

# ASPETTANDO ROBOT



La Robotica è una disciplina vastissima le cui radici culturali affondano nella storia e nella letteratura. L'impiego industriale dei Robot è iniziato negli anni '40 del secolo scorso; da allora, la sua evoluzione è andata almeno di pari passo con quelle dell'Informatica, dell'Elettronica e della Meccanica. Oggi si assiste probabilmente all'inizio di una nuova era, in cui il Robot opererà a stretto contatto con gli esseri umani, aiutandoli nei compiti della vita quotidiana e accompagnandoli nel tempo libero.

**Renato Zaccaria**

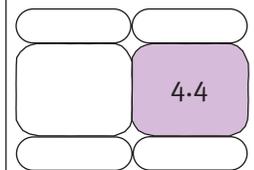
## 1. INTRODUZIONE

**P**ensare di riuscire a delineare in poche pagine un quadro sufficientemente completo, aggiornato, obiettivo, esauriente e allo stesso tempo chiaro della Robotica è presunzione forse per chiunque. Chi scrive ha iniziato a occuparsi di Robotica subito dopo la sua tesi di laurea, a metà degli anni '70, e ha avuto la non comune fortuna di appartenere fin dalle origini al miglior gruppo di ricerca possibile in questo campo: un gruppo pluridisciplinare con interessi diversificati in architetture informatiche, intelligenza artificiale, sistemi operativi, controlli automatici, bioingegneria, neuroscienze, scienze cognitive, telecomunicazioni; ognuno di questi argomenti, degnamente, ha che fare con la Robotica, senza esaurirne le esigenze scientifiche. Nessun settore della scienza, della tecnologia e del mercato ha mai avuto, infatti, un così ampio spettro di competenze e componenti disciplinari come la Robotica. A quelle che sono state già citate se ne devono aggiungere molte altre come le tecnologie dei materiali, l'elettronica – quella di potenza in particolare – e

l'elettrotecnica degli attuatori e degli impianti, l'ottica, l'acustica, l'elaborazione delle immagini; e, ultima ma non meno importante, la meccanica con le metodologie e tecnologie relative. In questo catalogo di discipline incompleto, che sarà approfondito nel paragrafo successivo, sta il grande limite alla crescita della Robotica; è difficilissimo se non impossibile, infatti, coltivare allo stesso livello di profondità le competenze necessarie, o anche solo poche fra tutte. Per questo la Robotica *non è*, ancora, una scienza, ma l'unione di scienze, discipline e tecnologie diverse ancora in attesa di una comune teoria, di un linguaggio che unifichi meccanica e *software*, algoritmi e materiali; insomma, di una matematica unica e di regole progettuali standard.

## 2. VERSO UNA SCIENZA

Molti passi in questo senso sono stati fatti, coronati anche da successo; il termine *meccatronica*, comune in Robotica, esprime la sintesi fra tecnologie meccaniche, attuatori ed elettronica di controllo; una teoria unifica-



ta della descrizione del movimento per robot mobili e manipolatori di generica geometria (il cosiddetto approccio del *configuration space*, teoria matematica basata sull'integrazione di meccanica razionale, calcolo vettoriale, geometria differenziale, algoritmi di ricerca e altri temi) è stata elaborata negli anni '80 ed è ora una vera metodologia della Robotica; lo stesso vale per metodi di visione artificiale, per linguaggi di descrizione dei compiti e dispositivi per controllo e regolazione di manipolatori e così via. Ma non appena si esce da situazioni standard, quali possono essere, ad esempio, la sequenza di operazioni di assemblaggio in una linea di produzione, il progettista robotico diventa rapidamente un artigiano che è costretto a inventare le metodologie di progetto e gli strumenti di sviluppo. Non c'è rimedio a questo problema, se non lavorare sulla ricerca di base e mantenere ad alto livello le *skill* del tecnico robotico.

### 3. L'AMPIEZZA DEL SETTORE

Se poi si pone attenzione alle tendenze sia di ricerca, sia di espansione dell'industria verso nuovi mercati, si deve, necessariamente, allungare l'elenco di discipline con nomi di recente acquisizione, per la Robotica, come la biomeccanica, le nanotecnologie, la modularità meccanica e funzionale, gli "ambienti intelligenti" (*smart house*), la "tecnologia dell'emozione" (*kansei*), la psicologia, l'*edutainment* (*education-entertainment*) e i modelli ad agenti.

Con un così ampio fronte tecnico-scientifico si capisce perché parlare di Robotica *tout court* non è possibile. In questo contributo ci si limiterà, anche per ragioni di spazio, a una serie di brevi paragrafi su alcuni temi specifici che si ritengono più attuali e significativi, legati, in particolare, al già vasto settore IT (*Information Technology*), puntualizzando che molti altri argomenti resteranno inesorabilmente assenti. Alcuni, in realtà pochi, dei termini dei tre elenchi tematici sopra riportati saranno espansi e commentati in questi paragrafi.

Quelli che non saranno citati sono altrettanti titoli dietro ai quali possono esser lette storie affascinanti, per le quali l'autore spe-

ra almeno di riuscire a suscitare la curiosità del lettore.

### 4. "STILL WAITING FOR ROBOTICS"

Il titolo di questo articolo, *Aspettando Robot*, sintetizza il pensiero, comune a tutti i ricercatori in Robotica, secondo cui una *scienza robotica* è, per le ragioni espresse poc'anzi, un'attesa colma di speranza. Allo stesso tempo, il Robot che si sogna (autonomo, intelligente, "simpatico", probabilmente *asimoviano*) è ancora lungi da venire, nella ricerca come nel mercato.

*Aspettando Robot* è anche un titolo "rubato" a un saggio del 1987 [3] in cui 32 ricercatori italiani esprimevano le loro considerazioni sull'Intelligenza Artificiale, con molti riferimenti alle applicazioni in Robotica, molto interessanti da rileggere oggi. Uno di questi ricercatori era Marco Somalvico, professore del Politecnico di Milano, dal quale dieci anni prima chi scrive, come altri amici e colleghi, aveva appreso le prime definizioni rigorose sulla Robotica; insegnamenti, che egli, appassionato come pochi, dispensava assieme alla passione per la ricerca, senza gelosie, sia agli allievi del suo gruppo che a chiunque altro dimostrasse sincero interesse. Marco Somalvico è scomparso inaspettatamente pochi mesi fa, e a lui che è stato pioniere della ricerca in Robotica in Italia è dedicato con affetto sincero e con vera commozione questo articolo, con l'augurio che nella storia di questa disciplina il suo nome venga sempre ricordato.

### 5. LA STORIA E LE FASI

Non c'è una sola data per la nascita ufficiale della Robotica. Anche se spesso vengono schematizzate fasi storiche simili a quelle delle "generazioni" del calcolatore e dell'elettronica digitale, l'evoluzione del robot – anche solo di quelli industriali più classici – è fatta di molti episodi significativi, in ambiti non necessariamente tecnico-scientifici. L'invenzione del termine stesso "Robotica" (*Robotics*) è normalmente attribuito a Isaac Asimov che lo introdusse nei suoi famosi racconti sui *robot positronici* alla fine degli anni '40. Fino a quell'epoca il robot era sostanzial-



mente un fatto letterario e cinematografico (il primo robot del cinema è *Maria*, in *Metropolis* di F. Lang, 1926). Contemporaneamente, però, i primi telemanipolatori venivano costruiti e usati per gestire in sicurezza sostanze radioattive. In essi erano già presenti le componenti base dei robot manipolatori attuali, cioè una catena cinematica articolata con un *end effector* (una pinza), con motori che azionavano i vari assi rotazionali nei giunti, un dispositivo di controllo analogico di posizione su di loro, e un'interfaccia di comando che oggi si possono definire *haptic*: un esoscheletro articolato, un *master*, azionando il quale si controllano identici movimenti del manipolatore remoto (*slave*). Si può quindi, presumibilmente, situare nella seconda metà degli anni '40 la nascita dei robot meccanici per uso pratico; nei telemanipolatori erano presenti le tecnologie e le metodologie di qualunque robot successivo a esclusione, ovviamente, di quella informatica: meccanica, elettrica, elettronica, teoria del controllo.

La prima idea di uso diverso e industriale di queste tecnologie è della metà degli anni '50; all'inizio degli anni '60 la statunitense *Unimation Inc.* introduce sul mercato *Unimate*, il primo manipolatore industriale *ante litteram*. Da quel momento, si sviluppa un'industria robotica e un mercato in veloce espansione, per molto tempo limitato ai bracci manipolatori, che anzi sono ancora oggi il robot industriale per antonomasia (la definizione "ufficiale" ISO è, infatti, quella del manipolatore per applicazioni di fabbrica). Negli stessi anni, viene sviluppata l'indispensabile teoria di base per la descrizione e il controllo del movimento in geometrie di bracci qualsivoglia: il contributo dell'informatica, in questa fase storica, è marginale.

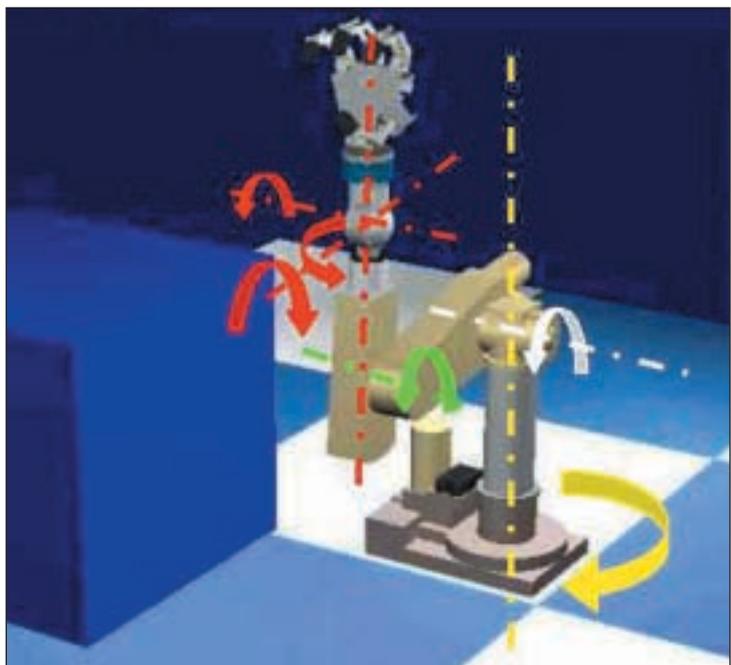
## 6. OLTRE LA FABBRICA

Il Giappone inizia a occuparsi di Robotica con quasi dieci anni di ritardo, alla fine degli anni '60, con le prime applicazioni nell'industria automobilistica (il Giappone importò ufficialmente il primo manipolatore statunitense nel 1968). Nel 1971, nasce la *Japanese Industrial Robotic Association* (JIRA), che ha influito sullo sviluppo di tutta l'industria ro-

botica per i successivi venticinque anni. È negli anni '70 che l'industria robotica mondiale assume una presenza non marginale, con un cospicuo mercato per i suoi prodotti: oltre a Unimation, AMF, Cincinnati, Adept, IBM negli USA; Fanuc, Yaskawa, Seiko, in Giappone e ABB, Kuka, DEA, Comau, in Europa. La guerra commerciale si svolge sul terreno della complessità e della versatilità dei bracci manipolatori, legata sostanzialmente al numero e al tipo dei suoi *gradi di libertà*, che determinano le capacità di articolarsi in posizioni e orientazioni diverse della sua "mano" (*end effector*). Come paragone comune, il braccio umano (antropomorfo per eccellenza) ha, articolazioni della mano escluse, sette gradi di libertà, mentre sei è il minimo teorico per portare l'organo terminale in qualunque posizione e orientazione all'interno dello spazio di lavoro. I robot manipolatori a sei gradi di libertà (Figura 1), che in industria sono detti comunemente *antropomorfi*, sono diventati uno standard alla fine degli anni '70, con il *Puma* della Unimation Inc., ancora oggi comune in molti laboratori di ricerca. Si tratta di un manipolatore a sei assi, con carico di poco più di un chilogrammo, con pinza intercambiabile, e spazio di lavoro di 2 m circa. A parte i miglioramenti nei materiali e nelle prestazioni (peraltro di non più di un ordine di grandezza, mentre le

**FIGURA 1**

Schema di un tipico braccio manipolatore a 6 gradi di libertà (o 6 assi) su cui è montato un particolare organo di presa



prestazioni dei sistemi di calcolo sono aumentate, contemporaneamente, di un fattore almeno  $10^6$ , si veda a tal proposito la Figura 2), questa geometria e queste caratteristiche di massima si ritrovano nella maggior parte dei modelli attuali (Figura 3).

Il Giappone costruì da zero nella decade degli anni '70 un enorme mercato interno per la Robotica industriale, influenzando in modo decisivo sul resto del mondo.

Nel 1982, la JIRA preparò, su richiesta del Ministero dell'Industria e del Commercio Internazionale, uno studio molto profondo sull'industria robotica nazionale dal titolo *The Robotic Industry of Japan – today and tomorrow* che fu al tempo stesso visionario, nel senso positivo di preveggenza e ambizioso, e miope. Quel documento fu il segno e il sostegno dell'influenza economica, tecnologica e culturale che la Robotica giapponese impose al resto del mondo. Il documento partì da un dato impressionante: in meno di

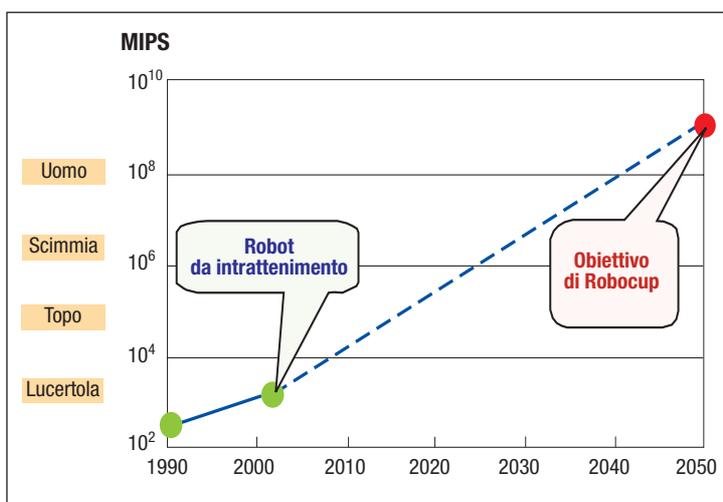
10 anni, il mercato interno aveva già assorbito quasi 80.000 robot, in gran parte manipolatori ma anche AGV (*Automated Guidance Vehicle*), impiegati nell'industria manifatturiera, di cui il 75% fra automobilistica, elettrodomestici, plastica.

L'industria robotica contava 290.000 addetti su 150 società con prevalenza di grandi imprese, mentre le PMI rappresentavano il 90% dei nomi ma solo il 9% degli addetti.

Per un rapido raffronto, basti considerare che 20 anni dopo (2000) il Giappone contava circa 400.000 robot al lavoro, l'UE 170.000 e gli USA 90.000.

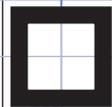
Il documento della JIRA fu visionario per diverse ragioni. Innanzi tutto, propose come strategia di sviluppo una definizione di robot industriale ben più ampia di quella vigente, e ancora oggi avanzata: "Un sistema [...] capace di eseguire funzionalità diversificate con molti gradi di libertà [...] con funzionalità sensoriali e di riconoscimento per comportamenti autonomi (*intelligent robot*)". In più, oltre ad affermare ufficialmente l'esistenza di un mercato per un simile *robot intelligente*, in netto contrasto con l'impostazione classica delle macchine a *Controllo Numerico* (NC), si spinse molto più in là di quanto studi analoghi facessero in America e in Europa, proponendo alla R&S in Robotica industriale un elenco stupefacente di settori non manifatturieri: *agricoltura, allevamento, foreste, oceano, edilizia, logistica e trasporti, gas, acqua e fognature, elettricità e telecomunicazioni, posta, energia, nucleare, spazio, medicale, riabilitazione, protesi, rifiuti, emergenze e disastri, difesa, ordine pubblico, servizi,*

**FIGURA 2**  
L'evoluzione della capacità di calcolo, secondo Moravec, per soddisfare gli obiettivi futuri di Robocup



**FIGURA 3**  
Una mano con quattro o cinque dita può avere 10 - 15 gradi di libertà e una ventina di motori





educazione e scuola, traffico urbano, manutenzione, pulizia.

Ognuno di questi settori fu previsto come settore di applicazione e di espansione del mercato. Terza lungimiranza fu l'analisi delle attività di R&S industriali e universitarie e la proposta del loro potenziamento mirato per far fronte ai mercati di nuova definizione.

A fronte di queste caratteristiche "illuminate", nel documento è presente una sorprendente miopia: l'autolimitazione di riferirsi unicamente al mercato interno, con totale assenza di studio e previsioni su quello internazionale. Questa visione autarchica è, probabilmente, il motivo per cui lo sviluppo non ha seguito i livelli di innovazione previsti. I ritorni dai mercati internazionali sarebbero dovuti avvenire grazie alla maggior competitività dei prodotti *low-tech* e *hi-tech* giapponesi realizzati da un'industria robotizzata, in una società robotizzata e, quindi, più competitiva, restituendo in via indiretta gli ingenti investimenti impiegati in una Robotica industriale "di quinta generazione" non venduta direttamente all'estero. La fragilità di questo assunto è evidente, e ad esso si può ricondurre il mancato sviluppo che non inficia la corretta analisi dei mercati potenziali della *Robotica intelligente* (si veda a tal proposito il paragrafo sulle prospettive future). Anche qui... si sta *aspettando robot*.

## 7. LA CULTURA E LA FICTION

La *fiction* ha sempre giocato un ruolo culturalmente significativo, nella Robotica, e questo non ha probabilmente paragoni in altre discipline o tecnologie. Un grande settore tecnologico molto legato alla letteratura da 2000 anni è, per esempio, quello della nave e del mare (a partire dall'Odissea, per esempio, il cui autore è considerato il primo ad aver introdotto dei robot nella narrazione, pur in un piccolo ruolo marginale). Eppure, la portata di quella sinergia tecnologia/fantasia non è paragonabile a questa fra automa e vivente, e soprattutto non è altrettanto bidirezionale nelle influenze. Oltre ai modelli logico-letterari di Isaac Asimov, dai quali pochi esperti in Robotica possono prescindere, negli anni '50 il cinema sviluppò modelli di pura fantasia ma con un impatto immaginifico che

nessun robotico ha mai potuto trascurare negli anni successivi.

L'icona forse più famosa di robot antropomorfo della storia è quella di Robby ne *Il Pianeta proibito* (*The Forbidden Planet*, 1954, film "dotto" ispirato a *La Tempesta di Shakespeare*). Robby è un robot asimoviano, cioè buono e dotato di personalità e integrato nella storia a pieno titolo: è il folletto Ariel del riferimento shakespeariano. Ancora oggi, esistono *club* di appassionati di quel robot e aziende che ne commercializzano modelli anche in grandezza naturale. Gli attuali robot da "compagnia" e antropomorfi giapponesi sono la realizzazione industriale, alla fine, di quell'idea di robot di cui 45 anni fa si producevano modellini giocattolo in latta, con i quali si giocava da bambini. In questo senso, si può parlare di "rivincita" dell'effimero in Robotica rispetto al concreto; oggi, esiste un emergente mercato per robot ideati puramente per il divertimento o la compagnia (*entertainment robotics*), o l'aiuto nella vita quotidiana (*service robotics*, *robot companion*, *edutainment robotics*) che sviluppa in un certo senso il modello del robot della letteratura rispetto a quello della fabbrica (che discende, a sua volta, dal modello tayloristico di radici settecentesche). Sul tema dell'*entertainment robotics* si accennerà in un paragrafo successivo.

Come è stato già ribadito, in Robotica l'immaginario e il letterario hanno sempre avuto un'importanza di gran lunga superiore a quanto accade in altre discipline, e continuano a contribuire in modo virtuoso allo sviluppo della conoscenza e addirittura, come accade nell'*entertainment*, a delineare nuove applicazioni. Ciò non deve sorprendere: il robot è l'Essere Artificiale, presente nei miti, nelle fantasie e nelle speculazioni dell'uomo da duemila anni e più. Questo mito si è espresso nei secoli della meccanica in un'inimmaginabile varietà di robot da intrattenimento apparentemente senza scopo, in cui si estrinsecava l'interesse puramente speculativo dell'uomo verso la definizione di cos'è la vita e verso la ricerca di una *vita artificiale* (*artificial life* è, non a caso, uno dei termini con cui oggi si riconoscono ricercatori che investigano l'*emergere spontaneo* di comportamenti complessi a partire dall'interazione



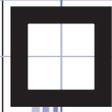
**FIGURA 4**  
 Alcune generazioni di robot giocattolo Sony. A sinistra l'umanoide SDR-4X: 38 gradi di libertà, riflesso di autoprotezione in caso di caduta, 2 videocamere e 7 microfoni

di comportamenti di semplici robot autonomi). La fantastica storia degli automi nei secoli (questo nome è stato sostituito nelle lingue occidentali dalla parola *robot*, a partire dagli anni '30; anche questa sostituzione, come è noto, viene dalla letteratura, anzi dal teatro) è narrata, per esempio, in *Storie di automi* [2]. La conoscenza del complesso, e per molti versi ancora inesplorato, universo "robot fantastico/robot reale" ha incuriosito molti autori sia tecnici, sia umanisti; si veda, per esempio, *Dizionario degli esseri umani fantastici e artificiali* [5], in cui è tentata una incredibilmente meticolosa classificazione dei percorsi culturali, immaginifici, scientifici e tecnologici degli "esseri" riconducibili a robot, reali e fantastici. Ancora per ribadire la forte presenza del cinema nella Robotica, oltre che citare la fortunata serie di robot che il cinema ha riscoperto negli ultimi 15 anni, è utile ricordare la serie di film anche di "categoria B" degli anni '40-'60 in cui l'idea di robot si formava a volte con rara preveggenza pur se con povertà di effetti speciali. Un film poco noto, *Attacco a base spaziale USA* (GOG, 1954) mostra con sorprendente rigore scientifico l'impiego di robot mobili, usando termini attuali, con manipolatori a bordo, in un centro di ricerca automatizzato, integrati in rete e controllati da un elaboratore centrale operante in *real time*. Non solo queste elaborazioni fantasiose partecipavano al processo di aggiornamento della cultura diffusa con scienza e tecnologia, ma facevano anche sviluppare, lentamente e poi sempre più velocemente, il settore della Robotica "effimera", quella dei modelli animati per il cinema - in cui pioniere fu l'italiano Carlo Rambaldi - per i parchi divertimenti a tema, fino agli at-

tuali robot "da compagnia" come il cane ALBO o il recente piccolo umanoide SDR-4X, entrambi della Sony, e simili (Figura 4).

## 8. INFORMATION TECHNOLOGY E ROBOTICA INDUSTRIALE

L'Informatica, o più in generale l'IT, ha iniziato a giocare un ruolo chiave relativamente tardi nella Robotica Industriale. Nella preistoria della Robotica, le tecnologie dominanti erano la meccanica, l'elettrica, l'elettronica e le tecniche di controllo automatico. Negli anni '60, le tecnologie informatiche raggiungevano, invece, l'apice in automazione di fabbrica con le macchine a *Controllo Numerico* (NC) in cui unità logiche di controllo (all'inizio semplici reti logiche, e poi reti programmabili, dispositivi flessibili come i *Programmable Logic Controller*, i microcontrollori...) regolavano ogni singolo asse delle macchine. Nel 1970, al *Machine Tool Show* in Chicago fu enunciato un principio che ha fatto molta strada: il *Direct Numerical Control* (DNC), o integrazione degli algoritmi di controllo numerico dei molteplici *assi* a livello dell'*intera fabbrica* (con un controllo centralizzato della produzione), entrando in sinergia con il giovane CIM (*Computer Integrated Manufacturing*). Il robot era per natura adatto a sviluppare questa perfetta integrazione; CIM e Robotica sono divenuti per trent'anni un binomio inscindibile, in cui l'IT è substrato teorico, metodologia e anche costo prevalente di sviluppo. L'evoluzione del robot ha perciò iniziato a seguire, dai primi anni '70, tutte le tappe fondamentali dei sistemi operativi, dei linguaggi, delle reti informatiche, degli standard e delle metodologie per l'automazione della produzione. Si sono create unità di controllo sempre più intercambiabili, con interfacce ai *fieldbus* per automazione. Si sono definiti standard, come STEP (*Standard for the Exchange of Product Model Data*) per la comunicazione fra modelli CAD (*Computer Aided Design*) e PDM (*Process Data Management*). Attualmente, si stanno diffondendo modelli OLE Microsoft (*Object Linking and Embedding*) per controllo di processi per l'interconnessione di macchine di produttori diversi, come pure standard basati su XML (*eXtensible Markup Language*).



Fra il 1997 e il 1999, il mercato internazionale ha avuto per la prima volta una leggera flessione, in cui solo il Giappone ha avuto un calo di vendite, seguito da una ripresa che ha riportato in aumento il 2000. Dedurre che il mercato della Robotica industriale sia saturo non è corretto. Piuttosto, il problema è da ricercarsi nelle necessità di rinnovare i modelli produttivi e vecchie o ingenuie politiche di impiego dell'IT (forse le stesse ingenuità che hanno portato al collasso delle imprese *dot-com* in poco più di un anno). Questa necessità sta comportando una sempre più raffinata integrazione fra le tecnologie di ingegneria del software, reti di automazione, applicazioni in real time, sistemi informativi, Internet e modelli di progettazione, pianificazione e gestione, anche basati su virtualità e simulazione. In queste si sta rivelando un valore aggiunto della componente IT cui gli stessi enti di formazione non sono pronti a far fronte, aprendo un problema di *digital divide* di cui parlerà nel seguito. Lo sviluppo dei modelli B2B (*Business to Business*), in particolare, si sposa molto bene con l'impiego di robot, flessibili e riconfigurabili. L'esempio della svedese *Sandvik Coromant*, produttrice di utensili da taglio per macchine automatiche, è interessante a questo proposito. Un cliente, in qualunque parte del mondo, può progettare il proprio utensile via *web*, tramite un CAD messo a disposizione in rete, con la possibilità di concludere il proprio ordine in Internet. Se il prodotto è standard viene spedito entro 24 h. Altrimenti, attraverso il sistema informativo, e i sistemi PDM e MRP (*Master Production Schedule*), che comportano la programmazione dei robot in automatico, viene avviata la produzione con una consegna garantita in due settimane.

Non è per niente vero, quindi, che il mercato della Robotica manifatturiera sia saturo. Il problema è far fronte a nuovi modelli che chiedono, oltre all'abbattimento dei costi, di aumentare la velocità e la flessibilità. Il classico settore dell'automobile, ancora fondamentale per la Robotica, richiede attualmente di migliorare qualità e velocità: negli USA, il tempo di attesa per un'automobile acquistata tende ad essere di una settimana, in Europa di sei. Tutte le proiezioni del mercato individuano ancora per almeno cinque anni,

a tecnologia costante, una crescita costante, con un mercato potenziale di 980.000 robot nel mondo contro i 750.000 del 2000. Benché in senso assoluto le prestazioni dei robot non siano evolute che di pochi ordini di grandezza dal 1970 (un'infima quantità rispetto ai miglioramenti dell'IT), solo negli ultimi 10 anni i prezzi dei robot sono calati fra il 45 e l'80%, mentre il costo del lavoro nei paesi tecnologicamente avanzati è aumentato del 40%; il tempo di obsolescenza di un robot va dai 10 ai 15 anni, mentre l'età media dei lavoratori aumenta in modo preoccupante in tutti i Paesi industrializzati. Anche l'aumento delle prestazioni, in assoluto non elevato rispetto ad altre tecnologie, mostra un tasso di crescita in aumento: fra il 1990 e il 2000, la capacità di carico manipolato (il *payload*) è cresciuta del 26%, la precisione del 61%, la velocità del 39%, il MTBF del 137%, il massimo numero di gradi di libertà del 45%. Tutte queste cifre si riflettono, ovviamente, sulla produttività. Mentre dal lato "informatico" (quello che determina realmente costi e produttività), cioè la programmazione e l'integrazione nella rete della fabbrica, il robot ha tutto il beneficio delle moderne tecnologie software e di sistema.

È facile concludere che l'ascesa prevista negli anni '80 da studi come quello citato del JIRA della Robotica nell'industria è stata rallentata dall'imprevedibile rottura delle barriere fra Est e Ovest e fra Nord e Sud del mondo, con la disponibilità temporanea di lavoratori qualificati a costi più bassi. È, tuttavia, molto probabile, e certamente anche socialmente auspicabile, che questa "concorrenza" fra uomini e robot in fabbrica sia destinata ad attenuarsi nel prossimo futuro.

## 9. NON SOLO MANIPOLATORI (I)

È stato citato il telemanipolatore negli anni '40, il braccio manipolatore negli anni '60, l'integrazione fra Robotica e CIM negli anni '70 e '80, i robot per il cinema e il divertimento. Altre nascite nella famiglia robotica sono state importanti. Non è noto quando sia nato il primo veicolo robotizzato autonomo (il primo *robot mobile*). Generalmente, questo primato viene attribuito a Shakey, prototipo presso lo *Stanford Research Institute* nel 1967, che era

dotato di sensori di tocco, telecamera, controllo digitale in real time e componenti di Intelligenza Artificiale. Da allora in poi, la tecnologia del veicolo robotizzato (AGV) si è sviluppata sia in ambito di ricerca sia in applicazioni industriali nella movimentazione, e fa ora parte a tutti gli effetti dell'automazione della fabbrica. In realtà, gli attuali robot industriali sono quasi essenzialmente bracci manipolatori applicati a vari settori specifici, più i veicoli mobili che realizzano tutte quelle attività di trasporto non convenienti da realizzare con macchine apposite (per esempio, convogliatori a nastro o a rulli). La fotografia dell'impiego dei robot industriali mostra attualmente (dati 2000) il 6,8% in lavorazioni (taglio, finitura...), il 12,2% nelle materie plastiche stampate, il 22,4% in assemblaggio, il 2,8% nel packaging, il 23,9% nella saldatura, e 11,8% nel trasporto, più un 20% in altre attività varie. L'AGV ha, quindi, un proprio settore ben definito, considerato che in tutti gli altri vengono impiegati bracci manipolatori. Il settore degli AGV è particolarmente interessante, visto che si basa su *know how* specifico con la possibilità di sviluppare un proprio mercato al di fuori della fabbrica: nella Robotica di Servizio, nell'automazione della guida delle automobili, dei sistemi di trasporto e movimentazione sulle banchine portuali e dei treni.

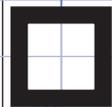
Un altro settore degno di nota è la saldatura robotizzata, che ha campi di applicazione diversi dalla fabbrica tradizionale, molti dei quali ancora da sviluppare; se è già presente nella cantieristica navale, può ulteriormente estendersi verso l'impiego a bordo, nello spazio, l'impiego subacqueo, le pipeline, le costruzioni metalliche in genere. Ed è da notare che, rispetto alle tecnologie robotiche industriali, la saldatura robotizzata richiede la gestione di sensori e l'integrazione con sistemi informativi e PDM, quindi lo sviluppo e l'impiego di conoscenze di maggior valore, tutte di tipo IT.

### 9.1. Non solo manipolatori (II)

L'evoluzione scientifica e tecnologica del Robot è costellata di molti episodi, piccoli e grandi: sarebbe impossibile elencarli tutti e dare a ciascuno di loro il giusto risalto. Ricordarne alcuni permette, però, di introdurre la differenza fra Robotica Industriale e la ricerca

in Robotica. Quest'ultima, è un'attività oggi estremamente differenziata nelle sue componenti disciplinari. Non è troppo semplificato affermare che, dal punto di vista storico, la ricerca in Robotica ha sviluppato soprattutto cinque settori: la *meccanica/meccatronica*; la *teoria del controllo*; le *tecnologie dei materiali* (per le parti strutturali, gli attuatori e i sensori); i *sistemi di programmazione*; l'*Intelligenza Artificiale*. In questi settori, fra l'altro, si è verificata la massima interazione con il contesto industriale. Non si parlerà, in questo contesto, per evidente mancanza di spazio, dei primi 4; d'altra parte, qualche accenno si trova, comunque, negli altri paragrafi. L'*Intelligenza Artificiale* (IA), invece, ha un legame così forte con la Robotica da richiedere alcune precisazioni.

L'IA [4] nasce indipendentemente dalla Robotica [1], anche se in quasi tutte le sue specifiche branche, anche le più teoriche, ha comunemente fatto riferimento a paradigmi robotici nello sviluppo delle sue teorie. La nascita ufficiale risale a una proposta di un gruppo di ricercatori (John McCarthy, Marvin Minsky, Alan Newell, Herbert Simon e Arthur Samuel) in uno storico convegno nel lontano 1956. Secondo una definizione estesa, il suo obiettivo è quello di studiare "tecniche che permettono di progettare [...] *hardware* e [...] *software* capaci di fornire [...] prestazioni che, a un osservatore comune, sembrerebbero essere di pertinenza esclusiva dell'intelligenza umana". Cioè si propone di "migliorare ed estendere le prestazioni" di un sistema informatico (e, quindi, anche un robot comandato da un calcolatore) rendendolo più flessibile, riconfigurabile, autonomo e con cui dialogare in modo migliore e più efficace (le citazioni fra virgolette in questo paragrafo sono tratte da Marco Somalvico, *Emula e non simula*, in "Aspettando Robot", [3]). Questo è stato fatto non necessariamente studiando e simulando i meccanismi propri della mente umana, bensì sperimentando e scegliendo "il modello [...] che garantiva [...] le migliori prestazioni". Il modello fondamentale era, e continua a essere, quello basato sulla *logica computazionale*, ovvero su teorie e rappresentazioni logiche nelle quali un dimostratore automatico universale, efficientemente imple-



mentato su una macchina digitale, ha il compito di risolvere il problema; con la particolarità che del problema non è necessario inventare una soluzione algoritmica ma “soltanto” una opportuna descrizione (rappresentazione della conoscenza) con un linguaggio di derivazione *logica* (dichiarativo anziché procedurale - quest’approccio era, fra l’altro, quello tentato e purtroppo fallito nel progetto della cosiddetta *quinta generazione* dei calcolatori). Rispetto alla rigidità iniziale, questi metodi hanno rapidamente prodotto capacità di gestire l’incertezza e il “rumore” dei dati del mondo reale, estendendo la logica con concetti di *fuzzyness* o di statistica, divenendo così adatti ad applicazioni in Robotica reale.

In Robotica, l’approccio dell’IA ha, in qualche modo, diviso le comunità dei ricercatori di cultura informatica, con i seguaci della IA “pura” orientati a risolvere problemi di più ampia portata e generalità (come la rappresentazione di ampie categorie di compiti, l’interpretazione di scene, la generazione di piani complessi di azioni, la diagnostica automatica, i sistemi di dialogo ad alto livello con un operatore, e simili), mentre altri robotici meno legati alle metodologie basate sulla logica computazionale si sono dedicati alla programmazione efficiente, alla risposta veloce ai sensori, alla definizione di algoritmi per problemi specifici di visione e movimento e così via. Questa distinzione è, però, molto schematica e vale per i primi trent’anni del settore. L’IA, in effetti, abbraccia oggi una grande varietà di approcci metodologici orientati al comportamento autonomo e intelligente del robot. All’inizio degli anni ’90, in un dibattito interno “fondazionale” suscitato da Rodney Brooks, è emersa una linea che negava la centralità alla *rappresentazione della conoscenza* e faceva *emergere* il comportamento intelligente dalla organizzazione, o autorganizzazione, di entità autonome semplici o elementari, dotate di scarsa o nulla capacità di rappresentazione. In molti casi, è stato introdotto l’apprendimento in forma di autorganizzazione con o senza *reward/punishment*. Senza entrare nel dettaglio e, quindi, con una certa dose di approssimazione, questo filone metodologico è rapidamente cresciuto

comprendendo macchine a stati distribuite (*subsumption, behavior architecture*), reti neurali, sistemi *genetici*, e altri approcci come quello dei sistemi *ibridi* in cui si cerca di far cooperare i due modelli computazionali alternativi (quello *logico* e quello *procedurale*). Il primo modello, infatti, è efficiente in attività più astratte come pianificazione, interazione uomo-macchina, memorizzazione; l’altro nel controllo ed esecuzione (i cosiddetti aspetti *reattivi*). Un recente settore dell’IA, che dedica molta attenzione nell’unire questi aspetti, è specialmente rivolto alla Robotica sotto il nome accattivante di *Cognitive Robotics*.

Robotica e IA nascono più o meno contemporaneamente, quaranta anni fa. C’è da chiedersi quanto l’IA abbia, fino ad oggi, influenzato la Robotica negli aspetti di mercato (che è appannaggio, si è detto, della Robotica Industriale). La risposta necessariamente riflette una visione personale, ed è inevitabilmente ambigua. Benché sia stato citato Shakey come primo caso di robot mobile che integrava sensori e movimento: Shakey era, infatti, un esperimento di IA applicata alla Robotica, chi scrive non crede che l’IA abbia contribuito *direttamente* e in modo rilevante alla Robotica “di mercato”. Non sembra che esista alcuna tecnologia che sia il risultato di un trasferimento diretto di un risultato di ricerca in IA alla Robotica Industriale. Ciononostante, si ritiene che il contributo dell’IA sia, e sia stato, fondamentale e non occasionale per almeno tre ragioni. La prima è che l’IA è un “generatore di obiettivi”: senza la spinta degli obiettivi ambiziosi che l’IA poneva innanzi alla Robotica *Intelligente* (o *autonoma*), non si sarebbe mantenuto il *trend* di ricerca e di innovazione che ha migliorato le tecnologie correnti. La seconda è che l’IA è un grande “generatore di *set-up* sperimentali” standard e avanzati (di cui Shakey è il primo) offerti alla comunità di ricerca in Robotica. Spesso questi *set-up* di riferimento – anche solo teorici, a volte tacciati di costituire *toy world* su cui costruire teorie – si affermano spontaneamente attraverso le pubblicazioni scientifiche; in molti casi, attraverso eventi internazionali, quali le *robot competition* dell’AAAI (*American Association for Artificial Intelli-*

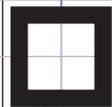
**RoboCup** ([www.robocup.org](http://www.robocup.org)) è un progetto internazionale rivolto all'Intelligenza Artificiale, alla Robotica e alle discipline collegate, che propone un problema di riferimento standard a tutti i ricercatori del mondo. In questo esperimento – una partita di calcio fra robot completamente autonomi – metodologie e tecnologie diverse debbono venir integrate al meglio, sotto la spinta della gara e del gioco. Il paradigma del gioco del calcio robotizzato si è rivelato perfettamente adatto a stimolare ricerca e innovazione trasferibili a problemi significativi nell'industria e nei servizi. RoboCup si pone come meta finale - entro il 2050 - un incontro fra una squadra di *umanoidi robot* e la squadra (di umani) campione del mondo, in cui la squadra di robot possa vincere. In questo senso, l'obiettivo è simile a quello che si è posto per molti anni la famosa *partita a scacchi* fra uomo e computer, vinta per la prima volta dal computer *Deep Blue* nel 1997. Proprio in quell'anno RoboCup è nato nella conferenza IJCAI-97 a Nagoya, dopo quasi 5 anni di preparazione. Mantiene al suo interno tre componenti: RoboCupSoccer (gioco del calcio), RoboCupJunior (educazione in Robotica rivolta ai giovani) e RoboCupRescue (applicazioni dei risultati tecnico-scientifici di RoboCup al campo dei grandi disastri e calamità naturali). Nella RoboCupSoccer si giocano partite in simulazione, fra robot mobili (su ruote), e fra robot "con zampe", bipedi compresi. L'Italia fino al 2000 partecipava, a differenza degli altri *team*, con un'unica squadra nazionale in cui robot diversi, e ricercatori di diverse Università si integravano, raggiungendo notevoli risultati. Attualmente, vi partecipano team di diverse sedi universitarie italiane.

genza) fino alla recente **RoboCup** (che quest'anno avrà il suo appuntamento internazionale a Padova) che costituisce, probabilmente, l'esperimento scientifico più grande mai organizzato per numero di studiosi partecipanti e per durata negli anni. E sull'importanza di esperimenti standard, ripetibili e accessibili alla comunità internazionale nessuno può essere in disaccordo. La terza e ultima ragione è che molti dei sogni dell'IA applicata alla Robotica potrebbero avverarsi quando i mercati si aprissero ai robot intelligenti proposti da quarant'anni dall'IA; e questo, come si vedrà nel paragrafo successivo, sembra prossimo (forse non si dovrà *aspettare robot* ancora molto a lungo). Con tutti i limiti che un'analogia può avere in questo campo, si potrebbe dire che l'IA sta alla Robotica come la Formula 1 sta all'industria automobilistica: spinta all'innovazione e terreno di prova per tecnologie che nessuna azienda potrebbe permettersi. E perché no: anche *business* e divertimento!

## 10. UNA EVOLUZIONE COMPLESSA

Diversi eventi tecnologici piccoli e grandi hanno fatto sviluppare la Robotica Industriale; in molti, indirettamente, c'è l'effetto di un risultato IT o IA in particolare. L'evoluzione dei linguaggi e ambienti software ha influito in modo determinante. In origine le *unità di governo* erano elaboratori dedicati con linguaggi di programmazione propri di bassissimo livello. Ciò era conseguenza soprattutto delle limitatezze

delle CPU usate. Il Personal Computer IBM del 1981 ha scatenato un processo evolutivo inarrestabile di miglioramento di prestazioni che si è riflesso anche sulle architetture di controllo. Maggior potenza di calcolo a basso costo significa flessibilità nel software e crollo delle barriere fra ambienti di programmazione diversi, interoperabilità, connessioni in rete. Anche se ancora molte difficoltà esistono per ragioni di politiche commerciali, oggi costa 10 volte meno programmare, riprogrammare, integrare il robot nell'architettura della fabbrica automatica. Come per la cinematica dei bracci robotizzati è stato fondamentale lo studio sistematico alla fine degli anni '60 (il noto *teorema di Pieper* è del 1968), per gli AGV è stato importante il formarsi di una teoria unificata del problema della *navigazione*, grazie a J. Borenstein (1995), con la conseguenza di sviluppare metodi algoritmici standard (un caso di tecnologia *soft*). Già dalla fine degli anni '80, si era diffuso il dispositivo tecnologico commerciale che ha permesso la diffusione degli AGV: la "testa *laser* rotante" che serve al calcolo della posizione mediante tecniche di triangolazione o di filtraggio adattativo. I robot mobili devono molto al famoso dott. Land della *Polaroid*, la cui rivoluzionaria macchina istantanea del 1978 possedeva un economico misuratore di distanze a ultrasuoni. Questo componente è lo standard da quindici anni per i sensori di prossimità a basso costo usati in tutti i robot da laboratorio e in non molte ancora applicazioni industriali, a causa della mancanza di sicurezza intrinseca. Un sensore che ha aggiunto, invece, le indispen-



sabili caratteristiche di sicurezza alla capacità di ricostruire l'ambiente circostante è l'altrettanto famoso *laser scanner* usato in molti AGV come in alcuni dei primi robot mobili per impieghi civili, diffuso alla fine degli anni '90. È inutile dilungarsi sugli effetti dell'evoluzione dell'elettronica e, in particolare, quella digitale: quale sia il progresso dall'elettronica di controllo di fine anni '50 a quella di oggi è conoscenza comune.

La visione artificiale è terreno tipico dell'IA, e di questa condivide la sorte di "componente di Formula 1"; è un campo che ha fornito immensi risultati scientifici a partire dagli anni '70 e pochi specifici sottoprodotti adatti al robot in fabbrica. Questi sono limitati sostanzialmente a misure bidimensionali rese accessibili dalle economiche tecnologie di *imaging* (telecamere, ottiche, schede di acquisizione e conversione) che provengono dal mercato di consumo. Ma occorre prestare molta attenzione alla visione artificiale disponibile al campo robotico (cioè di velocità tale da operare in tempo reale), e non bisogna commettere l'errore di ritenere lo stato dell'arte come adatto ad applicazioni soltanto di laboratorio. L'assenza *de facto* di applicazioni al campo della Robotica Industriale non sembrerebbe giustificata, vista la capacità delle tecniche consolidate di Visione Artificiale di risolvere problemi sensoriali comuni, come rilevare la posizione di un oggetto o individuare un ostacolo da evitare, o anche di localizzare con precisione un robot mobile mediante immagini dell'ambiente. Il vero problema è che le tecniche di visione, per loro natura, non permettono la quantificazione di specifiche di funzionamento certe. Per un robot la cui produttività dipende dalla ripetizione veloce di cicli di operazioni in ambienti stabili questo è un inconveniente insormontabile, specialmente se intervengono ragioni di sicurezza. In future applicazioni in cui questi vincoli siano rilasciati o invertiti come, per esempio, in un robot mobile in ambienti parzialmente ignoti in cui non il fattore velocità, ma la capacità di incontrare ostacoli sconosciuti è decisiva, lo stato dell'arte sulla visione artificiale si dimostrerà notevolmente maturo (Figura 5).



Un ultimo cenno va ai materiali. In quanto "corpo", contano almeno quanto la "mente", anche se molti ricercatori tendono a sottovalutarli. Molti dei miglioramenti dei robot attuali citati nel paragrafo "IT e Robotica Industriale", rispetto alle origini, deriva dal progresso nei materiali: struttura, meccanica, attuatori. Questi ultimi, ora sempre più basati su motori *brushless* con magneti ceramici ad altissima forza coercitiva, hanno migliorato la densità di potenza e le caratteristiche dinamiche determinando prestazioni e precisioni più elevate. Il miglioramento degli attuatori ha avuto, probabilmente, l'influsso più importante. Nuovi materiali hanno fornito sensori stabili e sicuri, in particolare quelli di contatto e di forza. Ma anche in questo campo ci sono risultati scientifici disponibili a tramutarsi in tecnologie solo per i nuovi settori di applicazione: sensori di contatto capaci di fornire "immagini tattili", materiali strutturali non metallici, per citare due risultati maturi, e, in futuro, biomateriali e nanomateriali che potrebbero alla fine far passare la tecnologia dei motori dall'attuale ferro-rame a una chimica (muscoli artificiali), con nuovi migliori rapporti potenza/peso.

Infine, è da dire che il robot è comunemente visto come "un calcolatore con sensori e motori": tutta l'evoluzione della tecnologia del *Personal Computer* (PC) si ribalta gratuitamente su di lui, dalle LAN (*Local Area Network*) *wireless* alla grafica, dal VRML (*Virtual Reality Modeling Language*) ai PC *disless*, dalla comunicazione attraverso ap-

#### FIGURA 5

*Frogbot per esplorazione di pianeti, comete e asteroidi. Pesa 1,3 kg, ha un solo motore e una molla, ed è in grado di muoversi a balzi*

plet Java alla rivista settimanale con gadget in edicola, e così via. E questo è parte di un piacevole circolo virtuoso appena iniziato.

## 11. L'ALTRA ROBOTICA

In più punti di questo articolo si è rimandato il lettore alle nuove applicazioni, ai nuovi mercati. Per parlarne è possibile basarci su un documento simile all'importante studio JIRA del 1980 citato in precedenza. Quello studio riguardava l'industria robotica giapponese e le sue prospettive; questo, a ventuno anni di distanza, e con gli stessi scopi, si rivolge all'industria robotica europea (*World 2001 Robotics* [6]), prodotto dalla Commissione Economica per l'Europa delle nazioni Unite (UNECE) e dalla IFR (*International Federation of Robotics*). Si tratta di un'autorevole pubblicazione periodica, di cui è disponibile anche l'edizione 2002 [7]. Si ritiene più esemplificativa la prima: pubblicata poco prima del tragico 11 settembre, presenta una più serena (e ottimistica) previsione degli sviluppi futuri, con un'analisi estesa dei nuovi mercati che non viene ripetuta ogni anno. Si citeranno poi i correttivi che sono da apportare a quei dati.

**FIGURA 6**  
Nursebot "Flo":  
progetto di robot  
assistente  
personale per  
anziani della  
Carnegie-Mellon  
University,  
Pittsburg, USA

Il dato più significativo è in qualche modo la definizione "ufficiale" di un nuovo mercato per la Robotica: la Robotica di Servizio (*Service Robotics*). Per questo nuovo tipo di robot viene proposta una definizione molto diversa da quella classica ISO: "Un robot che opera in modo autonomo o semi-autonomo per compiere servizi utili al benessere (*well*

*being*) di esseri umani (Figura 6) o di apparecchiature, con l'esclusione di operazioni di manifattura".

Vengono anche classificate una serie di attività che ricordano l'avveniristico elenco del JIRA riportato più sopra. È interessante analizzare le differenze fra i due elenchi: quello del 1980, in realtà, riguarda in gran parte attività lavorative industriali in sostituzione di mano d'opera – industria agroalimentare, miniere, costruzioni eccetera. Le prospettive dello studio IFR-UN/ECE sono molto più concentrate su impieghi "civili", in cui un robot interagisce con gli umani pur sostituendo compiti umani. Si tratta di una differenza non tecnologica, ma sociale ed economica. Non vi è dubbio che le stesse tecnologie, o tecnologie molto simili, potrebbero adattarsi ai campi d'applicazione diversi. È verosimile che questa differenza di prospettiva sia dovuta alle mutate disponibilità di forza lavoro, già citate in precedenza.

Fra le circa 35 categorie citate, alcune mostrano un notevole incremento previsionale di installazioni in Europa nei quattro anni successivi allo studio. Se ne citano alcune: robot per pulizia professionale (da 440 a 14.150); robot per chirurgia (da 1.600 a 4.800); di piccolo trasporto in edifici (da 60 a 140) (Figura 7); per rifornimento di carburante (da 50 a 1.100); in agricoltura e allevamenti (da 1.100 a 2.300); per sorveglianza e sicurezza (da 60 a 1.800); per pulizia domestica o tagliaerba (da 12.500 a 425.000); per compagnia e divertimento (da 100.000 a 200.000). Alcuni, come la sorveglianza e il rifornimento di carburante, prevedono un aumento percentuale notevole (200 - 300%). Sorprendentemente, applicazioni popolari in laboratori di ricerca non sono premiate da questo studio (robot nei musei: da cinque a 10; robot mobili per impieghi generali da 260 a 230; robot da laboratorio da 1.000 a 320; robot per trasporto di disabili: da 260 a 230). Le ragioni dipendono da accurati studi di mercato e devono far riflettere.

Anche i settori degli "ambienti estremi", come spazio e mare (*underwater Robotics*) (Figura 8 e 9) apparentemente non sono premiati dallo studio, nonostante l'interesse scientifico, la presenza di investimenti consi-



**FIGURA 7**

*Robot mobile di servizio staffetta per trasporto in ospedali (DIST e Genova Robot)*

stenti da parte delle Agenzie Spaziali e di applicazioni marine già eccellenti. In realtà, questo tipo di applicazioni, destinate per ora a poche importanti applicazioni, sfugge a studi di mercato basati sui numeri; la loro rilevanza però va considerata.

È davvero sconcertante il non trovare esplicitamente in questi studi applicazioni di sminamento umanitario (*de-mining*). Si spera che siano nascosti nelle voci *other type* (anche se si nutrono forti dubbi al riguardo), ma è da considerare che – non per colpa della Robotica, ma della diabolica semplicità con cui le mine anti-uomo sono costruite – questo resta, purtroppo, ancora un tema di ricerca.

Non v'è dubbio che l'apertura di questi mercati segnerà un cambiamento generazionale in Robotica. Ed è da notare che questi nuovi mercati si affiancheranno, senza interferire, alla Robotica Industriale. Verranno finalmente utilizzate tecnologie mature prodotte dai vari settori dell'IA, della visione artificiale, da quelle dei materiali, con una an-

**FIGURA 8**

*Robot mobile Sojourner basato su ruote che ha partecipato all'esplorazione di Marte (NASA)*

**FIGURA 9**

*Robotica subacquea: manipolazione mediante bracci in grado di operare ad elevate profondità; esplorazione dei fondali (DIST – Ansaldo – Movita)*

cora più profonda integrazione con le tecnologie IT e, in particolare, con quelle civili (*wireless*, Internet).

## 12. SMART HOUSING

Fra le tecnologie che sono state denominate “civili” e di cui ci si attende l’integrazione con i robot di servizio, un posto speciale spetta a quelle cosiddette di “automazione di edificio”, chiamata anche degli “edifici” o “case intelligenti” (*smart housing*). Le case intelligenti hanno un’infrastruttura di comunicazione a cui sono interfacciati i normali dispositivi di un’abitazione: luci, porte, finestre, condizionamento ambientale, elettrodomestici, dispositivi per disabili o anziani, apparecchi di comunicazione e così via. Gli scopi possono essere quello di realizzare alloggi protetti o semplicemente case automatizzate per migliorare la qualità della vita. In un prossimo futuro, elettrodomestici e cibi dialogheranno grazie a un’elettronica integrata per migliorare l’approvvigionamento domestico e automatizzare la preparazione. Ma l’automazione degli edifici è già comunemente diffusa, con compiti più tecnici, in grossi complessi edilizi in cui ascensori, impianti di allarme, aria, luci, servizi vari sono collegati via rete a sistemi informatici di controllo. Le tecnologie già diffuse e disponibili sono quelle di particolari bus di campo (*fieldbus*), o reti locali per automazione, simili a quelli usati in automazione di fabbrica, ma specializzati in questo tipo di applicazioni. In diversi centri di ricerca si studia l’integrazione di questa tecnologia disponibile con i robot di servizio. È chiaro, infatti, che se un robot agisce in un ambiente civile (un edificio commerciale, una casa d’abitazione), esso deve interagire con una quantità di installazioni e dispositivi – porte, ascensori, dispositivi di emergenza o allarme – e deve comunicare con qualche posto di controllo remoto. In più, per ragioni di sicurezza o sorveglianza, deve poter conoscere o influire sulle politiche con cui l’infrastruttura intelligente dell’edificio affronta un determinato problema – per esempio, un incendio. È innovativo in questo caso pensare al robot non come un’entità totalmente autonoma, ma come parte di un sistema distribuito intelligente (la rete di automazione dell’edificio) di cui fa parte e con cui è in comunicazione continua o periodica. Questo vale anche per i livelli meno complessi e co-

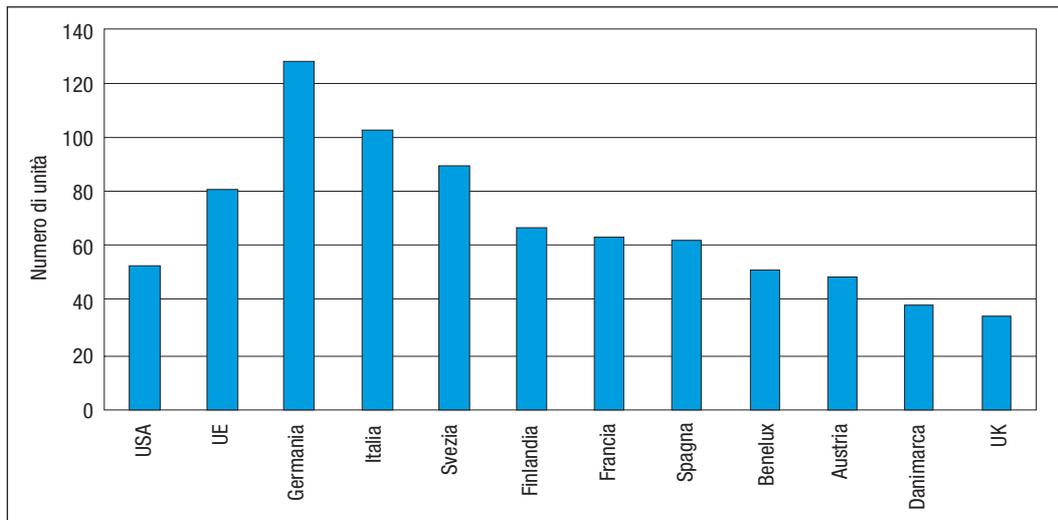
stosi, come un aspirapolvere robotizzato in una smart house. Si ritiene ragionevole che questo approccio, studiato in Europa da diversi centri di ricerca in Robotica, e che ben si sposa con recenti tecnologie software ad agenti e per sistemi embedded, possa godere di fortuna e acceleri l’ampliamento del mercato di una parte significativa dei robot di servizio.

## 13. DOPO IL 2001

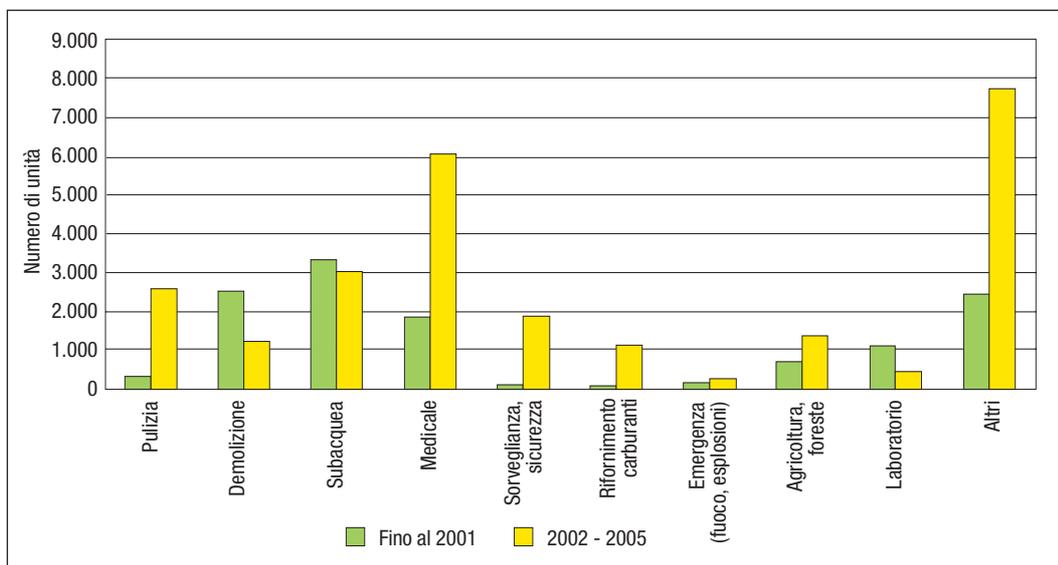
Il *report* IFR–UN/ECE del 2002 [7] non corregge in negativo gli scenari del 2001 (Figura 10), anche se gli obiettivi verranno spostati in avanti dalla crisi dei mercati internazionali. In particolare, viene notato che, a fronte di una diminuzione degli investimenti media del -3% mondiale in Robotica, di cui il 17% negli USA, l’Europa ha visto, invece, un record positivo di +2.5% (con punte di oltre il 25% in UK e Spagna). Dal punto di vista delle vendite, i mercati industriali USA e Giappone hanno avuto una forte flessione mentre l’Europa ha continuato a crescere, seppure di una piccola percentuale. Le previsioni a quattro-cinque anni sono ora diminuite in modo non rilevante, almeno per l’Europa.

Per quanto riguarda la Robotica di Servizio, ci sono solo poche variazioni significative nella previsione del 2005 (un anno dopo quella dello studio precedente): una diminuzione della pulizia industriale, un notevole aumento dei robot chirurgici, una ulteriore diminuzione del trasporto disabili, e una vera e propria esplosione dei robot da compagnia e divertimento (da 200.000 a 1.200.000, si veda a tal proposito il grafico in Figura 11).

Un’attenzione particolare spetta proprio a questi ultimi, di cui AIBO della Sony è stato il primo esemplare a larga diffusione. È curioso pensare a questo settore come quello che ha sostituito in Giappone, alla fine degli anni ’90, i settori previsti dal rapporto del 1980; e questa volta con lo sguardo ai mercati internazionali e a Internet, che ne ha decretato il successo. Parafrasando le parole del suo vicepresidente, Toshidata Doy, la Sony afferma che dal 1990 al 2000 il *personal computer* e Internet hanno dominato i mercati; dal 2000 al 2010 questo compito sarà svolto dai robot autonomi personali. Vero o no, l’inatteso ani-



**FIGURA 10**  
 Numero di robot per ogni 10.000 lavoratori impiegati nell'industria manifatturiera nel 2001: (Fonte: World Robotics 2002)



**FIGURA 11**  
 Mercato europeo della Robotica di Servizio: installazioni al 2001 e previsioni 2002-2005: (Fonte: World Robotics 2002)

maletto con 18 gradi di libertà e con una semplice simulazione di vita affettiva ha avuto un successo straordinario. Lo scenario futuro dei robot "gioiosamente inutili" come questo è previsto roseo. L'invecchiamento della popolazione, la vita solitaria, la diffusione della moda tecnologica grazie a Internet sembrano forti elementi a favore. È possibile che questo tipo di Robotica, con il suo obiettivo di simulare la vita, giochi un ruolo di sfida simile a quello dell'IA classica nei confronti della Robotica Industriale. Un comportamento simil-vivo è, necessariamente, un comportamento complesso e parzialmente imprevedibile, quale può emergere solo da una struttura teorica e tecnologica complessa, molto più complessa di quella pre-

programmata attuale. Che una simile complessità serva a impieghi ripetitivi ancorché autonomi (come verificare che nessun intruso sia in un edificio o raccogliere le mele) è dubbio; alcuni ricercatori sostengono sia indispensabile per comunicare *emozioni* e studiano modelli mutuati dalle neuroscienze o ispirati a una teoria della *consciousness* artificiale. L'interesse teoretico è formidabile, perché re-imposta lo studio dell'*artificial brain* e dell'*artificial mind*. Personalmente, chi scrive non sarebbe sorpreso se obiettivi *visionary* di questo genere di ricerca in Robotica fossero supportati da fiorenti mercati di animaletti artificiali per compagnia (anche se spera che i bambini umani non trascurino per loro i cani e i gatti biologici).

#### 14. ESISTE UN DIGITAL DIVIDE IN ROBOTICA

È forse opportuno terminare questo (sicuramente lacunoso) discorso attraverso la Robotica con un allarme. Il *digital divide*, il gap culturale tipico dell'ICT [Mondo Digitale, n. 2 giugno 2002], esiste già in Robotica, e potrebbe allargarsi con la nascita della Robotica di Servizio. Le motivazioni sono quelle generali, aggravate dalle competenze "di nicchia" dell'industria robotica, che la formazione istituzionale nazionale (secondaria, universitaria) potrebbe non fornire agli stessi ritmi della crescita del mercato (il solo mercato interno della Robotica Industriale classica è, da anni, il secondo in Europa, e le previsioni a cinque anni confermano questa posizione). Questo vale, in particolare, per le attività di R&S che in Italia dovrebbero essere particolarmente significative. Occorre tenere presente che, rispetto ad altri settori tecnologici, la Robotica richiede da 1 a 2 volte l'aggiornamento professionale degli addetti. Inoltre, il cambiamento del ruolo del robot in fabbrica verso un nuovo modello integrato B2B richiede l'incrocio di competenze IT finora diverse. Infine, la tradizione delle PMI *hi-tech* italiane richiede nuovi modelli di scambio e aggregazione, diversi da quelle in cui, in distretti industriali tradizionali, esse operavano tradizionalmente nell'indotto di una grande impresa di riferimento. Come è noto, la competitività riferita alla presenza nelle imprese di *highly skilled people* è molto bassa in Italia, se paragonata ad altri Paesi europei. Questo fatto, unito alla necessità "endemica" di riqualificare lavoratori usciti da posizioni *low-tech*, porrà seri problemi all'Italia se non saranno potenziati i meccanismi per colmare il digital divide anche in questo settore. Le soluzioni sono note: formazione permanente (con l'intervento mirato delle Regioni), raccordo fra Università e mondo del lavoro e sostegno a Distretti Tecnologi-

ci "virtuali" in cui aziende ed Enti fanno sistema per migliorare innanzi tutto l'aggiornamento del loro capitale umano. Da pochi mesi, in Liguria, una ventina di PMI (fra cui alcune *spin-off* universitarie) si sono associate con la Camera di Commercio, l'Associazione Industriali, l'Università di Genova, il Parco Scientifico e Tecnologico della Liguria, e l'associazione DIXET (che raggruppa a sua volta più di 100 PMI nel settore elettronico) dando vita all'Associazione Polo della Robotica ([www.polorobotica.it](http://www.polorobotica.it)). Questa associazione si pone come obiettivo di far crescere, dal tessuto industriale ligure a tradizione di medie e grandi imprese, un'attività verso la Robotica a tutto campo e ad alto contenuto di ricerca e innovazione. Si tratta della prima esperienza italiana di questo tipo, ed è una sfida che chi scrive e i suoi colleghi sono stati pronti a raccogliere – per ingannare il tempo *aspettando robot*.

#### Bibliografia

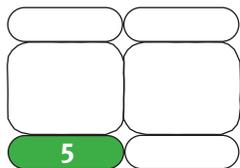
- [1] Gini G, Caglioti V: *Robotica*. Bologna, Zanichelli, 2003.
- [2] Losano G: *Storie di automi*. Torino, Einaudi, 1990.
- [3] Jacobelli J, (a cura di): *Aspettando Robot. Il futuro prossimo dell'Intelligenza Artificiale*. Bari, Laterza, 1987.
- [4] Russel S, Norvig P: *Intelligenza artificiale. Un approccio moderno*. Torino, UTET, 1998.
- [5] Tagliascio V: *Dizionario degli esseri umani fantastici ed artificiali*. Mondadori, 1999.
- [6] United Nations Economic Commission for Europe (UNECE): *World Robotics 2001 – Statistics, Market Analysis, Forecasts, Case Studies and Profitability of Robot Investment*. Ginevra, UNECE, 2001.
- [7] United Nations Economic Commission for Europe (UNECE): *World Robotics 2002 – Statistics, Market Analysis, Forecasts, Case Studies and Profitability of Robot Investment*. Ginevra, UNECE, 2002.

RENATO ZACCARIA Professore Straordinario presso l'Università di Genova. Insegna Robotica e Sistemi Operativi nella Facoltà di Ingegneria, e Informatica presso Facoltà umanistiche. Dirige un laboratorio di ricerca in Robotica. Si occupa di Robotica, Intelligenza Artificiale e Domotica in scenari di servizio, entertainment, sorveglianza e sicurezza.  
[renato.zaccaria@unige.it](mailto:renato.zaccaria@unige.it)



# FORMAZIONE INFORMATICA E PRODUTTIVITÀ

PierFranco Camussone



Nel corso dello studio sul costo dell'ignoranza informatica [3], sono emersi due importanti quesiti. Dal momento che tale costo è elevato e contribuisce a diminuire la produttività delle persone, si può puntare sulla formazione per ridurre il tempo non produttivo degli utenti? E inoltre, i corsi per il conseguimento della ECDL possono rappresentare lo strumento formativo adatto a ottenere tale scopo? Per rispondere a questi interrogativi è stata avviata una indagine empirica i cui risultati sono esposti in questo articolo.

## 1. INTRODUZIONE

**S**e la formazione sia un costo (e, quindi, un onere da ridurre), oppure un investimento che può dare i suoi ritorni, è un quesito su cui si è discusso a lungo in vari contesti [1, 2, 4]. Molto è stato scritto sul ritorno per l'azienda degli investimenti in formazione e, in particolare, sul ROI (*Return of Investment*) della formazione informatica degli utenti. Spesso, i dati non hanno confortato le attese. Però, dal momento che è stato rilevato [3] un costo considerevole derivante dall'ignoranza informatica, che si aggira sui 2000 € l'anno per ogni utente generico, il gruppo di ricerca della SDA Bocconi e dell'AICA ha ipotizzato che un intervento di formazione possa ridurre tale onere e ha cercato di misurare i ritorni della formazione.

L'analisi degli effetti della formazione sull'uso degli strumenti informatici è stata effettuata mediante misurazioni su un campione di soggetti che hanno frequentato i corsi ECDL (*European Computer Driving Licence*). I test hanno riguardato due tipologie di persone:

■ studenti dell'Università Bocconi (facoltà di economia) iscritti al 3° e 4° anno del corso di laurea che hanno frequentato i corsi per il conseguimento della patente ECDL (175 persone);

■ dipendenti di medie o grandi imprese che hanno seguito, anch'essi, il medesimo percorso formativo (27 persone).

Come è noto la patente ECDL riguarda la capacità di utilizzare in modo appropriato il PC come strumento di lavoro individuale. Si è trattato, quindi, di misurare i progressi dei partecipanti ai corsi nella abilità di lavoro nei seguenti ambienti:

■ funzionalità di base del computer (Sistema Operativo e *Utility*);

■ *Word Processor*;

■ foglio di lavoro elettronico (*Spreadsheet*);

■ strumenti di *Effective Presentation*;

■ *Internet ed e-mail*.

Un medesimo test, costituito da 37 domande riguardanti la soluzione di problemi pratici nei 5 ambienti, precedentemente citati, è stato affrontato dai partecipanti prima e dopo il corso. Per ogni soggetto e per ogni



ambiente sono state misurate due variabili:

1. il punteggio conseguito nel test, che misura il livello di padronanza della tematica;

2. il tempo impiegato nel test.

Su quest'ultima variabile occorre, però, fare una precisazione: non si tratta del tempo impiegato a "risolvere" il problema, ma del tempo che il partecipante ha "passato" sul problema, qualunque sia stato il punteggio conseguito.

## 2. FORMAZIONE E LIVELLO DI CONOSCENZE

La situazione del livello di conoscenze iniziali del campione è illustrata in figura 1. Da essa appare che prima del corso circa un terzo dei partecipanti aveva conoscenze inferiori al 50% di quanto previsto come massimo dai test. Ma gli altri due terzi avevano già una discreta conoscenza degli strumenti informatici. Ciò può essere spiegato dalla diffusione della cultura informatica tra gli studenti derivante dall'aver frequentato corsi di informatica di base negli istituti superiori, o nei primi anni di università, mentre per i dipendenti può essersi verificato un apprendimento di tipo autodidattico, fatto in azienda con l'aiuto di colleghi.

La ricerca si è, quindi, svolta su un campione di persone non totalmente "ignoranti" per quanto riguardava le tecnologie informatiche, come è naturale che sia, essendo l'Italia ormai entrata anch'essa nella cosiddetta società dell'informazione assieme agli altri Paesi più sviluppati.

In ogni caso, alla fine del corso il punteggio conseguito nei medesimi test è risultato decisamente incrementato (Figura 2), a riprova dell'utilità del corso dal punto di vista dell'apprendimento e, quindi, del miglioramento delle competenze, soprattutto sotto il profilo del *problem solving*. La grande maggioranza dei partecipanti ha avuto un miglioramento compreso tra il 40 e il 200% rispetto al punteggio iniziale, calcolato come segue:

$$\text{Variazione} = \frac{\text{punteggio finale} - \text{punteggio iniziale}}{\text{punteggio iniziale}}$$

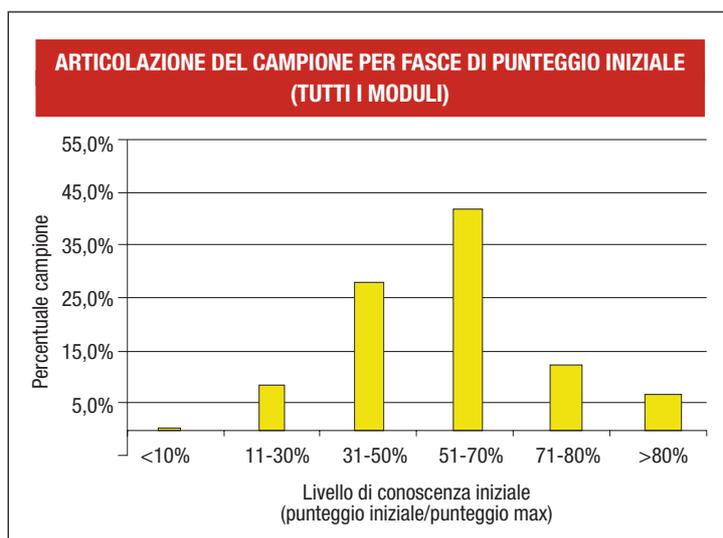


FIGURA 1

Situazione iniziale delle conoscenze del campione

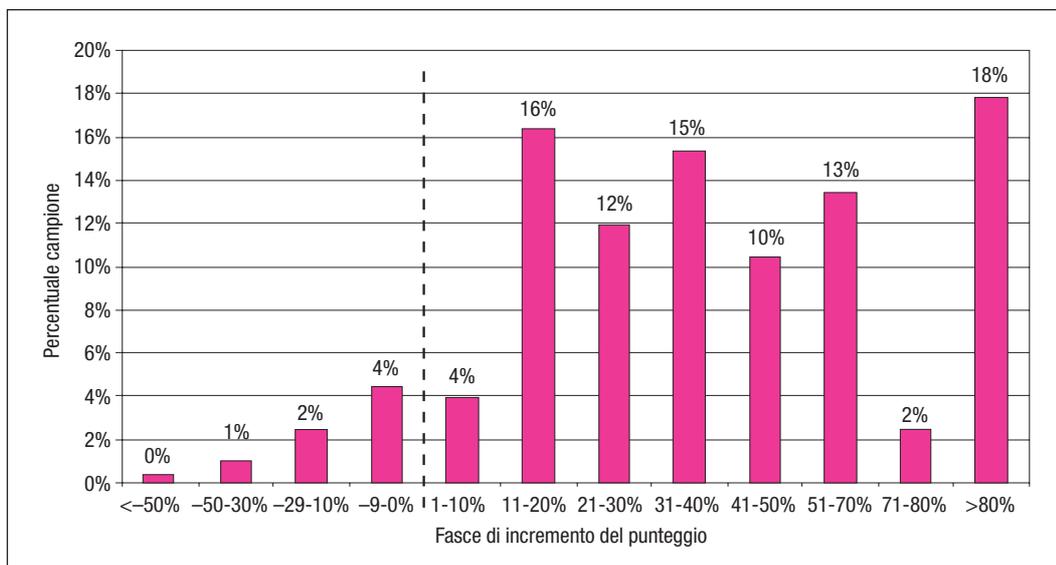
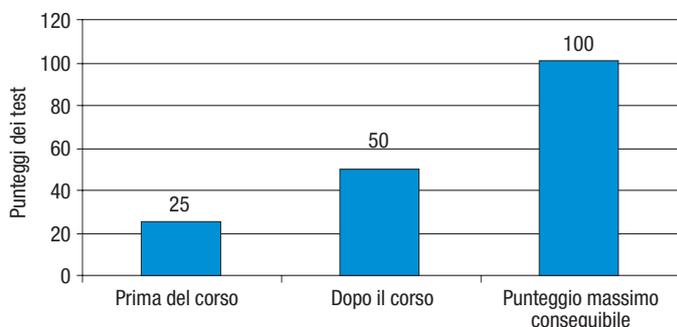


FIGURA 2

Miglioramento nel punteggio a fine corso

### LIVELLO DI CONOSCENZE MISURATO DAI TEST



#### Modalità di misura:

Variatione relativa delle conoscenze, cioè rispetto al livello iniziale	$(50 - 25)/25 = 100\%$
Variatione assoluta delle conoscenze, cioè rispetto al livello massimo	$(50 - 25)/100 = 25\%$

**FIGURA 3**  
*Misurazione del miglioramento delle conoscenze relativo e assoluto*

Naturalmente coloro che inizialmente partivano da una base modesta di conoscenze sono quelli che hanno conseguito gli incrementi percentualmente superiori. Ciò è confermato anche dall'indice di correlazione calcolato tra *punteggio iniziale e incremento del punteggio ottenuto dopo il corso*.

Il valore di tale indice è risultato complessivamente pari a  $-0,698$ , valore che indica una significativa correlazione inversa tra livello di partenza e miglioramento conseguito (chi parte da livelli bassi migliora di più).

Ridurre ogni considerazione all'incremento del punteggio iniziale è sembrato, tuttavia, troppo semplicistico al gruppo dei ricercatori. Dal momento che questo valore risente troppo del livello di partenza (ad esempio, un partecipante con punteggio iniziale pari a zero presenterebbe una variazione infinita dopo il corso), si è deciso di calcolare l'incremento di conoscenze anche in un altro modo.

Per chiarire questo aspetto si veda il caso illustrato in figura 3; in questo esempio l'incremento di conoscenze misurato rispetto al livello di partenza indicherebbe una crescita del 100%; mentre lo stesso incremento rapportato alla scala assoluta delle conoscenze, cioè riferito al massimo del punteggio, indicherebbe un miglioramento del 25%.

Si è, quindi, distinto l'*incremento relativo di conoscenze* (ovvero, a partire dal livello di conoscenze iniziali) rispetto all'*incremento*

*assoluto* (cioè rispetto al massimo delle conoscenze acquisibili). Sotto quest'ultimo aspetto, l'incremento massimo conseguibile non è più infinito, ma può arrivare al massimo al 100%.

In figura 4, sono rappresentati i risultati della ricerca, che provano comunque un incremento medio delle conoscenze assolute di circa il 20%, con una significativa concentrazione su valori superiori al 50% (oltre la metà dei partecipanti). Ciò significa che i corsi consentono di migliorare le competenze, qualunque sia il livello di partenza; come si vede bene dalla figura 5, in cui sono presentati i miglioramenti relativi e assoluti dei punteggi per fascia di conoscenze iniziali.

Come si può vedere, le fasce di individui con competenze iniziali elementari sono quelle che hanno presentato gli incrementi maggiori del livello relativo di conoscenze. Ma sono gli individui con conoscenze intermedie (punteggi iniziali compresi tra il 30 e il 70% rispetto al massimo previsto) che riescono ad approfittare meglio dell'opportunità di formazione, completando la propria preparazione (variazioni più significative rispetto al punteggio massimo; miglioramento delle conoscenze in assoluto).

Sembra, quindi, di poter desumere che i corsi possano avere una duplice valenza:

**a. alfabetizzazione di massa**, indicata dall'incremento del punteggio iniziale, per coloro che sono quasi "digiuni" dell'argomento e che, quindi, non possono aspettarsi dopo il corso di avvicinarsi ai valori massimi conseguibili nel test;

**b. affinamento e consolidamento delle conoscenze** per coloro che già conoscono la materia (punteggi iniziali più elevati) e che possono aspirare a raggiungere i punteggi massimi dei test.

Volendo approfondire l'analisi con riferimento ai diversi moduli del corso, si può riscontrare un esito significativamente differente del processo di apprendimento in relazione ai differenti ambienti funzionali oggetto di formazione. *In primis*, i livelli di conoscenze iniziali sono risultati diversi in relazione ai diversi ambienti: ad esempio, le funzioni del sistema operativo Windows sono più conosciute rispetto a quelle di uno strumento di

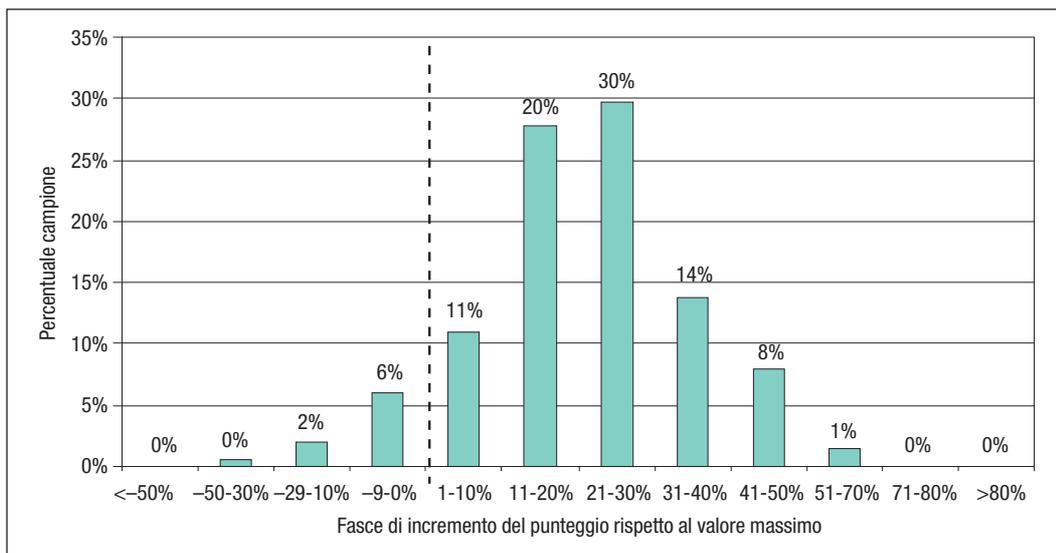


FIGURA 4

Miglioramento del livello assoluto delle conoscenze

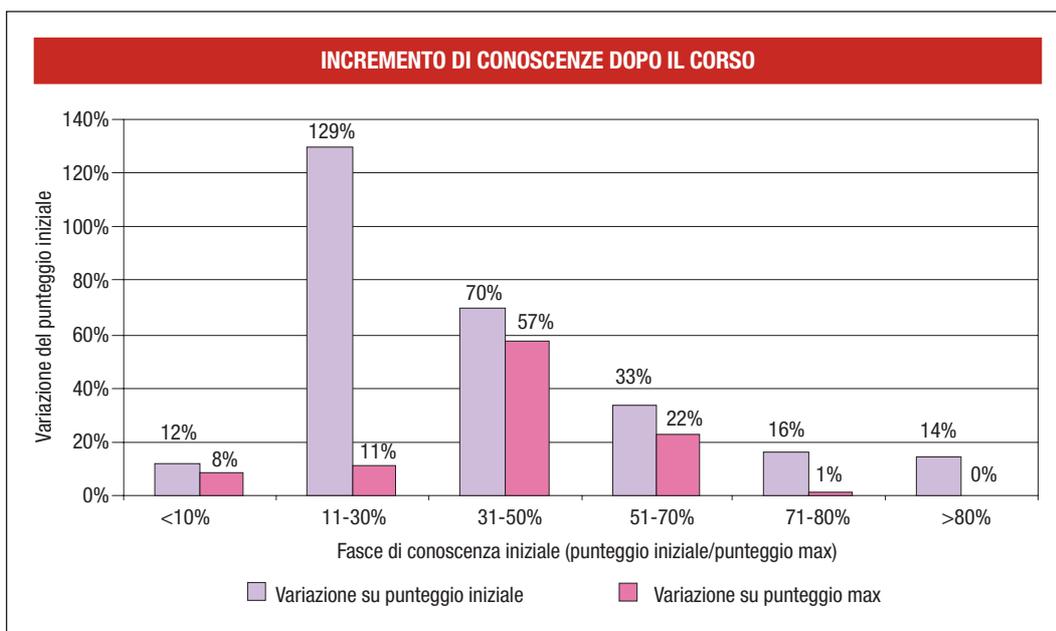


FIGURA 5

Miglioramento nel punteggio per fasce di conoscenze iniziali

Effective Presentation dai partecipanti ai corsi, che già dispongono di oltre il 60% delle conoscenze al riguardo (Figura 6). Una situazione analoga, anche se un po' meno soddisfacente, si riscontra per la posta elettronica e il *browser* di Internet.

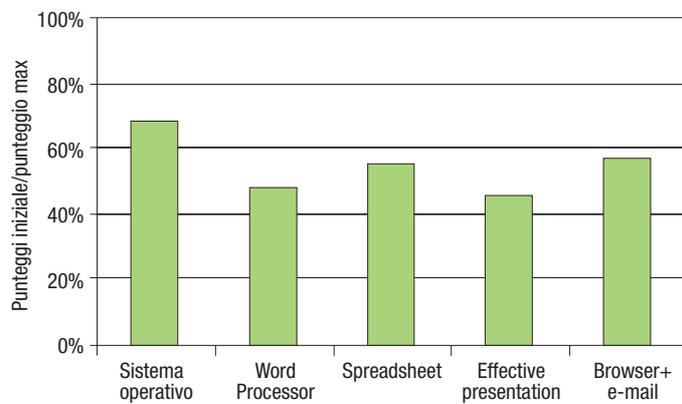
Meno note sono, invece, risultate le funzionalità del Word Processor. La sorpresa è rappresentata proprio dal *software* di preparazione dei testi; benché risultasse già usato dai partecipanti, le sue funzionalità più avanzate non erano note agli utenti, che si limitavano a usare le sue caratteristiche basilari, non riuscendo spesso a risolvere in

modo soddisfacente problemi abbastanza comuni.

I corsi hanno dimostrato un'efficacia differente in relazione ai vari moduli. Come si può vedere in figura 7, i miglioramenti maggiori sui punteggi iniziali e sui punteggi massimi conseguibili si sono avuti nel caso dello Spreadsheet, che notoriamente ha contenuti concettuali superiori agli altri moduli (rispettivamente +86% e +29%).

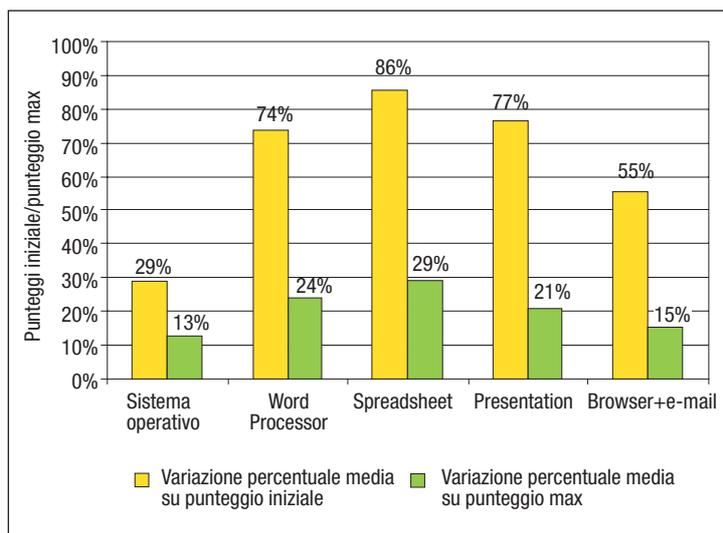
In seconda posizione si è piazzato il Word Processor con un progresso medio del 74% rispetto alle conoscenze iniziali e del 24% rispetto ai valori massimi conseguibili. Lo stru-

### CONOSCENZA INIZIALE PER SINGOLO MODULO



**FIGURA 6**

La situazione delle conoscenze iniziali per i diversi moduli



**FIGURA 7**

L'incremento delle conoscenze iniziali per i diversi moduli

Moduli applicativi	Coefficienti di correlazione tra punteggio iniziale e variazione % punteggio
Sistema Operativo (Windows)	-0,67
Word Processor	-0,65
Spreadsheet	-0,75
Effective Presentation tool	-0,56
Browser ed e-mail	-0,59

**TABELLA 1**

Coefficienti di correlazione tra punteggio iniziale e variazione percentuale media del punteggio iniziale

mento di Effective Presentation è circa nella medesima situazione.

Il sistema operativo, più conosciuto in partenza, è ovviamente il modulo che ha fatto registrare minori progressi assoluti, relativi cioè ai valori delle conoscenze massime.

Anche nel caso dei singoli moduli si è riscontrato che l'incremento di conoscenze si verificava, in genere, per tutti i partecipanti, ma non nella stessa misura. Sono stati, pertanto, calcolati gli indici di correlazione per ciascun ambiente applicativo tra il punteggio di partenza di ogni partecipante e il miglioramento ottenuto. Il valore più elevato dell'indice è stato riscontrato per l'ambiente del foglio di lavoro elettronico, a indicare che in tal caso le nozioni impartite sono risultate particolarmente efficaci (Tabella 1).

### 3. FORMAZIONE E TEMPI DI SVOLGIMENTO DEL LAVORO

Semberebbe logico aspettarsi che coloro che hanno raggiunto un livello più elevato di conoscenza degli strumenti informatici impieghino meno tempo per svolgere un lavoro che richiede l'uso degli strumenti medesimi. I risultati della ricerca hanno, invece, fornito indicazioni contrastanti (Figura 8): le persone che partivano da livelli di conoscenza bassi hanno svolto i test, dopo il corso, impiegando addirittura più tempo. Vi è stato un miglioramento tra il 7 e il 12% del tempo impiegato inizialmente per coloro che partivano da livelli di conoscenza intermedi e, infine, le persone più preparate hanno anch'esse allungato i tempi di esecuzione dei test. Quale spiegazione si può ipotizzare al riguardo?

Per prima cosa si deve sottolineare che i test non erano concepiti per premiare coloro che impiegavano meno tempo nell'esecuzione. Trattandosi di prove di esame vi era un tempo limite e gli studenti sono portati abitualmente a usufruire di tutto l'intervallo di tempo loro concesso. Il campione costituito dai dipendenti, invece, ha mostrato un atteggiamento differente rispetto al tempo e tra loro, sono di più coloro che hanno ridotto il tempo di esecuzione dei test.

In ogni caso, se si vuole riflettere sui risultati della ricerca e trovare una spiegazione, si

possono formulare le seguenti considerazioni.

**1.** Il risparmio di tempo nell'esecuzione del lavoro si verifica quando l'interessato ha sviluppato una discreta esperienza nell'uso degli strumenti, non quando ne sta apprendendo l'uso. In questa seconda circostanza, il lavoratore è ancora in fase di "metabolizzazione" delle nuove tecnologie, cerca di usarle bene e di non sbagliare, e sta sviluppando le proprie modalità d'uso che, con l'esperienza, si tradurranno in automatismi di comportamento (quasi dei riflessi condizionati) che lo porteranno ad accelerare molto i processi di uso degli strumenti in questione.

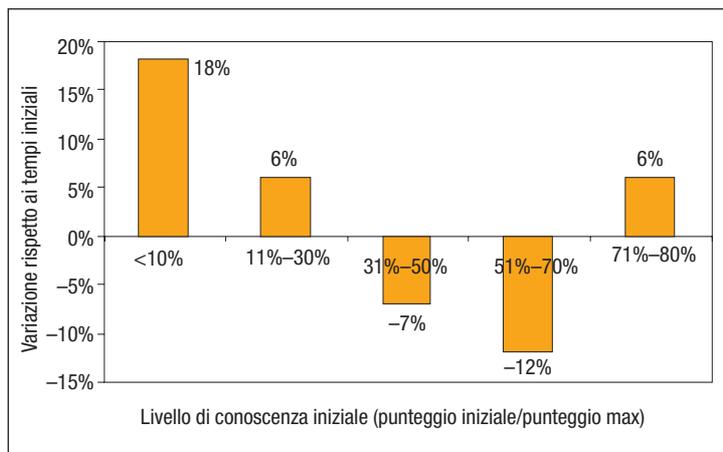
**2.** Coloro che partono da livelli bassi di conoscenza iniziale sono molto attenti all'uso corretto delle tecnologie appena apprese e, quindi, il loro comportamento determina addirittura un allungamento dei tempi di test. Va però osservato che il tempo iniziale era anche influenzato dalla loro ignoranza informatica, per cui abbandonavano velocemente un problema se non sapevano risolverlo. Mentre a fine corso si impegnavano per svolgerlo fino in fondo.

**3.** Chi invece partiva da livelli medi di conoscenza degli strumenti e, quindi, probabilmente aveva anche prima la capacità di risolvere in qualche modo il problema presentato dal test, è riuscito a fine corso a svolgere il compito in un tempo inferiore.

**4.** Per i più esperti, invece, può essersi verificato un fenomeno già noto in letteratura [6]. Con l'aumento delle conoscenze cresce il desiderio di rifinire meglio il lavoro, di completarlo in modo elegante e così via, portando a un allungamento dei tempi di lavoro, cui però si accompagna un risultato finale di qualità superiore.

#### 4. GLI EFFETTI SULLA PRODUTTIVITÀ

Si è già osservato che gli esiti della ricerca hanno fornito un risultato incoraggiante per quanto concerne il livello delle conoscenze, che dopo il corso ECDL sono cresciute di un 47%, se raffrontate con il livello iniziale, e del 20% rispetto a una scala assoluta di conoscenze ottimali.



Ciò significa che:

1. si fanno lavori che precedentemente non si era in grado di svolgere;
2. si fanno meglio i lavori che precedentemente si svolgevano in modo insoddisfacente;
3. dopo un periodo di "metabolizzazione" delle nozioni apprese si dovrebbe impiegare meno tempo a svolgere il lavoro;
4. si dovrebbe ridurre il tempo perso in modo improduttivo davanti al computer per effetto della ignoranza informatica [3].

Queste osservazioni fanno sorgere la curiosità di sapere se, in qualche modo, sia possibile avere un'idea dei ritorni degli investimenti in formazione (ROI della formazione). Pur riconoscendo che alcuni dei vantaggi appena ricordati sono di natura qualitativa e, quindi, difficilmente monetizzabili, si seguirà la strada tracciata da coloro che si sono cimentati in questa impresa [5].

La ricerca ha evidenziato un modesto miglioramento del tempo di esecuzione dei test per coloro che si trovavano a livelli intermedi di conoscenza iniziale (riduzione di tempo compresa tra -7% e -12%). Si è anche osservato che un miglioramento significativo può avvenire solo con la pratica e con l'esperienza, quando sono ormai stati metabolizzati gli insegnamenti del corso. Si può, quindi, stimare cautelativamente, che la riduzione del tempo di esecuzione di un lavoro su computer, tra coloro che non hanno fatto formazione specifica e coloro che hanno seguito i corsi, possa aggirarsi in media attorno al 10%. Questo assunto sembra anche troppo conservativo rispetto a quanto si

**FIGURA 8**

*Variazione nei tempi di svolgimento dei test*

riscontra a regime nella pratica aziendale (divario tra i “non formati” e i “formati”). Tuttavia, i ricercatori non se la sono sentita – in assenza di ulteriori elementi – di andare oltre questa aspettativa.

Nella società dell’informazione i lavoratori di concetto (impiegati) spendono davanti al video del computer il 60% del proprio tempo lavorativo teorico (40 h settimanali) come messo in evidenza dalla ricerca dell’Istituto di Statistica Norvegese [3]. Si tratta,

quindi, di 24 h settimanali su cui può manifestarsi l’effetto di aumento di efficienza (riduzione del 10% del tempo di esecuzione dei lavori).

Sulla base di queste premesse si può effettuare il calcolo del miglioramento della produttività di un soggetto che abbia fatto formazione sugli strumenti informatici, come illustrato nella tabella 2.

Un’altra considerazione che riguarda il ROI della formazione, può essere svolta con riferimento alla possibilità per l’azienda di ridurre i cosiddetti “costi dell’ignoranza”, cioè gli oneri nascosti derivanti dall’impreparazione informatica degli utenti. [3] Anche in questo caso, è possibile partire dai valori calcolati dall’Istituto di Statistica Norvegese (Figura 9): tolte le voci su cui la formazione ECDL non ha effetti (guasti degli strumenti, accesso ai sistemi *legacy*, accesso ai *data base*, *virus* e altro) sulle rimanenti cause di perdita di tempo ovviamente si può pensare a una riduzione del tempo perso.

Per stimare di quanto possano ridursi le voci rimanenti si sono misurati gli incrementi di conoscenza assoluta (cioè la variazione di punteggio nei test rispetto al livello massimo di conoscenze raggiungibili).

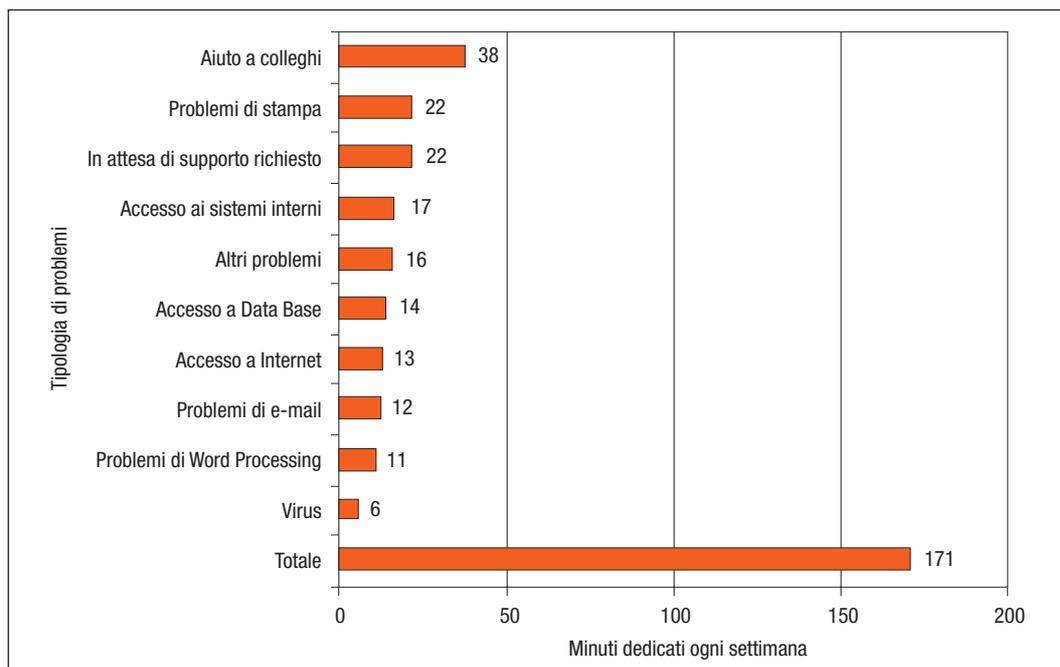
Per esempio nel caso del Word Processor la ricerca ha misurato un incremento di cono-

Tempo speso davanti al computer (ore/settimana davanti al computer)	24
Aumento della produttività (riduzione tempo di esecuzione lavoro)	10%
Ore disponibili per altri compiti	2,4
Miglioramento produttività annuale (ore totali su 48 settimane lavorative)	115,2
Pari a giorni	14,4
Costo giornaliero <sup>1</sup> (Euro)	136,6
Costo annuo (Euro) (miglioramento della produttività annuale)	1.967

<sup>1</sup> Secondo i dati di costo elaborati dal CNEL sulla base statistica dell’ISTAT il costo del lavoro in Italia per addetto nel 2001 ammontava a 136,6 euro al giorno.

**TABELLA 2**

Calcolo del miglioramento della produttività annuale di una persona formata



**FIGURA 9**

Tempo settimanale non produttivo per ogni utente di strumenti informatici

Voci sensibili alla formazione	Tempo settimanale perso	Incremento % conoscenze rispetto al max	Tempo risparmiato
Aiuto a colleghi	38	20%	7,6
Problemi di stampa	22	13%	2,86
In attesa di supporto	22	20%	4,4
Accesso a Internet	13	16%	2,08
Problemi di e-mail	12	16%	1,92
Problemi di WP	11	24%	2,64
minuti/settimana	118	18%	21,5
settimane lavorative all'anno: 48			
ore/anno			17,2
giorni/anno			2,15

**TABELLA 3**

*Riduzione del tempo perso per ignoranza informatica*

scenze assoluto (rispetto al valore massimo) del 24%; ciò potrebbe indurre a pensare che il tempo perso per problemi riguardanti questo ambiente possa subire una riduzione proporzionale.

L'applicazione di questo ragionamento è illustrata nella tabella 3.

Come si può constatare, il risparmio complessivo di tempo perso si aggira sui 21 min settimanali, ovvero su circa 2 giorni l'anno. Utilizzando i dati di costo del lavoro in Italia per addetto nel 2001<sup>1</sup> (136,6 € al giorno) la riduzione di tempo perso equivale a un risparmio di 293,69 € all'anno. Forse il ragionamento è troppo prudente, ma a chi scrive non pare opportuno spingersi oltre in mancanza di dati più precisi al riguardo.

In conclusione, se si sommano i due effetti valorizzati, aumento di produttività e riduzione del costo dell'ignoranza, si ottiene per ciascun dipendente soggetto a formazione del tipo ECDL i valori espressi nella tabella 4.

Anche in questo caso, si può valutare l'impatto che una formazione generalizzata di questo tipo potrebbe avere sul sistema economico italiano nel suo complesso.

Si tratta di estendere il ragionamento all'intera popolazione di *generic user* italiani. Il

<sup>1</sup> Si veda nota nella tabella 2.

Guadagno di produttività (Euro)	1.967
Riduzione dei costi dell'ignoranza informatica (Euro)	294
Totale	2.261

**TABELLA 4**

*Il ritorno annuale della formazione ECDL per individuo*

Risparmi per singolo individuo (Euro)	Generic user	Risparmi complessivi (Euro)
2.261	6.700.000	15.146.899.000

**TABELLA 5**

*Gli effetti positivi sul sistema economico italiano della formazione informatica (tipo ECDL)*

risultato ottenibile è quello esposto in tabella 5.

## 5. LA FORMAZIONE INFORMATICA NELLE IMPRESE MANIFATTURIERE ITALIANE

Le imprese italiane sono coscienti dell'importanza della formazione informatica? Investono nella formazione? Quali ritorni si attendono da queste spese?

Per rispondere a questi quesiti i ricercatori della SDA Bocconi hanno intervistato un campione di imprese manifatturiere in Italia, costituito da 58 aziende prevalentemente del centro nord, e composto per il 58% da società con un fatturato superiore a 50 milioni

di euro (con più di 250 dipendenti), per il 35% da società con un fatturato compreso tra 10 e 50 milioni di euro e per il restante 7% da aziende con un fatturato inferiore ai 10 milioni di euro.

Il campione, che riproduce in modo emblematico la situazione del contesto imprenditoriale manifatturiero italiano, ha fornito i seguenti elementi caratterizzanti:

- dipendenti: 41% impiegati e dirigenti  
59% operai;
- spesa in formazione<sup>2</sup>: 71% delle imprese inferiore allo 0,05% del fatturato;
- spesa in formazione: 29% delle imprese compresa tra 0,05% e 0,5% del fatturato;
- aziende interessate: 8% delle imprese alla certificazione ECDL.

L'indagine ha mostrato che nelle imprese manifatturiere italiane la spesa in formazione è, nel complesso, modesta e, se ripartita sull'intero personale, ammonta ad alcune decine di euro all'anno (spesa pro capite). Inoltre, la spesa per la formazione informatica non rappresenta, mediamente, che il 20% di tale spesa, anche se vi sono imprese (le più piccole) che dichiarano in proposito una spesa nulla, e altre (le maggiori) che dichiarano una percentuale vicina al 40% della

spesa informatica sul costo totale sostenuto per la formazione.

Si è, inoltre, constatato che la spesa in formazione informatica cresce con:

- la dimensione dell'azienda;
- la complessità organizzativa (più aree di affari, più linee di prodotto, più canali di vendita, maggior peso dell'export ecc.);
- gli investimenti fatti nelle tecnologie informatiche.

Infine, si è chiesto alle aziende di illustrare i risultati che, a loro giudizio, erano stati raggiunti con la formazione informatica (nel caso in cui l'avessero svolta). Il quadro che si è ottenuto è riassunto nella tabella 6.

La ricerca conferma, quindi, quanto ci si poteva aspettare in termini qualitativi come ritorno della formazione informatica. C'è un generale consenso sulla importanza e sulla necessità di fare formazione. Quasi tutte le aziende vorrebbero farla, ma le più piccole hanno confessato che la dimensione delle strutture di cui dispongono non consente loro di distogliere personale dal lavoro produttivo per mandarlo in formazione. Per ora, sembra che la formazione informatica sia praticata maggiormente dalle aziende più grandi o più complesse.

La ricerca empirica ha mostrato che nelle aziende italiane, anche se non si arriva a calcolare il ROI della formazione informati-

Risultati conseguiti: la formazione ha ...	% delle aziende che concordano
.. migliorato la capacità d'uso degli strumenti informatici	97%
.. aumentato la produttività del personale	91%
.. determinato soddisfazione e gratificazione del personale	87%
.. migliorato la qualità dei documenti prodotti	80%
.. ridotto i costi di assistenza agli utenti	61%
.. ridotto il personale di staff (segretarie e assistenti)	58%
.. ridotto l'uso errato degli strumenti informatici e dei disservizi conseguenti	58%

**TABELLA 6**

*Gli effetti della formazione informatica secondo le imprese manifatturiere italiane*

<sup>2</sup> Formazione in generale, che quindi comprende anche quella informatica. Sono esclusi i costi del personale in formazione lavoro e il costo dell'addestramento mediante affiancamento.

ca, certamente vi è, però, un generale consenso sugli effetti positivi dell'*information technology training*.

### Bibliografia

- [1] Bassi L, McMurrer: *Training investment can mean financial performance*. American society for training and development 1998.
- [2] Carroll JM, Rosson MB: *Managing evaluation goals for training*. Communication of ACM; July 1998.
- [3] Camussone PF: Il costo dell'ignoranza nella società dell'informazione. *Mondo Digitale*, Anno II, giugno 2003, p. 3-14.
- [4] Crooks JW: *How important is training?* Communication of ACM; september 1994.
- [5] Danziger JN, Jenning JA, Park SC: *ICT Training*. Center for research on ICT and Organizations; University of California Irvine, 1999.
- [6] Danziger JN, Wang YC: *Enhancing end users' ICT skills in the new economy*. Center for research on ICT and Organizations; University of California Irvine, 2000.
- [7] Desmarais MC, Leclair R, Fiset J, Talby H: *Cost-justifying electronic performance support system*. Communication of ACM, July 1997.
- [8] Shapiro C, Varian: *Information Rules. A Strategic Guide to Network Economy*, Harvard Business School Press, Boston, 1999.

PIERFRANCO CAMUSSONE è professore di "Organizzazione e sistemi informativi" presso l'Università di Trento. Direttore dell'Area Sistemi Informativi della Scuola di Direzione Aziendale (SDA) della Bocconi. Membro di comitati scientifici di diverse riviste (tra cui *Economia e Management*, *Mondo Digitale*). Autore di numerose pubblicazioni sugli aspetti economici ed organizzativi dell'informatica.  
pierfranco.camussone@uni-bocconi.it

# CHIMICA COMPUTAZIONALE: PROGETTAZIONE DI MOLECOLE AL CALCOLATORE

La chimica computazionale ha compiuto progressi enormi negli ultimi due decenni grazie allo sviluppo di metodi avanzati per la determinazione della struttura elettronica di solidi, liquidi e molecole, di algoritmi di calcolo parallelo efficienti nonché all'aumentata potenza di elaborazione. Oggi, la chimica computazionale si affianca a pieno titolo ad altre discipline per l'interpretazione e la comprensione di dati sperimentali e per la progettazione di nuovi sistemi con proprietà ben definite.

## 1. INTRODUZIONE

**L**a chimica è sicuramente una scienza antichissima e il suo sviluppo è intimamente connesso a quello del genere umano, anche se per molto tempo la possibilità di controllare la trasformazione di sostanze in altre sostanze non è stata percepita in modo chiaro dall'uomo.

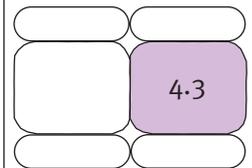
Per molti secoli la chimica, o meglio l'alchimia, è stata vista come una sorta di magia in grado di trasformare gli elementi gli uni negli altri, come dimostrato dai tanti cercatori della pietra filosofale che si sono cimentati in questa improbabile impresa. Comunque la si veda, non c'è alcun dubbio che da quando l'*homo sapiens* si è affermato come specie dominante, tutte le attività connesse con la chimica (dalla cottura dei cibi alla fusione dei metalli e delle leghe, dalla messa a punto di medicamenti e tisane, alla tintura dei tessuti ecc.) sono state di natura esclusivamente empirica, basate su tentativi ripetuti più volte, che oggi si definiscono sperimentali. Al punto che nel suo trattato "Corso di Filosofia Positiva" del 1830, il filosofo francese Augu-

ste Comte scriveva che "ogni tentativo di utilizzare metodi matematici nello studio di problemi chimici deve essere considerato profondamente irrazionale e contrario allo spirito della chimica. Se l'analisi matematica dovesse un giorno assumere un ruolo prominente nella chimica – una aberrazione che fortunatamente è quasi impossibile – ciò produrrebbe una rapida e generale degenerazione di questa scienza". Questo la dice lunga su come era visto l'uso di strumenti matematici per lo studio di problemi chimici in un secolo, l'Ottocento, che pure ha prodotto profonde e importantissime rivoluzioni in questo campo, basti pensare alle leggi di Faraday sull'elettrolisi, alla formulazione della termodinamica, al sistema periodico di Mendeléeu o ai composti organici otticamente attivi di Pasteur ecc..

In effetti, molti di questi sviluppi, su cui si basano le conoscenze attuali, sono stati raggiunti senza sapere che esistessero gli atomi e le molecole né come queste fossero fatte in realtà. Questa, infatti, è una conquista relativamente recente nella storia della scienza,



Gianfranco Pacchioni  
Cristiana Di Valentin



che risale ai primi tre decenni del 1900, quelli che George Gamow definì con una felice espressione “i trent’anni che sconvolsero la fisica”. Nei tre decenni che vanno dal 1900 al 1930, infatti, vennero poste le basi matematiche della teoria dei *quanti* su cui si basa l’attuale comprensione dei fenomeni atomici e molecolari, ossia della chimica e della fisica della materia. Al punto che il grande fisico inglese Paul A. M. Dirac nel 1929, ossia un secolo dopo la frase di Compton, riferendosi alle leggi della meccanica quantistica recentemente scoperte, poteva dichiarare che “*le leggi fisiche necessarie per una teoria matematica di tutta la chimica e di parte della fisica sono note completamente, e l’unica difficoltà è che l’applicazione esatta di queste leggi porta a equazioni troppo complicate per poter essere risolte*”. È interessante notare che in questa famosa affermazione Dirac parlava consapevolmente del fatto che le leggi della teoria dei quanti spiegano solo parte della fisica, in quanto esistono fenomeni che possono essere interpretati solo con la teoria della relatività. Dirac riteneva, erroneamente, che i fenomeni relativistici non avessero alcun effetto sui processi chimici, da cui l’affermazione per cui la meccanica quantistica spiegherebbe “tutta la chimica” e solo parte della fisica. In realtà, oggi si sa che anche molti fenomeni chimici sono strettamente legati a effetti relativistici. Un aspetto interessante della affermazione di Dirac resta, comunque, che la teoria matematica che è alla base delle trasformazioni chimiche, la teoria dei quanti, è sì completamente formulata, ma anche talmente complessa che le equazioni che andrebbero risolte per arrivare a dei risultati pratici sono di fatto troppo complicate. Se nel corso del 1900 non ci fosse stata la scoperta del *transistor*<sup>1</sup> e il successivo sviluppo della elaborazione elettronica, la frase di Dirac avrebbe costituito una sorta di suggello definitivo allo sviluppo di una teoria formale per la spiegazione dei fenomeni naturali atomici e molecolari elegante, ma di scarsa rilevanza pratica.

Le cose non sono andate così e i circa settant’anni che sono seguiti alla formulazione

della meccanica quantistica hanno visto da una parte, lo sviluppo vorticoso e continuo della potenza di elaborazione dei calcolatori elettronici, dall’altra la messa a punto di metodi e algoritmi sempre più avanzati ed efficienti volti alla soluzione delle equazioni di cui parlava Dirac. La combinazione di questi due fattori ha portato alla nascita di una nuova importante disciplina, la chimica computazionale. La definitiva consacrazione di questo campo è avvenuta verso la fine del secolo appena trascorso con l’assegnazione del Premio Nobel per la chimica 1998 a due “padri” di questa disciplina, John Pople e Walter Kohn. Da notare che Pople di formazione è un matematico e Kohn un fisico, a dimostrare una volta di più la natura interdisciplinare della scienza moderna e il ruolo fondamentale che la matematica e la fisica hanno nei fondamenti della chimica.

Cos’è allora la chimica computazionale o chimica al *computer*? È semplicemente una branca di un settore più ampio in rapida evoluzione che va sotto il nome di simulazioni al calcolatore o *scientific computing*. Questa nuova disciplina sta affiancando sempre di più l’approccio tradizionale alla soluzione di problemi scientifici e tecnologici basati sulla sperimentazione diretta (senza comunque sostituirla). La simulazione al calcolatore e il calcolo scientifico non vanno confusi con le aree più generali della chimica o della fisica teoriche. Gli esperimenti producono nuovi fatti e nuove scoperte e dischiudono all’uomo i segreti della natura. Il ruolo della teoria è quello di fornire un quadro di riferimento generale di spiegazione dei fenomeni osservati tramite un insieme di leggi matematiche. Il celebre Lord Kelvin diceva nel lontano 1900 che “*quando si può misurare ciò di cui si parla ed esprimerlo in numeri se ne sa qualcosa; ma quando non si può esprimerlo in formule o numeri allora la nostra conoscenza è scarna e insoddisfacente*”. Il calcolo scientifico è qualcosa a metà tra teoria ed esperimento: è basato su teorie e formalismi ben definiti e sviluppati ma è usato per produrre nuovi fatti e risultati, in un modo sempre più simile a quello in cui vengono svolti gli esperimenti. L’uso combinato di *software* e algoritmi avanzati e di potenza di calcolo elevata permette, oggi, di simulare un esperimento al computer

<sup>1</sup> Avvenuta nel 1947.

prima di effettuarlo, con costi più bassi e risposte più rapide. Ovviamente, la capacità predittiva di questi calcoli è molto legata agli algoritmi utilizzati e alle approssimazioni introdotte e, quindi, al livello di affidabilità della simulazione. Dopo alcuni decenni in cui i progressi in questa direzione sono stati piuttosto lenti, si è assistito nell'ultima decade del secolo scorso a una definitiva affermazione della chimica computazionale, al punto che oggi una parte consistente dei lavori originali che appaiono sulla letteratura scientifica internazionale si basano su questo metodo. Questo articolo intende fornire alcuni elementi per meglio comprendere come si è arrivati a ciò e quali sono i problemi che si pensa di poter risolvere in questo modo.

## 2. TEMPO E TEMPERATURA

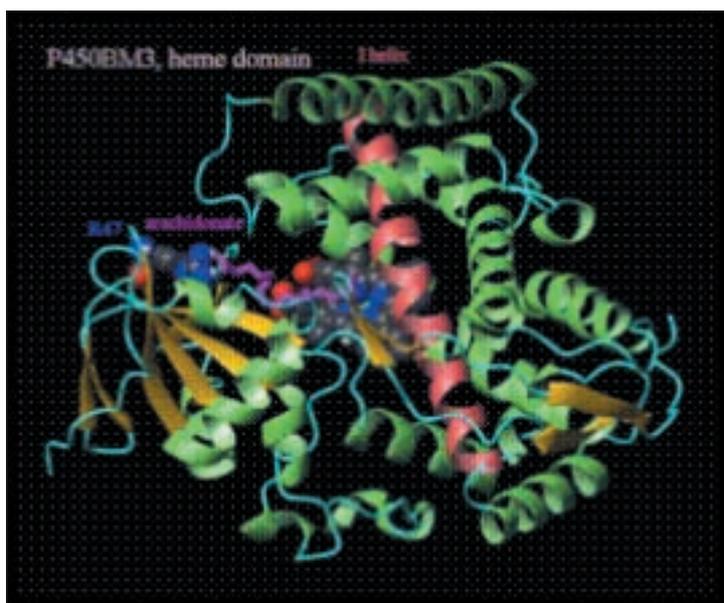
Simulare i processi chimici significa, di fatto, riprodurre in un computer fenomeni estremamente complessi che vanno dalla combinazione di due o più atomi isolati a formare una molecola gassosa, sino a trasformazioni in molti stadi che coinvolgono molecole di grandi dimensioni in soluzioni a molti componenti come i processi biochimici. In tutti i casi, il livello di complessità cresce via via che si introducono nuove variabili da cui dipendono le trasformazioni chimiche. Tra queste, due sono particolarmente importanti, il tempo e la temperatura. La totalità dei processi chimici avviene su scale temporali che vanno da frazioni infinitesimali di secondo sino a tempi geologici; si è di fronte a dei processi dinamici, in cui l'evoluzione temporale gioca un ruolo fondamentale. Più sono lunghi i tempi del processo, più sono lunghe le simulazioni e i relativi calcoli. L'altra variabile fondamentale dei processi dinamici è la temperatura. La temperatura misura l'energia termica di un sistema, a sua volta legata al moto molecolare. La materia a temperature bassissime tende a bloccare il moto dei nuclei atomici e il limite fisico dello zero assoluto ( $-273,16\text{ }^{\circ}\text{C}$ ) corrisponde, in pratica, ad assenza di movimento nucleare. Per considerare questi fattori è necessario fare delle simulazioni dinamiche, in cui si tiene conto della temperatura e l'evoluzione del sistema viene seguita per tempi sufficientemente lunghi. È questo l'ambito della

dinamica molecolare, di fatto l'obiettivo finale di tutte le simulazioni chimiche.

Per arrivare a una buona descrizione dei fenomeni dinamici è però necessario avere come base una conoscenza accurata dei campi di forze che governano le interazioni tra atomi e molecole, ossia dei processi elementari in cui legami chimici vengono rotti per formarne dei nuovi. È questa la vera base della teoria quantistica della chimica, ossia la capacità di prevedere e calcolare a priori la stabilità di nuovi composti, il costo energetico necessario per la loro preparazione, la loro reattività. Per fare ciò non è necessario studiare il fenomeno su scale temporali o introdurre effetti di temperatura. È, invece, necessario risolvere le equazioni fondamentali della quanto-meccanica (quelle di cui parlava Dirac) possibilmente con il minor numero possibile di approssimazioni, senza cioè introdurre nessun "ingrediente" dedotto dai dati sperimentali nei calcoli e partendo solo da costanti universali come massa e carica di nuclei ed elettroni.

Esiste, quindi, una gerarchia di livelli di simulazione, a complessità crescente, dove si passa da un esame delle proprietà fondamentali di una singola molecola al suo modo di interagire con altre molecole anche in situazioni complesse. Lo studio delle proprietà fondamentali senza uso di parametri empirici e di semplificazioni prende il nome di metodi *ab initio* o anche da primi principi, proprio per indicare l'assenza di assunzioni di sorta. Questi calcoli considerano le proprietà elettroniche e strutturali delle molecole alla temperatura dello zero assoluto e senza considerarne l'evoluzione nel tempo. Il fatto però di risolvere le complesse equazioni che descrivono la struttura elettronica in modo "esatto" rende questi metodi assai pesanti dal punto di vista computazionale, tanto che oggi essi sono ancora limitati a sistemi che contengono al massimo alcune centinaia di atomi (sino a pochi anni fa si parlava al massimo di poche decine di atomi!). Solo verso la metà degli anni '80 due ricercatori italiani, Roberto Car e Michele Parrinello, riuscirono a provare la possibilità di estendere l'uso dei metodi *ab initio* allo studio di problemi dinamici, dando così l'avvio al fertile campo della dinamica molecolare in cui evoluzione temporale e temperatura vengono inclusi senza approssimazioni empiriche. Og-

gi, questo tipo di simulazioni rappresenta una realtà, anche se ristretta ancora a tempi molto brevi e a insiemi di poche decine di atomi per via degli enormi costi computazionali. Quando si rende necessario simulare situazioni molto complesse, l'uso dei metodi *ab initio* si rivela, allo stato attuale, impraticabile. Il problema può venire affrontato mediante l'introduzione di approssimazioni, come ad esempio l'uso di parametri empirici derivati da misure sperimentali in calcoli che vengono, quindi, detti semi-empirici o più drasticamente abbandonando la trattazione quantistica del problema ricorrendo a delle descrizioni approssimate delle energie in gioco nella rottura e formazione di legami. Essendo basate sulle leggi della fisica classica, queste simulazioni vanno sotto il nome di dinamica molecolare classica. In questo modo è possibile effettuare simulazioni su scale temporali significative in tempi accettabili, al prezzo di rinunciare alla massima accuratezza dei risultati. Quando poi si debbono descrivere sistemi ancora più complessi (e questo è il caso reale della maggioranza dei processi chimici) la simulazione diviene di tipo sostanzialmente statistico e rinuncia in parte all'ambizioso progetto di partire dalla leggi fondamentali che descrivono la stabilità degli atomi per giungere a capire fenomeni chimici complessi.



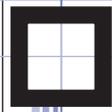
Struttura tridimensionale cristallina del centro emico dell'enzima P450 BM3  
Questa e le figure successive dell'articolo presentano esempi di molecole progettate al computer

A seconda del problema che si vuole descrivere esistono, quindi, strumenti computazionali specifici, dai più complessi e accurati, ma anche molto "costosi", metodi *ab initio* statici o dinamici, sino ai più semplici metodi approssimati classici o statistici. Nel seguito del presente articolo, si cercherà di illustrare gli aspetti fondamentali di questi approcci alle simulazioni in chimica.

### 3. LA TEORIA QUANTISTICA

I primi esempi di calcoli di chimica quantistica risalgono agli anni '30, al tempo in cui la meccanica quantistica era stata ormai formulata nelle sue grandi linee. Come noto, la meccanica quantistica affonda le sue radici nella teoria dei quanti enunciata da Max Planck all'inizio del secolo. Fu solo tra il 1920 e il 1930 che la nuova e rivoluzionaria teoria, secondo cui l'energia viene emessa o assorbita dai sistemi atomici o molecolari secondo quantità ben definite dette "quanti", ebbe un suo pieno riconoscimento nonché un proprio apparato matematico grazie ai contributi di scienziati quali de Broglie, Pauli, Schödinger, Heisenberg, Dirac.

Le leggi della meccanica quantistica si distinguono da quelle della fisica classica per il loro carattere fondamentale: infatti, mentre le leggi quantistiche sono in grado di descrivere altrettanto bene il moto di un elettrone attorno a un nucleo atomico quanto quello di un pianeta attorno al sole, ciò non è più vero per le leggi classiche. Per mezzo delle leggi classiche si può descrivere il comportamento (moto) di un meccanismo composto da molle, leve ecc., se si conoscono alcune costanti di materiali quali la densità, il calore specifico, l'elasticità. Però, se ci si chiede perché le costanti elastiche hanno i valori che hanno, perché una barra si spezza se la tensione cui è sottoposta supera un certo limite, e così via, la fisica classica non fornisce risposta. Essa non dice perché il rame fonde a 1083 °C, perché il vapore di sodio emette luce gialla, perché l'idrogeno ha certe proprietà chimiche, perché il sole brilla, perché il nucleo dell'atomo di uranio si disintegra spontaneamente, perché l'argento conduce l'elettricità, perché lo zolfo è un isolante. A queste domande può dare una risposta la meccanica quantistica.



Sulla base delle elementari interazioni tra le particelle costituenti la materia si può via via risalire ai comportamenti più complessi che danno vita ai fenomeni macroscopici che ogni giorno si osservano. Occorre, però, fare una precisazione: il fatto che la teoria dei quanti possa dare una risposta in teoria a tali domande non sempre significa che la possa dare anche in pratica. Qui entrano in gioco, infatti, la complessità del problema e la necessità di servirsi dei calcolatori elettronici.

#### 4. L'EQUAZIONE DI SCHRÖDINGER

Sempre negli anni '30, la formulazione matematica della fisica quantistica proposta da Schrödinger con la sua funzione d'onda si prestava particolarmente per lo studio di sistemi atomici o molecolari. Alcune grandezze fisiche nei sistemi microscopici sono in relazione con l'energia totale del sistema, ossia con il lavoro che occorre compiere per distruggere il sistema stesso. Se si parla di una molecola, ad esempio, tale quantità corrisponde al lavoro necessario per "scollare" le une dalle altre tutte le particelle presenti (nuclei ed elettroni) e portarle a distanza infinita privandole della loro energia cinetica. Ad ogni sistema atomico o molecolare è possibile associare un operatore hamiltoniano, indicato con  $H$ , che descrive l'energia del sistema. Applicare l'operatore hamiltoniano alla funzione d'onda  $\Psi$  porta alla celebre equazione di Schrödinger che nella forma più generale è nota come:

$$H \Psi = E \Psi$$

Più in dettaglio si può scrivere:

$$[-\nabla^2 + U(x, y, z)] \Psi(x, y, z) = E \Psi(x, y, z)$$

omettendo alcune costanti moltiplicative. Il termine in parentesi quadra è l'operatore hamiltoniano ed  $E$  è l'energia totale del sistema. Schrödinger dimostrò che tale equazione è risolvibile solo per valori quantizzati di  $E$ . Nell'operatore hamiltoniano sono contenuti due termini che rappresentano l'energia cinetica e quella potenziale delle particelle presenti nel sistema. Nel caso di un atomo, composto

da un nucleo carico positivamente e da un certo numero di elettroni, l'energia totale del sistema è data dalla somma di alcuni contributi. Innanzitutto, va considerata l'energia cinetica degli elettroni che, essendo in continua rotazione attorno al nucleo, possiedono una propria energia di movimento; poi va considerata l'attrazione elettrostatica nucleo-elettrone nonché la repulsione che ogni elettrone esercita sugli altri elettroni. Quindi l'operatore hamiltoniano conterrà tre termini, ciascuno dei quali rappresenta uno di questi contributi all'energia totale del sistema. L'operatore  $\nabla^2$ , descrive l'energia cinetica, mentre le interazioni elettrostatiche, attrattive o repulsive, sono racchiuse nel potenziale  $U(x, y, z)$ . Le informazioni relative al numero di elettroni e alla loro maggiore o minore distanza dal nucleo, alla loro disposizione spaziale sono, invece, contenute nella funzione d'onda  $\Psi$ . Il significato fisico di tale funzione è che essa, o meglio, il suo quadrato, descrive la probabilità di trovare un elettrone in una certa regione di spazio a un dato istante.

Da questi "ingredienti" di partenza è possibile ricavare *ab initio* tutta una serie di informazioni sulla struttura dell'atomo e sulle sue proprietà. Perfettamente simile è il caso molecolare, dove l'operatore hamiltoniano con-

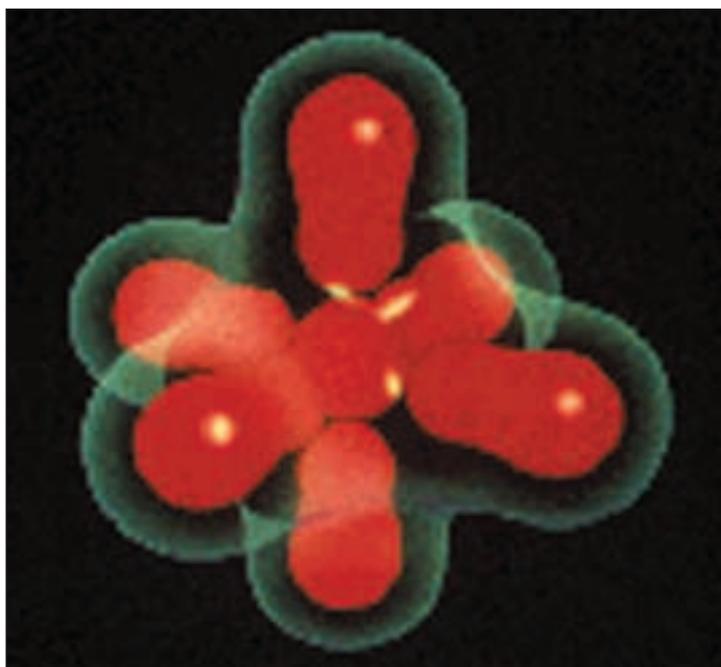
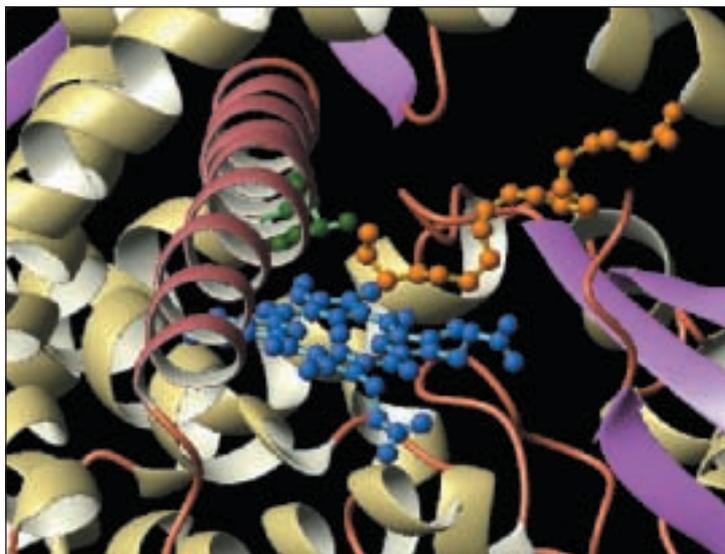


Immagine di una molecola di  $Cr(CO)_6$ . Le superfici rappresentate sono a densità elettronica costante



Particolare della struttura cristallina del centro eme (complesso ferro-porfirina) dell'enzima P450 BM3

terrà nuovi termini relativi alle interazioni non presenti nell'atomo come, per esempio, la repulsione reciproca dei nuclei atomici o l'attrazione che il nucleo A esercita sugli elettroni del nucleo B e viceversa.

Purtroppo l'equazione di Schrödinger, esattamente risolvibile nel caso dell'atomo di idrogeno e, con grandi sforzi, di altri sistemi molto semplici, si trasforma in un complicato insieme di equazioni differenziali man mano che il sistema in esame aumenta le sue dimensioni. Questo comporta a sua volta la risoluzione di un certo numero di integrali il cui numero cresce grosso modo come  $N^4$ , dove  $N$  è il numero di funzioni matematiche usate per descrivere il sistema (di fatto  $N$  cresce con il numero di elettroni  $e$ , quindi, con la complessità della molecola). Sino alla fine degli anni '50, quindi, la possibilità di studiare teoricamente il comportamento chimico-fisico della materia è stato limitato dalla scarsa diffusione di calcolatori sufficientemente potenti. Per dare un'idea di come l'analisi della struttura elettronica di una molecola richieda uno sforzo computazionale enorme basta dire che gli integrali da calcolare nel caso di una molecola non particolarmente grande come il glucosio sono circa tre milioni e mezzo. Essi possono però diventare molti di più quando si vogliono ottenere dei risultati particolarmente affidati.

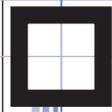
L'aumento progressivo della potenza di cal-

colo, dagli anni '80 a oggi, ha permesso di lavorare con sistemi molecolari di dimensioni via via maggiori. Grazie al calcolo parallelo e allo sviluppo di software in grado di sfruttare al massimo il parallelismo è possibile raggiungere grande efficienza di calcolo e risultati accurati per sistemi molecolari di dimensioni piuttosto grandi. La sfida però non è ancora conclusa: per esempio, lo studio *ab initio* di sistemi proteici può risultare impraticabile anche per i più moderni computer. Tanto per fare un esempio, lo studio di una molecola di interesse biologico come la ferro-porfirina, un'unità di base dell'emoglobina, richiede come minimo il calcolo di circa 220 milioni di integrali, ognuno dei quali viene valutato mediante un buon numero di operazioni.

## 5. DAI PRIMI CALCOLI QUANTOMECCANICI AI PROGRAMMI USER-FRIENDLY

La complessità delle equazioni da risolvere e del formalismo quantistico ha limitato notevolmente lo sviluppo della chimica teorica almeno sin verso la metà degli anni '60, quando la diffusione dei primi programmi di calcolo di una certa efficienza nonché di elaboratori più potenti ha permesso a questa disciplina di staccarsi dalla fase puramente teorico-matematica per entrare in quella di interpretazione, verifica e infine previsione di dati sperimentali.

Il tipo di calcolo prevede come unici dati di partenza le costanti fisiche del sistema, come la carica e la massa di elettroni e nuclei, e, ovviamente, il numero di elettroni e di nuclei presenti nella molecola nonché le posizioni spaziali dei nuclei stessi. Fatta l'approssimazione che i nuclei siano fissi, dato che la loro velocità di movimento risulta estremamente più bassa rispetto a quella degli elettroni, il calcolo viene effettuato, senza introdurre cioè ulteriori approssimazioni. Secondo la teoria MO (*Molecular Orbital*), un orbitale molecolare è espresso come combinazione lineare di orbitali atomici e la forma, nonché l'energia, di tali orbitali molecolari viene determinata risolvendo un'equazione matriciale ad autovalori-autovettori in cui compaiono matrici di ordine  $n$ , dove  $n$  è il numero di orbitali di base del siste-



ma. È questo l'approccio noto come metodo di Hartree-Fock, dal nome dei due scienziati che lo hanno sviluppato.

Tra i primi calcoli eseguiti in questo modo vi sono quelli che l'italiano Enrico Clementi fece dal 1960 al 1964 negli Stati Uniti. Il programma di calcolo molecolare IBMOL, che venne messo a punto per questo genere di studi, è uno dei primi che sia circolato negli ambienti scientifici, ed è stato utilizzato per almeno i successivi 10-15 anni. Questo e altri codici basati sul metodo di Hartree-Fock sono stati al centro dell'attività in chimica quantistica con metodi *ab initio* negli anni '60 e '70. Va detto, però, che la soluzione dell'equazione di Schrödinger e la determinazione della funzione d'onda del sistema mediante l'approssimazione di Hartree-Fock soffre di un problema. Il metodo di Hartree-Fock non include, infatti, nel calcolo gli effetti noti come correlazione elettronica, una correzione all'energia totale del sistema piccola in termini relativi ma importantissima ai fini della descrizione accurata del legame chimico. Per fare un esempio, l'energia di legame di una molecola semplice come l'ossigeno molecolare, O<sub>2</sub>, è circa un quinto di quella sperimentale quando determinata mediante l'approssimazione di Hartree-Fock. L'introduzione della correlazione elettronica porta, invece, a un risultato praticamente identico a quello sperimentale. Purtroppo l'introduzione della correlazione elettronica a partire dalla funzione d'onda Hartree-Fock rappresenta un problema complesso che ha impegnato i chimici quantistici per oltre trent'anni nel tentativo di mettere a punto algoritmi e metodi efficienti per introdurre questo termine. Va detto che questi sforzi hanno condotto a metodi avanzati per risolvere l'equazione di Schrödinger (teoria delle perturbazioni, interazione di configurazione ecc.) che portano praticamente alla soluzione esatta al punto di dover rivedere e re-interpretare in alcuni casi particolari gli stessi dati sperimentali.

Con lo sviluppo di nuovi formalismi e metodi, e lo scambio nell'ambito scientifico accademico dei primi **codici di calcolo** (si veda a tal proposito il riquadro a p. 38) si è assistito così allo sviluppo via via più rapido anche delle applicazioni della chimica quantistica. In questo senso, un ruolo importante lo ha avu-

to una istituzione nota come QCPE (*Quantum Chemistry Program Exchange*)<sup>2</sup>, presso l'Università dell'Indiana, che per almeno due decenni ha costituito un punto di riferimento per mettere a disposizione della comunità scientifica programmi e software di chimica quantistica, in un'era dominata dalla mancanza di standard, se si eccettua il linguaggio di programmazione, il *Fortran*. Da alcuni di questi codici e *subroutine*, poi inglobati in programmi via via più ricchi, versatili e complessi sono nati alcuni dei programmi oggi disponibili a chiunque voglia fare del calcolo quantistico. Tra questi il più noto è certamente il programma Gaussian, nelle sue molteplici versioni, sviluppato a partire dai primi anni '70 dal gruppo di John Pople alla Carnegie-Mellon University a Pittsburgh. Questo programma è divenuto poi di ampio utilizzo anche per non-esperti e il premio Nobel per la chimica del 1998 a Pople ha, di fatto, sancito l'importante ruolo di questo codice nel rendere popolare e accessibile la chimica quantistica anche a molti ricercatori di formazione sperimentale.

Si può dire, quindi, che l'obiettivo iniziale che ci si è posti nell'affrontare lo studio delle interazioni chimiche per via computazionale, il riprodurre i dati sperimentali esistenti al fine di verificare l'affidabilità del metodo, è stato pienamente raggiunto già a metà degli anni '80. Da allora in poi, la chimica computazionale viene comunemente utilizzata come strumento di modellizzazione, interpretazione, razionalizzazione e previsione e ha, quindi, assunto, a tutti gli effetti, un ruolo attivo e insostituibile nello sviluppo della chimica moderna. La maggior parte delle reazioni di interesse chimico, fisico o biologico, coinvolge però sistemi di grandi dimensioni. Basti pensare ai processi di sintesi di composti organici naturali, alla struttura dello stato solido o alle reazioni del DNA. In questo senso, i metodi basati sul formalismo di Hartree-Fock e sulla introduzione della correlazione elettronica si sono rivelati molto costosi, richiedendo tempi di elaborazione che crescono

<sup>2</sup> Quantum Chemistry Program Exchange: [http://www.chem.indiana.edu/facilities/qcpe\\_frontend.htm](http://www.chem.indiana.edu/facilities/qcpe_frontend.htm)

## I codici di calcolo di simulazione quantistica

### **GAUSSIANO3** (<http://www.gaussian.com/>)

*Gaussian 03* è la versione più recente della serie *Gaussian* di programmi per il calcolo della struttura elettronica. *Gaussian 03* è usato da chimici, ingegneri chimici, biochimici, fisici e altri scienziati per la ricerca in ambiti consolidati ed emergenti di interesse chimico. Partendo dalle leggi fondamentali della meccanica quantistica, *Gaussian* è in grado di prevedere le energie, le strutture molecolari, e le frequenze di vibrazione di sistemi molecolari, insieme a numerose altre proprietà molecolari derivanti da questo tipo di conti di base. Può essere applicato allo studio di molecole e reazioni in un'ampia finestra di condizioni, che include sia le specie stabili sia i composti che non possono o possono solo con difficoltà essere osservati sperimentalmente, come, ad esempio intermedi a vita breve e strutture di transizione.

### **GAMESS-UK** (<http://www.cse.clcr.ac.uk/qcg/gamess-uk/>)

*GAMESS-UK* è un codice per il calcolo *ab initio* della struttura elettronica. È in grado di svolgere calcoli Hartree-Fock, DFT e multi-configurazionali, oltre che tutta una serie di calcoli utilizzando le varie tecniche post Hartree Fock. *GAMESS-UK* è basato sul software originale *GAMESS* (1981) e il suo sviluppo è stato coordinato negli ultimi dieci anni dal Daresbury Laboratory.

### **ADF** (<http://www.scm.com/>)

*ADF* è un pacchetto software basato sulla Teoria del Funzionale della Densità sviluppato presso l'Università di Amsterdam. Permette di svolgere calcoli di struttura elettronica a principi primi ed è usato con profitto da chimici teorici e computazionali di tutto il mondo, sia nell'industria che nell'accademia. *ADF* è usato in effetti in tutti i campi della chimica, ed è particolarmente popolare in catalisi, chimica inorganica, in quella dei metalli pesanti, come in spettroscopia. Può essere applicato a tutti i sistemi, proteine, solventi, polimeri, superfici, e solidi, alle semplici molecole in fase gassosa.

### **NWCHEM** (<http://www.emsl.pnl.gov:2080/docs/nwchem/>)

*NWChem* è un pacchetto software di chimica computazionale, progettato per essere eseguito sia su supercomputer paralleli ad alta prestazione, sia su convenzionali *cluster* di personal computer. L'obiettivo della progettazione è quello di creare un codice in grado di scalare al meglio sia nell'abilità di trattare efficientemente problemi di grosse dimensioni, sia nell'uso di risorse di calcolo parallele. *NWChem* è stato sviluppato presso la *Pacific Northwest National Laboratory* (PNNL).

### **TURBOMOLE** (<http://www.turbomole.com/>)

*TURBOMOLE* è stato sviluppato presso l'Università di Karlsruhe ed è progettato con speciale attenzione per stazioni di lavoro UNIX e personal computer, sfruttando in modo efficiente le caratteristiche e capacità di questo tipo di hardware. Una caratteristica eccellente di *TURBOMOLE* è costituita dagli algoritmi semi-diretti adattabili alla memoria e alle richieste di spazio-disco.

### **Q-CHEM** (<http://www.q-chem.com>)

*Q-Chem 2.0* è in grado di condurre calcoli a principi primi sia sullo stato fondamentale che sugli stati eccitati delle molecole. Include tutta una serie di metodi che non sono disponibili in altri pacchetti di chimica quantistica computazionale.

### **CPMD** (<http://www.cpmc.org>)

*CPMD* (Dinamica Molecolare Car-Parrinello) è un codice a onde piane/pseudopotenziali basato sulla Teoria del Funzionale della Densità, progettato specialmente per condurre calcoli di dinamica molecolare a principi primi. È applicato principalmente a problemi di stato solido.

### **DeMon** ([http://www.demon-software.com/public\\_html](http://www.demon-software.com/public_html))

Il programma *deMon-KS*, sviluppato presso l'Università di Montreal, permette di condurre calcoli *ab initio*, basati sulla Teoria del Funzionale della Densità, su sistemi di grosse dimensioni, caratterizzati anche dalla presenza di metalli di transizione, ottenendo risultati di alta precisione in tempi relativamente brevi.

### **CRYSTAL** (<http://www.cse.clcr.ac.uk/cm/g/CRYSTAL>)

Il programma *CRYSTAL*, sviluppato presso l'Università di Torino, calcola la struttura elettronica di sistemi periodici con i metodi Hartree Fock e DFT. Il codice può essere utilizzato per condurre studi approfonditi su struttura e proprietà fisiche, elettroniche e magnetiche di molecole, polimeri, superfici e solidi cristallini.

### **VASP** (<http://cms.mpi.univie.ac.at/vasp/>)

*VASP* è stato sviluppato presso l'Università di Vienna e progettato per condurre calcoli di dinamica molecolare (MD) quantistica da primi principi utilizzando pseudopotenziali e funzioni di base costituite da onde piane. È particolarmente indicato per lo studio e la descrizione di solidi cristallini.

Un sito in cui è possibile trovare un elenco dettagliato di pacchetti software per simulazioni in aree di interesse chimico è il seguente:

### **SAL** (<http://sal.kachinatech.com/Z/2>)

*SAL* (Applicazioni Scientifiche su Linux) è una collezione di informazioni e collegamenti virtuali a pacchetti software di interesse in campo scientifico e ingegneristico.

come  $N^7$  dove  $N$  è il numero funzioni usate per descrivere il sistema. È a questo punto che nel panorama delle simulazioni quantistiche in chimica si è affacciata, in modo decisivo, una diversa metodologia per affrontare il problema.

## 6. LA TEORIA DEL FUNZIONALE DELLA DENSITÀ

Dall'inizio degli anni '90, la teoria del funzionale della densità, che sino ad allora occupava una posizione periferica in ambito chimico essendo stata sviluppata per lo studio di solidi cristallini, ha svolto un ruolo da protagonista nell'ambito della modellistica chimica. Lo sviluppo è dovuto essenzialmente all'eccellente accuratezza raggiunta dai metodi basati su questa teoria, ma anche al loro costo computazionale, decisamente inferiore rispetto ai metodi tipo Hartree-Fock. La teoria del funzionale della densità ha trovato inizialmente applicazione nell'ambito della fisica dello stato solido, e solo in un secondo momento, anche in quello della chimica. La transizione non è stata semplice in quanto il linguaggio della comunità dei fisici dello stato solido era completamente estraneo a quello ormai consolidato nella comunità dei chimici quantistici.

Alla base della teoria del funzionale della densità c'è il teorema di Hohenberg e Kohn del 1964 secondo cui l'energia elettronica dello stato fondamentale di un solido o di una molecola,  $E$ , è determinata completamente dalla densità elettronica  $\rho$ , una grandezza che descrive la distribuzione degli elettroni attorno ai nuclei atomici. In altre parole, nota la densità elettronica dello stato fondamentale, l'energia totale del sistema (come anche le altre proprietà elettroniche) è determinata in modo univoco. La conseguenza di questo teorema può essere meglio illustrata per confronto con la teoria degli orbitali molecolari. Mentre la complessità della funzione d'onda molecolare  $\Psi$  aumenta con il numero degli elettroni, la densità elettronica, che dipende solo dalle tre coordinate spaziali, ha lo stesso numero di variabili, indipendentemente dalle dimensioni del sistema.

Nonostante sia stato dimostrato che a cia-

scuna differente densità elettronica corrisponde una differente energia dello stato fondamentale, esiste ancora un problema non risolto per questa teoria: il funzionale che mette in relazione le due quantità,  $E$  e  $\rho$ , non è noto. L'obiettivo dei vari metodi basati sulla teoria del funzionale della densità, la DFT (*Density Functional Theory*), è la formulazione di funzionali in grado di mettere in relazione la densità elettronica con l'energia.

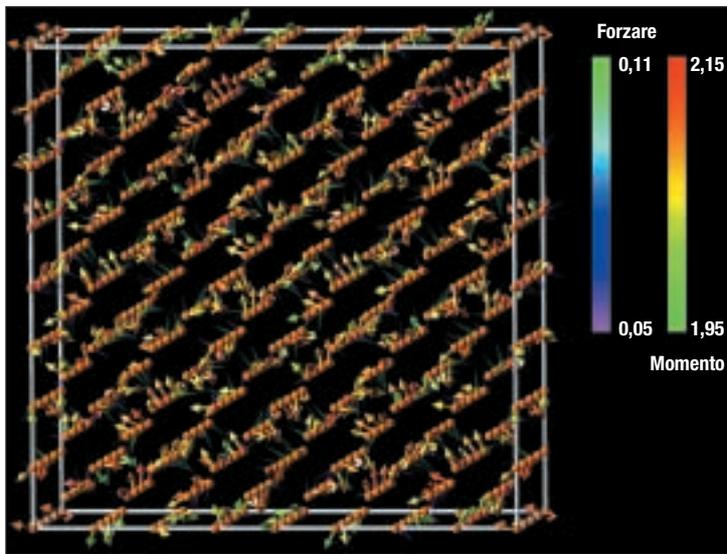
I metodi DFT sono stati adottati dalla comunità dei chimici computazionali anche grazie all'introduzione degli orbitali di Kohn e Sham (1965). Alla base della teoria di Kohn-Sham vi è il calcolo dell'energia cinetica  $E_c$  nell'ipotesi che gli elettroni non interagiscano. Poiché nel sistema reale gli elettroni interagiscono, l'energia cinetica calcolata in questa approssimazione non è quella totale. Tuttavia, la differenza tra il valore esatto e quello calcolato è piccola e viene assorbita in un termine definito di scambio-correlazione.

Un'espressione generale per l'energia DFT può essere scritta come:

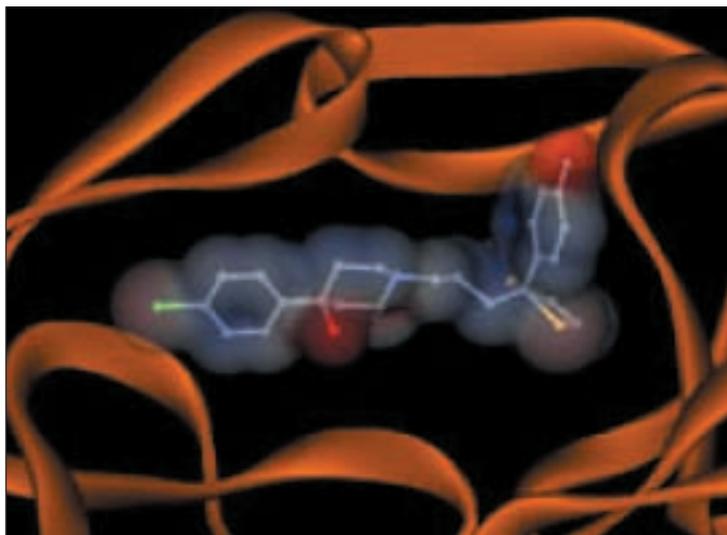
$$E_{\text{DFT}} = E_c(\rho) + E_{\text{ne}}(\rho) + J(\rho) + E_{\text{xc}}(\rho)$$

dove  $E_c$  è l'energia cinetica nell'ipotesi di elettroni non-interagenti,  $E_{\text{ne}}$  è l'energia di attrazione nucleo-elettrone,  $J$  è l'energia di interazione coulombiana e  $E_{\text{xc}}$  è l'energia di scambio-correlazione. In questo modo, la correlazione elettronica, che con i metodi basati sulla determinazione della funzione d'onda tipo Hartree-Fock richiede calcoli estremamente onerosi, viene introdotta in modo molto diretto e senza sostanziale aggravio. Il risultato è una accuratezza elevata a un ragionevole costo di elaborazione.

Il maggior problema nell'ambito della teoria DFT è ottenere funzionali in grado di descrivere bene il termine di scambio-correlazione. Una volta determinato il funzionale, il problema risulta molto simile a quello incontrato per la teoria degli orbitali molecolari. Nonostante si possano riscontrare molte somiglianze, esistono anche notevoli differenze rispetto alla teoria basata sugli orbitali molecolari. La forma dei funzionali è progettata in modo che essi presentino un



Il risultato del calcolo dei momenti magnetici per una cella unitaria di 512 atomi di ferro mediante calcoli di tipo DFT. Il momento magnetico di ciascun atomo è descritto da un vettore



Modello a nastro per l'enzima proteasi HIV-1 complessato a un inibitore non peptidico

determinato comportamento limite, e che riproducano determinati parametri noti. Nella *Local Density Approximation* (LDA) si assume che la densità possa essere trattata localmente come un gas uniforme di elettroni. Gli sviluppi dell'approccio LDA hanno portato a considerare un gas di elettroni non-uniforme. Ciò comporta una dipendenza dell'energia del sistema anche dalla derivata prima della densità. Questi metodi sono noti come *Gradient Corrected or Generalized Gradient Approximation* (GGA) e sono

quelli che hanno consentito l'applicazione con successo della teoria DFT ai problemi di interesse chimico dove sono richieste accuratissime molto elevate. Oggi, la quasi totalità delle simulazioni da primi principi fatte in ambito chimico si basano sul metodo DFT, mentre solo 10-15 anni fa non più del 10-15% degli studi si basavano su questo approccio.

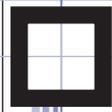
## 7. I METODI SEMIEMPIRICI

Dalla seconda metà degli anni '60, hanno trovato largo impiego i metodi semiempirici, così detti in quanto oltre alle informazioni di base già presenti nei metodi *ab initio* si introducono altri dati di carattere sperimentale (da cui il nome di metodi semiempirici). In questo modo, molti degli integrali da calcolare, corrispondenti ad altrettante interazioni all'interno della molecola vengono trascurati o parametrizzati in modo da ridurre drasticamente le dimensioni del calcolo. Inoltre, questi metodi tendono a considerare espressamente solo alcuni degli elettroni presenti nella molecola e in particolare quelli che determinano le proprietà chimiche.

Grazie ai metodi semiempirici è stato possibile per la prima volta gettare un ponte tra teoria e sperimentazione negli anni in cui la complessità dei conti *ab initio* era fuori portata, sia per la mancanza di algoritmi efficienti sia per la ridotta capacità dei processori (si parla degli anni '60 e '70). Oggi, questi metodi vengono spesso usati per descrivere la porzione più esterna rispetto al sito reattivo di una molecola o di un solido. In pratica, la molecola viene suddivisa in due regioni trattate a livello differente. La porzione più interna viene trattata ad un elevato livello di teoria (*ab initio*), mentre la porzione più esterna viene trattata con un metodo semiempirico.

## 8. DINAMICA MOLECOLARE

Fino ad ora, si è analizzato in dettaglio come calcolare l'energia totale e le proprietà di una molecola, senza considerare la sua evoluzione nel tempo. Si è parlato, infatti, dei metodi "statici", in cui le posizioni nucleari del siste-



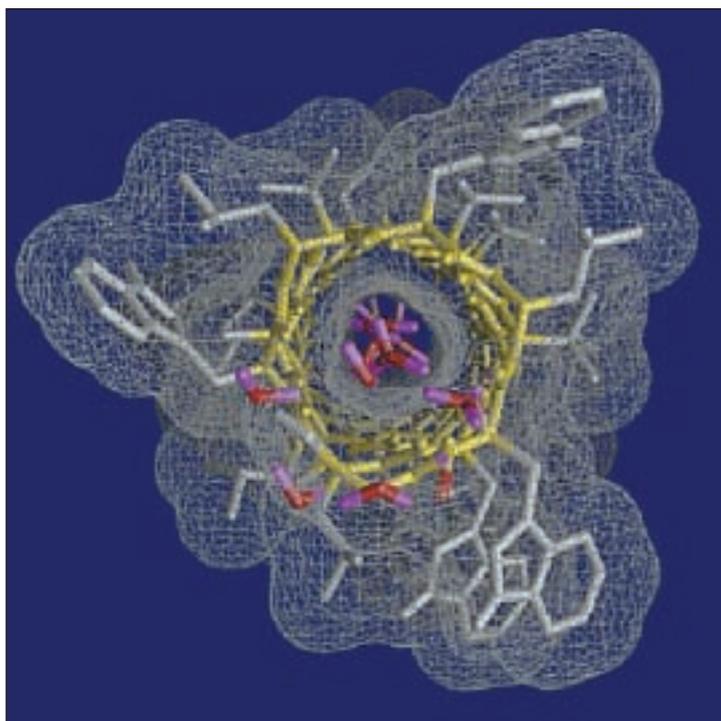
ma sono fisse. Si è però detto all'inizio che il maggior interesse sta nella trattazione di fenomeni dinamici.

I metodi di simulazione tradizionale possono essere suddivisi in due classi:

1. simulazioni stocastiche;
2. simulazioni deterministiche, che sono ampiamente rappresentate dal metodo di Monte Carlo (MC) e dal metodo di Dinamica Molecolare (MD), rispettivamente.

Le simulazioni Monte Carlo si basano su un approccio statistico. Varie configurazioni del sistema vengono generate per spostamenti casuali delle particelle e un algoritmo stabilisce se accettare o rifiutare una nuova configurazione sulla base della variazione di energia. Cammini verso energie inferiori sono sempre accettati, quelli a più alte energie sono accettati con una probabilità determinata dalla statistica di Boltzmann. In questo modo, le proprietà del sistema possono essere calcolate come media su tutti gli spostamenti. Al contrario, i metodi di dinamica molecolare si basano sulle equazioni del moto definite dall'hamiltoniana del sistema. Il metodo consiste nell'integrazione di queste equazioni per ottenere le nuove posizioni e velocità delle particelle. Le simulazioni richiedono sempre la definizione di un modello (campo di forze) per le interazioni tra le varie componenti del sistema. Tale modello deve rispettare alcune leggi fondamentali e va testato attraverso un confronto con i dati sperimentali disponibili. Gli ingredienti indispensabili per una simulazione di dinamica molecolare sono: il modello, un integratore per propagare le posizioni e le velocità in un intervallo di tempo prestabilito e un *ensemble statistico*. Il risultato della simulazione è corretto solo rispetto al modello scelto e va, quindi, analizzato sulla base delle previsioni teoriche e delle osservazioni sperimentali.

Una questione importante relativa alle simulazioni è la scala dei tempi e delle lunghezze percorribili. Più sofisticata è la tecnica di simulazione, più breve è l'intervallo di tempo e delle lunghezze accessibili. La *performance* di una simulazione di dinamica di particelle dipende fortemente dalle attrezzature computazionali disponibili. Metodi di dinamica molecolare classica sono oggi applicati a una vasta gam-



*La gramicidina costituisce il canale proteico più studiato. La struttura riportata è stata ottenuta attraverso un calcolo di dinamica molecolare*

ma di problemi, come, ad esempio, proprietà dei liquidi, difetti nei solidi, proprietà di superficie, grandi aggregati molecolari, biomolecole ecc.. Negli ultimi anni molti programmi sono stati progettati anche per il calcolo su computer paralleli, accelerando, quindi, in modo considerevole, i tempi di calcolo.

Il regno dei metodi di dinamica molecolare tradizionale e di struttura elettronica si è esteso ampiamente grazie allo sviluppo della famiglia di tecniche chiamata dinamica molecolare *ab initio* di cui si diceva al paragrafo 2. L'idea alla base di questi metodi è di calcolare le forze che agiscono sui nuclei determinando la struttura elettronica del sistema da primi principi. Il problema, in questo caso, non è più selezionare il campo di forze ma selezionare il livello di approssimazione a cui risolvere l'equazione di Schrödinger. Questo permette di studiare anche sistemi chimicamente complessi. Ovviamente, i notevoli vantaggi sono accompagnati da un alto prezzo da pagare a causa del collocamento della dinamica molecolare in una cornice *ab initio*: i tempi e le lunghezze accessibili sono molto più piccoli. Tuttavia, quest'ultima tecnica, non influenzata da modelli predisposti, per-

mette di riscontrare eventuali fenomeni inattesi e, quindi, presenta una potenzialità di previsione molto elevata.

## 9. LE RISPOSTE DELLA CHIMICA TEORICA

Chiariti pregi e difetti dei principali metodi di calcolo di cui si avvale oggi la chimica computazionale si può tentare di illustrare le domande a cui l'uso combinato dei principi della meccanica quantistica e dei calcolatori elettronici può dare una risposta adeguata. Per prima cosa è possibile uno studio conformazionale di molecole, ioni, radicali ecc., ossia della disposizione geometrica ottimale dei nuclei atomici all'interno della struttura molecolare. Si tratta di un aspetto piuttosto importante in quanto dalla "forma" delle molecole spesso dipende la reattività. Per determinare la geometria di una molecola si spostano gli atomi dalle loro posizioni valutando le conseguenti variazioni dell'energia totale del sistema. La geometria che corrisponde alla minima energia è quella più favorita.

Una seconda importante risposta riguarda

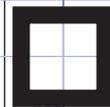
la stabilità dei sistemi chimici. Per fare un semplice esempio, si consideri la molecola di idrogeno,  $H_2$ , la cui energia è stata valutata per via teorica in 31,8 eV (elettronvolt). L'energia dell'atomo di idrogeno è di 13,6 eV, per cui dalla combinazione di due atomi a dare la molecola si ha un guadagno di 4,6 eV. Questa è la forza del legame H-H e corrisponde all'energia che occorre fornire per "rompere" la molecola  $H_2$  e riottenere i due atomi separati. L'importanza della valutazione delle energie di legame è notevole. Ogni reazione chimica, partendo da certi reagenti iniziali, conduce a nuovi prodotti finali coinvolgendo la rottura di alcuni legami e la formazione di altri. Se i legami di partenza sono deboli è evidente che la reazione sarà più facile in quanto tali legami si romperanno senza che si debba fornire troppa energia (per esempio, sotto forma di riscaldamento). Inoltre, se si formano nuovi legami particolarmente forti il prodotto finale sarà stabile e non tenderà a decomporre o ad alterarsi.

Va fatta, però, una distinzione tra gli aspetti termodinamici e quelli cinetici. La variazione di energia nel passaggio da reagenti a prodotti caratterizza l'energetica della reazione ma può accadere che un processo favorito dal punto di vista energetico presenti una velocità di reazione talmente bassa che la reazione non può essere osservata. La chimica computazionale permette di calcolare non solo le quantità termodinamiche ma anche le barriere di attivazione per il passaggio da reagenti a prodotti e, quindi, la cinetica o velocità della reazione.

L'altro grande campo di applicazione è quello della determinazione delle proprietà elettroniche del sistema. Un numero enorme di informazioni è, infatti, racchiuso nella funzione d'onda elettronica del sistema o, il che è di fatto la stessa cosa, nella sua densità totale nel caso si usino metodi tipo DFT. Per esempio, è possibile assegnare delle cariche atomiche, un parametro molto utile per la discussione e la razionalizzazione della struttura e della reattività molecolari. Da queste derivano poi i potenziali elettrostatici delle molecole che determinano molte delle interazioni con quello che sta attorno, per esempio, molecole di solvente o altre molecole.



Copertina del numero di Marzo 2003 della rivista "Science" dedicata allo studio della catalisi sulla base del calcolo quantistico



Basti dire che dalla forma del potenziale elettrostatico dipendono molte delle interazioni biologiche e farmacologiche. Esiste poi la risposta di molecole e solidi alla radiazione elettromagnetica che sta alla base delle moderne spettroscopie. Grazie a queste accurate tecniche di indagine è possibile acquisire un gran numero di informazioni sulle sostanze chimiche, nonché riconoscerle anche quando sono presenti in tracce. A questo proposito, basta accennare a una tecnica spettroscopica sviluppata in origine a puro scopo analitico, la *Risonanza Magnetica Nucleare* (NMR) ma che viene oggi utilizzata comunemente in diagnostica medica. Spettri NMR possono essere simulati mediante il calcolo quantistico e, attualmente, è piuttosto comune affiancare alla misura sperimentale i valori calcolati in modo da poter effettuare assegnazioni precise e prive di ambiguità. Queste risposte riguardano tantissime proprietà, da quelle vibrazionali e quelle ottiche (il che permette di spiegare perché e quando certi composti sono colorati), da quelle di conducibilità elettrica ai comportamenti magnetici ecc.. Si può dire che oggi la previsione delle proprietà elettroniche e spettroscopiche dei composti chimici è forse una delle conquiste più importanti della chimica computazionale.

## 10. CONCLUSIONI

La teoria dei quanti ha da poco compiuto i 100 anni, essendo stata formulata da Max Planck ai primi del Novecento, ma solo da poco più di un decennio le simulazioni quantitative in chimica hanno assunto un ruolo veramente importante. Questo è dovuto, da una parte, al vorticoso sviluppo della capacità di elaborazione e dall'altra al continuo progresso della teoria nel fornire strumenti adeguati ed efficaci per la soluzione del problema. I grandi progressi ottenuti su questo secondo fronte hanno permesso di raggiungere livelli di affidabilità delle simulazioni tali da promuovere, a tutti gli effetti, la chimica computazionale da un ruolo ancillare a quello di disciplina complementare al metodo sperimentale. Il confronto diretto dei risultati computazionali con i valori sperimentali permette non solo di convalidare il modello teorico proposto, ma anche di fare previsioni e suggerire eventuali altri esperimenti da condurre. Recenti progressi nel campo dei fenomeni dinamici, lo sviluppo di algoritmi sempre più efficienti e accurati, insieme al prevedibile aumento delle capacità di elaborazione garantiscono che il ruolo delle simulazioni in chimica è destinato ad aumentare nei prossimi anni a una velocità maggiore di quella sostenuta sin qui.

GIANFRANCO PACCHIONI è Ordinario di Chimica dello Stato Solido presso il Corso di Laurea in Scienza dei Materiali dell'Università di Milano-Bicocca. Ottenuto il Dottorato di ricerca a Berlino nel 1984, è stato Visiting Professor al centro ricerche IBM di Almaden in California, al Politecnico di Monaco di Baviera e all'Università di Barcellona. È autore di oltre 250 lavori sulla struttura elettronica mediante calcolo quantistico di complessi e materiali inorganici. Dal 1999, dirige la Scuola di Specializzazione in Scienza e Tecnologia dei Materiali dell'Università di Milano-Bicocca.  
gianfranco.pacchioni@unimib.it

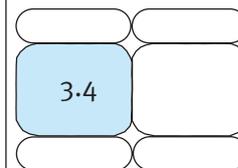
CRISTIANA DI VALENTIN ha conseguito il Dottorato di ricerca nel 2000 presso l'Università di Pavia, svolgendo una tesi di chimica computazionale in collaborazione con la Technische Universität di Monaco. Nel 2002, si è specializzata in Scienza e Tecnologia dei Materiali presso l'Università di Milano-Bicocca. Da dicembre 2002 è ricercatore universitario presso il Dipartimento di Scienza dei Materiali dell'Università di Milano-Bicocca.  
cristiana.divalentin@unimib.it

# UBIQUITOUS COMPUTING



Roberto Saracco

Con l'espressione *Ubiquitous Computing*<sup>1</sup> si possono intendere più aspetti: la pervasività di sistemi che offrono capacità di elaborazione locale, la presenza di capacità elaborativa in una molteplicità di oggetti attorno a noi, la possibilità di accedere da qualsiasi punto a capacità elaborative in grado di soddisfare qualunque (o quasi) necessità. L'obiettivo del presente articolo è di fare una panoramica sulle tendenze evolutive dell'Ubiquitous Computing con una maggiore attenzione alla terza accezione.



## 1. UNO PER TANTI, UNO CIASCUNO, TANTI PER UNO

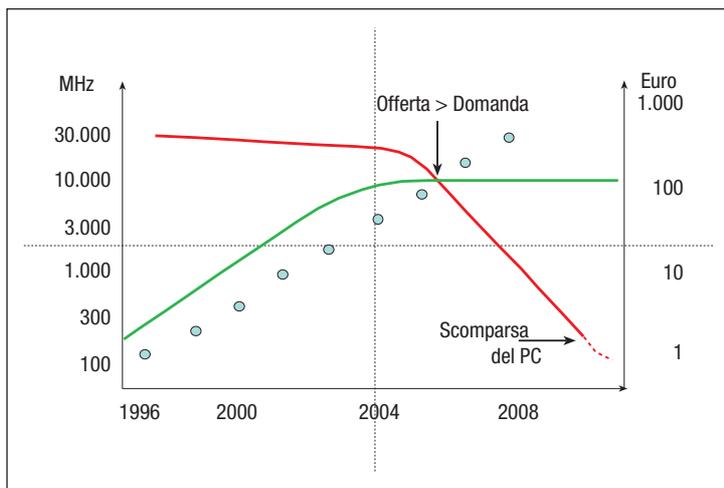
In principio l'uomo creò il *mainframe*. Era così costoso e così complicato che il suo accesso era consentito a pochi che ne condividevano le risorse. Quindi "uno per tanti", ma non solo: era, infatti, difficile anche immaginare che molti potessero volere accedere a risorse elaborative<sup>2</sup>. Poi venne il PC e il paradigma divenne "una persona, un PC". Quindi arrivarono i *laptop* e i PDA (*Personal Digital Assistant*) e ben presto il paradigma è diventato "tanti elaboratore per ogni persona".

Ci si limita per il momento a considerare "i tanti per uno" dove i tanti sono il PC, laptop e PDA ma anche l'agenda elettronica, la calcolatrice tascabile, il telefonino, il navigatore, il telecomando dell'*home entertainment system*: ovvero, tutti quegli oggetti che, in qualche modo, elaborano, su richiesta, le informazioni.

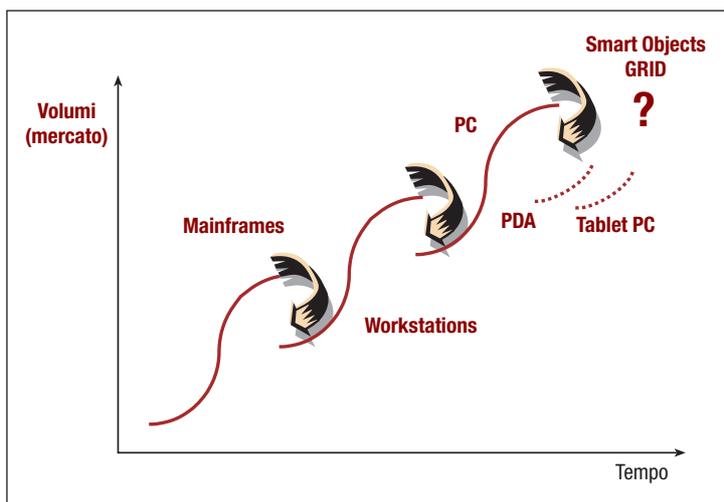
L'evoluzione di questi sistemi ha sostanzialmente cavalcato la legge di Moore, un raddoppio di capacità elaborativa (a costi quasi costanti) ogni 18 mesi; detto in altri termini, ciò significa che nei prossimi diciotto mesi si assisterà a un aumento della capacità

<sup>1</sup> Il termine *Ubiquitous Computing*, in genere, è attribuito a Mark Weiser, direttore scientifico delle ricerche tecnologiche allo Xerox PARC (*Palo Alto Research Centre*). Secondo quanto disse: "ci sarà un'orchestra di oggetti interattivi, non invasivi, dalla presenza discreta, ovunque".

<sup>2</sup> Come non sorridere ricordando la risposta di Watson, capo della IBM, ai suoi ingegneri che proponevano un ingresso nel business dei computer a inizio anni '50: "al mondo ci sarà un mercato per 4 o 5 computer...". O alla affermazione di Ken Olsen, capo della Digital, che venti anni dopo diceva a chi gli proponeva di entrare nel mercato dei PC: "non riesco ad immaginare perché mai una persona dovrebbe volere un PC sulla sua scrivania...". Forse sarebbe il caso di sorridere anche all'obiettivo enunciato da Al Gore: "nel futuro nelle case di tutti gli americani avremo una connessione ad Internet". E sarebbe il caso di sorridere perché nel futuro avremo probabilmente decine di connessioni ad internet in ogni casa, non una!



**FIGURA 1**  
La proiezione degli effetti sul mercato del continuo aumento della capacità elaborativa



**FIGURA 2**  
Le cosiddette "curve S" di penetrazione dei prodotti

elaborativa analogo a quello che si è verificato negli ultimi 35 anni<sup>3</sup>! Ora il punto cruciale non è determinare quando la legge di Moore urterà contro il muro dei limiti fisici [2, 6], piuttosto chiedersi fino a che punto ha senso un aumento della capacità elaborativa dal punto di vista del mercato. Nella figura 1, è possibile vedere una proiezione degli effetti sul mercato del continuo aumento della capacità elaborativa. Fino a che la domanda di capacità elaborativa è stata maggiore dell'offerta (spesso grazie ai gio-

chi e ai programmi di grafica) il prezzo non è sceso in modo considerevole (per i PC di fascia alta, cioè quelli che offrono la maggiore capacità elaborativa).

Tuttavia, la continua crescita di capacità inizia a non essere più accompagnata da una crescita di domanda e, pertanto, si vedrà nei prossimi anni un netto decremento del costo della capacità elaborativa (la crescita della fotografia e cinematografia digitale a livello *consumer* rallenteranno questa caduta). Non si deve, comunque, pensare che tutta la scatola PC crolli di prezzo in quanto questa non contiene solo il *chip* di elaborazione ma diverse altre componenti. Quello che crolla è il prezzo del microprocessore. E questo rende possibile il suo utilizzo in moltissimi altri dispositivi, così come la caduta del prezzo dell'8080 lo ha fatto inserire in oggetti come serrature di alberghi e controlli per innaffiare il giardino.

La vendita di PC si è stabilizzata intorno ai 130 milioni di pezzi all'anno<sup>4</sup>, la vendita dei palmari è intorno agli 11 milioni di pezzi e quella dei telefonini intorno ai 400 milioni. In Italia, i PC sono circa 13 milioni di cui circa 6 milioni sono quelli nelle famiglie, una diffusione, quindi, intorno al 28% delle famiglie italiane. Negli USA la penetrazione si sta assestando sul 50% e si sono iniziati a notare segni di un allungamento della vita media del PC (le persone sono meno motivate a cambiarlo dato che le prestazioni offerte sono ritenute sufficienti) e di una raggiunta maturità del mercato per le connessioni a Internet che sta mostrando segni di flessione.

I PC hanno rappresentato un elemento di forte discontinuità nel mercato, nell'uso e nelle tecnologie (si pensi al passaggio ad architetture aperte come il concetto di *bus* che ha creato un nuovo mondo di industrie).

Nel grafico della figura 2 si riportano le discontinuità che si sono verificate nel settore della capacità elaborativa, le cosiddette curve S di penetrazione dei prodotti e come ciascuno abbia in qualche modo "spiazzato" l'altro. Ciò che è interessante notare è il fatto che il parametro che porta a un effetto di disconti-

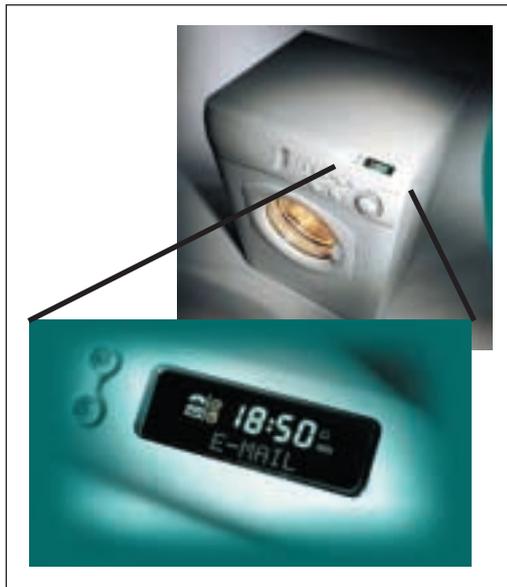
<sup>3</sup> Dal 1968 al 2003 si è passati da 4 MHz a 2 GHz, nei prossimi 18 mesi si passerà da 2 a 4 GHz.

<sup>4</sup> Rapporto Assinform 2003; PC: 130 nel 2000, 123 nel 2001, 128 nel 2002, Palmari: 11 nel 2000, 12.8 nel 2001, 11.7 nel 2002; telefonini: 410 nel 2000; 390 nel 2001, 405 nel 2002.

nuità non riguarda tanto le prestazioni elaborative, e quindi il progresso della tecnologia in termini prestazionali, quanto piuttosto i volumi. Una delle implicazioni di questo è che i PDA non rappresentano, ne rappresenteranno, un elemento di discontinuità ma semplicemente un satellite nel *big business* della capacità elaborativa. Una trasformazione (attualmente, improbabile<sup>5</sup>) del telefonino nel senso di *gateway* verso capacità elaborativa potrebbe viceversa avere un effetto dirimente dati i volumi in gioco. La prossima discontinuità sarà probabilmente rappresentata dagli *smart object* e poi, forse, dal GRID.

## 2. A COSA SERVE UN FRIGORIFERO O UNA LAVATRICE CON UN PC EMBEDDED?

Il quesito è più che pertinente. La Ariston ha messo sul mercato Margherita (Figura 3) una lavatrice che a tutti gli effetti ha un PC all'interno. È possibile programmare in modo sofisticato i vari lavaggi, è possibile collegarla a una LAN (*Local Area Network*) domestica e permettere il dialogo con il frigorifero, è possibile collegarla a Internet e ricevere, e inviare, *e-mail* mentre si fa il carico del bucato. Perché collegare la lavatrice a una LAN in modo che si parli con il frigorifero? Non è un'idea così balzana. La connessione tra i diversi elettrodomestici consente di minimizzare i consumi di energia, di mantenere il consumo complessivo entro i limiti di potenza del contratto e, dal punto di vista del fornitore di energia, permette di minimizzare gli sfasamenti dovuti alle induttanze. Se, per esempio, il frigorifero si coordina con la lavatrice non farà partire il suo motore mentre è attivo quello della lavatrice (diminuendo, così, lo sfasamento induttivo). Se la lavatrice è in fase di riscaldamento dell'acqua, la lavastoviglie aspetterà prima di procedere a scaldare la sua evitando, in questo



**FIGURA 3**

*La lavatrice Margherita della Ariston può interagire con gli altri elettrodomestici*

modo, di far scattare l'automatico per il supero di potenza richiesta. Il collegamento a Internet non serve, ovviamente, a soddisfare l'impulso di leggersi le *mail* mentre si carica il cestello, quanto piuttosto a segnalare eventuali anomalie a un centro servizi permettendo una pianificazione di intervento (magari da remoto) prima che si manifesti un disservizio<sup>6</sup>. Nel giro di qualche anno, quella stessa connessione a Internet potrà essere utilizzata per richiedere istruzioni su come lavare un certo capo di abbigliamento che la lavatrice riconoscerà dalla *tag* RFID (*Radio Frequency Identification*)<sup>7</sup> che questo ha nelle trame del tessuto o magari per rimuovere un certo tipo di macchia rilevata da appositi sensori nel cestello. Ma quanto costa dotare gli elettrodomestici di un *computer*? Uicom [7] offre un microprocessore ai costruttori di elettrodomestici (dai proiettori di lucidi a sistemi di ventilazione e riscaldamento) a 13 dollari. Oltre a essere in grado di elaborare informazioni il microprocessore, ha anche la possibilità di collegarsi in rete. Questo permette ai centri-servi-

<sup>5</sup> La scarsa probabilità deriva dal limite posto dalle batterie e dalla percezione di funzionalità insita nell'uso del telefonino. Questo serve per parlare, non per gestire applicazioni di elaborazione.

<sup>6</sup> Non solo. C'è anche il proposito di trasformare la lavatrice da un prodotto a un servizio, un *pay per use* che permette un dialogo continuo con il cliente...

<sup>7</sup> Entro il 2006 il costo di questo tipo di *tag* dovrebbe essere paragonabile ai codici a barre che identificano i prodotti oggi e si dovrebbe assistere a una loro massiccia diffusione.

zio di monitorare da remoto il corretto funzionamento di questi sistemi. Emware [3] offre, invece, un *software* che permette ai costruttori di mantenere una visibilità sui loro prodotti quando sono utilizzati dai clienti fornendo loro una supervisione da remoto come servizio aggiunto al prodotto.

La pervasività della capacità elaborativa in ambienti industriali sta rapidamente diventando una realtà<sup>8</sup>. La progressiva disponibilità di sensori intelligenti (in grado, cioè, di elaborare localmente le informazioni e di comunicarle a un ambiente che si auto-configura) spingerà ulteriormente in questa direzione.

La Microsoft ha lanciato il programma SPOT (*Smart Personal Object Technology*) in cui prevede che una molteplicità di oggetti disporrà di capacità elaborativa consentendo una loro interazione con l'ambiente, la ricezione di informazioni e il loro processamento. Un portachiavi potrebbe, ad esempio, ricevere via FM (*Frequency Modulation*) le condizioni di traffico ed elaborarle in modo tale che quando si inseriscono le chiavi nella accensione dell'auto si illumini automaticamente un *led* colorato che indichi quale è il percorso con minor traffico. Le informazioni provenienti dall'ambiente vengono elaborate all'interno del portachiavi che ha preregistrati quattro percorsi alternativi per andare dalla casa all'ufficio e viceversa<sup>9</sup>.

Una cornice sul tavolino di un soggiorno potrebbe, invece, contenere una fotografia che cambia a seconda di chi è nell'ambiente, dell'ora del giorno e anche di cosa sta succedendo. Le fotografie sono contenute, in parte, nel chip annegato nella cornice e, in parte, in

qualche computer (o *server*) collegato a Internet<sup>10</sup>. Una di queste cornici posta sul tavolino vicino al telefono potrebbe mostrare le foto del nipote, scattate durante una vacanza, per poi fare apparire la foto dell'amico che sta facendo squillare il telefono in quel momento. La stessa cornice potrebbe mandare le foto anche sullo schermo del televisore quando viene acceso, toccando semplicemente con una mano la cornice e con l'altra il televisore. Piccole cose, che però richiedono notevoli capacità elaborative "distribuite" nell'ambiente e gestite in modo "semplice". Questo è un aspetto fondamentale per il successo dell'Ubiquitous Computing.

Questa semplicità, che poi significa utilizzabilità e bassi costi di esercizio (in soldi o in sforzi di uso), porta direttamente al *focus* di questa panoramica sull'Ubiquitous Computing: il GRID.

### 3. DA INFORMAZIONI AD APPLICAZIONI

Internet era nata per fornire un Ubiquitous Computing. Nei primi anni '60 i computer erano molto cari e le università americane continuavano a chiedere finanziamenti per poterne acquistare uno. Ci si chiese allora perché non realizzare una rete per consentire di accedere da remoto a questi (pochi) computer evitando di moltiplicare gli investimenti. È per rispondere a questa necessità che nasce Internet. L'idea che il valore potesse essere in una rete che condivideva informazioni o che consentisse di scambiare informazioni non era parte della cultura di quegli anni<sup>11</sup>.

<sup>8</sup> Anche settori come quello della salute si stanno evolvendo rapidamente sfruttando la pervasività dei computer. Beckman Coulter, [www.beckman.com](http://www.beckman.com), produce strumenti medicali dotandoli della possibilità di supervisione da remoto, Vivometrix propone delle magliette che inglobano sensori per permettere di seguire l'evoluzione di pazienti appena dimessi dall'ospedale aumentando un questo modo la qualità della cura e diminuendo i costi di ospedalizzazione. Il settore chimico è un altro utilizzatore di questi sistemi ad esempio per verificare lo stato dei silos, così come il settore della distribuzione e della gestione degli inventari di magazzino. In quest'ultimo settore un ulteriore impulso al pervasive computing è dato dalle tag RF ID.

<sup>9</sup> Questo portachiavi è stato realizzato dal Media Lab di Cambridge, USA.

<sup>10</sup> Un servizio di questo genere è offerto da CEIVA, negli USA. Con 149\$ si acquista la cornice e con 5\$ al mese la cornice attinge automaticamente via Internet a foto che stanno su un server e che possono essere "programmate" per apparire in certi istanti.

<sup>11</sup> Anzi: i resoconti di quegli anni indicano che l'idea di condividere i propri dati con altre università venisse visto come il fumo negli occhi da professori gelosi delle loro ricerche. Similmente i realizzatori della prima Internet si stupirono nel vedere che molti utilizzatori cominciavano a utilizzare Internet per scambiarsi messaggi al posto di farsi una telefonata.



Internet ha poi avuto il suo merito maggiore nella condivisione di informazioni. La condivisione gratuita, addirittura, è diventata una specie di religione, ovviamente scontrandosi con molti "atei", come le case discografiche, che insistono sul fatto che la condivisione "non sa da fare". Il concetto dei *property right* è stato portato da Internet a livelli completamente sconosciuti negli anni '80.

A metà degli anni '90 la presenza di migliaia di PC agganciati a Internet e, sostanzialmente, inoperosi per gran parte del tempo ha fatto venire l'idea di utilizzarli per effettuare una elaborazione distribuita dei segnali radio catturati da radiotelescopi per scoprire una eventuale presenza di segnali artificiali emessi da qualche remota civiltà nello spazio cosmico: nasceva SETI@home; era il luglio del 1996, la data che probabilmente si può prendere come nascita di una elaborazione massicciamente distribuita. I calcolatori connessi erano centinaia di migliaia e la capacità elaborativa superava quella dei più grandi supercomputer.

In realtà, erano state molte le iniziative nel settore della elaborazione distribuita e molteplici anche i risultati conseguiti. Tra le tante chi scrive ne ricorda una, in particolare, che non ha avuto successo ma che tra tutte è stata quella più lungimirante e che più, concettualmente, assomiglia al GRID che forse si avrà nel prossimo futuro: TINA.

La *Telecommunication Information Networking Architecture* (TINA) era nata nel 1992 da una serie di incontri tra varie società di telecomunicazioni<sup>12</sup>, che vedevano nell'esistenza delle loro reti e nei molteplici sistemi che si collegavano a questa (usandola o per farla usare), un ambiente distribuito che avrebbe potuto evolvere verso una rete di servizi in cui il costo di ogni nuovo servizio avrebbe potuto essere ridotto drasticamente utilizzando varie funzionalità e servizi già presenti in qualche punto della rete.

Le difficoltà per risolvere i molteplici problemi sul tavolo con requisiti che, da un lato, portavano ad aprire al massimo i sistemi e,

dall'altro, si scontravano con l'esigenza di protezione degli stessi non consentì di raggiungere i risultati che ci si era prefissi.

Negli anni '90, si sono avute altre iniziative che, pur non essendo direttamente collegate a un'idea di GRID, hanno portato avanti numerose tecnologie che oggi sono alla base dell'idea del GRID.

Nel 1993, Mosaic rivoluziona l'accesso a Internet rendendo facile raggiungere le informazioni e introduce il concetto, in rete, del *plug in*: diventa possibile a un fornitore di dati procurare insieme ai dati anche una piccola applicazione che ne consente la gestione. La compatibilità di queste piccole applicazioni con una qualunque piattaforma elaborativa diventa realtà nel maggio del 1995 con il lancio di Java da parte di SUN. I precedenti sforzi con CORBA<sup>13</sup> (*Common Object Resource Broker Architecture*) erano stati importanti ma non avevano portato a una reale convergenza da parte dei costruttori.

A livello applicativo forse il primo esempio di utilizzo di una rete distribuita è quello realizzato da una azienda di avvocati in Arizona che effettua il primo *spam* di tipo commerciale.

Il 1997 segna, tuttavia, il vero inizio del GRID: in quell'anno, inizia lo sviluppo di GLOBUS, un insieme di applicazioni (*toolkit*) che consentono l'accesso ad applicazioni distribuite e viene fondata, inoltre, Entropia Inc. per commercializzare un sistema di utilizzo dei cicli "morti" di PC su di una rete.

Il toolkit GLOBUS contiene 3 protocolli fondamentali per il GRID: il *connection protocol*, il *resource protocol* e il *collective protocol*.

Il **connection protocol** fornisce quell'insieme di "primitive" che consentono a un'applicazione di presentarsi a un'altra, presente in rete, stabilendo una connessione in cui viene autenticata l'identità dei partecipanti oltre a una negoziazione del modo con cui verrà effettuata la comunicazione.

Il **resource protocol** consente a un'applicazione di scoprire quali applicazioni di un certo tipo siano in rete, quali di queste siano disponibili e cosa facciano. Allo stesso tempo,

<sup>12</sup> Tra i fondatori del consorzio, che venne formalizzato a inizio del 1993 ma che iniziò a operare nel 1992, ci furono BT, Bellcore, CSELT (Telecom Italia), France Telecom, NTT, TELSTRA.

<sup>13</sup> Nel 1991.

stabilisce un meccanismo tramite il quale ogni applicazione che si rende disponibile in rete riesce a dichiarare che cosa fa.

Il **collective protocol** permette, infine, di aggregare insieme più applicazioni distribuite in rete trattandole, dal punto di vista di chi accede, come se fossero un'unica applicazione che fornisce un servizio di alto livello (se si considerano di basso livello quelli forniti dalle singole applicazioni).

Nimrod, un progetto lanciato nel 1994 per utilizzare sistemi altamente paralleli<sup>14</sup> (non necessariamente distribuiti), viene portato sul GRID utilizzando il toolkit GLOBUS e a settembre di quell'anno viene organizzata dall'*Argonne National Lab* (in Illinois, vicino a Chicago) un *workshop* che ha l'obiettivo di costruire un GRID computazionale.

A questo punto, per maggior chiarezza di esposizione conviene definire i diversi tipi di GRID. Ce ne sono almeno 3: il P-GRID, il D-GRID e l'A-GRID.

Questa nomenclatura non è ancora standard e viene utilizzata da chi scrive per comodità di presentazione. In effetti, il GRID è ancora un oggetto in parte inesplorato e anche i vari addetti ai lavori hanno idee diverse su ciò che sia. Una classificazione, anche se non universalmente accettata, può quindi essere utile.

### 3.1. Il Processing GRID

Il P-GRID, (o Processing GRID), si propone di offrire una infrastruttura che fornisca, a chi vi accede, una capacità computazionale su richiesta (teoricamente grande quanto è la somma delle capacità delle macchine collegate in un certo istante). Si può dire che questo sia il tipo di GRID più semplice da realizzare e, in effetti, utilizzi di questo tipo sono stati i primi ad apparire. Esistono due

sfumature, non di poco conto, di questo tipo di GRID. Una si può definire il supercomputer GRID, cioè una infrastruttura che mettendo a fattor comune molte capacità elaborative singole è in grado di erogare una capacità enorme, equivalente, se non superiore, a quella di un supercomputer.

Il tutto a prezzi significativamente inferiori, così come mostrato nella tabella 1 estrapolata da un'analisi [5].

IBM si è in qualche modo candidata a essere un punto di riferimento su questa concezione di P-GRID offrendo sia servizi di capacità di elaborazione *on demand* sia vere e proprie strutture assimilabili come capacità ai supercomputer<sup>15</sup>.

L'altro tipo di P-GRID, che si può definire di *processing commodity*, consiste nell'offerta di capacità elaborativa come se questa fosse acqua che sgorga aprendo il rubinetto. Non si tratta, quindi, di grandi capacità ma di piccole "portate" rese però disponibili a un numero potenzialmente enorme di richiedenti. Non si è, concettualmente, lontani dall'ormai lontanissimo *time sharing* degli anni '60, e neppure dal concetto di *Network Computer* proposto da SUN negli anni '90. Probabilmente non è una coincidenza casuale che sia ancora SUN a proporre questo tipo di GRID<sup>16</sup>.

Il Network Computer era una bella idea: si potevano abbattere i costi di possesso dei PC, e soprattutto quelli che derivano dal mantenerli aggiornati e funzionanti, utilizzando un terminale di basso costo che consentiva comunque di immagazzinare dati e fare elaborazioni sofisticate appoggiandosi alla rete, che a questo punto diventava "il computer". L'idea non ha avuto successo sostanzialmente perché nel frattempo il costo dei PC è sceso al di sotto del costo di questi

<sup>14</sup> Si pensi a una applicazione che deve essere applicata a vari insiemi di dati, per esempio studiare come la chiglia di una nave si comporta in presenza di una varietà di tipi di onde e di carico...Coppa America insegna! <http://www.csse.monash.edu.au/~davida/nimrod/>.

<sup>15</sup> In questo caso le richieste di capacità devono essere fatte con un anticipo di almeno un mese per predisporre l'infrastruttura. Un tempo lungo ma sempre molto più breve di quello che passa dall'ordine di un supercomputer alla sua consegna e attivazione, e il tutto a costi molto più bassi. La stima fornita da IBM sulle dimensioni di questo mercato è dell'ordine di una sessantina di aziende in USA che possono essere interessate a utilizzare la potenza elaborativa equivalente a quella di un supercomputer.

<sup>16</sup> L'offerta è chiamata Progetto Orion dalla SUN.

	Capacità			Costo (\$)
	Locale	In rete	Totale	
<b>Soluzione Tradizionale</b>				
Base	500	0	500	2.818.728
Media	1.000	0	1.000	5.037.456
Grande	2.000	0	2.000	9.474.912
<b>Soluzione GRID</b>				
Base	100	400	500	2.753.550
Media	100	900	1.000	3.179.921
Grande	100	1.900	2.000	3.073.328

**TABELLA 1**  
I vantaggi economici del P- GRID

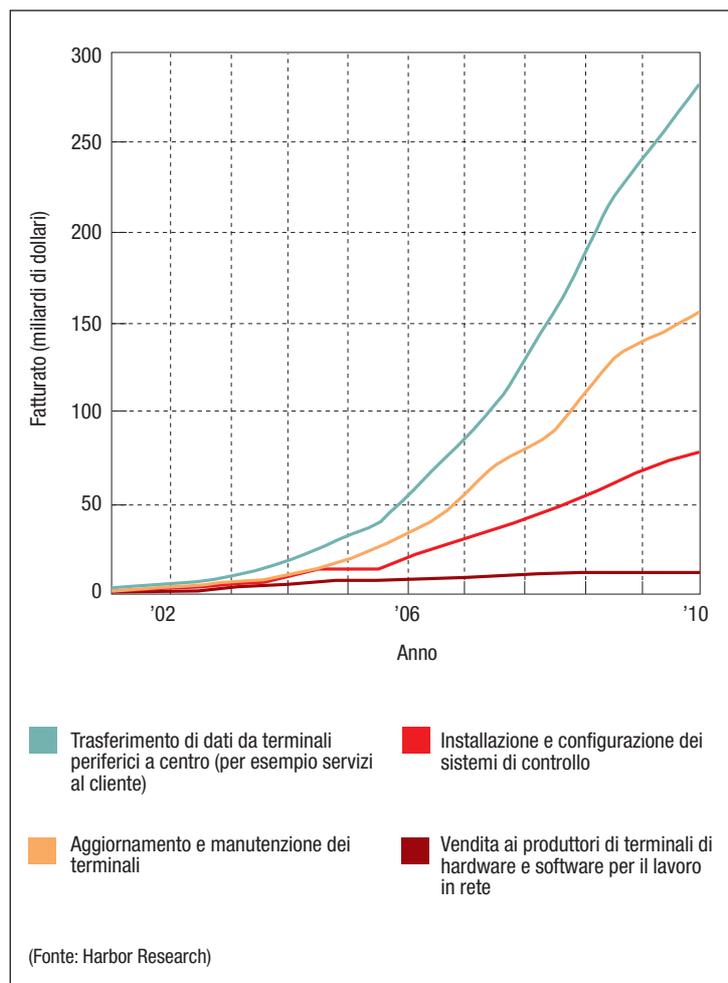
mini terminali facendo perdere molto dell'interesse originariamente suscitato.

Nella figura 4, un'analisi della *Harbor Research*, mostra i ricavi attesi dalla connessione di una molteplicità di oggetti a una rete che fornisce la capacità elaborativa. Si noti il volume dei ricavi conseguenti alle funzionalità di mantenimento e aggiornamento di questi oggetti effettuato tramite servizi di rete.

Se questa stima si dimostrerà vera, SUN avrà fatto la scommessa giusta. IBM, per contro, entra ovviamente in questo tipo di mercato passando dagli oggetti "più grandi" e, quindi, dalla porta dei servizi alle aziende medio grandi.

Esistono, attualmente, diversi progetti che si basano sul P-GRID.

Lo *Smallpox Research GRID Project* [4] vedrà impegnati IBM, United Devices e Accelrys, oltre alla partecipazione di ricercatori delle università di Essex e Oxford in Inghilterra ed esperti del *Robarts Research Institute* e dello *Sloan Kettering Cancer Center*. I due milioni di PC che si prevede di utilizzare saranno, invece, forniti da volontari che scaricando un piccolo applicativo renderanno disponibile la potenza elaborativa che non usano via Internet. Questa potenza è stimata in circa 1.100 teraflop (oltre un milione di miliardi di operazioni al secondo), equivalente a 30 volte la capacità elaborativa del più veloce supercomputer oggi esistente. Questo approccio è di particolare interesse in quanto permette di effettuare ricerche a basso co-



**FIGURA 4**

Il fatturato potenziale derivante dall'uso pervasivo dei computer. Gli analisti prevedono una crescita esplosiva del numero di terminali connessi a Internet nella seconda metà del decennio 2000-2010.

sto e, quindi, di rendere possibile trovare soluzioni a problemi che hanno un basso ritorno economico (il vaiolo è tecnicamente sradicato e la paura deriva solo da un suo risorgere a seguito di impiego di armi chimiche). Si aprono, dunque, prospettive concrete per lo studio di molte patologie che affliggono molte persone in Paesi in via di sviluppo (soprattutto in Africa) e che oggi le aziende farmaceutiche non affrontano in quanto i ritorni economici sarebbero insufficienti a ripagare le spese.

A fine novembre 2002, il progetto CISS (*Canadian Internetworked Scientific Supercomputer*) ha annunciato l'interconnessione di 1360 computer distribuiti in 21 città: questi, per un giorno, il 4 novembre scorso, hanno fornito una potenza di calcolo equivalente a quella ottenibile dal quinto supercomputer al mondo. La potenza elaborativa messa a disposizione è stata utilizzata per modellare l'interazione tra due molecole in 20.000 diverse configurazioni misurando i legami energetici che si formano. Questo calcolo, se effettuato tramite un PC di ultima generazione, avrebbe comportato un tempo di elaborazione di più di 6 anni. La modalità di aggregazione e di elaborazione parallela utilizzata dal progetto CISS, condotto dalla Università di Alberta è di tipo proprietario e costruita appositamente per affrontare certi tipi di problemi. Diversa, quindi, da quella sviluppata per il GRID che si basa sul concetto di interfacce standardizzate ma pur sempre un esempio di GRID.

Il progetto [Folding@Home](#) ha fornito nuovi indizi su come risolvere il problema del ripiegamento delle proteine. Le proteine iniziano la loro esistenza sotto forma di una lunga catena di amminoacidi all'interno di una cellula, e poi si ripiegano per raggiungere la loro forma caratteristica. Questo processo può richiedere tempi anche molto diversi, da qualche nanosecondo fino a varie decine di microsecondi, a seconda di quali e quanti passaggi sono necessari. Anche se l'intero processo è incredibilmente veloce, un computer moderno ha bisogno di anni per simulare anche un solo microsecondo. Per risolvere questo problema, un gruppo di ricercatori ha tentato un approccio diverso. Piuttosto che simulare le proteine per vari microsecondi

consecutivamente, Vijay Pande e i suoi colleghi della Stanford University hanno studiato per periodi brevi, da 5 a 10 ns, una catena di 23 amminoacidi chiamata BBA5. Le probabilità di vedere il ripiegamento in un simile tempo sono molto scarse, così i ricercatori hanno assegnato il compito a più di 30.000 calcolatori distribuiti in tutto il mondo, che hanno provato le brevi simulazioni. Mettendo insieme un tempo di calcolo equivalente a 2000 anni si è previsto che la proteina dovrebbe ripiegarsi in circa 10  $\mu$ s. Per verificare questa previsione, Martin Gruebele, dell'Università dell'Illinois ha usato dei laser per costringere la proteina a dispiegarsi e ha poi misurato il tempo richiesto per il ripiegamento usando amminoacidi fluorescenti, visibili quando la proteina è in forma lineare e vengono nascosti uno a uno durante il processo. Il tempo misurato, 7,5  $\mu$ s, è un buon accordo con la previsione numerica. Il successo del progetto è di stimolo alla creazione di reti di applicazioni, come il GRID, ed è ovviamente una prospettiva di grande interesse per gli operatori di telecomunicazioni che possono sfruttare le loro reti come infrastrutture di elaborazione.

L'efficacia dimostrata da progetti come questi che sono stati citati sta spingendo anche verso una messa in rete dei grandi sistemi di elaborazione che, quindi, possono mettere a disposizione di molti la loro potenza elaborativa. In un certo senso, questo è più vicino al concetto di capacità elaborativa come commodity.

L'IBM ha ricevuto una commessa dalla Germania per realizzare un supercomputer che non sarà dedicato a una singola organizzazione ma entrerà a far parte del GRID, sarà quindi accessibile da chiunque (o quasi). Questo supercomputer avrà una capacità elaborativa di solo 5.800 miliardi di operazioni al secondo utilizzando 37 eServer IBM (contro le 7300 del supercomputer USA basato su 44 server).

### 3.2. Il Data GRID

Il D-GRID (o Data GRID) si propone di rendere possibile la condivisione di enormi quantità di dati che possano essere analizzati in parallelo influenzandosi a vicenda. Questa possibilità di influenza reciproca è la chiave che distingue il Data GRID dal *web* dove, in effetti,

esistono enormi quantità di dati che sono condivisi da milioni di navigatori.

Il Data GRID nasce dal CERN [1] per permettere a circa 8000 ricercatori distribuiti in centinaia di centri di ricerca e università di accedere ai dati che saranno prodotti dal LHD (*Large Hadron Collider*).

A luglio 2003, il Data GRID che è stato costruito a partire da metà 2002 collegando 15 nazioni, sarà aperto ai ricercatori. Questo Data GRID [1] contiene al momento 5 Tbyte di dati; a regime questa rete fornirà una capacità elaborativa equivalente a 200.000 PC e una capacità di memoria in grado di memorizzare fino a 8 Pbyte (8 milioni di Giga/byte) all'anno.

Al momento, le difficoltà di ottenere un insieme omogeneo a cui attingere per disporre di queste capacità sono ancora notevoli. Circa il 20% delle attività non riesce a concludersi per problemi di incompatibilità tra i diversi sistemi. In questa fase, il GRID non sarà disponibile a chiunque ma solo a quelle organizzazioni che si impegnano a fornire garanzie sui loro sistemi e sulle procedure di accesso.

La Oracle ha annunciato l'intenzione di rendere disponibili interfacce (*GRID development kit*) che consentano l'interfacciamento dei suoi *data base* al GRID. La loro speranza è che i *data base* Oracle possano divenire il sistema di memorizzazione di eccellenza per il GRID. Questo è interessante vista la diffusione di questi sistemi a livello delle imprese che, in questo modo, si troverebbero potenzialmente inserite nel GRID. La visione di Oracle è quella in cui i clienti non sono più interessati a sapere dove stia il server con i dati e neppure il computer su cui vengono effettuate le elaborazioni. Electronics Arts ha sviluppato un gioco di realtà virtuale basato su queste nuove interfacce di Oracle e sulla architettura GRID consentendo l'accesso contemporaneo a 100.000 giocatori.

A inizio ottobre 2002, la Oxford University ha annunciato l'utilizzo del GRID per il progetto *eDiamond*, un progetto iniziato a dicembre del 2001, per aggregare e rendere disponibili informazioni mediche relative alla cura del tumore al seno. Il nome del progetto vuole sottolineare come questo presenti molte

sfaccettature tra cui quella di servire la ricerca avanzata e quello del supporto alla cura giorno per giorno.

Il progetto fa parte di un programma più ampio, ([www.research-councils.ac.uk/escienze](http://www.research-councils.ac.uk/escienze)), che si propone di rendere accessibile: capacità elaborativa, informazioni scientifiche e strumentazioni sperimentali così come avviene per le informazioni su Internet. Per questo viene utilizzato il GRID. L'eDiamond rende possibile non solo l'accesso (a istituzioni, ricercatori, medici e pazienti) a informazioni come mammografie, analisi citologiche e quant'altro in forma che ne garantisce la riservatezza, mette anche a disposizione strumenti di analisi per indirizzare verso la migliore cura. Applicazioni software presenti sul GRID consentono di effettuare analisi comparate di migliaia di situazioni patologiche e delle relative cure derivando da queste le indicazioni sul miglior approccio terapeutico. Interessante, dal punto di vista tecnologico ed economico, il fatto che questo progetto per la prima volta si avvalga di applicazioni commercialmente disponibili la cui utilizzabilità ed efficacia è moltiplicata dalla loro interconnessione tramite GRID.

### 3.3. L'Application GRID

L'A-GRID (o Application GRID) si propone di rendere disponibili applicazioni presenti sul GRID a chiunque ne abbia bisogno (se questa disponibilità venga fornita gratuitamente o a pagamento è indipendente dal meccanismo che fornisce anche la possibilità di negoziare il costo per l'utilizzo). Non viene solo messa a disposizione un'applicazione residente in qualche punto del GRID che opererà in modo remoto da questo punto, ma viene anche resa possibile l'integrazione di più applicazioni per ottenere il servizio di elaborazione desiderato. Si supponga, per un momento, di essere un ricercatore che vuole verificare l'efficacia di una certa medicina che sta progettando simulando il suo comportamento nel corpo umano. Questo ricercatore esprimerà quello che vuole ottenere tramite una applicazione di interfaccia sul GRID. Il GRID (il *resource protocol*) andrà a cercare se esistono delle applicazioni che possono fornire il servizio richiesto o almeno una parte di questo. È

quasi certo che non esisterà un tale servizio (è una nuova medicina) ma è altrettanto certo che esisteranno diverse applicazioni che possono essere utilizzate per affrontare parte del problema. Per esempio, un'applicazione che può simulare come quel particolare tipo di medicina può essere veicolato (dalla pillola alle molecole di medicinale che raggiungono le cellule), un'altra che permette di studiare il meccanismo di penetrazione attraverso la membrana cellulare di cellule *target* e di altre cellule (che non devono essere attaccate), un'altra che studia la denaturazione delle sostanze medicinali e la loro espulsione, e così via. La creazione di un'applicazione distribuita a partire da un certo insieme di componenti (ciascuno trovato tramite il resource protocol) è la chiave dell'A-GRID. Questo è realizzato tramite una interazione tra la modalità di descrizione di ciò che si vuole e il collective protocol che mette insieme quanto è stato trovato sul GRID. Questa interazione è molto complicata e, attualmente, rappresenta il maggiore scoglio a una effettiva utilizzazione dell'A-GRID. Manca, in altre parole, quello che HTML (*Hyper Text Markup Language*) e *browser* fanno per la condivisione efficace delle informazioni sul *www* (*World Wide Web*). In effetti, l'esempio del ricercatore nel settore farmaceutico non è un esempio ipotetico ma reale e descrive quanto si è iniziato a fare. Il problema è che questo è possibile in quel settore perché, da un lato, si ha un ricercatore che è preparato a descrivere il suo problema con dei linguaggi che non sono alla portata di tutti e dall'altro è stato delimitato in modo preciso il settore con un accordo tra chi opera nella descrizione delle applicazioni.

Un primo esempio di A-GRID, noto a tutti, è quello collegato ai giochi *on-line*. In questo caso, la distinzione tra P-GRID, D-GRID e A-GRID non è semplice ma tendenzialmente si può dire che questo tipo di utilizzo va nella direzione dell'A-GRID.

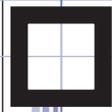
Sony insieme a IBM e Butterfly svilupperà un GRID per la PlayStation 2. Questa evoluzione è in linea con i tempi che queste aziende avevano dichiarato qualche mese fa a inizio progetto. Portare il GRID nel campo dei giochi significa renderne l'utilizzo più semplice ed

economico e questo contribuirà alla sua affermazione. Inizialmente, l'obiettivo è quello di aggregare capacità elaborativa in cui far vivere le comunità dei giochi "massicciamente" distribuiti. I passi successivi, tuttavia, sono nella direzione di permettere a ciascun giocatore di fruire di applicazioni disponibili nella comunità del gioco per ...giocare meglio!

Un altro esempio, forse sorprendente, è quello di fornire una applicazione per aumentare la velocità di trasporto delle informazioni.

A fine novembre 2002, è stato raggiunto un nuovo record di velocità di trasmissione in un collegamento tra il Giappone e gli Stati Uniti. La velocità di trasmissione sulla fibra si sta avvicinando ai 10 Tbit/s (ovvero, 10 mila miliardi di bit al secondo). Il record annunciato riguarda una velocità di 707 Mbit/s: sembra un passo indietro! In realtà, questo annuncio è "una notizia" in quanto occorre distinguere tra la velocità bruta di trasporto ottenibile su di una fibra, che come si è detto ha raggiunto i Terabyte al secondo, e quella utile realizzata tra due applicazioni che per scambiarsi dati devono utilizzare una varietà di *link* in sequenza. È opportuno, a questo punto, "fare due conti". La trasmissione di un pacchetto di dati su una distanza di 10.000 km, e il riscontro che il pacchetto è stato ricevuto, richiede all'incirca due decimi di secondo. Il tipico pacchetto ha una dimensione di 64 kbit per cui in un secondo si riescono a inviare solo 320 kbit su un singolo *stream* Internet. Multiplandone diversi si raggiungono velocità dell'ordine dei 2.5 Mbit/s. Velocità più alte richiedono la parallelizzazione di stream da parte di molti elaboratori a entrambe le estremità. È quanto hanno realizzato in questo caso i ricercatori utilizzando le capacità del GRID di eseguire elaborazioni distribuite. I 7 *cluster* di PC all'Istituto di Tecnologie di Tokyo e negli USA hanno copiato nelle diverse sedi varie centinaia di Gigabyte di dati e, quindi, hanno lavorato insieme su un complessivo di 18 Tbyte.

L'interessante, dal punto di vista dell'A-GRID, è che questo potrebbe essere utilizzato per aumentare la capacità di trasmissione tra due applicazioni su di una rete con la curiosa proprietà che in un certo senso gli instradamenti sono gestiti all'esterno della rete.



#### 4. UBIQUITOUS COMPUTING: LE SFIDE PER IL SUCCESSO

Questa carrellata sull'Ubiquitous Computing è stata in realtà effettuata a cavallo tra passato, presente e futuro dando per scontato che molte delle sfide che ancora attendono questo settore saranno superate.

Vale la pena, pertanto, concludere proprio facendo una riflessione sui problemi che devono essere ancora risolti per far progredire l'Ubiquitous Computing e anche quelli che potranno essere generati da questo.

La crescita tumultuosa delle applicazioni, della capacità elaborativa e la diminuzione del costo oltre ai processi produttivi che consentono di impacchettare dei PC in qualunque oggetto con costi marginali portano in modo naturale ad ambienti in cui l'intelligenza è locale. L'evoluzione delle telecomunicazioni e dell'accesso porta alla interconnessione di questi ambienti e potenzialmente ne rende possibile il controllo da remoto. Tuttavia, queste due forze, una centrifuga (elaborazione locale) e una centripeta (telecomunicazioni), non si bilanciano, e quella centrifuga tenderà a prevalere. Tutto sommato que-

sta dovrebbe essere una buona notizia (o un'ottimistica previsione) per chi teme l'avvento del Grande Fratello.

La localizzazione della capacità elaborativa porta con sé il problema della gestione. Nuove architetture e strutture sono necessarie. L'evoluzione verso gli *autonomic system*<sup>17</sup> è un passo in questa direzione.

Proiettandosi ancora più nel futuro l'esistenza di strutture elaborative altamente distribuite e comunicanti potrebbe cambiare radicalmente quella che oggi si considera come rete di telecomunicazione. Infatti, se ciascun telefonino diventa al tempo stesso un terminale, un gateway e un punto di commutazione la rete scompare, sostituita dalla maglia di decine di migliaia di telefonini che comunicano direttamente tra loro (o sfruttando altri telefonini come punti di transito). Questo tipo di reti (*ad hoc network*) troverà una prima applicazione a livello delle reti di sensori che costituiscono reti locali e potrà espandersi a livello di reti metropolitane (questa volta con i telefonini) dopo il 2010<sup>18</sup>.

Un'applicazione più futuribile, ma forse an-

<sup>17</sup> La Fujitsu Siemens Computer ha presentato al CeBIT la propria strategia di sviluppo e implementazione degli Autonomic Systems, denominati SysFrame, capaci di monitorarsi e ottimizzarsi in modo pressoché indipendente. L'idea alla base del sistema SysFrame parte dallo studio di come un organismo vivente automaticamente reagisce di fronte ai cambiamenti che lo riguardano e che riguardano l'ambiente circostante. Questo concetto applicato all'IT, significa sviluppare funzioni di self-management per piattaforme IT. Queste funzioni comprendono quelle di auto configurazione, auto ottimizzazione, auto recovery e auto protezione.

SysFrame copre l'intera offerta orientata ai sistemi critici per il business, Primergy (sistemi Intel), Primepower (sistemi Solaris), BS2000/OSD mainframe e integra i sistemi desktop e laptop.

IBM, il 4 aprile, ha parimenti annunciato l'introduzione di un *blueprint* per aiutare i suoi clienti a sviluppare autonomic systems. L'impegno IBM in questo settore è nato nel 2001 con la costituzione di un apposito gruppo di ricerca e si inquadra nella sua iniziativa eBusiness on Demand.

<http://www-3.ibm.com/autonomic/index.shtml>

Al di là dell'annuncio di Fujitsu e IBM questo tipo di sistemi crescerà nel futuro, andando anche oltre l'area dell'IT ed entrando sicuramente in quella delle telecomunicazioni. Inoltre, gli Autonomic Systems rappresentano una evoluzione nella direzione della capacità elaborativa vista come "commodity" (al pari dell'elettricità, dell'acqua) in sinergia con il GRID.

Ci si attende una diminuzione dei costi di esercizio e manutenzione dei sistemi informatici oltre alla capacità di gestire livelli di complessità maggiori quali quelli che deriveranno dalla moltiplicazione dei processori, interconnessi tra loro, applicati a una miriade di oggetti.

Quello annunciato rappresenta un primo passo verso sistemi realmente autonomi. Il controllo locale e distribuito avviene tramite tecnologie di tipo predittivo che forniscono le correlazioni tra le diverse componenti del sistema. A queste si aggiungeranno tecnologie cosiddette adattative in grado di variare le strategie delle azioni di risposta al variare del contesto in modo completamente autonomo.

<http://www.fujitsu-siemens.com/atbusiness/1020071.html>

<http://www.mcpressonline.com/mc/.6ae62969!more=1#more>

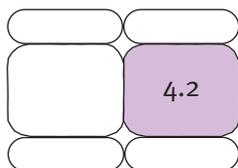
<sup>18</sup> Il fattore limitante è oggi costituito dal consumo delle batterie, troppo grande per rendere pratico un tipo di rete di questo tipo.





# OPEN SOURCE E PUBBLICA AMMINISTRAZIONE

Giovanna Sissa



Imprese, mondo della ricerca, utenti di tecnologie informatiche e telecomunicazioni, Pubbliche Amministrazioni guardano con interesse al software open source e al modello di sviluppo a esso collegato. Da molte parti si ritiene che il paradigma open source possa favorire il rilancio dell'industria informatica nazionale ed europea e il dibattito sul software open source concerne l'impatto che esso può avere sullo sviluppo del Sistema Paese.

## 1. LA STORIA

L'origine di quel complesso fenomeno che va sotto il nome di *Open Source Software* (OSS) è intimamente connessa con la storia del sistema operativo UNIX.

La comunità scientifica creatasi con la prima fase di sviluppo di UNIX quando tramontò l'idea di un suo sviluppo non proprietario aveva maturato al suo interno una forte coesione intellettuale e un modello culturale di collaborazione. Si ricostituì, infatti, parzialmente nella **Free Software Foundation** (FSF) [4] e poi nella **Open Source Initiative** (OSI) [12]. I due movimenti che ne sono derivati, quello del *software libero* e quello del codice a *sorgente aperto*, condividono entrambe l'idea guida della *disponibilità completa e senza limitazioni dei sorgenti*, ma si diversificano, oltre che per la terminologia, per alcuni aspetti filosofici e di principio [8, 9].

I sostenitori del software libero scelgono questo termine per insistere sulle "libertà" associate a un software: tali *libertà* sono dei *criteri*, ma soprattutto delle *motivazioni etiche*. I sostenitori dell'uso del termine "codice

a sorgente aperto" insistono sulle caratteristiche operative, e in particolare sulla disponibilità del codice sorgente, senza tentare di fornire motivazioni etiche.

Fatte salve queste precisazioni, ricche peraltro di implicazioni [4, 8, 12], per non appesantire il lettore meno esperto e per concentrarsi, invece, sulle specificità nella Pubblica Amministrazione, nel seguito di questo articolo si useranno indifferentemente i termini "open source software" e "software libero" per indicare l'insieme delle tematiche e delle proposte che si rifanno alle interpretazioni citate.

## 2. IL CUORE DELL'OPEN SOURCE: LE LICENZE

In funzione del livello di contribuzione richiesto all'utente (gratuità), di disponibilità del codice sorgente (trasparenza) o di importanza data alla nozione di proprietà (diritti d'autore), i software sono designati con i termini di *dominio pubblico*, *liberi*, *freeware*, *shareware* o *proprietary*. È la natura della licenza

Per la **Free Software Foundation** (FSF), un software è considerato libero se la licenza accorda all'utilizzatore le seguenti quattro libertà:

- *Libertà di eseguire il programma per qualunque uso.*
- *Libertà di studiare il funzionamento del programma e di adattarlo ai propri bisogni. Per questo l'accesso al codice sorgente è condizione necessaria.*
- *Libertà di ridistribuire delle copie.*
- *Libertà di migliorare il programma e di pubblicare le modifiche, per farne profittare tutta la comunità di utenti e di sviluppatori.* A tal fine è condizione indispensabile l'accesso al codice sorgente.

La FSF ha creato il concetto di "copyleft". Questo termine definisce una licenza che riprende le quattro libertà suddette e i cui termini devono essere ripresi in modo identico in caso di nuova distribuzione. Ciò permette di evitare che una distribuzione di software modificato restringa i diritti iniziali.

I criteri, invece, dell'**Open Source Initiative** (OSI) che permettono di determinare la natura libera o proprietaria di una licenza di software si articolano in nove punti:

- I** *Libera redistribuzione*
- II** *Codice sorgente*
- III** *Prodotti derivati*
- IV** *Integrità del codice sorgente dell'autore*
- V** *Assenza di discriminazione nei confronti di persone o gruppi*
- V** *Assenza di discriminazione nei confronti di sfere di attività*
- VII** *Distribuzione di licenza*
- VIII** *La licenza non deve essere specifica di un prodotto*
- IX** *La licenza non deve imporre limitazioni ad altri software: ovvero, esigere che gli altri programmi distribuiti sullo stesso supporto fisico siano anch'essi software liberi.*

associata al software a determinare l'appartenenza a una delle categorie suddette. Il concetto di software libero è, quindi, associato all'insieme di software coperti da un tipo di licenza particolare: le licenze di software libero [4, 12].

Giuridicamente, il software libero non è un software senza diritti: esso resta governato dalle disposizioni della licenza e l'autore del software resta il titolare dell'insieme dei diritti di autore.

Le licenze di software libero consistono in una messa a disposizione del software con l'intento di permettere la libera evoluzione del software medesimo. La licenza non ha per scopo il trasferimento di un diritto di proprietà o la rinuncia al diritto di autore o di "far cadere" il software nel pubblico dominio; diffondendo il proprio software libero l'autore può assicurarsi che la libera utilizzazione del software non sia perturbata dalle azioni dei soggetti alla licenza.

Richard Stallman e la FSF [9] definiscono il concetto di *copyleft* in contrapposizione al tradizionale *copyright*: ove il copyright tende a tutelare il diritto d'autore, anche attraverso limitazioni all'accesso della conoscenza, mentre il copyleft intende tutelare il più generale diritto della collettività a fruire dei prodotti dell'innovazione.

I principi del copyleft vengono formalizzati

dalla FSF nella cosiddetta *General Public License* (GPL). Il cliente di accordo GPL è vincolato a utilizzare a sua volta la GPL e dovrà, quindi, fornire il codice sorgente delle estensioni realizzate. In altre parole, la GPL è un modello di licenza ricorsivo. Il codice e le "libertà" a esso associate diventano così legalmente inseparabili.

Il mondo dell'open source non coincide con la GPL. Esistono modelli di licenza alternativi quali ad esempio, LGPL (*Lesser General Public License*), Artistic, BSD (*Berkeley Software Distribution*), MPL (*Mozilla Public License*) ecc., che prevedono, in forme differenti, l'apertura del codice sorgente.

Alcune licenze, come la GPL, vietano la realizzazione di soluzioni proprietarie a partire dal software libero. Altre licenze, come le licenze tipo MIT o BSD, consentono le derivazioni proprietarie. Questa possibilità offerta allo sfruttamento proprietario di una base di codice costituisce la principale distinzione fra le differenti licenze di software libero.

Secondo l'Open Source Initiative trentuno licenze rispondono ai criteri dell'associazione stessa (Tabella 1).

Le licenze OSS non prescrivono che il software debba essere ceduto gratuitamente. Il software open source non è alternativo al software commerciale: il modello OSS non preclude la presenza di distribuzione com-

Apache Software License	<a href="http://www.opensource.org/licenses/apachepl.html">http://www.opensource.org/licenses/apachepl.html</a>
Apple Public Source License	<a href="http://www.opensource.org/licenses/apsl.html">http://www.opensource.org/licenses/apsl.html</a>
Artistic license	<a href="http://www.opensource.org/licenses/artistic-license.html">http://www.opensource.org/licenses/artistic-license.html</a>
BSD license	<a href="http://www.opensource.org/licenses/bsd-license.html">http://www.opensource.org/licenses/bsd-license.html</a>
Common Public License	<a href="http://www.opensource.org/licenses/cpl.html">http://www.opensource.org/licenses/cpl.html</a>
Eiffel Forums License	<a href="http://www.opensource.org/licenses/eiffel.html">http://www.opensource.org/licenses/eiffel.html</a>
GNU General Public License (GPL)	<a href="http://www.opensource.org/licenses/gpl-license.html">http://www.opensource.org/licenses/gpl-license.html</a>
GNU Library or "Lesser" Public License (LGPL)	<a href="http://www.opensource.org/licenses/lgpl-license.html">http://www.opensource.org/licenses/lgpl-license.html</a>
IBM Public License	<a href="http://www.opensource.org/licenses/ibmpl.html">http://www.opensource.org/licenses/ibmpl.html</a>
Intel Open Source License	<a href="http://www.opensource.org/licenses/intel-open-source-license.html">http://www.opensource.org/licenses/intel-open-source-license.html</a>
Jabber Open Source License	<a href="http://www.opensource.org/licenses/jabberpl.html">http://www.opensource.org/licenses/jabberpl.html</a>
MIT license	<a href="http://www.opensource.org/licenses/mit-license.html">http://www.opensource.org/licenses/mit-license.html</a>
MITRE Collaborative Virtual Workspace License (CVW License)	<a href="http://www.opensource.org/licenses/mitrepl.html">http://www.opensource.org/licenses/mitrepl.html</a>
Motosoto License	<a href="http://www.opensource.org/licenses/motosoto.html">http://www.opensource.org/licenses/motosoto.html</a>
Mozilla Public License 1.1 (MPL 1.1)	<a href="http://www.opensource.org/licenses/mozilla1.1.html">http://www.opensource.org/licenses/mozilla1.1.html</a>
Nethack License	<a href="http://www.opensource.org/licenses/nethack.html">http://www.opensource.org/licenses/nethack.html</a>
Nokia Open Source License (NOKOS License) Version 1.0a	<a href="http://www.opensource.org/licenses/nokia.html">http://www.opensource.org/licenses/nokia.html</a>
Open Group Test Suite License	<a href="http://www.opensource.org/licenses/opengroup.html">http://www.opensource.org/licenses/opengroup.html</a>
Python license (CNRI Python License)	<a href="http://www.opensource.org/licenses/pythonpl.html">http://www.opensource.org/licenses/pythonpl.html</a>
Python Software Foundation License	<a href="http://www.opensource.org/licenses/PythonSoftFoundation.html">http://www.opensource.org/licenses/PythonSoftFoundation.html</a>
Qt Public License (QPL)	<a href="http://www.opensource.org/licenses/qtpl.html">http://www.opensource.org/licenses/qtpl.html</a>
Ricoh Source Code Public License	<a href="http://www.opensource.org/licenses/ricohpl.html">http://www.opensource.org/licenses/ricohpl.html</a>
Sleepycat License	<a href="http://www.opensource.org/licenses/ricohpl.html">http://www.opensource.org/licenses/ricohpl.html</a>
Sun Industry Standards Source License (SISSL)	<a href="http://www.opensource.org/licenses/sisslpl.html">http://www.opensource.org/licenses/sisslpl.html</a>
Sun Public License	<a href="http://www.opensource.org/licenses/sunpublic.html">http://www.opensource.org/licenses/sunpublic.html</a>
University of Illinois/NCSA Open Source License	<a href="http://www.opensource.org/licenses/UoI-NCSA.html">http://www.opensource.org/licenses/UoI-NCSA.html</a>
Vovida Software License v. 1.0	<a href="http://www.opensource.org/licenses/vovidapl.html">http://www.opensource.org/licenses/vovidapl.html</a>
W3C License	<a href="http://www.opensource.org/licenses/W3C.html">http://www.opensource.org/licenses/W3C.html</a>
X.Net License	<a href="http://www.opensource.org/licenses/xnet.html">http://www.opensource.org/licenses/xnet.html</a>
zlib/libpng license	<a href="http://www.opensource.org/licenses/zlib-license.html">http://www.opensource.org/licenses/zlib-license.html</a>
Zope Public License	<a href="http://www.opensource.org/licenses/zpl.html">http://www.opensource.org/licenses/zpl.html</a>

**TABELLA 1**

*Le trentuno licenze che rispondono ai criteri dell'Open Source Initiative*



merciale, di fornitori di valore aggiunto o di servizi di supporto.

È corretto definire l'OSS come alternativa al modello di licenza proprietario (*closed source*), in cui l'accesso al codice sorgente non è concesso e il fornitore del software vende all'utente una "licenza d'utilizzo", temporanea o illimitata, che consente l'uso del prodotto, ma non implica in nessun modo che l'utente acquisisca la proprietà del software.

### 3. I MATTATORI DELL'OPEN SOURCE

Il grosso pubblico ha preso familiarità con il concetto di software libero dopo il successo del sistema Linux, un sistema operativo *Unix-like* il cui *kernel* fu realizzato nel 1991 da Linus Torvalds e da un folto gruppo di collaboratori volontari sparsi in tutto il mondo. Questo kernel, in virtù di un set di componenti "liberi" rilasciati sotto la licenza GPL del progetto GNU (*Gnu's Not Unix*), costituisce ora un software completamente libero. Per estensione e semplificazione il sistema viene chiamato LINUX (più esattamente è il sistema GNU/Linux).

GNU/Linux viene distribuito da alcuni facilitatori commerciali, che forniscono servizi di supporto e *package* a basso costo, definiti comunemente "distribuzioni".

Gli esempi più famosi e citati di OSS includono, oltre a Linux, il *server web* Apache [10] che è l'OSS più diffuso, anche in quanto disponibile su vari sistemi operativi: l'ultima rilevazione di *Netcraft Web Server Survey* [11] di febbraio 2003 segnala che Apache ha raggiunto il 66,75% di presenza su circa 35 milioni di siti analizzati.

Da citare, inoltre, BIND, che opera la risoluzione del DNS per l'intera Internet, Sendmail, il più importante e diffuso software di trasporto e-mail su Internet e Openoffice.org, la suite open source di *office automation*.

#### 3.1. I motori della produzione del software open source

Le entità che contribuiscono a livello mondiale alla produzione del software libero afferiscono sia al settore pubblico che privato.

Nel settore pubblico, i primi contributori so-

no stati i laboratori di ricerca e le università, che sono stati poi affiancati dalle amministrazioni centrali e locali. Per un ricercatore di un laboratorio o università, il quadro del software libero permette di diffondere dei lavori innovativi a una ampia comunità o anche di fornire un contributo personale che va ad aggiungersi a un insieme già elaborato. Il contributo del ricercatore resta identificabile e può accrescerne la notorietà. Per le amministrazioni centrali e locali, una delle motivazioni è la messa in opera di interfacce o di strumenti utilizzabili da entità connesse, ma non in rapporto gerarchico.

Un fenomeno diffuso nell'OSS è la cosiddetta *Advocacy*, teorizzata dalle figure di spicco del movimento. Si tratta di una sorta di *marketing ono-to-one*, in base al quale gli utenti dei programmi OSS sono invitati a convincere altri membri del loro collettivo di riferimento a fare altrettanto e ad abbandonare il mondo commerciale. Il software open source costituisce, quindi, un paradigma di sviluppo, di diffusione e di cooperazione nel campo della *Information Technology* (IT).

### 4. STATO DELL'ARTE DELL'OPEN SOURCE SOFTWARE NELLA PUBBLICA AMMINISTRAZIONE

I dati relativi all'uso di OSS nella Pubblica Amministrazione (PA) sono pochi e frammentari. Come già detto, le informazioni più dettagliate riguardano l'uso di Apache come HTTP (*Hyper Text Transfer Protocol*) server.

I report sull'argomento di alcuni analisti, quali Forrester, META, IDC o Gartner, sono contraddittori a causa della complessità del rilevamento delle informazioni e in quanto non esistono dati per il solo settore pubblico.

La Commissione Europea ha promosso alcuni studi sul tema. I più articolati e significativi sono senz'altro *Study into the use of Open Source Software in the Public Sector - An IDA Study - Interchange of Data between Administrations* [2] e *FLOSS (Free/Libre and Open Source Software: Survey and Study)* [7], ai quali molta parte del presente articolo si ispira.

Le amministrazioni pubbliche nei vari stati, e

in Europa, in particolare (Figura 1), hanno propensioni e *policy* sul tema dell'OSS molto diverse: dalla forte adesione di Francia e Germania, al pragmatismo inglese fino allo scarso interesse di altri. L'attenzione sembra, comunque, in ascesa ovunque.

#### 4.1. L'open source in Europa

La citata ricerca dalla Commissione Europea IDA Project [2] riporta i dati di una rilevazione effettuata mediante un questionario compilato dai responsabili dei sistemi informativi delle pubbliche amministrazioni dei paesi comunitari. Anche se la rilevazione presenta limiti quantitativi, come evidenziato nel report stesso poiché si tratta degli unici dati a livello europeo relativi al settore pubblico, tali risultati sono comunque interessanti.

Risulta che l'utilizzo di OSS nella PA è concentrato nella fascia "server", dove Linux spesso completa o sostituisce precedenti versioni proprietarie di Unix. Risulta vincente il duo "Apache/Linux".

Sul lato *client*, invece, le suite di Office Automation OSS costituiscono il fenomeno più interessante, anche se con dimensioni quanti-

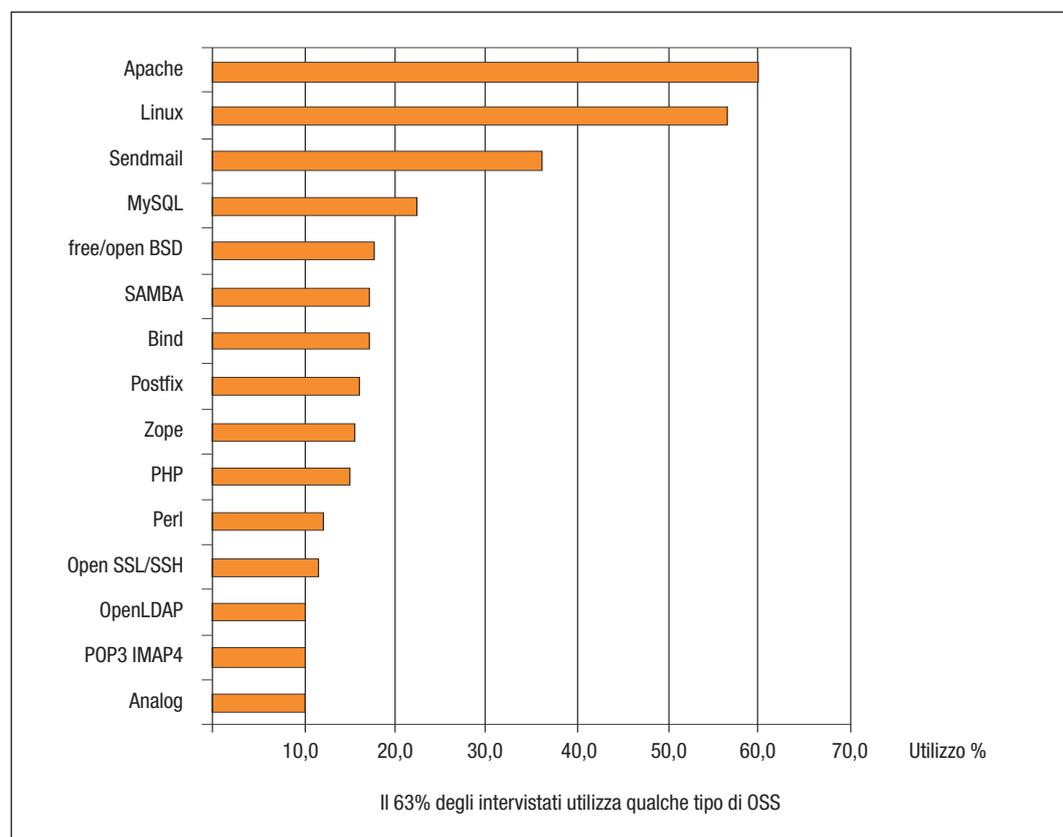
tativamente ben più contenute rispetto a Linux nella fascia server.

#### 4.2. Alcuni dati della Pubblica Amministrazione Locale italiana

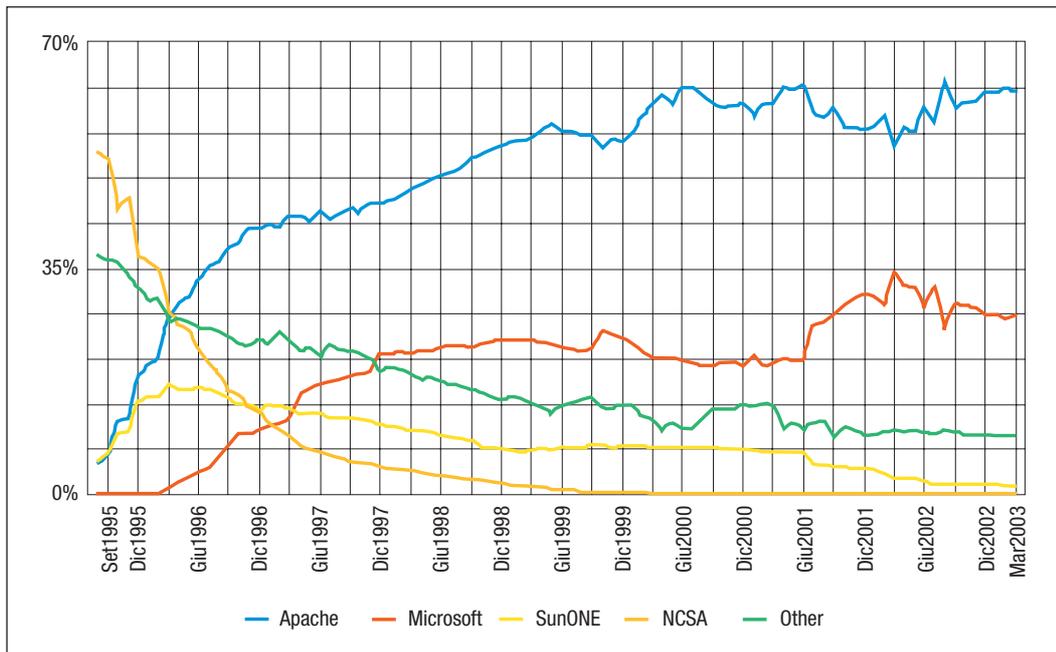
Alcuni ricercatori dell'Osservatorio P.A.O.S. (*Software Opensource nella Pubblica Amministrazione* -Università di Bologna) effettuano una rilevazione sistematica relativa al tipo di server HTTP utilizzato dalle PAL italiane [3] (Figura 2). Tale rilevazione è effettuata "interrogando", per mezzo di una procedura automatizzata, i server relativi a una lista di domini riservati alla amministrazione pubblica locale e riguarda esclusivamente i siti dei nomi di dominio di comuni, province e regioni.

Da un confronto con i risultati della citata rilevazione periodica realizzata da Netcraft sull'intera popolazione dei siti Web mondiale emerge un utilizzo relativamente ridotto dei server (open source) Apache nell'amministrazione pubblica italiana: meno del 40% nella PA italiana contro oltre il 60% nel mondo.

La rilevazione indica una stima della percentuale di adozione dei diversi tipi di server HTTP all'interno di comuni, province e regioni



**FIGURA 1**  
OSS nelle Pubbliche Amministrazioni europee – lato server (Fonte IDA study, European Commission, 2001)



**FIGURA 2**  
Percentuale di server HTTP nella PAL italiana – fonte P.A.O.S.

e quindi, in termini strettamente rigorosi, non è interpretabile come una misura dell'adozione del software open source in generale all'interno di tali amministrazioni.

Secondo P.A.O.S. "Considerato però che i server Apache, con una frequenza di utilizzo a livello mondiale di oltre il 60% (secondo i dati forniti dalla rilevazione di Netcraft), rappresentano uno dei prodotti OS di più facile e ovvio utilizzo, argomentiamo che un'organizzazione che non utilizza neanche un server HTTP OS, mostra in media di possedere una scarsa propensione a utilizzare software OS in generale, e a maggior ragione verosimilmente non farà uso di applicativi OS di più difficile utilizzo, meno documentati e consolidati, rispetto alla famiglia dei server Apache". In questo senso, i dati della rilevazione, pur non riferendosi all'utilizzo del software OS in generale, forniscono un'indicazione utile per valutare la "propensione" dell'amministrazione pubblica all'utilizzo di software open source.

## 5. RUOLO DEL SOFTWARE OPEN SOURCE NELLA PUBBLICA AMMINISTRAZIONE

Nella PA la sfida dell'OSS si gioca sul *software applicativo custom* prodotto dalle amministrazioni pubbliche per far fronte a esigenze amministrative specifiche o, più in generale,

di amministrazione telematica, quali la gestione di strade o ospedali, l'istruzione, il pagamento d'imposte, la giustizia e la gestione del territorio [15].

Il software prodotto da o per le amministrazioni pubbliche non è costituito da "pacchetti standard" che possono essere utilizzati "tali e quali" da altri utenti.

La presunta assenza di finalità commerciali per quanto riguarda i canoni di licenza e la necessità di fornire il software con il codice, per consentire di adattarlo alle realtà locali prima di applicarlo e ridistribuirlo, conducono naturalmente ad aderire al modello OSS [6].

Fra i requisiti della PA vanno ricordati, fra gli altri, *economicità, indipendenza dai fornitori, sicurezza, riusabilità e interoperabilità*.

### 5.1. Benefici del modello open source nella pubblica amministrazione

L'adozione di OSS porta normalmente a un *risparmio iniziale* in termini di costi per licenze. Un confronto economico corretto deve essere però compiuto non solo sulla spesa iniziale, ma tra il TCO (*Total Cost of Ownership*) delle soluzioni open source e il TCO delle soluzioni proprietarie. Oltre al costo delle licenze, nel TCO confluiscono le spese dei servizi di supporto, della formazione, i costi di migrazione, d'installazione e di gestione. Le linee guida del governo inglese evidenziano molto prag-

maticamente il *value for money* come criterio di scelta per il software open source [13]. Un elemento a favore dell'OSS nella PA è l'*indipendenza dai fornitori*, consistente nel poter affidare il supporto di un prodotto open source a un'azienda scelta dal cliente, laddove nel mondo del software proprietario solo il produttore (o un suo partner autorizzato) può supportare il proprio software. Dispone, inoltre, del codice sorgente dei programmi utilizzati all'interno della propria organizzazione permette (anche se non garantisce) un grado maggiore di *sicurezza*. Sono, infatti, più agevoli i controlli interni (ove nei software proprietari ci si deve affidare ai produttori) alla ricerca di eventuali *back door* o debolezze sfruttabili da attacchi esterni. L'amministrazione dispone di un miglior controllo sulla politica di evoluzione del proprio parco applicativo e, in maniera generale, sul governo della gestione del patrimonio pubblico. Questo argomento assume una rilevanza particolare in aspetti connessi con la sicurezza, quali l'autenticazione e l'identificazione del cittadino o quelli relativi all'integrità, confidenzialità

I formati "aperti" e "standard" sono:

1. Un "formato aperto" può essere definito come la "modalità di rappresentazione dei dati in forma elettronica, deliberatamente resa pubblica, completamente documentata ed utilizzabile da chiunque". In questo senso, per esempio il formato utilizzato da "OpenOffice.org" è un formato aperto in quanto:
  - È una modalità di rappresentazione dei dati in forma elettronica.
  - È esaustivamente documentato e utilizzabile da chiunque: i dati vengono rappresentati nativamente in XML (*eXtensible Markup Language*) e salvati come documenti XML la cui struttura è definita in una DTD, grammatica di una classe di documenti XML, pubblica.
2. Un **formato è standard** quando è definito da un ente di standardizzazione (come nel caso di HTML) o è di fatto condiviso da una comunità (come nel caso di PDF, *Portable Document Format*).
3. Un **formato è uno standard aperto** quando soddisfa il requisito di pubblicità e di normazione (per esempio, XML e HTML sono standard aperti perché le loro specifiche sono pubblicamente documentate, definite e mantenute da un ente di standardizzazione, il W3C).
4. Un **formato testo** è un formato appartenente al sottoinsieme dei formati, caratterizzato dalla corrispondenza biunivoca fra un carattere alfanumerico ("c" minuscola) di un determinato insieme (latino, greco, cirillico, arabo, devanagari ecc.) e il valore del gruppo di bit costituenti l'unità di informazione di quel formato (p.e. 1 byte ASCII, 1-6 byte UTF-8 ecc.).
5. I **formati testo aperti standard** consentono l'indipendenza dall'evoluzione tecnologica; pertanto, le informazioni rappresentate con questo formato sono recuperabili anche molto tempo dopo la generazione, senza necessità di pesanti riconversioni. Questa caratteristica è ancor più vera per quei formati come SGML e XML che al dato associano la relativa descrizione (metadato) in linguaggio naturale: fra cinquanta anni, purché sia possibile leggere un *file* di testo con codifica ASCII o UNICODE, sarà sempre possibile recuperare i dati di un documento XML e capirne semantica e struttura.

e all'accessibilità dei dati nel corso del tempo. In generale, il software open source è più adatto a essere personalizzato o esteso come funzionalità rispetto a un software proprietario e quindi *riusato*.

Il ricorso al software libero può fungere da leva per la modernizzazione dei sistemi informatici dello Stato. La possibilità di ricorrere sia a software libero che proprietario aumenta le possibilità di scelta delle amministrazioni e consente [1]:

■ di accedere a un patrimonio considerevole di software spesso di qualità e conforme agli standard;

■ di governare il rapporto costo totale della soluzione/rispondenza ai bisogni attraverso il rafforzamento della concorrenza, allo scopo di mantenere questo rapporto al livello più basso possibile;

■ di governare il software e di avere la possibilità di assicurarne la perennità. In altre parole, l'amministrazione è messa in grado di capire e modificare il software per facilitare la sua integrazione e/o la sua evoluzione.

Cruciale per le Pubbliche Amministrazioni è l'interscambio di dati, da cui scaturisce la necessità di usare **formati aperti e standard**, scelta che assicura:

■ **Indipendenza**. La documentazione pubblica e completa del formato consente l'indipendenza da uno specifico prodotto e fornitore; tutti possono sviluppare applicazioni che gestiscono un formato aperto.

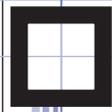
■ **Interoperabilità**. Usando formati aperti standard, sistemi eterogenei sono in grado di condividere gli stessi dati.

■ **Neutralità**. I formati aperti non obbligano a usare uno specifico prodotto, lasciando libero l'utente di scegliere sulla base del rapporto qualità/prezzo.

Inoltre, i *formati testo aperti standard* comportano l'ulteriore beneficio della persistenza, caratteristica importante per la tutela del patrimonio informativo nel tempo a fronte del mutamento tecnologico.

## 5.2. Limiti e rischi

Si noti che l'interoperabilità non comporta anche la portabilità, che invece è un punto a sfavore degli OSS. Le diverse distribuzioni di Linux, per esempio, contengono versioni differenti delle stesse librerie, per cui gli svilup-



paratori hanno difficoltà nel garantire la portabilità delle loro applicazioni.

Sempre a sfavore si citano la bassa compatibilità con standard commerciali.

Infine, la principale difficoltà, come riconosciuto dalla stessa comunità internazionale di utenti Linux, è la mancanza di *driver*: la maggioranza dei produttori di periferiche non forniscono driver per Linux, dunque la lista dell'*hardware* compatibile è limitata ai dispositivi cui la comunità degli sviluppatori open source ha accesso. Quando viene lanciata sul mercato una nuova periferica, occorrono mesi prima che i driver siano disponibili, ammesso che i produttori forniscano le interfacce necessarie per lo sviluppo dei driver stessi; il problema è particolarmente evidente per le schede video e per i *modem*.

Qualsiasi sviluppo e messa in produzione di software comporta dei rischi. L'utilizzazione del software libero, lasciando un più grande margine di manovra agli utilizzatori, necessita anche da parte loro di una comprensione chiara delle nuove implicazioni legate al loro maggior controllo sul software; si enunciano di seguito i principali fattori di rischio [1]:

□ sviluppare utilizzando dei componenti di software libero le cui licenze sono fra loro incompatibili.

La disponibilità del codice sorgente permette tecnicamente di comporre un software utilizzando il codice sorgente dei componenti o dei software completi soggetti a licenze differenti. I responsabili di progetto devono vigilare sulla compatibilità dei componenti al fine di garantire la legalità del prodotto finale.

□ Implicazione dell'amministrazione, in quanto responsabile del software sviluppato o modificato, nella problematica del rispetto del diritto d'autore o della garanzia.

La diffusione pubblica del software realizzato o modificato dall'amministrazione rende necessario assicurarsi che i diritti degli autori siano rispettati e precisare le garanzie che vengono fornite con il software (sia libero che proprietario). La disponibilità del codice sorgente rende più facile la verifica di eventuali violazioni del diritto di autore e richiede, pertanto, una maggiore attenzione all'origine del codice sorgente incorporato nel software.

## 6. CONCLUSIONI

Un IT manager di una PA si domanderà dunque: "In che misura o in che condizioni il software sviluppato da un'amministrazione corrisponde al modello di sviluppo a codice sorgente aperto"?

Se si analizzano i processi di sviluppo, acquisizione, installazione del software e le risorse coinvolte (umane, organizzative, progettuali, finanziarie, distributive e d'informazione), proprie del nuovo modello di sviluppo OSS - definito *bazar*, in contrapposizione al modello tradizionale di costruzione della *cattedrale* [14] - si riscontra che esso si basa sui contributi di migliaia di programmatori che lavorano in modo totalmente volontario, scambiandosi idee e file attraverso Internet, e giungono in assenza di autorità centrale a realizzare programmi complessi.

In realtà, i progetti di successo (uno per tutti Linux) si sono aggregati intorno a figure carismatiche, come appunto Linus Torvalds. La constatazione che i progetti più riusciti e i software più affermati sviluppati nel paradigma OSS siano software di sistema (Sistemi Operativi, web server, *mail system*, linguaggi ecc.) spinge, inoltre, a pensare che tale modello abbia retto poiché di fatto la comunità spontanea degli sviluppatori condivideva le conoscenze dei requisiti e dell'architettura oggetto dello sviluppo [5].

In sintesi, è semplicistico pensare che nella PA sia possibile, senza sforzo iniziale e competenze interne, beneficiare di aggiornamenti e potenziamenti gratuiti, nel quadro di un sistema a codice sorgente aperto, a meno che non siano rispettate precise condizioni di sviluppo, quali [2, 6, 7]:

■ numero ragionevole di operatori "con lo stesso problema";

■ ripartizione iniziale e flessibile dei compiti di gestione/direzione tra persone diverse appartenenti a organizzazioni diverse;

■ documentazione esauriente;

■ niente codici monolitici;

■ software articolato in più segmenti di codice di dimensioni relativamente ridotte, così da agevolarne una gestione individuale;

■ chiara identificazione delle parti mature/stabili e di quelle che vanno ottimizzate in base al principio "dei rilasci frequenti".

D'altronde molte di queste considerazioni valgono anche per gli sviluppi proprietari. Forse il paradigma OSS può contribuire ad applicare buone regole troppo spesso solo enunciate e ad attivare un ciclo virtuoso di sviluppo del software, in un modello di *business* originale e promettente. La partita è aperta e, specialmente nella Pubblica Amministrazione, avvincente.

### Bibliografia

- [1] ATICA (Agency for Information and Communications Technologies in the Civil Service): *Guide to choosing and using free software licences for government and public sector entities*. December 2002.
- [2] European Commission - DG Enterprise: *Study into the use of Open Source Software in the Public Sector An IDA Study - Interchange of Data between Administrations*. June 2001.
- [3] Faglioni G, et al: P.A.O.S - *Osservatorio Software Opensource nella Pubblica Amministrazione*. <http://ei.unibo.it/paos/>
- [4] Free Software Foundation <http://www.fsf.org>.
- [5] Fuggetta A: Open source software: an evaluation. *Journal of Systems and Software*, Vol. 66, Issue 1, 2003, p. 1-90.
- [6] Grasso F: Autorità per l'informatica nella pubblica amministrazione: *Il Software Open Source (OSS) - scenario e prospettive*, Giugno 2002 - n. 10 - supplemento al n. 3/2002 di Informazioni.
- [7] International Institute of Infonomics University of Maastricht, Berlecon Research GmbH Berlin: *FLOSS (Free/Libre and Open Source Software: Survey and Study)*. June 2002.
- [8] Meo A, Berra A: *Informatica solidale*. Bollati Boringhieri, 2001
- [9] Meo AR: Software libero e Open Source. *Mondo Digitale*, Anno I, n. 2, 2002, p.3-17.
- [10] Mockus A, et al: Two Case Studies of Open Source Software Development: Apache and Mozilla. *ACM Transactions on Software Engineering and Methodology*, Vol. 11, n. 3, July 2002, p. 309-346.
- [11] Netcraft survey <http://www.netcraft.com/>
- [12] Open Source Initiative <http://www.opensource.org>.
- [13] *Open Source Software - USE Within UK Government*. Version 1 Date: 15/7/02 [http://www.govtalk.gov.uk/documents/oss\\_policydocument\\_2002-07-15.doc](http://www.govtalk.gov.uk/documents/oss_policydocument_2002-07-15.doc).
- [14] Raymond E: *The Cathedral and the Bazaar*. Paperback edition, 2001.
- [15] Russo P, Sissa G: *Il Governo Elettronico*. Apogeo 2000.

GIOVANNA SISSA Ha partecipato ai lavori della "Commissione sul software a codice sorgente aperto nella PA" del Ministero per l'Innovazione e le Tecnologie. È esperta indipendente per la Commissione Europea - DG INFOS dal 1997.

Consulente per la ricerca e sviluppo nell'Industria e nella PA, ha diretto in passato lo sviluppo di sistemi esperiti per Ansaldo Trasporti e si è occupata di trasferimento tecnologico dell'ICT e di UNIX.

Presso l'Università di Genova, dove è stata professore a contratto di Intelligenza Artificiale, dirige dal 2001 l'Osservatorio Tecnologico del MIUR, un servizio di trasferimento tecnologico dell'ICT alle scuole. È coautrice del libro "Il Governo Elettronico".

Membro del gruppo italiano di definizione della certificazione ECDL Advanced - IT Administrator, ha contribuito alla stesura del volume "Uso avanzato delle reti".

[giosissa@tin.it](mailto:giosissa@tin.it)