRSS 1996] F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal - *Pattern-Oriented Software Architecture, Volume 1, A System of Patterns* - Ed. Wiley 1996

[De Marco 2000] M. De Marco - *Sistemi Informativi Aziendali* - Franco Angeli Edizioni, Milano 2000

[Desio 2006] C. Desio - *Il pattern Model View Controller* – su Web <http://www.claudiodesio.com/ooa&cd/mvc.htm>

[LL 2004] K Laudon, J Laudon – *Management dei sistemi informativi* – Ed. Pearson Education Italia, Milano, 2004

[Maciaszek 2002] L. A. Maciaszek – *Sviluppo di sistemi informativi con UML* – Ed Addison-Wesley, 2002

[TOGAF 2006] Capitolo sui pattern, su Web <http://www.opengroup.org/architecture/togaf8-doc/arch/chap28.html>

[ZBGM 2004] W. Zuser, S. Biffl, T Grechenig, M. Kohle - *Ingegneria del Software con UML e Unified Process* - Ed McGraw-Hill, 2004

Case study

Uno schema di analisi parziale e completa

In quanto segue vengono trattati alcuni esempi reali ed analizzate tutte le loro caratteristiche. Lo schema è sempre il seguente:

- Identificare il “sistema impresa” ed i suoi confini
- Definire le caratteristiche da evidenziare
- Definire il business e la generazione del valore, identificando il cosiddetto core business, cioè l’insieme di prodotti o servizi che maggiormente contribuiscono ai ricavi dell’azienda ed intorno alla fornitura dei quali l’attività dell’azienda ruota
- Definire i processi business associati a tale core business e le loro caratteristiche
- Definire i flussi informativi richiesti da tali processi per il loro funzionamento
- Definire i requisiti informativi ed informatici che essi comportano e producono
- Stabilire quali requisiti funzionali sono traducibili in soluzioni informatiche, ossia programmi applicativi
- Identificare quali infrastrutture informatiche tali programmi richiedono per il loro funzionamento, rispettando i vincoli tecnici imposti
- Scegliere se la soluzione da adottare deve essere implementata attraverso
 - make (costruzione interna),
 - buy (acquisizione sul mercato)
 - o customize (adattamento, più o meno spinto, di una soluzione presente sul mercato),

tenendo presente

- il ritorno di investimento (ROI),
- l’insieme di competenze, interne all’azienda o reperibili come servizio sul mercato, necessarie per l’uso e la manutenzione di tale soluzione,
- i conseguenti costi di gestione ed i costi di acquisto e della soluzione (TCO).

L’analisi completa può essere applicata in questa sede solo a contesti non troppo complessi. Per sistemi ad elevata complessità (es. grandi aziende, banche, enti governativi...) è conveniente dividere da subito il modello in sotto-sistemi, per esempio servendosi della scomposizione basata sulla catena del valore di Porter, e poi applicare i punti dell’analisi al sotto-sistema, previa identificazione dei flussi di informazione e valore che lo legano agli altri macrocomponenti.

Studio associato di professionisti

Si consideri uno studio associato di 10 avvocati professionisti, specializzati in vari campi giuridici, che a volte lavorano insieme collaborando sul medesimo cliente, ma che possono anche condurre un'intera causa in modo individuale. Accanto ai professionisti sono anche due impiegati, aventi il ruolo di gestire amministrativamente e contabilmente lo studio, gestire le anagrafiche dei clienti e verificare l'andamento del business, per potere dare ogni volta che si rende necessario, un rendiconto ai consigli di amministrazione dei soci.

Il business primario in questo caso è la conduzione della causa e (potenzialmente) la sua vittoria, per ottenere il ricavo corrisposto come parcella, o, in alternativa, la consulenza legale, anch'essa con parcella associata.

E' interessante notare che, analizzando il processo relativo alla conduzione di una causa, si vede che esso si compone di attività di ricerca informazioni, verifica di fatti, stesura di documenti, partecipazione ai dibattiti nei tribunali. Il tutto però è quasi sempre condotto da una sola persona o, al più, da due o tre dei soci dello studio, che si coordinano fra loro, qualora sia necessaria l'esperienza di specializzazioni diverse. Simile, con meno tipologie di attività componenti, è il processo per la costruzione della consulenza. Si può affermare quindi che i processi primari (ossia quelli che generano il valore) di questo tipo di impresa sono, per così dire, "collassati" entro un singolo operatore (o una piccola associazione di singoli operatori), in questo caso il professionista che li svolge in prima persona, pur essendo possibile definire la conduzione di una causa come successione di attività ben precise, svolte tutte dalla stessa persona (o da un numero molto limitato di persone). Inoltre queste attività hanno un elevato contenuto informativo e intellettuale e sono scarsamente automatizzabili. La gestione della conoscenza intrinseca alle attività può però beneficiare in modo fondamentale dell'uso degli strumenti informatici applicati al knowledge management, per quanto riguarda il reperimento di informazioni, la generazione di nuova informazione ed il suo efficace immagazzinamento.

Accanto ai suddetti processi primari, vi sono processi secondari (e quindi che non producono direttamente valore, ma che sono indispensabili per vincoli strutturali) che coinvolgono i professionisti, come la partecipazione alle attività dell'Ordine degli Avvocati, ai corsi di aggiornamento o l'autoaggiornamento attraverso consultazione di archivi di informazione presenti su supporto informatico (es. CD-ROM) o accessibili in rete.

Ovviamente per il sistema "Studio Professionale" sono anche indispensabili i processi secondari, gestionali ed amministrativi, svolti dai due impiegati, che possono essere raggruppati come:

- Gestione degli incassi e delle spese, della redistribuzione degli utili ed, in generale dei flussi di cassa (spesso chiamato processo Finance);
- Verifica dei pagamenti ed invio dei solleciti ai clienti morosi (potrebbe anche essere considerato come una parte del processo precedente);
- Gestione delle anagrafiche dei clienti;

- Gestione delle comunicazioni verso i clienti (es. posta, documenti ufficiali dello studio...);
- Altri processi simili...

Per le varie attività è possibile misurare parametri qualitativi e quantitativi dell’efficienza, tra cui spiccano:

1. Tempo di esecuzione complessivo di un’attività;
2. Tasso di errori associato all’attività (e quindi misurazione della qualità, ma anche connesso con il punto precedente, in quanto la correzione dell’errore e le conseguenti attività parassite sprecano tempo);
3. Efficienza dell’output ottenuto, che per la conduzione di una causa può essere calcolato valutando l’effetto positivo per la vittoria della causa dell’arringa o delle informazioni raccolte ed aggregate ecc...;
4. Rapidità nel reperire le informazioni necessarie al processo di consulenza;
5. Capacità di riutilizzare le informazioni raccolte o la nuova informazione generata (si pensi al riuso di un documento di perizia legale, che è costato molto tempo per la sua creazione, ma diviene adattabile con poche modifiche a vari casi, rivendendolo allo stesso costo iniziale, permettendo di ammortizzare efficacemente il tempo inizialmente speso).

I passi precedenti definiscono abbastanza chiaramente quelli che sono i flussi informativi associati ai processi:

1. Ricerca di informazioni in rete;
2. Scambio di dati tra i clienti e lo studio, dati che possono essere trasmessi in formato elettronico o attraverso telefono, fax, posta cartacea;
3. Scambio di dati tra i professionisti e tra i professionisti e le due impieghi;
4. Scambio di dati tra i professionisti ed enti esterni come l’Ordine degli Avvocati, i tribunali, altri organismi statali ecc....

L’informatica può intervenire efficacemente in tutto questo: i flussi informativi possono essere implementati, anche se magari in modo non esclusivo, come flussi informatici, ossia le informazioni possono comunque continuare a fluire anche attraverso canali più tradizionali, come la posta cartacea ed il telefono. Il sistema informatico risultante può avere la seguente struttura, visualizzata in figura 10.1:

1. Computer server con disco molto capiente ospitante i servizi di
 - a. file server, con le cartelle personali contenenti le copie dei documenti personali di ciascun professionista, nonché cartelle ad accesso comune per la condivisione di documenti ed informazioni in genere; tali cartelle sono soggette a backup, ovvero al salvataggio periodico, per esempio, ogni notte;
 - b. print server, con le funzionalità di stampa (pilotaggio di stampanti laser di qualità) accessibili da tutte le postazioni;
 - c. domain server, con le funzionalità di gestione degli accessi alla rete ed alle singole cartelle od altre risorse presenti, associate ai profili dei vari professionisti ed impiegati;

- d. database server, che ospita la base di dati del programma di contabilità usato dalle impieghi, e anche dati strutturati, come l'archivio completo dei clienti;
- e. antivirus;
2. Computer server con disco molto capiente ospitante i servizi di posta elettronica, sia interna, che garantisce efficacemente comunicazioni asincrone tra i professionisti e tra questi e le impieghi, sia da e verso la rete esterna; eventualmente anche il fax può essere ospitato entro tale server, consentendo di spedire fax direttamente dalla propria postazione o di consultare l'archivio elettronico dei fax ricevuti, senza alcuno spreco di carta; viene spesso usato un server a parte in quanto il carico indotto dal servizio di posta elettronica può essere ampio;
3. Computer dedicato al compito di firewall, ossia di controllo del traffico da e verso la rete pubblica esterna;
4. Accesso alla rete con modem ADSL ad alta velocità, che consente sia la ricerca di informazioni, sia lo scambio di posta elettronica tra il server del provider del servizio internet e il server di posta interno, consentendo comunicazioni e-mail efficaci con il mondo;
5. PC dei professionisti, che ospitano una dotazione di suite Office, con programmi di videoscrittura, foglio elettronico, creazione presentazioni ecc..., il client di posta elettronica, il browser per la navigazione in Internet; da queste postazioni è possibile realizzare documenti e salvarli sul server o consultare gli archivi propri o comuni presenti sul server stesso;
6. PC degli impieghi, che, oltre alla dotazione software dei professionisti, ospitano anche il client del programma di contabilità.

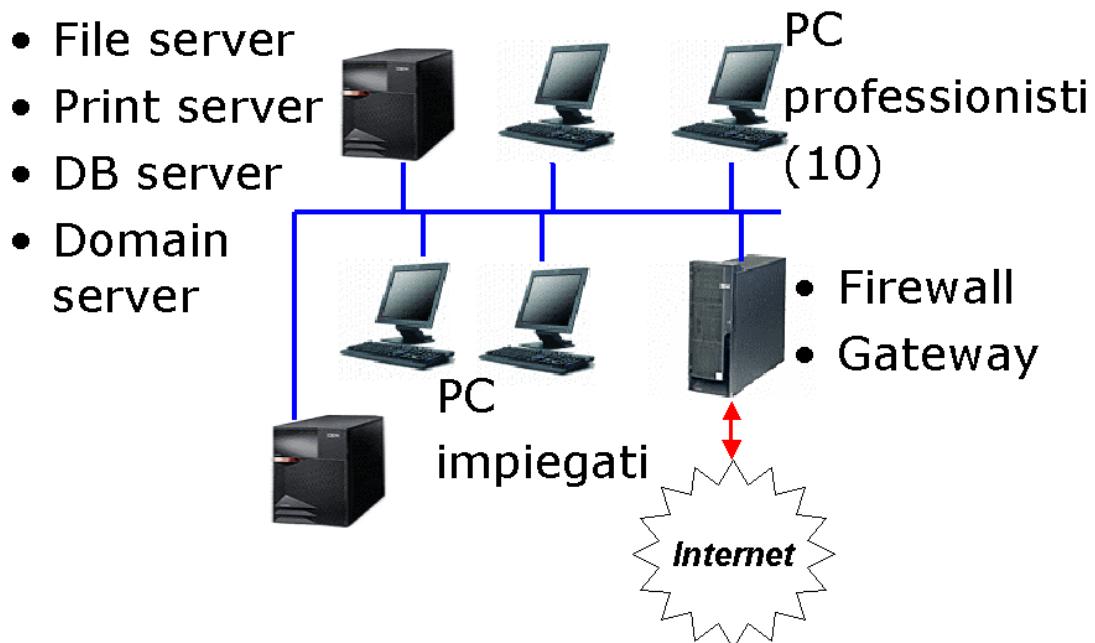


Figura 10.1: Schema della possibile rete informatica di uno Studio Associato di Avvocati.

In apparenza tutto quanto si è detto in precedenza per lo Studio Associato di Avvocati dovrebbe essere vero per qualsiasi altra categoria professionale organizzata in modo simile. In realtà in alcuni casi esistono delle differenze. Per comprendere la natura di queste differenze occorre esaminare in dettaglio il processo di esecuzione del lavoro che genera il ricavo per l'impresa.

Quasi tutti i professionisti possono compiere delle consulenze, per le quali il processo di reperimento di informazioni e di stesura di documenti (perizie, studi di fattibilità, pareri legali ecc...), entro cui sta il contenuto informativo oggetto della prestazione per il quale il cliente è disposto a pagare, è quasi identico a quello visto. Ma per l'erogazione di altri servizi possono esistere delle differenze, per capire le quali serve suddividere in diverse categorie i servizi professionali:

1. Professionisti in cui il lavoro è puramente intellettuale, ossia l'informatica aiuta per il trattamento testi, stesura documenti, ossia nella rappresentazione e memorizzazione dell'informazione, ma non interviene ulteriormente sul processo; in pratica possiamo considerare che il fatto di scrivere un testo con il computer consente di ridurre tempo, eliminare errori, riusare più volte uno stesso testo, ma, a parte questo, non ha nulla di diverso rispetto a scriverlo a mano o con la macchina da scrivere, e allo stesso modo la consultazione di informazione in rete è molto più rapida, ma concettualmente non diversa dal consultare testi cartacei; i suddetti avvocati rientrano in questa categoria;
2. Professionisti per i quali l'informatica interviene in parte ad automatizzare il lavoro, ossia non soltanto rende più rapida la creazione del documento con l'informazione associata, ma ne automatizza alcune parti; esempi di questa categoria possono essere Ingegneri, Architetti e Geometri, per i quali i sistemi CAD (Disegno assistito dal Calcolatore) forniscono uno strumento utile per realizzare i progetti in tempi molto più brevi, e inoltre i sistemi di calcolo strutturale consentono di validare tali progetti, evidenziando eventuali imperfezioni presenti, i sistemi di rendering consentono di visualizzare in forma realistica tridimensionale tali progetti, per poterli mostrare ai clienti prima della loro realizzazione, ecc...; altri appartenenti a questa categoria possono essere i grafici;
3. Professionisti per i quali l'informatica può intervenire anche in supporto alle decisioni; si pensi ad esempio ai medici che consultano un sistema esperto per avere conferme alla propria diagnosi;
4. Professionisti per i quali l'informatica interviene massicciamente ad automatizzare alcuni dei processi, rendendoli molto più rapidi e meno soggetti ad errori; un esempio sono i commercialisti, per i quali i programmi appositi automatizzano tutti i processi di stesura dei documenti di dichiarazione dei redditi, consentendo al professionista di dedicare maggiore tempo a processi consulenziali, potenzialmente anche più redditizi.

In pratica quindi questo significa che ci potranno essere più componenti hardware e software nel sistema informatico, per esempio, in uno studio di architettura troveremo un plotter, strumento per stampare i disegni attraverso una penna computerizzata, nello

studio del commercialista troveremo il client del programma di contabilità sulle postazioni di tutti i soci ecc...

A questo punto siamo in grado di definire la scelta di architettura informatica.

Per la parte “generale”, che comprende il file server, le postazioni di lavoro con la dotazione di suite Office, sono possibili diverse soluzioni seguendo gli standard di mercato. In particolare è possibile implementare tutto, a parità di hardware, attraverso le soluzioni Microsoft (sistema operativo Windows 2000 o XP e Windows2003 Server, MS-Office, Exchange Server per la posta elettronica, SQL Server come DB Server) oppure attraverso soluzioni completamente basate su Linux (sistema operativo Linux, OpenOffice, qmail server per la posta elettronica, MySQL o PostgresSQL o Oracle come DB Server) o anche soluzioni ibride (es. Linux solo sul server).

La parte specifica (es. il programma del commercialista) sarà quella determinante. In base alla disponibilità delle componenti client e server degli applicativi specifici sulle varie piattaforme software infrastrutturali (Linux, Windows o altro) presenti sul mercato, sarà possibile una scelta anche di tali componenti del sistema informatico. Ovviamente la disponibilità di un servizio di manutenzione a costi ragionevoli, o le competenze interne per provvedere in proprio alla manutenzione, saranno un fattore determinante per la scelta finale.

Molte delle caratteristiche esaminate in questo esempio sono comuni anche ad altri sistemi, come, per esempio, gli uffici legali, gli uffici marketing o gli uffici tecnici interni alle aziende medio-grandi.

Agenzia di lavoro interinale

Un'agenzia di lavoro interinale ha lo scopo di procurarsi un insieme di risorse umane con varie specializzazioni e livelli di professionalità e di collocare tali risorse sul mercato, allocando loro giornate lavorative, per periodi più o meno lunghi, presso i propri clienti.

Il core business è quindi la vendita di questa attività, cercando di alzare il più possibile il margine di guadagno, ossia la differenza fra il costo per l'agenzia di una giornata lavorativa di una determinata risorsa umana e il ricavo che il cliente paga per tale giornata. Accanto a questo possono essere presenti anche servizi più di "consulenza", quali, ad esempio, una selezione di risorse fatta per un cliente che poi assumerà direttamente tali risorse, che potrebbe essere pagata all'agenzia in base al tempo speso per l'opera di selezione stessa, o la collaborazione alla stesura di questionari di selezione, o l'elaborazione dei risultati di tali questionari.

Il ciclo passivo in questo caso è praticamente coincidente con un processo di reclutamento e gestione delle risorse umane (per essere precisi, la gestione delle risorse umane qui si limiterebbe ai soli dipendenti propri dell'agenzia di lavoro interinale), su larga scala. Deve in particolare avvenire anche una appropriata classificazione ed archiviazione delle competenze delle risorse.

Il marketing viene compiuto pubblicizzando le professionalità delle proprie risorse e quindi ha bisogno di una appropriata gestione della conoscenza associata all'informazione sulla professionalità delle risorse umane disponibili.

Il ciclo attivo è rappresentato dalla vendita delle giornate di prestazione d'opera delle risorse umane. Il margine, fra quanto il cliente esterno paga per una risorsa (di solito l'unità è la giornata-uomo) ed il costo giornaliero della risorsa per l'agenzia, genera il ricavo primario dell'azienda di lavoro interinale, da cui poi devono essere detratti tutti i costi dei processi ausiliari necessari al funzionamento dell'azienda stessa.

Per i processi ausiliari valgono considerazioni molto simili allo Studio Professionale esaminato nell'esempio precedente, solo qui la scala sarà tipicamente molto più grande e il numero di impiegati addetti a tali processi anche. Se poi l'agenzia fa parte di un circuito occorre valutare anche l'organizzazione interna di quest'ultimo. Per esempio, ogni filiale opera in un territorio, ma può rendere disponibili anche alle altre sedi le proprie risorse umane disposte a trasferirsi o a viaggiare. Le altre sedi potranno quindi collocarle presso i propri clienti. L'attività di selezione ed inserimento delle risorse dovrà essere in qualche modo riconosciuta economicamente alla filiale da cui esse provengono.

Ma ora focalizziamo l'attenzione sui processi primari di acquisizione e rivendita delle risorse, nonchè sulle relazioni con i clienti finali del servizio di lavoro interinale. Il rapporto con i propri clienti e l'andamento del mercato devono guidare l'agenzia nella ricerca e selezione di nuovi collaboratori, nonchè eventualmente anche nella conversione verso nuove professionalità di propri collaboratori esistenti che abbiano dimostrato capacità intrinseche (es. grinta, intelligenza, laboriosità, onestà...) tali da giustificare un investimento in una loro riqualificazione. In pratica quindi l'agenzia, in funzione del bacino territoriale in cui si trova, seleziona in preferenza risorse con certe

professionalità, ove esiste una maggiore probabilità di rivendita di tali professionalità. Ovviamente nel processo di rivendita il marketing e la cura del cliente rivestono un ruolo fondamentale.

Il processo di reclutamento si può dividere nelle seguenti fasi:

- Definizione dell'oggetto della ricerca e pubblicizzazione della ricerca, attraverso canali di vario tipo (es. giornali, riviste, radio, TV, internet);
- Definizione delle prove di selezione, per esempio questionari di gruppo con relative matrici di punteggio, colloqui individuali e di gruppo;
- Fase di ricezione dei curriculum vitae e della loro valutazione e classificazione entro un archivio delle risorse umane;
- Convocazione dei candidati per le prove, con contemporanea pianificazione e coordinamento degli addetti alle selezioni (es. operatori, intervistatori, psicologi, esperti funzionali del settore per il quale la professionalità viene selezionata), che può coinvolgere anche l'intervento di esperti esterni all'agenzia, spesso appartenenti al/ai clienti finali;
- Esecuzione delle varie prove, con registrazione di tutti i risultati e loro associazione al profilo dei vari candidati, in modo tale da definire delle graduatorie di interesse per l'agenzia, in relazione alla professionalità cercata, o eventualmente ad altre;
- Selezione dei candidati finali ed avvio delle pratiche amministrative per inquadrarli ed inviarli presso il cliente finale, rendendoli operativi e quindi produttivi per l'agenzia.

Appare evidente da quest'analisi che il processo di selezione è lungo, complesso e dispendioso e va quindi il più possibile ottimizzato in tutte le sue fasi. In particolare devono essere definite delle procedure operative che ottimizzino i tempi delle prove, consentendo di raccogliere informazioni sufficienti relativamente alla valutazione del singolo candidato nel minor tempo possibile. Più candidati un operatore di selezione riesce a valutare nell'unità di tempo (tipicamente la giornata lavorativa) e più il processo è efficiente. Nello stesso tempo però occorre cercare anche di rendere il più possibile fruttuose e redditizie tutte le informazioni sui candidati raccolte nell'ambito delle prove di selezione. Infatti, in uno scenario caratterizzato dalla presenza di moltissime agenzie concorrenti, diviene fondamentale non solo la possibilità di soddisfare il cliente da un punto di vista della qualità della professionalità, ma anche del minor tempo impiegato a reperire e fornire una risorsa al cliente che la richiede.

Questa capacità di essere tempestivi nella soddisfazione della domanda non si può limitare al lato delle vendite, ma deve influenzare tutto il processo di gestione delle risorse. In particolare la profilazione dei candidati deve fornire tutte le informazioni relative alle loro possibilità operative, ovvero a tutte le loro potenzialità lavorative, in modo tale che, se anche una persona è stata selezionata per una determinata posizione e poi non l'ha occupata, qualora emergano altre professionalità in possesso di tale persona, esse possano essere usate qualora il mercato lo richieda. In pratica quindi, per ogni risorsa che viene registrata nell'archivio, devono essere note tutte le mansioni che tale risorsa è in grado di espletare e deve essere tracciato al meglio anche il piano di impiego presso i clienti, in modo tale da pianificare e preparare anche il trasferimento

da un cliente ad un altro nel minor tempo possibile di una risorsa umana, magari con una mansione diversa da quella espletata presso il primo cliente.

I processi di vendita esaminati sono simili a quelli che sono presenti entro le varie agenzie di servizi, come, per esempio, le aziende di pulizia industriale, di disinfezione, di manutenzione giardini ecc..., anche se di solito tali agenzie hanno già un proprio insieme di collaboratori, dipendenti e non, e tendono a reclutarli sul mercato con minore frequenza.

Pertanto i processi richiedono risorse informatiche in grado di archiviare e reperire facilmente tutte le professionalità e capacità relative a tutti i candidati passati per le varie selezioni tenute dall'agenzia, e anche di tracciarne i piani di attività presso i vari clienti, in modo tale da potere trovare nel minor tempo possibile la risorsa più adatta per la richiesta di un cliente, meglio ancora se fra risorse le cui informazioni sono già in possesso dell'agenzia o che già stanno lavorando per l'agenzia.

Si impone la creazione di un archivio informatico con la sua base di dati, consultabile efficacemente per le ricerche di personale dagli addetti alla vendita e, nello stesso tempo, aggiornabile efficacemente da parte dei selezionatori. Anche le schede di raccolta dati dovranno essere studiate in modo tale da rendere l'operazione di registrazione su supporto informatico delle informazioni raccolte su una determinata risorsa umana il più veloci ed efficienti possibile.

Il contatto con i canali sopra citati per le pubblicazioni di inserzioni e richieste dovrà anche esso essere il più efficiente possibile. Ci saranno quindi procedure codificate per giungere alla costruzione rapida di inserzioni standardizzate.

Dal punto di vista informatico si richiede quindi una base di dati potenzialmente molto grande, ossia contenente i dati relativi a molte persone, cui si possa accedere simultaneamente da un alto numero di clienti per la consultazione e la ricerca e da un numero più limitato per gli inserimenti delle nuove risorse umane o per l'aggiornamento dei dati esistenti. Se l'agenzia fa parte di una rete e l'archivio è comune, i dati saranno centralizzati e l'accesso avverrà anche tramite rete geografica oltre che locale.

In base a quanto finora esaminato siamo in grado di definire le necessità del sistema informatico che l'agenzia deve implementare. Occorre un programma ad accesso multiutente per la gestione sia dei profili delle risorse umane sia dei loro periodi di occupazione. La base di dati sottostante a tale programma dovrà essere centralizzata e ad essa dovranno avere accesso i vari operatori (nel caso di una rete di agenzie sarà nel database server della sede centrale e dovrà essere organizzata una rete informatica sicura, ossia al riparo da accessi fraudolenti ed intercettazioni, che consenta l'accesso da tutte le agenzie periferiche a tale database server). La soluzione è quindi un programma client-server che copra queste necessità funzionali, mantenendo nel contempo la coerenza dei dati, evitando quindi, per esempio, che la stessa risorsa umana possa essere allocata su due clienti diversi da due operatori commerciali diversi. Il sistema dovrà consentire le ricerche di professionalità con parole chiave e essere facilmente aggiornabile, in modo tale da registrare nuove capacità professionali acquisite dalle risorse umane.

Accanto a tale sistema dovrà funzionare anche un programma gestionale per le attività contabili ed amministrative.

Se pensiamo ad una rete di agenzie, abbiamo quindi a che fare con un sistema informatico distribuito, in cui il server centrale si trova presso la sede centrale dell'agenzia, in una stanza sicura e chiusa, e i dati in esso contenuti sono accessibili solo attraverso i programmi applicativi. La rete che connette le varie filiali deve essere sicura, quindi implementata come una rete privata (che ha lo svantaggio degli alti costi) o attraverso una rete privata virtuale o VPN, che, usando canali protetti da crittografia dei dati, consente gli stessi livelli di sicurezza pur transitando attraverso la rete pubblica.

Un sistema di posta elettronica interno alla rete di agenzie completerà le necessità di scambio dati ed informazioni fra i vari operatori.

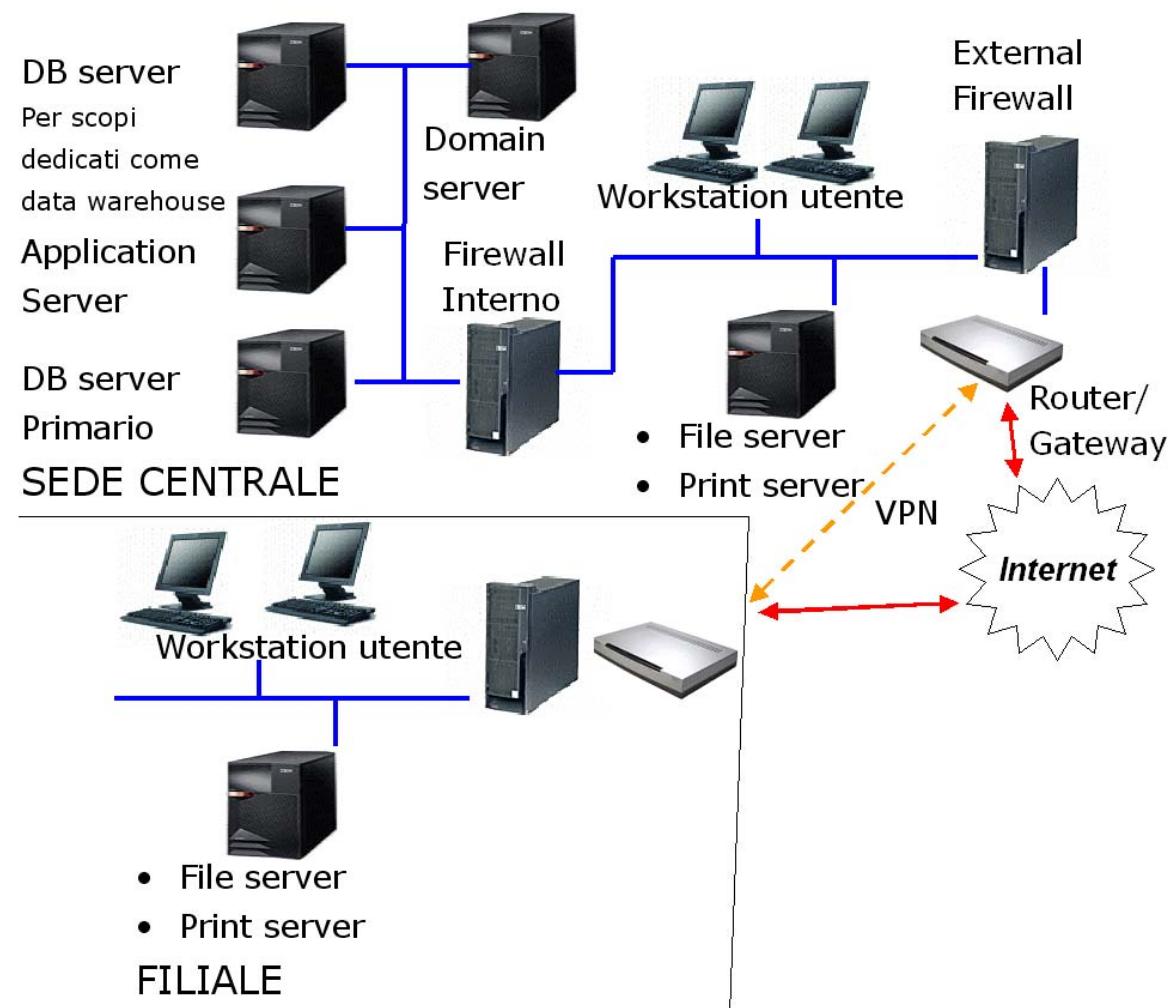


Figura 10.2: Un possibile sistema informatico di un'agenzia di lavoro interinale distribuita in varie sedi sul territorio. Viene riportata in basso a destra una filiale, ove troviamo un file server e un print server. Per evitare interruzioni del lavoro, in caso di caduta del collegamento in rete, potrebbe essere anche presente una copia locale di alcuni dei dati del database centrale, anche se questo pone ovviamente problemi di sincronizzazione fra i due archivi di dati.

Azienda vinicola

Questo esempio è più complesso dei precedenti ed è estremamente importante, in quanto presenta molte caratteristiche comuni a una piccola o media impresa di tipo produttivo di vari settori.

Il “sistema azienda” in esame è costituito da una impresa vinicola con la sua cantina, che lavora sia le uve provenienti dalla propria vigna che quelle di alcuni produttori, per generare un vino a marchio doc. Dovendo sottostare ai regolamenti di qualità definiti dal consorzio di gestione del marchio, l’azienda deve garantire il controllo di filiera, ossia il rispetto dei parametri qualitativi previsti sia per la materia prima (uva) sia per i vari prodotti intermedi sino al vino, rispettando le procedure previste per le varie fasi di lavorazione ed invecchiamento ed i valori previsti per i parametri che le caratterizzano. La gestione di filiera è tipica dei processi dell’industria alimentare, ma spesso si estende anche ad altri settori, come l’industria farmaceutica e cosmetica e alcuni casi particolari di aziende meccaniche con prodotti di alta qualità.

La catena del valore di Porter è chiaramente mappabile sull’attività dell’azienda vinicola. Le materie prime sono sia le uve acquisite da altri produttori più piccoli, sia tutto quanto è necessario per la coltivazione di uva interna all’azienda. L’ottimizzazione del processo dovrà portare a stabilire prezzi accettabili sia per le uve sia per le altre materie prime. Inoltre i dipendenti (o i collaboratori temporanei) addetti alla cura della vigna e alla vendemmia avranno un costo, così come le macchine agricole e gli altri strumenti necessari per la coltivazione ed il trasporto dell’uva.

Le fasi intermedie di lavorazione dovranno essere eseguite nel rispetto delle procedure stabilite dal consorzio. La filiera obbliga a controlli di qualità sulle materie prime, sulle procedure e sui semilavorati eseguiti durante tutte le fasi della lavorazione, e il flusso informativo prevede quindi sia l’informazione associata al processo, sia una informazione proveniente dalle varie fasi e destinata all’ufficio controllo qualità. I processi amministrativi e gestionali associati dovranno interagire con l’ufficio qualità, in modo tale da intraprendere azioni immediate (es. reclami verso i fornitori o variazione delle procedure), qualora si rilevino problematiche legate alla qualità.

Sul lato vendite troviamo le necessità più importanti: il marketing, che potrebbe essere condotto anche in collaborazione con le altre aziende facenti parte del consorzio, dovrà rivolgersi non soltanto all’Italia ma anche all’estero, e quindi dovrà studiare nuovi mercati e tendere a rafforzare la propria presenza entro quelli esistenti, facendo largo uso dei vari canali di comunicazione (Web, e-mail, telefono, fax, TV...) sia per campagne, sia come strumento di comunicazione con i clienti esistenti, integrando il tutto in una politica di CRM ben studiata, tendente a mantenere il cliente fidelizzato. Il reparto vendite dovrà essere rapido, se i clienti sono anche agenzie di ristorazione o grande distribuzione, dovrà essere in grado di soddisfare tempestivamente anche richieste ad elevato volume.

I flussi informativi sono quindi raggruppabili come segue:

- Informazioni scambiate con i fornitori esterni di uve, relative soprattutto alla qualità delle uve ricevute, e alla normale procedura di acquisto;

- Informazioni scambiate con i fornitori dei prodotti per la coltivazione; anche questi fornitori devono garantire il rispetto delle regole previste per i loro prodotti;
- Informazioni sulla qualità delle procedure di coltivazione, che devono pervenire all'ufficio qualità;
- Informazioni sulla procedura di vendemmia e pigiatura, e sulla qualità del mosto ecc... che devono pervenire all'ufficio qualità per essere verificati e consentire tempestive correzioni qualora sia necessario, oltre che permettere di prevedere qualità e quantità della produzione dell'annata, per consentire una più efficiente e mirata campagna delle vendite;
- Informazioni sulla fase di invecchiamento in botte e su tutti i parametri che la caratterizzano, per i quali vale lo stesso principio;
- Informazioni sulla fase di imbottigliamento, per cui valgono gli stessi principi;
- Informazioni sulle bottiglie disponibili entro la cantina, che devono essere sempre a disposizione dell'ufficio vendite e del marketing;
- Scambio di informazioni con i clienti, sia come messaggio dall'azienda ai clienti, sia come ordini, sia come richieste di informazioni da parte dei clienti; dovendo competere sul mercato globale è importante potere evadere tempestivamente queste richieste di informazioni;
- Feedback sulla qualità e sul gradimento del prodotto da parte dei clienti, anche allo scopo di pianificare eventuali cambi di quantità dei vari tipi prodotto (es. maggiori quantità di vino invecchiato per più anni) per gli anni successivi.

Il flusso informatico deve quindi garantire la presenza di un archivio centrale di informazioni, cui pervengono i dati provenienti dalle varie fasi di lavorazione, e di un archivio sul prodotto presente in magazzino, a disposizione delle vendite, un archivio clienti per il CRM. Questi archivi devono essere intercomunicanti. Occorre quindi una serie di programmi applicativi connessi fra loro che implementino le funzionalità suddette. Occorre anche una serie di sensori (es. termometri e misuratori di umidità nelle cantine) e di strumenti di raccolta dati (es. un lettore di codice a barre per le cassette di uva in arrivo, per catalogare rapidamente i vari lotti, oppure un misuratore di acidità del mosto connesso via rete wireless con i sistemi centrali), che possano permettere una rapida ed efficiente raccolta delle informazioni sulle varie fasi della lavorazione. Questa necessità di integrazione fra le componenti informatiche e di automazione dei sistemi di produzione e le componenti informatiche dei sistemi informativi, allo scopo di automatizzare completamente i flussi di informazioni che devono andare dai primi ai secondi, è ormai sempre più presente in ogni azienda di produzione, in special modo sulle aziende che devono rispettare regolamenti di filiera.

Allo stesso modo il lato sell-side dovrà usare tutti gli strumenti di comunicazione per gestire le relazioni con i clienti.

I sistemi informatici potrebbero essere quindi composti come segue:

1. Server centrale, contenuto in una stanza con condizioni di funzionamento ottimale (la “sala macchine” dei Centri Elaborazione Dati o CED), che ospita il servizio DBMS con tutte le basi di dati corrispondenti ai vari archivi;

2. Server di posta elettronica, probabilmente nella stessa stanza; per esso valgono le stesse considerazioni fatte negli esempi precedenti;
3. File server, che ospita tutti i documenti ed è soggetto a backup; potrebbe non essere nella stessa stanza ma essere negli uffici;
4. Server di stampa; tipicamente più di uno;
5. Domain server (magari coincidente con il file server) che gestisce i diritti di accesso alla rete ed alle varie cartelle dati;
6. Postazioni utente, tutte con una dotazione di Office, più i programmi applicativi necessari per il lavoro dell'operatore che le usa, connessi come client al Database server centrale;
7. Collegamento in rete wireless, che permette agli operatori nella vigna o nei reparti di lavorazione di interagire con i sistemi centrali per leggere o modificare dei dati senza bisogno di avere un cavo di connessione;
8. Collegamento ad internet, via modem ADSL (se la zona è raggiunta);
9. Sito Web esterno, posto entro il data center di un Internet Service Provider, molto probabilmente inserito nella rete di siti Web del consorzio doc di cui l'azienda fa parte; i contenuti di tale sito vengono periodicamente aggiornati da parte degli addetti al marketing e al CRM, che monitorano anche gli accessi di clienti e potenziali clienti al sito (processo indicato spesso come observation).

Gli applicativi potrebbero essere stati realizzati ad hoc (make), in quanto non è facile trovare soluzioni software preconfezionate che supportino i processi molto specifici dell'azienda vinicola. Probabilmente il produttore software ha realizzato l'applicativo non solo per l'azienda in questione, ma per tutte le aziende del consorzio, lavorando in stretta collaborazione con il consorzio stesso ed accompagnando il design e la progettazione del software con uno studio ed una ottimizzazione dei processi business delle aziende vinicole (ossia applicando ad esse il Business Process Reengineering). In base alle caratteristiche tecniche di tale software saranno stati scelti anche i componenti infrastrutturali del sistema informatico.

In figura 10.3 sono rappresentate le componenti del sistema informatico descritto.

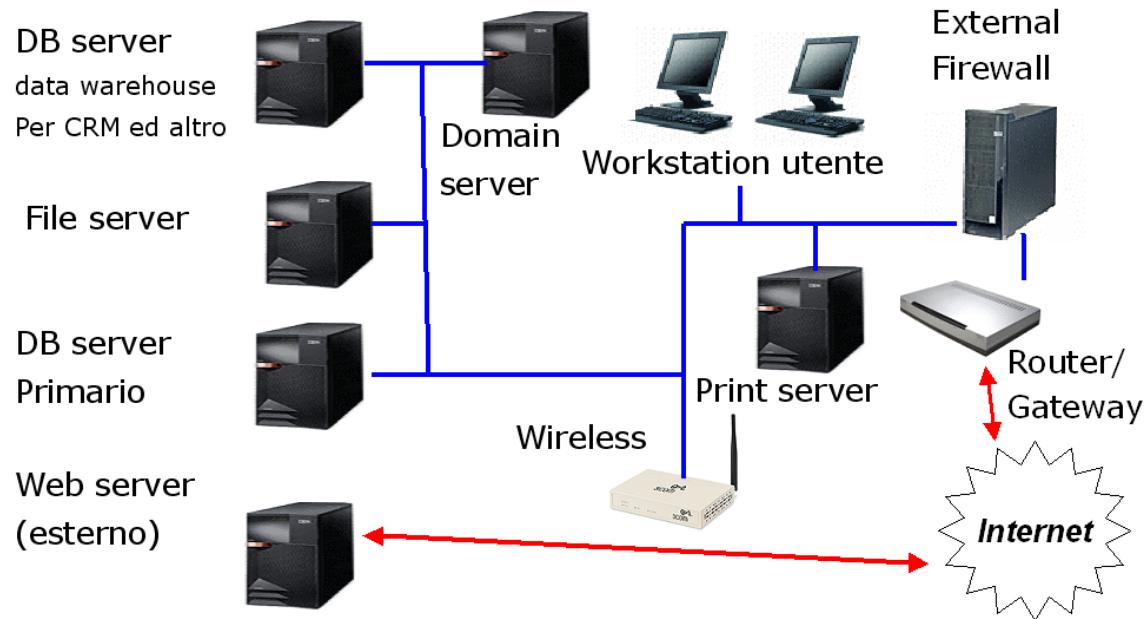


Figura 10.3: Un possibile sistema informatico di un'azienda vinicola. Non sono rappresentati esplicitamente gli strumenti wireless con cui gli operatori accedono al sistema dalla vigna o dai reparti di lavorazione.

Azienda vendita CD e libri con sito Web

Questo modello rappresenta, su piccola scala, il funzionamento di una grande azienda di distribuzione prodotti su Web, come Amazon.Com.

Il “sistema azienda” in esame è costituito dall’azienda e in particolare:

- dal sito Web, tipicamente ospitato non entro l’azienda stessa, ma nella Web farm (ossia nel data center pubblico allacciato alla rete Internet) di un fornitore di servizio;
- dai suoi uffici amministrativi
- dai suoi magazzini
- dal reparto logistica, in merito al quale occorre tenere presente che in molti casi il servizio di logistica viene esternalizzato ed affidato ad aziende specializzate (corrieri e simili).

Sono inoltre presenti altri attori che partecipano ai processi business associati all’azienda, tra cui:

il cliente, ossia colui che è disposto a pagare per ricevere beni e/o servizi e quindi provvedere all’alimentazione del valore associato al processo;

la banca, attraverso cui avvengono sia i controlli di disponibilità finanziarie relativi al cliente, sia le operazioni di pagamento vere e proprie.

Obiettivo dell’azienda che vende è far sì che i clienti acquistino il più possibile i prodotti da essa distribuiti. Saranno quindi poste in atto tutte le metodologie di marketing, CRM ed altro associate ai clienti, come visto nel capitolo 6. Ma in questo modello si vuole analizzare i componenti base del sistema azienda ed i processi in modo semplificato.

In figura 10.4 sono presentati i flussi informativi associati ai sottoprocessi del ciclo attivo (e, nel caso della consegna al cliente finale, anche flussi di prodotti), sotto forma di collaboration diagram in cui le varie componenti dell’azienda associate al processo vengono rappresentate come entità. Come indicato prima, sono presenti il sito Web, la banca (esterna all’azienda), l’amministrazione, il magazzino, il reparto logistica (il cui servizio potrebbe essere in outsourcing presso un corriere) e, ovviamente, il cliente.

Il valore è fornito dal cliente, mentre le spese delle fasi del processo sono legate:

- ai processi ausiliari dell’amministrazione
- al costo della struttura del sito Web
- al costo delle operazioni con la banca
- al costo della gestione del magazzino
- al costo della logistica.

Inoltre occorre considerare il costo del ciclo passivo: l’approvvigionamento dei prodotti ha i costi ausiliari ed il costo dei prodotti stessi, per cui il management dell’azienda dovrà operare per ridurre il più possibile tutti questi costi.

Per completezza va detto che il ciclo di pagamento ha diverse possibilità. Qui si suppone che si segua il modello in cui il cliente accetta di pagare solo a merce ricevuta, fermo restando che le eventuali spese di reso devono essere a suo carico.

Nel dettaglio le componenti informatiche dei flussi sono:

1. Interazione via Web tra cliente finale e sito dell'azienda, che danno avvio al processo quando il cliente conferma la richiesta di ordine;
2. Interazione tra il sito Web ed il database dei prodotti per verificare la disponibilità di questi ultimi, che ha luogo in un momento successivo all'ordine del cliente e deve essere fatta in quanto ci possono essere state variazioni tra i dati di disponibilità che il cliente ha visto sul sito Web e la situazione effettiva al momento dell'invio dell'ordine; nel modello considerato si suppone che il database di riferimento si trovi presso il sito stesso, ma, nel caso in cui esista un collegamento in tempo reale fra il magazzino ed il sito, il database potrebbe essere presso il magazzino; in questo caso invece la sincronizzazione tra le due basi di dati avviene ad intervalli regolari ma non in tempo reale;
3. Trasmissione dell'ordine dal sito Web verso l'amministrazione; l'ordine viene trasferito elettronicamente e quindi, per evitare la perdita di tempo di dovere reinserire dei dati entro i programmi gestionali in uso nell'amministrazione, deve esistere una interfaccia che consenta di acquisire direttamente i dati entro di essi;
4. L'amministrazione procede ai controlli sulla storia passata del cliente, se il cliente ha già acquistato prodotti presso l'azienda, e sulla presenza di credito sufficiente a coprire il valore dell'acquisto sul conto associato al numero di carta di credito fornito dal cliente interrogando la banca;
5. Se i risultati dei controlli sono positivi, l'amministrazione istituisce la pratica d'ordine e invia al magazzino le istruzioni per creare la spedizione con il lotto di prodotti acquistati dal cliente;
6. Il magazzino riceve la comunicazione in forma elettronica e, valutata la disponibilità di prodotti, che dovrebbe comunque essere sincronizzata con quanto presente nel sito, procede alla preparazione della spedizione;
7. La spedizione fisica e le informazioni ad essa associate (bolla ecc...) vengono consegnate al vettore logistico, che provvede alla consegna al cliente entro i tempi concordati dal contratto;
8. Il cliente riceve la merce, la verifica e firma il documento di accettazione; viene notificata all'amministrazione la consegna da parte del vettore logistico;
9. L'amministrazione emette fattura e la invia al cliente (ad esempio, tramite posta elettronica);
10. Il cliente procede al pagamento, oppure istruisce la banca affinché proceda.

Nel processo descritto sono presenti due punti di debolezza che possono essere rimossi con opportune modifiche:

- Il cliente dovrebbe inviare il proprio numero di carta di credito all'amministrazione per la verifica di disponibilità di contante sul conto corrente, ma questo implica che l'amministrazione possa registrare tale numero e quindi un rischio potenziale per il cliente stesso; la soluzione sta nel cosiddetto "Virtual POS", ossia in una Web form inserita entro il sito Web che invia i dati direttamente alla banca, senza passare in alcun modo per l'amministrazione; la banca ritorna all'amministrazione solo il risultato della verifica, senza sapere a quali prodotti il valore monetario da controllare era associato (salvaguardando quindi l'azienda dal rischio che qualche impiegato

disonesto della banca possa trasferire i suoi dati ad un'azienda concorrente) ed allo stesso tempo proteggendo il cliente da usi impropri del suo numero di carta di credito; ulteriori informazioni sul Virtual POS sono reperibili in [OTP 2002];

- Il fatto che il pagamento avvenga solo alla consegna espone l'azienda al rischio reale di non essere pagata, pertanto normalmente l'addebito viene fatto al cliente direttamente a partire dalla carta di credito, prima dell'avvio delle fasi successive del processo; il pagamento alla consegna è invece tipico del rapporto B2B in cui entrambi gli attori sono aziende o comunque entità giuridiche dotate di partita IVA; esistono vari meccanismi correttivi per salvaguardare entrambe le parti dai rischi, presentati in [OTP 2002].

Il processo di Amazon.Com è presentato in dettaglio in [BFM 2001], cui si rimanda per approfondimenti.

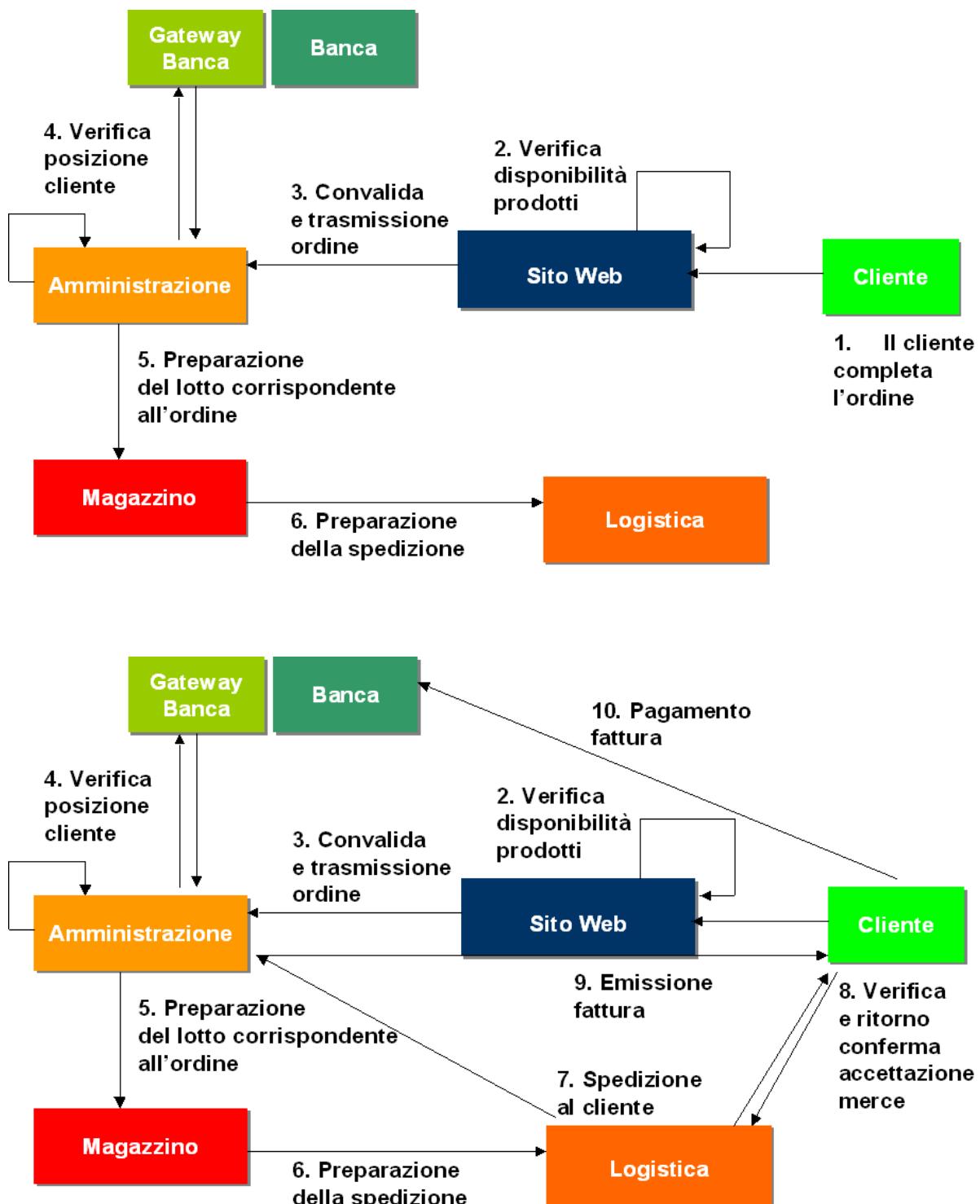


Figura 10.4: Il ciclo attivo di vendita su Web ed i suoi flussi informativi.

Domande ed esercizi

1. Quali sono i passi necessari per l'analisi di un sistema informativo?
2. Come si possono individuare i flussi informativi associati ad un "sistema azienda"?
3. Analizzare il sistema informativo necessario per una media azienda metalmeccanica.
4. Analizzare il sistema informativo necessario per un piccolo comune.

Bibliografia

[BFM 2001] G. Bracchi, C. Francalanci, G. Motta - *Sistemi Informativi e aziende in rete* – Ed. McGraw-Hill Italia, Milano, 2001

[OTP 2002] D. O'Mahony, H. Tewari e M. Peirce - *Electronic Payment Systems for E-Commerce (2nd Edition)* - Ed. Artech House, 2002

Giulio Destri si occupa di progettazione di soluzioni ICT e di formazione aziendale con Area Solutions Providers (www.areasp.com) ed i suoi partner.

Docente di "Sistemi Informativi I" ed "Ingegneria del Software" presso il Corso di Laurea in Informatica dell'Università degli Studi di Parma e di "Internet Security" e "Electronic Payment Systems" presso il Master of Management in Network Economy (MiNE) dell'Università Cattolica di Piacenza /U.C. Berkeley, ha pubblicato numerosi articoli sulle riviste Infomedia e su importanti riviste accademiche.

Si è laureato in Ingegneria Elettronica ed ha conseguito il Dottorato di Ricerca in Ingegneria Informatica presso l'Università degli Studi di Parma, specializzandosi in elaborazione distribuita e reti. Ha operato presso importanti realtà aziendali italiane come il Gruppo Barilla, il Gruppo Telecom Italia, il Gruppo CRIF, Marina Rinaldi.

Facoltà di Scienze MM. FF. NN. – Corso di Laurea in Informatica
Università degli Studi di Parma
Viale G.P.Usberti, 53/A, 43100 Parma

Euro 15,00

ISBN 978-88-7847-135-1

A standard one-dimensional barcode is located in the bottom right corner of the page. It is used to represent the book's ISBN number (978-88-7847-135-1) in a compact, machine-readable format.