

PIATTAFORME PER SERVIZI APPLICATIVI IN RETE

Le *Service Delivery Platform* costituiscono un valido strumento per supportare il rapido sviluppo, l'efficiente esecuzione e l'efficace gestione dei servizi applicativi nelle reti di telecomunicazione sia fisse che mobili. Queste piattaforme sono particolarmente interessanti in un momento in cui le tecnologie e i modelli di business legati alla fornitura dei servizi a valore aggiunto stanno evolvendo molto rapidamente. L'articolo presenta quindi le principali caratteristiche delle *Service Delivery Platform* e delle tecnologie di supporto.

1. INTRODUZIONE

La ricerca di modelli innovativi di fruizione e di nuovi modelli di business in relazione alla fornitura di servizi in rete favorisce e stimola sia l'evoluzione dell'architettura delle reti telematiche sia l'adozione delle nuove tecnologie che vengono costantemente introdotte.

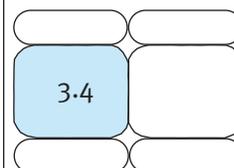
Nel settore IT (*Information Technology*) questa evoluzione ha per oggetto principalmente nuove piattaforme di livello applicativo, tipicamente basate sul paradigma *peer-to-peer*, e nuove modalità di fruizione della rete, tipicamente basate sul paradigma del Web2.0. La rete Internet, in particolare, evolve dal paradigma della "rete dell'informazione", a supporto cioè degli scambi informativi *one-to-one* per la comunicazione interpersonale (prevalentemente e-mail) e *one-to-many* per la fruizione di contenuti e servizi su siti e portali Web, ai paradigmi della "rete della conoscenza" e della "rete dei servizi" nei quali gli utilizzatori non sono più soltanto soggetti passivi fruitori di contenuti ma diventano invece soggetti attivi, e cioè autori o comunque fornitori

di contenuti, per esempio attraverso la condivisione di video (esempio, Youtube), immagini (esempio, Flickr), dati (esempio, Google), messaggistica ed informazioni di contesto (esempio, Skype, MS Messenger) e servizi o applicazioni (esempio, Facebook).

Nel settore TLC (*Telecommunications*), invece, l'evoluzione ha per oggetto principalmente la migrazione del cosiddetto "piano di controllo" delle reti telefoniche, tradizionalmente basato sull'architettura e sul protocollo *Signaling System #7 - SS7* [1], verso nuovi piani di controllo appoggiati sul *layer IP* (come per esempio nell'*IP Multimedia Subsystem - IMS* [2]), finalizzati a supportare sia l'integrazione delle reti e dei servizi su rete fissa e su rete mobile sia la realizzazione di servizi applicativi evoluti, spesso basati su paradigmi simili a quelli del mondo IT, all'interno di piattaforme e ambienti integrati per la fornitura di servizi (esempio, *Service Delivery Platform - SDP* o *Service Delivery Framework - SDF*). L'attenzione delle aziende che operano nel mondo dei servizi in rete è oggi focalizzata prevalentemente sull'integrazione tra il mondo IT e il



Pierpaolo Baglietto
Massimo Maresca
Francesco Moggia
Michele Stecca
Alberto Giordano



mondo TLC. Più specificatamente l'entrata in scena delle nuove realtà del mondo Internet, come Skype, YouTube o Google, ha portato gli operatori di TLC a ripensare alle proprie strategie in modo coordinato con Internet e non in contrapposizione ad essa. In altre parole la progressiva diminuzione dei ricavi provenienti dal traffico voce determinata dalla concorrenza derivante dalla liberalizzazione del mercato della fornitura dei servizi di telecomunicazione ha spinto gli operatori di TLC ad incrementare la propria offerta di servizi innovativi da un lato per limitare la riduzione dell'*Average Revenue per User* - ARPU, che con il solo servizio di trasporto sarebbe ridottissimo, e dall'altro per estendere la propria clientela e fidelizzare quella esistente. Vengono quindi introdotti servizi a valore aggiunto (*Value Added Service* - VAS) basati sull'integrazione di video, voce e dati come per esempio: chiamate multimediali, video-conferenze, *instant messaging* ecc..

In questo scenario evolutivo caratterizzato da una situazione di elevatissima variabilità compare un elemento infrastrutturale che ricopre un ruolo determinante nella fornitura dei servizi a valore aggiunto: la *Service Delivery Platform*. La SDP costituisce un elemento innovativo in rapido e continuo sviluppo ed incarna in modo specifico l'integrazione fra mondo IT e reti di telecomunicazione.

Nella prima sezione di questo articolo viene riassunta l'evoluzione delle reti telefoniche dal punto di vista dell'erogazione dei servizi. Vengono successivamente presentate le caratteristiche salienti delle SDP, elencati e categorizzati i servizi funzionali e applicativi che questo tipo di piattaforme forniscono ed infine si accennerà ad alcune delle tecnologie maggiormente utilizzate per la realizzazione delle SDP.

2. EVOLUZIONE DELL' EROGAZIONE DEI SERVIZI NELLE RETI TELEFONICHE

Per comprendere meglio come si è giunti alle attuali architetture per la fornitura dei servizi e quali "driver" hanno portato alla loro definizione, è opportuno ripercorrere brevemente l'evoluzione delle tecnologie utilizzate per l'erogazione dei servizi in rete partendo dalle reti telefoniche nell'era pre-IN (*Intelligent*

Network), in cui non era presente un particolare componente dedicato alla loro gestione, fino a giungere alle SDP convergenti.

Le caratteristiche peculiari della *Public Switched Telephone Network* - PSTN sono in estrema sintesi di essere progettata essenzialmente per il supporto del servizio telefonico e di non prevedere una gestione centralizzata dei servizi che erano invece gestiti da apposite applicazioni nei nodi terminali (switch telefonici) che offrivano dei servizi telefonici di base chiamati *Plain Old Telephone Service* - POTS (per esempio *voice mail*, *call waiting* ecc.). Questo richiedeva la replicazione degli apparati di controllo della rete e la replicazione degli apparati di tariffazione dei servizi, con la conseguente difficoltà di introdurre nuovi servizi data la limitata flessibilità del sistema.

A partire dalla metà degli anni '80 la rete PSTN cambia radicalmente grazie anche all'utilizzo del sistema di segnalazione SS7 (1) su canale comune (CCS, *Common Channel Signalling*) che ha introdotto la distinzione tra i livelli di trasporto, segnalazione e applicazioni. Nascono così le *Intelligent Network*, che rappresentano un punto di svolta rispetto al passato in particolare per quanto riguarda la fornitura dei servizi telefonici perché non prevedono più una gestione dei servizi decentrata e hard-coded nel sistema di *switching* ma centralizzata nel *Service Control Point* - SCP - e flessibile.

Purtroppo però le reti IN non permettevano ancora di raggiungere gli alti livelli di flessibilità richiesti dal mercato in quanto gli SCP erano *application-dependent* ovvero ogni nuova applicazione che si intendeva aggiungere richiedeva un sistema dedicato. Le *Advanced Intelligent Network* (AIN) superano parzialmente questo problema definendo SCP indipendenti dai servizi.

Parallelamente all'evoluzione della rete PSTN, la cui obsolescenza è segnata sia dall'invecchiamento tecnologico sia dalla mancanza di apertura verso il supporto di servizi a banda larga, si è assistito allo sviluppo della telefonia sulle reti a commutazione di pacchetto e in particolare al cosiddetto VoIP (*Voice over IP*). In questo ambito si colloca l'architettura *Soft-switch* e cioè quell'insieme di prodotti, protocolli e applicazioni che permettono ad un dispositivo di accedere a servizi di telecomunicazione su una rete IP.

Infine, vi è l'*IP Multimedia Subsystem - IMS* (2) che rappresenta il futuro delle reti per le telecomunicazioni (attualmente gli operatori telefonici si trovano in un momento di migrazione verso questa architettura). IMS è la specifica standard del settore per le comunicazioni definita dal 3GPP (*3-rd Generation Partnership Project*) che descrive il modo in cui gli elementi di una rete interagiscono allo scopo di fornire servizi e applicazioni vocali, dati e video in qualsiasi combinazione, su qualsiasi rete fissa o mobile. Esso utilizza dei protocolli standard quali *Session Initiation Protocol - SIP* [3], *Diameter*, *Real Time Protocol - RTP*, e si basa sulla stratificazione orizzontale delle reti in cui i vari livelli di trasporto, di segnalazione (o controllo) e delle applicazioni sono distinti (Figura 1). In IMS i servizi applicativi vengono forniti dagli *Application Server* e da piattaforme che mettono disposizione degli utilizzatori un ambiente finalizzato a supportare la creazione e l'esecuzione efficiente di servizi applicativi: la *Service Delivery Platform - SDP*.

3. SERVICE DELIVERY PLATFORM

Con il passaggio dal modello verticale o "stovepipes" (in cui ogni diversa rete di accesso ha una struttura di gestione dedicata) a quello orizzontale di IMS (che permette di ridurre la replicazione delle funzionalità nella rete in quanto è presente un unico layer di controllo comune a tutte le differenti tecnologie di accesso), il modo di concepire la fornitura di servizi cambia completamente perchè la visione diventa di tipo convergente ed è possibile prevedere un'unica piattaforma - la SDP - che possa supportare vari servizi relativi anche a reti diverse.

3.1. Funzionalità

Nonostante non esista una definizione universalmente riconosciuta di SDP, è possibile identificare un insieme di funzionalità che essa fornisce agli sviluppatori di servizi:

- permettere la rapida realizzazione dei *Value Added Services - VAS* (riduzione del *time-to-market*); si va sempre più affermando l'utilizzo di *Service Creation Environment - SCE* di tipo grafico che permettono di sviluppare i VAS mediante operazioni di *drag&drop* di blocchetti

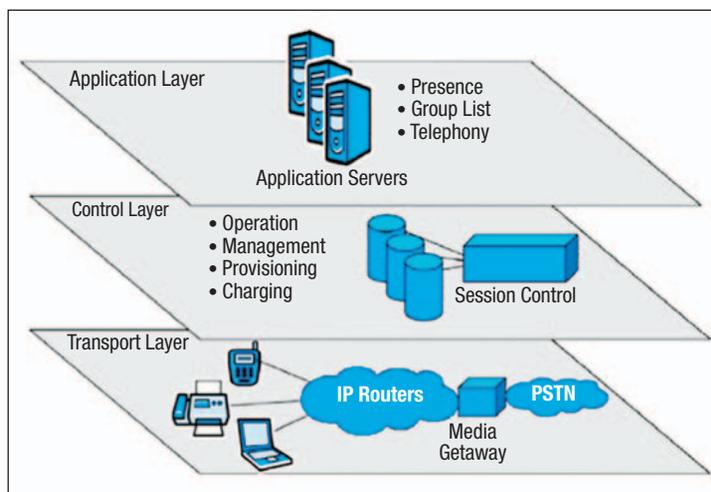


FIGURA 1
L'architettura a tre livelli di IMS. (Fonte: spie.org)

che rappresentano delle funzionalità di base altamente riutilizzabili;

- fornire un completo framework per il *deployment*, esecuzione, gestione e tariffazione dei VAS; favorire la portabilità dei servizi attraverso differenti fornitori (concetto "write once, run anywhere"); questo permette agli operatori di acquisire più libertà nella scelta dei *Network Equipment Provider - NEP*;

- fornire un meccanismo di astrazione dello strato di accesso; essendo la SDP un mezzo attraverso cui raggiungere la convergenza fissa-mobile, essa deve assicurare l'indipendenza dalla rete di accesso attraverso l'utilizzo di interfacce aperte che permetta agli sviluppatori di utilizzare le varie funzioni in modo uniforme;

- consentire una facile integrazione con gli altri elementi presenti nelle reti degli operatori con particolare attenzione per gli *Operational Support System - OSS* e i *Business Support System - BSS*;

- supportare la *Service Exposure* e cioè prevedere una modalità di fornitura di alcune funzionalità e/o servizi a terze parti che sia standard e sicura;

- effettuare *run-time monitoring, logging and reporting*; tracciare il comportamento della piattaforma attraverso un *Service Management Engine - SME* è molto importante per poter risolvere i malfunzionamenti e per effettuare un adattamento dei parametri della stessa di tipo *context-aware*;

- prevedere l'interoperabilità con i sistemi *legacy* ma anche essere estensibile per quanto

riguarda l'aggiunta di nuove tecnologie (importanza dell'utilizzo di interfacce standard).

3.2. Modello di riferimento per i servizi

Prima di passare ad esaminare l'architettura che può essere presa come riferimento per il funzionamento di una SDP, si ritiene opportuno chiarire il concetto di "servizio" perchè in questo ambito esso può assumere diversi significati a seconda del contesto a cui ci si riferisce. Si possono identificare tre differenti tipologie di servizi:

- **Servizi Infrastrutturali:** si occupano della gestione degli aspetti di più basso livello e sono gestiti dalla piattaforma;
- **Servizi Abilitanti:** si occupano di fornire delle funzionalità standard che possono essere utilizzate dagli sviluppatori di servizi applicativi; possono essere visti come dei mattoncini di base che vengono assemblati opportunamente per creare delle entità più complesse;
- **Servizi Applicativi:** questi rappresentano i servizi finali che vengono effettivamente utilizzati dagli utenti.

Il modello proposto prevede che lo sviluppatore definisca i Servizi Applicativi che verranno resi disponibili agli utenti finali mediante l'utilizzo combinato di alcuni Servizi Abilitanti con parti di codice peculiari di ogni specifica applicazione. I vari Servizi Applicativi saranno eseguiti e gestiti nella piattaforma la quale assicura il corretto ed efficiente funzionamento degli stessi grazie ai Servizi Infrastrutturali.

3.2.1. SERVIZI INFRASTRUTTURALI

I Servizi infrastrutturali risultano forniti in modo trasparente agli sviluppatori di Servizi Applicativi, che possono quindi concentrare la loro attenzione solo sulla logica applicativa, lasciando in carico alla piattaforma tutte quelle funzionalità di più basso livello che vengono ora elencate:

- **Scalability:** essa deve gestire opportunamente il *load balancing* delle richieste e prevedere un meccanismo automatico per l'aggiunta di nodi di elaborazione in modo che il sistema sia in grado di auto-riconfigurarsi senza dover sospendere la sua esecuzione.
- **Fault Tolerance:** le SDP devono continuare a funzionare anche in caso di malfunzionamenti in quanto sospendere l'erogazione dei servizi può causare ingenti perdite economiche per

l'operatore; è quindi necessario che la piattaforma fornisca un meccanismo automatico di *failure-detection* che individui gli eventuali guasti e che riorganizzi "a caldo" il bilanciamento del traffico sui nodi disponibili.

- **Low Latency/High Throughput:** questi parametri sono tipici del mondo delle telecomunicazioni e sottolineano il fatto che è molto importante che le richieste in entrata alla SDP siano servite in modo rapido.
- **Service Management:** deve essere realizzata una facile integrazione con i sistemi di OSS e BSS.
- **Security Management:** è richiesta un'opportuna gestione degli aspetti di *Authorization, Authentication e Accounting* - AAA.

Il supporto dei Servizi Infrastrutturali è imprescindibile in una SDP di qualità *Carrier Grade* ovvero che assicuri un'affidabilità pari al 99.999% (i famosi *five-nines*). Questa denominazione implica il fatto che la SDP riesca ad erogare i suoi servizi anche quando gli utenti raggiungono un numero considerevole.

3.2.2. SERVIZI ABILITANTI

I Servizi Abilitanti, o *Service Enabler*, sono quelli che vengono resi disponibili dalla SDP allo sviluppatore di Servizi Applicativi per consentire a quest'ultimo di contare su un insieme di funzionalità trasversali che possono essere assemblate secondo le esigenze specifiche di ogni applicazione e, dove possibile, riutilizzate, per creare le applicazioni che vengono proposte all'utente finale e che costituiscono i *Value Added Services*. Esempi di Servizi Abilitanti sono: *Session Enabler* (gestione sessioni utente), *AAA Enabler* (gestione aspetti di sicurezza, spesso bastato sul protocollo DIAMETER), *Presence Enabler* (gestione dello "stato/presenza" degli utenti), *Media Enabler* (gestione del mixing e della gestione di media differenti), *Messaging Enabler* (gestione unificata della messaggistica), *Charging Enabler* (gestione della tariffazione dei servizi che si divide in *online* e *offline charging*) ecc..

3.2.3. SERVIZI APPLICATIVI

3.2.3.1. Servizi User-to-User

Per quanto concerne le comunicazioni tra singoli utenti, si tratta di estendere le classi di servizi già disponibili per gli utenti come le tradi-

zionali chiamate, gli SMS, gli MMS ecc.. Alcuni servizi di esempio possono essere:

- i *Rich Voice Services*, nei quali si aggiunge alla classica chiamata telefonica la possibilità di scambiare contenuti come immagini o file di vario tipo. Una SDP dovrebbe anche essere in grado di gestire le *video-chiamate* oppure sessioni di *chat (Instant Messaging)*;

- il *Multi Device Service* permette all'utente chiamante di essere identificato dall'utente chiamato sempre con lo stesso identificativo (per esempio un SIP-ID) indipendentemente dall'apparecchio che esso utilizza per la comunicazione (telefono fisso, cellulare, IP-phone ecc.);

- il *Multi Ring Service* permette di distribuire simultaneamente le chiamate entranti su vari apparecchi;

- il *Follow-Me /Line Hunting Service* permette di decidere in che ordine devono suonare i differenti apparecchi (per esempio si può impostare che suoni per primo il telefono cellulare e, qualora non vi sia risposta, dirottare la chiamata sull'IP-Phone).

3.2.3.2. Servizi Multi-User

In questo scenario sono disponibili diversi servizi che prevedono il coinvolgimento di più utenti contemporaneamente:

- le Conferenze Multimediali che costituiscono un importante servizio che viene utilizzato dalle aziende per effettuare incontri a distanza e diminuire i relativi costi; è possibile inoltre classificare in questa categoria anche il servizio Mobile Gaming che, per sua natura, richiede il coinvolgimento di più utenti;

- il *Multi Channel Televoting System* che permette di gestire le operazioni di voto collegate, per esempio, ai programmi televisivi;

- il *Convergent Messaging* che è un sistema di messaggistica che gestisce tutti i vari formati (testuali, SMS, e-mail, *instant messaging*) attraverso cui gli utenti possono comunicare tra di loro (*buddy list*).

3.2.3.3. Servizi Server-to-User

Per quanto riguarda le applicazioni *Server-to-User*, risulta particolarmente interessante la categoria di servizi che viene definita *Dynamic Push Service* che è possibile grazie anche alla presenza di una funzionalità prevista dall'architettura IMS, anche se non necessariamente

supportata, che provvede alla *localizzazione geografica* dell'utente. Sottoscrivendosi a questo tipo di servizi, gli utenti possono ricevere dei contenuti in modo particolarmente personalizzato in quanto il loro invio si basa su molti fattori come la localizzazione geografica, lo stato di presenza, l'orario, profili di preferenze pre-caricate ecc.. Il tipo di contenuto che viene ricevuto può essere estremamente variabile e dipende dallo specifico servizio (per esempio, informazioni stradali, il filmato del gol della propria squadra, l'indirizzo di un determinato negozio ecc.). Gli operatori possono progettare i più diversi tipi di *Dynamic Push Service* basandosi su appropriate indagini di mercato (per un esempio si veda [8]).

3.3. Architettura di riferimento

Il problema degli standard nell'ambito delle SDP si riflette anche nella mancanza di un'architettura che definisca nel dettaglio le componenti che devono essere presenti nella piattaforma e le relative funzionalità.

Dalla consultazione della letteratura a disposizione, sembra ormai imporsi l'architettura proposta in [7] e mostrata nella figura 2 che, sebbene molto generica, viene riconosciuta come riferimento da molti fornitori. Essa include i seguenti elementi:

- *Network Abstraction Layer*: fornisce interfacce standardizzate verso i vari tipi di reti. Lo scopo principale di questo componente è

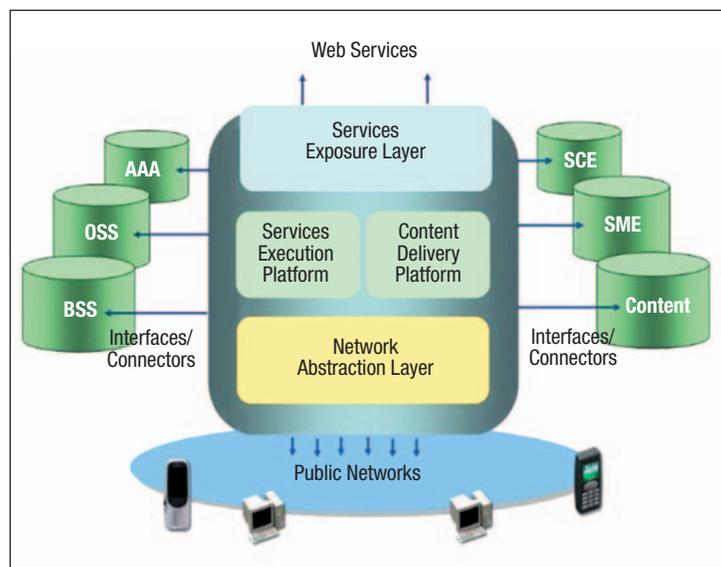


FIGURA 2
Architettura delle SDP. (Fonte: Rapporto Moriana [7])

quello di rendere l'accesso ai servizi indipendente dalla tecnologia di rete sottostante.

□ *Service Exposure Layer*: componente opzionale che serve a mettere a disposizione a terze parti alcune funzionalità. Esso permette agli operatori di "aprire" le proprie reti attraverso l'utilizzo di interfacce standardizzate e sicure.

□ *Content Delivery Platform*: componente che gestisce i contenuti multimediali degli apparecchi mobili.

□ *Service Execution Platform*: costituisce l'ambiente in cui sono presenti i servizi e in cui vengono eseguiti. Questa parte della piattaforma deve assicurare delle elevate prestazioni per poter raggiungere dei sufficienti livelli di "user experience".

4. TECNOLOGIE A SUPPORTO DELLA SERVICE EXECUTION PLATFORM

Per quanto concerne gli aspetti tecnologici, risultano particolarmente interessanti tre standard utilizzati nelle *Service Execution Platform* che sono stati definiti dal *Java Community Process* - JCP mediante la divulgazione di specifiche chiamate *Java Specification Request* - JSR:

- la specifica J2EE ([5]) descrive una piattaforma che utilizza un modello applicativo *multi-tier* e distribuito. La logica applicativa è suddivisa in componenti che rispecchiano differenti funzioni che devono essere svolte e che vengono installati talvolta in diverse macchine. J2EE è costituita da un insieme di specifiche (per esempio Java Transaction API-JTA, Java DataBase Connector-JDBC ecc.) che standardizzano i molti aspetti relativi alle applicazioni enterprise.

- La specifica JSLEE ([4]) descrive un'architettura che permette di creare delle applicazioni multiprotocollo orientate agli eventi (*Event Driven Applications*, EDA). L'ambiente gesti-

sce tutti gli aspetti legati alle prestazioni, alla concorrenza, alle transazioni, ai fallimenti e alla persistenza in modo tale da assicurare i due requisiti chiavi alle applicazioni orientate agli eventi: bassa latenza e alto *throughput*.

- Le applicazioni basate sulle SIP Servlet ([6]) sono implementate attraverso API che permettono di gestire i messaggi del protocollo SIP e viene effettuato il loro "deploy" all'interno di un *Application server* che ne dà supporto equivalente a quello fornito alle HTTP Servlet.

Una comparazione dei tre standard citati precedentemente è proposta nella tabella 1.

Vista l'estrema eterogeneità delle tecnologie proposte, si deduce che non esiste una scelta ottima valida in assoluto e, infatti, le diverse aziende manifatturiere si stanno muovendo in direzioni diverse per la scelta delle piattaforme di supporto per l'esecuzione dei servizi. Le motivazioni che stanno alla base di questa eterogeneità vanno ricercate nelle diverse esperienze e nei diversi modelli di business a cui fanno riferimenti le aziende produttrici di piattaforme ma anche nell'imaturità di alcune tecnologie.

5. I SERVIZI APPLICATIVI NELL'ERA DEGLI SMARTPHONE: GLI APPLICATION STORE

La recente diffusione degli *Smartphone*, ovvero dei dispositivi mobili forniti di elevate capacità computazionali, numerose periferiche, connessione ad Internet ecc. - il più famoso dei quali è l'*iPhone* prodotto da Apple - ha portato numerose novità sia sotto l'aspetto tecnologico che dei modelli di business.

In particolare gli *Smartphone* sono dei dispositivi dotati di sistema operativo su cui è possibile installare delle applicazioni che vengono rese disponibili in un apposito sito chiamato *Application Store*. Gli utenti finali visitano que-

TABELLA 1
Tabella comparativa degli standard a supporto dell'esecuzione dei servizi

	J2EE	JSLEE	SIP-SERVLET
Gestione efficiente degli eventi	NO	SI	Solo protocollo SIP
Gestione concorrenza	SI	SI	NO
Multiprotocollo	NO	SI	NO
Esistenza di prodotti open source	SI (JBoss, Sun GlassFish ecc.)	SI (Mobicents)	SI (Red Hat Comm. Platform, Saifin ecc.)

sto negozio virtuale, identificano l'applicazione di loro gradimento (che può essere gratuita oppure a pagamento) e, dopo averla scaricata, la installano sul proprio terminale dove l'applicazione viene eseguita. Il *Service Execution Environment* è quindi parzialmente (e in alcuni casi completamente) localizzato nel dispositivo mobile dell'utente e questo costituisce un importante cambiamento di paradigma che ha delle importanti ripercussioni sulla progettazione delle SDP (per esempio si deve gestire opportunamente un aumento di carico sul *Service Exposure Layer*).

È interessante sottolineare il fatto che le applicazioni disponibili negli *Application Store* possono essere sviluppate dai fornitori degli Smartphone ma nella grande maggioranza dei casi sono sviluppate da "terze parti" (aziende oppure privati) che realizzano dei servizi sfruttando gli appositi SDK (*Software Development Kit*) che vengono messi a disposizione. Emerge quindi la necessità di definire modelli di business che tengano conto dei diversi attori che entrano in gioco ovvero il gestore dell'*Application Store*, lo sviluppatore dell'applicazione e l'utente finale che scarica e utilizza l'applicazione. Il modello proposto dall'*Application Store* di Apple, che rappresenta il punto di riferimento in questo ambito, prevede che il ricavo ottenuto dalla vendita di una determinata applicazione sia suddiviso secondo il seguente schema: 70% del ricavo allo sviluppatore e 30% ad Apple.

Esempi di *Application Store* attualmente esistenti sono, oltre *AppStore* di Apple, *Ovi* di Nokia, *Android Market* per i dispositivi basati sul sistema operativo Android di Google e *360* di Vodafone.

6. CONCLUSIONI

In questo articolo è stata ripercorsa la storia dell'evoluzione della fornitura dei servizi telefonici partendo dalla più datata architettura della rete PSTN, in cui erano disponibili pochi semplici servizi, fino ad arrivare alla rete IMS (la cui definizione da parte degli enti di standardizzazione non è ancora giunta ad una versione definitiva) che dovrebbe ampliare l'insieme dei servizi disponibili grazie alla stratificazione orizzontale della rete che permette di velocizzare la fornitura

di servizi convergenti (e quindi non solo telefonici).

In questo nuovo scenario ricoprono un ruolo cruciale le *Service Delivery Platform* per le quali non è ancora disponibile la descrizione di una architettura standard universalmente riconosciuta ma per le quali si possono identificare una serie di requisiti comuni come favorire il veloce sviluppo di nuovi servizi applicativi (al fine di abbassare il *time-to-market* dei servizi), assicurare la facile integrazione con i sistemi già presenti presso gli operatori (con particolare attenzione ai sistemi di OSS e BSS), fornire delle funzionalità trasversali come la scalabilità e la tolleranza ai guasti (fondamentali per poter definire una SDP *carrier grade*) e il rispetto dei più importanti standard tecnologici. Per quanto riguarda quest'ultimo punto, si è discusso della presenza di una certa varietà di possibilità (J2EE, JSLEE, SIP Servlet) e di come la tecnologia più adatta da scegliere possa variare a seconda del tipo di servizi che si intende fornire.

Oltre alla classificazione dei numerosi standard presenti per l'implementazione delle varie componenti delle SDP, rimane ancora aperta la questione di categorizzare precisamente l'insieme dei servizi che si possono realizzare e fornire mediante queste piattaforme software (applicazioni telefoniche, Internet, *MashUp* ecc.) e quale collocazione possano trovare sul mercato. Per quanto riguarda quest'ultima questione, più legata agli aspetti di business che a quelli tecnologici, è stato introdotto il tema degli *Application Store* che si sta affermando prepotentemente sulla scia del successo dell'*Apple Store* e che dimostra come il coinvolgimento mediante appropriati schemi di *revenue-sharing* delle cosiddette "terze parti" possa essere un fattore fondamentale per la realizzazione di servizi applicativi che siano più appetibili per gli utenti finali.

Bibliografia

- [1] Cisco: *SS7 Fundamentals Cisco Tutorial*. [disponibile sul Web all'indirizzo: http://ciscosystems.com/univercd/cc/td/doc/product/tel_pswt/vco_prod/ss7_fund/index.htm]
- [2] 3 GPP Group: IMS Releases, [<http://www.3gpp.org/article/ims>]

- [3] Internet Engineering Task Force - IETF, Session Initiation Protocol (RFC 3261).
- [4] JAVA Community Process: JAIN SLEE 1.1 Specification, Final Release. <http://jcp.org/en/jsr/detail?id=240>, Luglio 2008.
- [5] JAVA Community Process: JAVA Platform Enterprise Edition v1.4 Specification, Final Release. <http://jcp.org/en/jsr/detail?id=151>, Novembre 2003.
- [6] JAVA Community Process: SIP Servlet 1.1 Specification, Final Release. <http://jcp.org/en/jsr/detail?id=289>, Agosto 2008.
- [7] Moriana Group: *Service Delivery Platform and Telecom Web Services*. Giugno 2004.
- [8] Metafacile: *Guida turistica GPS*. [http://www.metafacile.com/ita/itinerari_gps_franciacorta.php]

PIERPAOLO BAGLIETTO si è laureato in Ingegneria Elettronica presso l'Università di Genova dove nel 1994 ha conseguito il Dottorato di Ricerca in Ingegneria Elettronica ed Informatica. È attualmente Professore Associato presso la Facoltà di Ingegneria dell'Università di Genova dove è docente dei corsi di Calcolatori Elettronici e di Sistemi Informativi e Servizi in Rete. I suoi interessi di ricerca sono centrati sui sistemi distribuiti, i sistemi di monitoraggio e gestione delle reti di nuova generazione e le piattaforme per lo sviluppo e l'erogazione di servizi.
E-mail: p.baglietto@cipi.unige.it

MASSIMO MARESCA si è laureato in Ingegneria Elettronica presso l'Università di Genova dove nel 1986 ha conseguito il Dottorato di Ricerca in Ingegneria Elettronica ed Informatica. Ha svolto attività di ricerca presso IBM T.J. Watson Research Centre di New York (USA) e presso l'International Computer Science Institute di Berkeley (USA). È attualmente professore ordinario presso l'Università di Padova e Direttore del Centro di Ricerca Interuniversitario sull'Ingegneria delle Piattaforme Informatiche (CIPi). I suoi interessi di ricerca sono centrati sulle piattaforme per la composizione e l'orchestrazione di servizi e sui sistemi di monitoraggio e gestione delle reti a banda larga.
E-mail: m.maresca@cipi.unige.it

FRANCESCO MOGGIA ha conseguito la laurea in Ingegneria delle Telecomunicazioni nel 2000 e il titolo di Dottore di Ricerca in Ingegneria Elettronica ed Informatica presso l'Università di Genova nel 2006. Ha svolto attività di ricerca presso il National Institute of Standards and Technology (Gaithersburg, MD) dal 2003 al 2005 contribuendo allo sviluppo del progetto open source Mobicents. Dal 2005 al 2007 è stato ricercatore in Telecom Italia, contribuendo alla JSR-289 (Servlet Sip) e JSR-240 (JSLEE). È attualmente responsabile tecnico di M3S srl. I suoi interessi di ricerca attuali sono le Service Delivery Platforms, le architetture SOA e REST e le tecnologie per i protocolli di messaggistica e VoIP.
E-mail: f.moggia@m3s.it

MICHELE STECCA si è laureato in Ingegneria Informatica all'Università degli Studi di Padova nel 2007. Attualmente è impegnato come dottorando di ricerca presso il CIPi (Centro Interuniversitario di Ricerca sull'Ingegneria delle Piattaforme Informatiche - Università di Padova e di Genova). In passato ha partecipato al progetto OPUCE (Open Platform for User-centric service Creation and Execution) finanziato dalla Comunità Europea nell'ambito del Sesto Programma Quadro. I suoi interessi di ricerca sono centrati sulle piattaforme per la composizione e l'orchestrazione di servizi (Mashups) e sulle piattaforme per la fornitura di servizi in ambito telecom.
E-mail: m.stecca@cipi.unige.it

ALBERTO GIORDANO si è laureato in Ingegneria Elettronica presso l'Università di Genova nel 1977. Ha successivamente svolto attività di ricerca presso l'Università di Karlsruhe e presso la Fraunhofer Gesellschaft di Karlsruhe. Dal 1984 ha coperto il ruolo di ricercatore ed attualmente è Professore Associato presso la Facoltà di Ingegneria dell'Università di Genova dove è docente dei corsi di Fondamenti di Informatica e Sistemi di Elaborazione. I suoi interessi di ricerca riguardano le architetture dei sistemi di elaborazione, le reti di telecomunicazione wired e wireless ed i sistemi orientati all'erogazione di servizi.
E-mail: a.giordano@cipi.unige.it

ANALISI E IDENTIFICAZIONE DEL TRAFFICO INTERNET

Per gestire in modo efficiente le risorse della sua rete, l'*Internet Service Provider* (ISP) ha la necessità di conoscere le caratteristiche dei flussi di traffico trasportati attraverso la sua infrastruttura e di saper individuare l'applicazione che ha generato i flussi stessi. Con queste informazioni, l'ISP può stabilire come gestire ogni flusso. Inoltre, conoscendo l'applicazione che sta generando un determinato flusso, si possono stabilire con precisione i requisiti di qualità del servizio ad esso associati che, se rispettati, determinano un maggior grado di soddisfazione degli utenti. L'analisi del traffico Internet e la sua identificazione possono fornire all'ISP queste informazioni di importanza strategica.

1. LE MOTIVAZIONI DELL'ANALISI E DELL'IDENTIFICAZIONE DEL TRAFFICO INTERNET

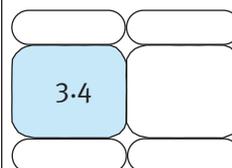
Per un *Internet Service Provider* (ISP), l'analisi e l'identificazione del traffico generato dai suoi clienti è propedeutica ad un insieme di operazioni critiche relative alla gestione delle risorse della rete e del suo rapporto, anche economico, con i clienti stessi. Se, da un lato, per un ISP è relativamente semplice misurare il volume complessivo di traffico che transita sui collegamenti (*link*) della propria rete, l'informazione ottenuta da una misurazione così aggregata e priva di dettagli si restringe ad una valutazione complessiva del carico al quale sono sottoposti i link e i nodi (*router*). Eventualmente, tramite una collezione di serie storiche di questo carico, l'ISP è in grado di eseguire un'analisi della tendenza del carico e di decidere se è opportuno potenziare qualche *link/router* che si sta avvicinando ad una soglia di carico che esso ritiene critica. Da questo punto di vista, l'analisi del carico complessivo delle risorse della rete abilita una basilare attività di *capacity planning* (pianificazione della

capacità) e *resource provisioning* (approvvigionamento delle risorse).

Per gestire in modo più preciso ed efficiente le risorse della sua infrastruttura, l'ISP ha bisogno di informazioni più dettagliate. In primo luogo, l'ISP deve conoscere le caratteristiche dei flussi di traffico che fa transitare nella sua rete, in particolare: la durata dei flussi, il volume di traffico che ognuno di essi trasporta, la loro velocità di trasmissione e il grado di variabilità di questa velocità. Questo tipo di analisi è denominata *flow analysis* (analisi a livello di flusso) e permette all'ISP una gestione delle risorse più efficiente di quella abilitata da una mera misurazione del livello di carico complessivo dei link. Il secondo problema che deve affrontare l'ISP è quello di conoscere la tipologia di applicazione il cui traffico è trasportato in rete dai flussi. L'analisi che permette di ottenere questa informazione è l'*identificazione* (o anche classificazione) del traffico che mette in grado l'ISP di gestire al meglio i flussi nella sua rete, tenendo conto dei requisiti di qualità e delle caratteristiche specifiche delle applicazioni utilizzate dagli utenti finali.



Paolo Giacomazzi



La conoscenza dell'aliquota di traffico generata da ogni tipologia di applicazione fornisce all'ISP uno strumento per meglio pianificare le proprie strategie di offerta di servizi e della relativa tassazione. Per esempio, è proprio grazie all'identificazione del traffico che si è potuto capire che attualmente il traffico generato dalle applicazioni di tipo *Peer-to-Peer file sharing* (per esempio eMule) **ammonta a circa l'80% del complessivo traffico in Internet** [1]. Visto che molto di questo traffico è generato da utenti dotati di connessioni ADSL con tassazione di tipo *flat* (a canone, indipendentemente dal volume di traffico scambiato dall'utente) gli ISP sono consci, grazie all'identificazione del traffico, di trovarsi in una situazione nella quale l'occupazione delle risorse di rete è in crescita veloce e, tale aumento (che comporta costi di investimento per il potenziamento dell'infrastruttura) non corrisponde ad un commisurato incremento del fatturato, con conseguente erosione dei margini operativi. Anche l'identificazione di applicazioni di telefonia *peer-to-peer* (per esempio Skype [2]) è critica per gli ISP. In questo caso, il traffico telefonico transita sulla linea ADSL del cliente, e non attraverso la rete telefonica, privando il provider della possibilità di tassare la chiamata. Questi due esempi dimostrano come l'ISP debba necessariamente conoscere la tipologia di applicazioni utilizzate dai propri clienti per individuare le criticità e le motivazioni che portano ad una diminuzione dei margini operativi e per poter elaborare contromisure. Da questo punto di vista, l'identificazione del traffico è per l'ISP un'attività di importanza strategica.

L'identificazione del traffico è anche un utile strumento per una gestione della Qualità del Servizio (QoS) differenziata per le diverse applicazioni. Infatti, se si riconosce l'applicazione che sta generando un dato flusso di traffico, si possono identificare i requisiti di QoS specifici per quell'applicazione (per esempio, in termini di *throughput* e ritardo) e quindi allocare le risorse necessarie per garantire che questi requisiti siano rispettati, con conseguente soddisfazione dei clienti. Questa attività è usualmente denominata *QoS management*.

Un'altra attività di gestione del traffico realizzata dagli ISP è il cosiddetto *traffic engineering*, un'operazione che consiste nel determi-

nare, per un dato flusso di traffico o per una data categoria di flussi, il percorso migliore (*route*) all'interno della rete, per meglio rispettare i requisiti di qualità del servizio delle applicazioni. Un'attività di *traffic engineering* mirata alla gestione della QoS richiede che i requisiti di qualità del servizio siano noti e, quindi, che si conduca un'attività di identificazione del traffico.

Dal punto di vista della gestione della sicurezza, l'identificazione del traffico può rendere più efficiente l'attività di *anomaly detection*, che consiste per esempio nell'individuare un carico di traffico anomalo per una data applicazione in rete. Una situazione di questo tipo può essere generata da attacchi di tipo *Denial of Service* o *Distributed Denial of Service*, nel quale un insieme di computer "infettati" genera un carico di traffico focalizzato verso risorse mirate (per esempio un sito web che si intende inabilitare). Se si rileva una distribuzione anomala del traffico, si possono intraprendere misure reattive per controbattere questo tipo di attacchi.

È opportuno citare infine la possibilità che gli ISP hanno di fornire risorse differenziate a diverse tipologie di applicazioni. È chiaro, infatti, che se il traffico *peer-to-peer* consuma la maggior parte delle risorse di rete senza produrre un fatturato commisurato, un ISP potrebbe fornire meno banda alle applicazioni di tipo *peer-to-peer*, a favore di applicazioni più remunerative. Questo è un argomento molto controverso che vede entrare nel dibattito la questione della *network neutrality*. La *network neutrality* è un principio in base al quale un ISP non dovrebbe discriminare il tipo di applicazione utilizzato dai suoi clienti, quando fornisce ad essi un collegamento ad Internet. In pratica, l'ISP dovrebbe agire secondo il principio "*un bit è un bit*", cioè, tutti i bit vanno trattati allo stesso modo, senza sfavorire il traffico di alcune tipologie di applicazioni per favorirne altre più convenienti dal punto di vista dei profitti. D'altra parte, la *network neutrality* è un principio non ancora formalizzato da normative, oggetto di dibattito, con sostenitori e oppositori. Si cita il caso (negli USA) di Comcast [3] che ha rallentato il traffico delle applicazioni di *file sharing*. Nel 2008, la Federal Communications Commission (FCC) sanzionò Comcast per questo comportamento, ma recentemente (Aprile

2010) la corte d'appello federale del distretto di Columbia ha ribaltato la decisione. Dunque, da un punto di vista tecnico, l'identificazione del traffico permette all'ISP di penalizzare alcune categorie di applicazioni, ma l'effettivo utilizzo di una politica di questo tipo non è ancora stato regolato dal Legislatore.

2. ANALISI DEI FLUSSI DI TRAFFICO

L'analisi dei flussi e l'identificazione del traffico Internet avvengono tramite l'osservazione dei pacchetti IP che costituiscono il flusso stesso. Un flusso di traffico è definito come una sequenza di pacchetti IP che condividono gli stessi indirizzi IP di sorgente e di destinazione (*Source Address e Destination Address* nella Figura 1), gli stessi numeri di porta di sorgente e di destinazione (*Source Port e Destination Port*) e lo stesso protocollo (il campo *Protocol* dell'*header IP*) di trasporto. Questi cinque campi, che nella figura 1 sono evidenziati con uno sfondo arancione, sono usualmente denominati la "quintupla" e identificano efficacemente il flusso al quale un pacchetto IP appartiene. Come conseguenza di questa definizione, un flusso di traffico è monodirezionale e, se si intende esaminare il comportamento complessivo di un'applicazione (traffico di andata e traffico di ritorno), si dovranno esamina-

re congiuntamente i due flussi monodirezionali che trasportano il traffico dell'applicazione nei due sensi (nel caso di un'applicazione *client-server*, i due flussi sono quello da *client a server* e quello da *server a client* che, insieme, formano il flusso logico bidirezionale di collegamento remoto tra *client* e *server*).

2.1. Caratteristiche dei flussi di traffico in Internet

Dal punto di vista degli ISP, le caratteristiche più critiche di un flusso di traffico sono la durata, il volume complessivo di traffico trasportato, la velocità di trasmissione e il grado di variabilità di questa velocità. Queste caratteristiche presentano una grande variabilità [3] e sulla base di queste si usa suddividere i flussi di traffico in un insieme di categorie. La categorizzazione più utilizzata è quella illustrata nella figura 2. Almeno il 45% dei flussi ha un tempo di vita di meno di 2 s - sono i flussi denominati *dragonfly* (libellula) - e circa il 98% dei flussi dura meno di 15 min. Il restante 2% dei flussi ha una durata di parecchie ore o giorni, sono denominati *tortoise* (tartaruga), e trasportano il 50%-60% dei byte trasmessi su un link. Per quanto riguarda il volume di byte trasportati, un flusso è un *mouse* (topo) o un *elephant* (elefante). Un *mouse* trasporta pochissimo traffico, ma i *mouse* sono decisamente numerosi. D'altra parte, i pochi flussi *elephant* sono

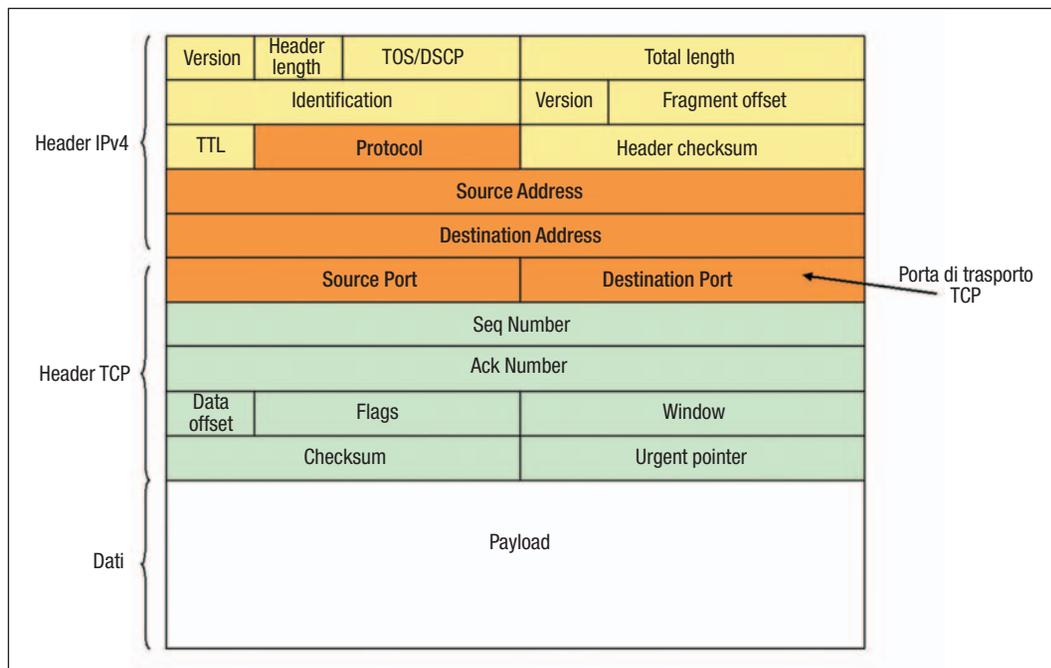


FIGURA 1
La porta di trasporto in un pacchetto IPv4/TCP

<p style="text-align: center;">Durata</p> <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>Tortoise</p>  <p>Durano anche per giorni</p> </div> <div style="text-align: center;"> <p>Dragonfly</p>  <p>Durano pochi secondi</p> </div> </div> <p style="text-align: center;">Le tortoise sono poche (meno del 2%), ma costituiscono il 50%-60% del traffico sui link di Internet</p>	<p style="text-align: center;">Volume</p> <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>Elephant</p>  </div> <div style="text-align: center;"> <p>Mouse</p>  </div> </div> <p style="text-align: center;">Gli elephant sono pochi, ma determinano la maggior parte del traffico in Internet</p>
<p style="text-align: center;">Velocità</p> <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>Cheetah</p>  </div> <div style="text-align: center;"> <p>Snail</p>  </div> </div> <p style="text-align: center;">Le Cheetah sono flussi ad elevata velocità, che generano rapidamente grandi volumi di traffico</p>	<p style="text-align: center;">Burstiness</p> <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>Porcupine</p>  </div> <div style="text-align: center;"> <p>Stingray</p>  </div> </div> <p style="text-align: center;">Il porcupine è un flusso di traffico con profilo di velocità molto variabile. Allocare e gestire le risorse per un porcupine è molto più difficile che per un flusso a profilo di velocità regolare (stingray)</p>

FIGURA 2
 Tipologie di flussi in Internet, classificati secondo il volume, la durata, la velocità e la burstiness

responsabili della maggior parte del traffico sui *link* della rete. La terza caratteristica significativa di un flusso di traffico è la sua *burstiness*. Un flusso con elevata *burstiness* presenta una velocità molto variabile nel tempo e passa rapidamente da uno stato in cui lavora a bassa velocità ad uno stato in cui trasmette a velocità molto elevata, e viceversa. Un flusso ad elevata *burstiness* è un *porcupine* (porcupino). Al contrario, flussi che presentano un profilo di velocità molto regolare e con poche variazioni sono denominati *stingray* (razza) [5]. Infine, un flusso può essere caratterizzato da una velocità media molto elevata, e in questo caso è denominato *cheetah* (ghepardo), o molto piccola, è il caso dello *snail* (lumaca). L'ISP, individuando le categorie di appartenenza dei flussi di traffico presenti sui link della propria rete, potrà evidenziare i flussi più critici, ponendo per esempio particolare attenzione agli *elephant*, che determinano l'aliquota maggiore del traffico in rete, per poterli gestire nel migliore dei modi. Per esempio, è logico instradare gli *elephant* attraverso i link provvisti di più risorse disponibili. In pratica, i flussi di traffico che devono destare più attenzione sono quelli ad alto volume (*elephant*), quelli veloci (*cheetah*) perché richiedono l'allocazione di molta banda sui link, quelli lunghi (*tortoise*) perché le risorse allocate rimango-

no impegnate a lungo e quelli molto variabili (*porcupine*), perché risulta molto delicata la determinazione delle risorse richieste per servirli con un livello di qualità adeguato (al contrario, determinare la banda richiesta per servire uno *stingray* è semplice, in quanto il profilo di velocità è poco variabile). Flussi che presentano tutte le quattro caratteristiche al livello di criticità massimo sono molto rari e sono più frequenti situazioni intermedie come per esempio flussi ad alto volume (*elephant*) e lunghi (*tortoise*), ma non particolarmente veloci o *bursty*. Anche questi flussi sono critici e da trattare in modo adeguato. Vi sono diversi metodi per individuare le quattro categorie di un flusso di traffico [1]. Partendo dalla constatazione che in Internet una piccola percentuale dei flussi è responsabile della maggior parte del traffico, una tecnica utilizzata per determinare se un flusso è un *elephant* o un *mouse* consiste nel calcolare il volume medio e la deviazione standard del volume di traffico generato dai flussi sul link in esame. Un flusso è classificato come *elephant* se il suo volume è più grande del volume medio dei flussi, più tre volte la deviazione standard, ed è classificato come *mouse* altrimenti. Secondo lo stesso principio, si classifica un flusso come *tortoise* se la sua durata è più grande della durata media dei flussi, più tre volte la deviazione stan-

dard della durata stessa, altrimenti il flusso è un *dragonfly*. Nello stesso modo si procede per classificare un flusso come *cheetah/snail e porcupine/stingray*.

3. L'IDENTIFICAZIONE DEL TRAFFICO INTERNET

Come già evidenziato, l'identificazione del traffico Internet si differenzia sostanzialmente dall'analisi dei flussi, in quanto la prima intende individuare l'applicazione che ha generato un flusso di traffico e la seconda ha lo scopo di misurare alcune caratteristiche complessive del traffico trasportato dai flussi.

3.1. L'identificazione del traffico Internet tramite il numero di porta

Tradizionalmente, l'identificazione dell'applicazione che genera un flusso di traffico sotto osservazione è stata eseguita tramite il riconoscimento delle porte a livello di trasporto. L'analisi delle porte di trasporto è stata (fino a che la si è potuta utilizzare) un metodo semplice ed efficace per identificare le applicazioni tramite l'osservazione dei pacchetti appartenenti al flusso. Come mostrato nella figura 1, un pacchetto IPv4 che trasporta i dati di un'applicazione che utilizza il *Transmission Control Protocol* (TCP) come livello di trasporto, riporta esplicitamente il numero di porta di destinazione (*Destination Port*) nell'*header* del TCP. La porta di destinazione ha lo scopo di identificare esplicitamente l'applicazione che utilizza i dati contenuti nel pacchetto. Esistono porte allocate ufficialmente dall'*Internet Assigned Numbers Authority* (IANA) [6]; per esempio, la porta 25 è assegnata al protocollo *Simple Mail Transfer Protocol* (e-mail), la porta 23 al *The Secure Shell* (SSH) *Protocol*, la porta 80 ad HTTP, originariamente utilizzata per il *web browsing*, e così via. Se un'applicazione utilizza una delle porte assegnate per raggiungere l'applicazione prevista (per esempio, il World Wide Web sulla porta 80), il riconoscimento dell'applicazione alla quale appartiene un pacchetto è direttamente desumibile dall'ispezione diretta della porta di destinazione. Anche nel caso in cui un'applicazione non sia assegnataria di una porta ufficiale, ma usi sempre una porta o un insieme di porte ben definito (come facevano per esempio

alcuni sistemi *peer-to-peer* di prima generazione), il riconoscimento è immediato tramite lo stesso metodo. Si nota infine che le stesse considerazioni si applicano nel caso in cui il protocollo di trasporto utilizzato sia lo *User Datagram Protocol* (UDP) invece che il TCP.

Questo metodo, purtroppo, oggi è applicabile in un numero ridotto di casi, in quanto molte applicazioni, e proprio quelle che generano la maggior parte del traffico, selezionano le porte dinamicamente e, soprattutto, utilizzano porte generiche, per esempio la porta 80, originariamente dedicata all'applicazione World Wide Web, con l'obiettivo di mascherarsi e di rendere un'applicazione (per esempio, di *peer-to-peer file sharing*) indistinguibile da un normale web browsing. Intorno agli anni 2003-2004, quando queste tecniche entrarono massivamente in campo, sembrò di registrare un calo del traffico *Peer-to-Peer*. Questa fu una deduzione errata, ma presto riconosciuta come tale: in realtà le applicazioni *Peer-to-Peer* avevano cominciato a nascondersi (da qui il titolo significativo "Is P2P dying or just hiding?" dell'articolo [7] pubblicato nel 2004, quando si iniziò a riscontrare questo fenomeno).

In conclusione, la mera analisi del numero di porta di destinazione ormai non fornisce all'ISP un'identificazione affidabile dell'applicazione che un flusso di traffico sta trasportando, quindi, è necessario utilizzare metodi diversi e più complessi.

3.2. La packet inspection

Un'altra metodologia tradizionale per l'identificazione del traffico Internet è la *packet inspection*, che consiste nell'osservare il contenuto dei pacchetti per identificare gli scambi protocollari che avvengono attraverso i flussi e, quindi, identificare l'applicazione. Potenzialmente questo è un metodo molto efficace che, esaminando una molteplicità di caratteristiche dei pacchetti, può superare il problema del mascheramento della porta di trasporto. D'altra parte, sussistono alcune problematiche che, in realtà, rendono la *packet inspection* in generale insufficiente. In primo luogo, un'analisi protocollare deve essere di tipo *stateful*, cioè, è necessario memorizzare e tenere aggiornato uno stato per ogni flusso esaminato, al fine di registrare lo stato corrente del protocollo ipotizzato e, quindi, verificare la

bontà dell'ipotesi. Questo produce chiaramente problemi di scalabilità su link ad alta capacità e per un elevato numero di flussi. Inoltre, si deve tenere presente che molte applicazioni cifrano il *payload* dei pacchetti e, di conseguenza, gli scambi protocollari tendono a diventare sempre meno osservabili. Infine, l'osservazione del *payload* dei pacchetti si scontra con problematiche di privacy dei dati personali degli utenti in molti Paesi.

La *packet inspection* può essere resa più scalabile tramite un approccio *stateless*, rinunciando ad intercettare esplicitamente gli scambi protocollari e ricercando particolari stringhe nel *payload* dei pacchetti. Per esempio, l'applicazione *peer-to-peer* eDonkey contiene la stringa '\xe3\x38' nel *payload* del pacchetto IP, alcune *query* Web contengono la stringa '\GET', e così via. Una tale ispezione del *payload* dei pacchetti permette una buona precisione dell'identificazione, ma presenta gli stessi svantaggi dell'ispezione *stateful*, per quanto riguarda la cifratura dei pacchetti e le problematiche relative alla privacy.

Vi sono diversi strumenti, anche aperti, disponibili per l'esecuzione della *packet inspection*. Per esempio, Snort [8] è un software aperto che può eseguire analisi del traffico in tempo reale, effettuando ricerche di particolari stringhe o sequenze di stringhe nei pacchetti. Snort, in tal modo, è anche in grado di rilevare la presenza di vari tipi di attacco nel momento in cui l'attacco stesso si sviluppa. In conclusione, la *packet inspection* è uno strumento efficace, ma la cui validità non è completa so-

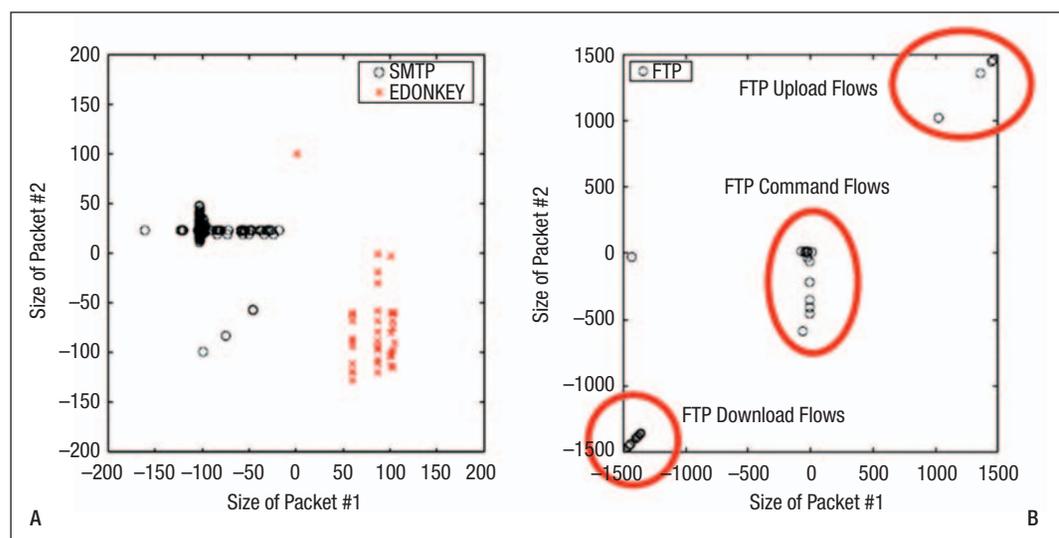
prattutto a causa dell'impossibilità di individuare stringhe caratteristiche nel *payload* di un pacchetto criptato.

3.3. Classificazione basata su caratteristiche statistiche del traffico

Sia la mera analisi della porta di trasporto che (in misura minore) la *packet inspection*, presentano limiti nell'identificazione del traffico Internet i quali possono essere superati da approcci più moderni che esaminano le caratteristiche statistiche dei flussi di traffico, rilasciando la necessità di esaminare il *payload* applicativo trasportato dai pacchetti. Caratteristiche statistiche dei flussi di traffico possono essere le distribuzioni dei tempi di interarrivo dei pacchetti e la loro correlazione, nonché la distribuzione delle lunghezze dei pacchetti, e la loro correlazione. Il presupposto di questo tipo di analisi è che diverse applicazioni presentino differenze osservabili nel processo di generazione dei pacchetti. Questa ipotesi è stata verificata con successo [9]; si è rilevato, per esempio, che diverse applicazioni TCP/IP sono contraddistinte da caratteristiche significativamente diverse nel processo di interarrivo dei pacchetti, delle lunghezze dei pacchetti [10] e della distribuzione delle lunghezze dei pacchetti [11, 12, 13].

Un esempio pratico di questo fatto è illustrato nella figura 3 A [14], che mostra il risultato della misurazione della lunghezza dei primi due pacchetti di flussi di traffico generati dalle applicazioni *Simple Mail Transfer Protocol* (SMTP) ed eDonkey. I grafici della figura riportano sul-

FIGURA 3
Lunghezze dei primi due pacchetti di
A - flussi SMTP ed eDonkey e
B - flussi FTP
(Figura tratta da [14])



l'asse X la lunghezza del primo pacchetto di ogni flusso esaminato e, sull'asse Y , la lunghezza del secondo pacchetto. In questo modo, nella figura ad ogni flusso osservato corrisponde un punto la cui ascissa e ordinata sono rispettivamente la lunghezza del primo e del secondo pacchetto del flusso. Si nota che i punti relativi alle due applicazioni tendono a concentrarsi in regioni decisamente separate, pertanto, l'insieme dei due attributi selezionati costituisce una buona base per la definizione di regioni di decisione efficaci per distinguere SMTP da EDonkey. La figura 3 B [14] mostra, per la stessa coppia di caratteristiche, i punti misurati per un insieme di flussi *File Transfer Protocol* (FTP). In questo caso, si nota che i flussi *upload* (da *client* a *server*) e *download* (da *server* a *client*) occupano regioni separate del piano e altrettanto si può stabilire per i flussi di controllo (che sono necessari per il funzionamento di FTP).

Sulla base di questa constatazione, sono stati introdotti nuovi metodi di classificazione che osservano il processo di generazione dei pacchetti in un flusso e ne identificano alcune caratteristiche statistiche distintive. Questo tipo di approccio risolve contemporaneamente il problema della cifratura dei pacchetti e le questioni di privacy legate all'ispezione dei *payload* dei pacchetti.

Se si desidera identificare l'applicazione che ha generato un flusso di traffico misurandone e valutandone alcune caratteristiche (dette anche attributi) statistiche, è importante tenere conto del fatto che, in generale, un'applicazione stabilisce almeno due flussi unidirezionali. Per esempio, anche nel semplice caso del Web Browsing, allo scaricamento di una pagina che genera un traffico da Web Server a utente, corrisponde un flusso di riscontri TCP da utente verso Web Server. Per ottenere una buona precisione nell'identificazione dell'applicazione è vantaggioso considerare contemporaneamente i due flussi applicativi nelle due direzioni. Tali flussi sono chiamati *forward* (per esempio un file scaricato) e *backward* (per esempio, i riscontri TCP relativi allo scaricamento nella direzione *forward*). Alcuni attributi usualmente esaminati per identificare i flussi di traffico sono riportati nella tabella 1.

La tabella 1 riporta quattro gruppi di attributi. Si osserva che i gruppi 1 e 2 fanno riferimento a caratteristiche complessive di un flusso, che pos-

sono essere determinate solo dopo che il flusso ha terminato la sua fase attiva e viene abbattuto. Infatti, l'attributo 1 (valore minimo, massimo, medio e deviazione standard della lunghezza dei pacchetti nella direzione "*forward*") può essere quantificato solo dopo che si è registrata la lunghezza di tutti i pacchetti del flusso esaminato e si è elaborata la statistica complessiva di queste lunghezze. Le stesse considerazioni valgono per gli attributi del gruppo 2. Al contrario, gli attributi dei gruppi 3 e 4 richiedono l'osservazione di pochissimi pacchetti (dei primi tre o dei primi cinque) di un flusso. Quindi, gli attributi dei gruppi 3 e 4 permettono l'identificazione dei flussi "*on the fly*", velocissima e in tempo reale, che può essere determinata già nei primi istanti di vita del flusso. Gli attributi dei gruppi dei gruppi 1 e 2 corrispondono ad un'identificazione in generale molto più lenta e che non può essere considerata in tempo reale. Come si vedrà nel seguito, gli attributi dei gruppi 3 e 4 sono (forse sorprendentemente) molto efficaci - soprattutto quelli del gruppo 4 - e quindi permettono un'identificazione sia veloce che precisa.

Un sistema di classificazione dei flussi seleziona un insieme di N attributi, definendo così uno spazio N -dimensionale dove la i -esima dimensione quantifica il valore che assume l' i -esimo attributo. Osservando un flusso di traffico si possono misurare gli attributi selezionati e, quindi, attribuire al flusso esaminato una coordinata (cioè, un punto) nello spazio N -dimensionale degli attributi. Se gli attributi sono selezionati in modo appropriato, si osserva che i flussi di traffico generati da una specifica applicazione tendono a concentrarsi, nello spazio N -dimensionale degli attributi, in certe regioni e non in altre occupate da altre applicazioni. L'esempio riportato nella figura 4 illustra il procedimento nel caso di due attributi. Nella figura 4 A è mostrata la regione dello spazio bidimensionale, relativo ai due attributi selezionati, nella quale vanno a concentrarsi i punti relativi ai flussi di traffico generati da una specifica applicazione A . Questa regione può essere ottenuta mediante osservazione di flussi di traffico per i quali è noto che l'applicazione che li ha generati è proprio A .

Una volta che la regione relativa all'applicazione A è definita, si è diviso il piano di identificazione in due regioni: la regione associata

Gruppo di attributi 1	
1.	Valore minimo, massimo, medio e deviazione standard della lunghezza dei pacchetti nella direzione "forward";
2.	Valore minimo, massimo, medio e deviazione standard della lunghezza dei pacchetti nella direzione "backward";
3.	Valore minimo, massimo, medio e deviazione standard dei tempi di interarrivo dei pacchetti nella direzione "forward";
4.	Valore minimo, massimo, medio e deviazione standard dei tempi di interarrivo dei pacchetti nella direzione "backward";
5.	Valore del campo "protocol" dell'header dei pacchetti IP;
6.	Numero totale di flag TCP URG e PUSH nella direzione "forward";
7.	Numero totale di flag TCP URG e PUSH nella direzione "backward";
Gruppo di attributi 2	
8.	Durata del flusso;
9.	Numero totale di byte e pacchetti nella direzione "forward";
10.	Numero totale di byte e pacchetti nella direzione "backward";
Gruppo di attributi 3	
11.	Lunghezza dei primi tre pacchetti nella direzione "forward";
12.	Lunghezza dei primi tre pacchetti nella direzione "backward";
13.	Tempi di interarrivo dei primi tre pacchetti nella direzione "forward";
14.	Tempi di interarrivo dei primi tre pacchetti nella direzione "backward";
Gruppo di attributi 4	
15.	Lunghezza dei primi cinque pacchetti nella direzione "forward";
16.	Lunghezza dei primi cinque pacchetti nella direzione "backward";
17.	Tempi di interarrivo dei primi cinque pacchetti nella direzione "forward";
18.	Tempi di interarrivo dei primi cinque pacchetti nella direzione "backward".

TABELLA 1
Gruppi di attributi
frequentemente
utilizzati nella
classificazione del
traffico Internet

all'applicazione *A* e la regione complementare, associata a tutte le applicazioni diverse da *A*. A questo punto, osservando un flusso per il quale l'applicazione che lo ha generato è ignota, si misureranno le due caratteristiche selezionate e si identificherà quindi un punto nel piano. Se tale punto cade all'interno della regione associata all'applicazione *A*, si deciderà che il flusso è stato generato da *A*, altrimenti, si decide che il flusso è stato generato da un'applicazione diversa. La figura 4 B esemplifica il procedimento, mostrando con punti neri i flussi di traffico che **realmente** sono stati generati da *A* e con punti bianchi i flussi di traffico che **in realtà** sono stati gene-

rati da un'applicazione diversa da *A*. I punti neri che sono compresi nella regione associata ad *A* corrispondono a classificazioni corrette, così come i punti bianchi che cadono al di fuori della regione associata ad *A*. D'altra parte, sono possibili errori. Infatti, un flusso generato da *A* potrebbe essere classificato come "non-*A*". Nella figura 4 B quest'evenienza è rappresentata dal punto nero fuori dalla regione associata ad *A*, ed è denominato "Falso Negativo". Un altro tipo di errore è il "Falso Positivo", cioè, un flusso che viene classificato come originato dall'applicazione *A*, ma in realtà non lo è. Nella figura 4 B ciò è rappresentato dal punto bianco compreso nella re-

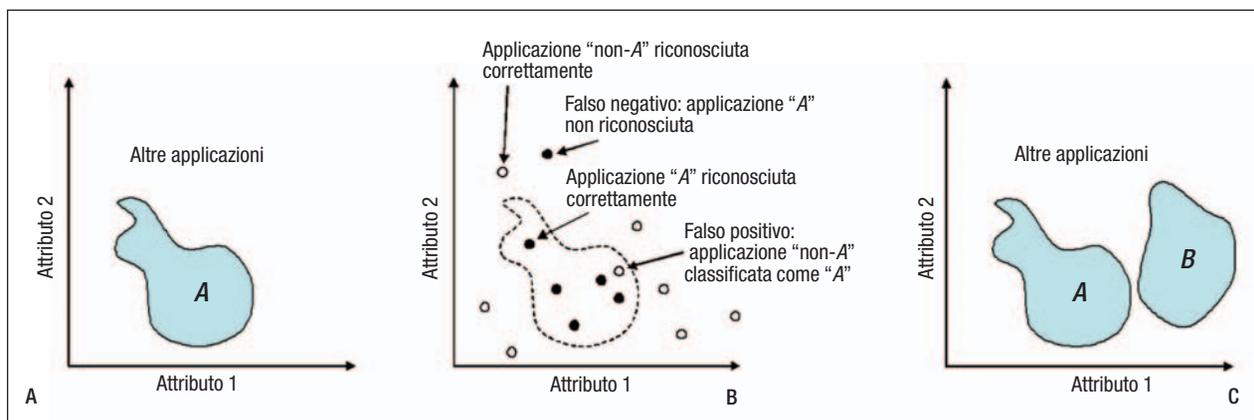


FIGURA 4
 Classificazione delle applicazioni sulla base di caratteristiche preselezionate dei flussi

gione associata ad A. In generale si desidera identificare contemporaneamente più di un'applicazione e ciò si ottiene, come mostrato nella figura 4 C, definendo nello spazio N -dimensionale le regioni associate ad ognuna delle applicazioni di interesse (le applicazioni A e B nell'esempio).

Due importanti metriche per la quantificazione della precisione di un sistema di identificazione delle applicazioni sono la percentuale di falsi negativi e di falsi positivi, definiti rispettivamente *False Negative Rate* (FNR) e *False Positive Rate* (FPR), che dovrebbero essere piccoli. A queste metriche si aggiungono: il *True Negative Rate* (TNR) e il *True Positive Rate* (TPR).

3.4. Il machine learning supervisionato

Una delle principali problematiche nell'applicazione delle tecniche di identificazione basate sull'analisi degli attributi dei flussi di traffico è la costruzione delle regole tramite le quali si identifica l'applicazione che ha generato un flusso di traffico. Alcuni metodi di *machine learning* costruiscono effettivamente lo spazio N -dimensionale degli attributi e le relative regioni di decisione. Altre tecniche elaborano insiemi di regole di decisione che non coinvolgono direttamente la costruzione delle regioni nello spazio degli attributi. Anche in quest'ultimo caso, la selezione di attributi che distinguono chiaramente le differenze tra le applicazioni in esame è un prerequisito essenziale.

Alcune tecniche di *machine learning* utilizzano un insieme di esempi pre-classificati per inferire automaticamente una serie di regole di clas-

sificazione tramite le quali si procede all'identificazione dei flussi di traffico non compresi nell'esempio fornito inizialmente. Questo tipo di tecnica è detta "*supervisionata*", in quanto l'algoritmo di identificazione del traffico è preliminarmente addestrato con informazioni preconfezionate. Tramite le regole di classificazione costruite sulla base degli esempi forniti, l'algoritmo di classificazione fornisce, a fronte di un input costituito da un flusso di traffico, una classificazione dello stesso in una delle possibili categorie (applicazioni) definite nell'esempio iniziale utilizzato per l'addestramento. L'inizializzazione di un algoritmo di *machine learning* supervisionato prevede:

- la fase di *addestramento*, nella quale il motore di classificazione esamina l'esempio fornito ed elabora il modello di classificazione;
- la successiva fase di *testing*, nella quale si forniscono al motore di classificazione alcuni flussi generati da applicazioni note, per verificare la precisione della classificazione operata. Una volta esaurita la fase di *testing*, il motore di classificazione può essere messo in produzione.

Nel caso della figura 4 C, un esempio d'addestramento potrebbe essere strutturato come una serie di flussi f_i , ad ognuno dei quali è associata la terna (x_{1i}, x_{2i}, y_i) , dove x_{1i} e x_{2i} sono il valore dell'attributo 1 e 2, rispettivamente, dell' i -esimo flusso dell'esempio, e y_i assume il valore A o B, a seconda che il flusso f_i sia stato generato dall'applicazione A o da B. Sulla base di un esempio sufficientemente corposo, il motore di *machine learning* è in grado di costruire le regioni relative ad A e B mostrate

nella figura 4 C. La generalizzazione nel caso di un insieme di N caratteristiche è intuitiva. Un problema significativo degli algoritmi di *machine learning* supervisionati è che gli esempi per l'addestramento devono essere costituiti da flussi (numerosi) correttamente pre-identificati. Anche la fase di test deve avvenire alimentando il motore di classificazione con flussi non appartenenti all'esempio iniziale e anch'essi correttamente pre-identificati. La creazione di collezioni di flussi correttamente identificati per l'addestramento e per il testing è una fase lunga e costosa. Esempi di tecniche di machine learning supervisionate sono il *Naïve Bayes*, le *Bayesian Network*, il *C4.5 Decision Tree*, e le *Support Vector Machines*.

3.4.1. IL NAÏVE BAYES

Il metodo *Naïve Bayes*, come dice il nome, si basa sul noto teorema sulle probabilità condizionate di Bayes, che è utilizzato in quest'ambito per individuare l'applicazione che più verosimilmente ha generato un flusso di traffico osservandone alcuni attributi selezionati, per esempio, tra quelli della tabella 1. Per ipotesi, si assuma di avere due classi (applicazioni), C_1 e C_2 , e un insieme di due attributi, X_1 e X_2 , che possono essere le lunghezze in byte dei primi due pacchetti di un flusso. Durante la fase di addestramento, osservando un insieme di flussi noti generati dalle applicazioni C_1 e C_2 , si determinano le distribuzioni di probabilità delle lunghezze dei primi due pacchetti dei flussi generati dall'applicazione C_1 , e il calcolo è ripetuto per l'applicazione C_2 . In seguito, dato un flusso ignoto, se ne misurano gli attributi X_1 e X_2 (cioè, le lunghezze dei primi due pacchetti) e applicando il teorema di Bayes si individua l'applicazione che più verosimilmente ha generato il flusso.

Attributo	Possibili valori
Tempo atmosferico	Sole, nuvoloso, pioggia
Temperatura	Variabile continua
Umidità	Variabile continua
Vento	Sì, no

TABELLA 2

Attributi e i possibili valori ad essi associati

3.4.2. L'ALGORITMO C.45 "DECISION TREE"

L'algoritmo C4.5 crea una struttura decisionale ad albero, nel quale i nodi rappresentano gli attributi e gli archi rappresentano i valori che connettono gli attributi. La costruzione dell'albero decisionale è la fase più complessa di C.45 e avviene durante l'addestramento applicando procedure complesse per ottimizzarne la costruzione, tentando di minimizzare i casi ambigui, che portano ad errori di classificazione. In questa sede si propone un semplice esempio per illustrare i concetti generali applicati dall'algoritmo. Si suggerisce la lettura di [15] per un approfondimento.

Si consideri una casistica di giocatori di golf che, in base alle condizioni atmosferiche, decidono se giocare o non giocare. Gli attributi del tempo atmosferico considerati sono illustrati nella tabella 2 e ad essi sono associati i possibili valori, che possono essere variabili discrete o continue. La sequenza di addestramento registra le decisioni effettivamente prese da alcuni giocatori ed è illustrata nella tabella 3.

L'elaborazione dei dati da parte di C.45 porta all'albero decisionale mostrato nella figura 5 A. Si nota che, se il tempo è nuvoloso si prenderà sempre la decisione di giocare, mentre se c'è il sole o se piove la decisione dipende da altri fattori come umidità e vento. La temperatura è stata eliminata come variabile decisionale, in quanto l'algoritmo di costruzione dell'albero ha stabilito, sulla base della sequenza di addestramento, che il guadagno informativo che si otterrebbe includendo tale variabile non è sufficiente a giustificarne l'utilizzo e, quindi, a rendere più complesso e ramificato l'albero. Questa eliminazione di un attributo è generata da speciali procedure di "potatura" dell'albero decisionale operate da C.45.

La figura 5 B mostra che la costruzione dell'albero decisionale di C.45 porta a definire, nello spazio degli attributi, delle zone di decisione. Lo spazio degli attributi è tridimensionale e non quadridimensionale, in quanto la variabile temperatura è stata eliminata da C.45.

3.4.3. BAYESIAN NETWORKS

Una *Bayesian Network* è una struttura decisionale che consiste in un grafo dove i nodi rappresentano attributi o classi e gli archi de-

Tempo atmosferico	Temperatura	Umidità	Vento	Si gioca?
Sole	29.4	85%	No	No
Sole	26.6	90%	Si	No
Nuvoloso	28.3	78%	No	Si
Pioggia	21.1	96%	No	Si
Pioggia	20	80%	No	Si
Pioggia	18.3	70%	Si	No
Nuvoloso	17.7	65%	Si	Si
Sole	22.2	95%	No	No
Sole	20.5	70%	No	Si
Pioggia	23.8	80%	No	Si
Sole	23.8	70%	Si	Si
Nuvoloso	22.2	90%	Si	Si
Nuvoloso	27.2	75%	No	Si
Pioggia	21.6	80%	Si	No

TABELLA 3
Dati di addestramento

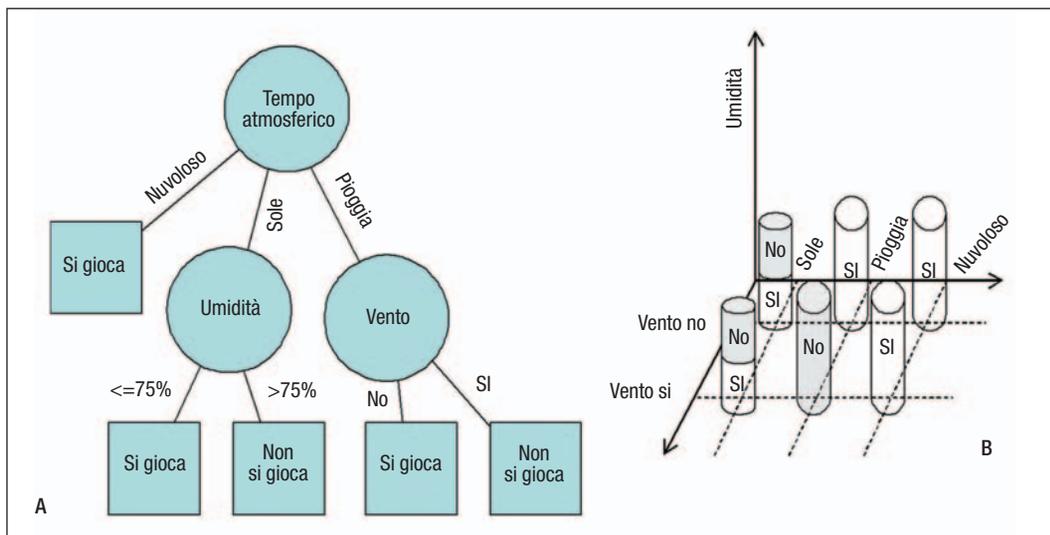


FIGURA 5
A - Albero decisionale nell'esempio di applicazione dell'algoritmo C.45; B - rappresentazione delle zone di decisione costruite dall'albero decisionale nello spazio degli attributi

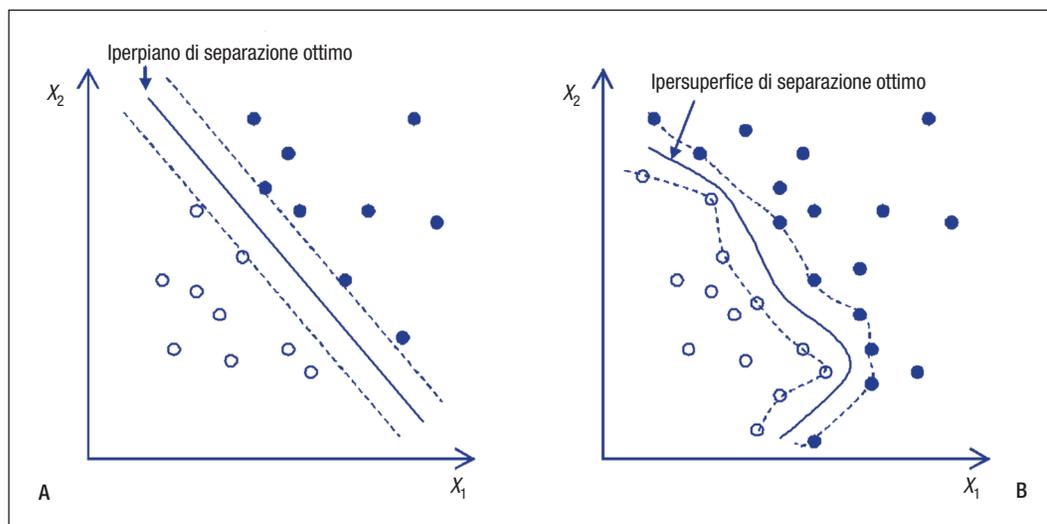
scrivono le mutue relazioni tra i nodi. Il peso di un arco è una probabilità condizionata e quantifica l'entità della relazione tra i due nodi che esso collega. Dato il grafo decisionale (costruito nella fase di addestramento) e le probabilità condizionate che pongono in relazione i nodi del grafo, la classificazione di un flusso è un'operazione relativamente semplice e, anche in questo caso, la vera criticità è la

costruzione del grafo decisionale in fase di addestramento. Per un approfondimento sulle Bayesian Networks si veda [16].

3.4.4. SUPPORT VECTOR MACHINES

Le *Support Vector Machines* (SVM) sono una tecnica generale di *pattern recognition* che può essere utilizzata per la classificazione del traffico Internet mediante *machine learning*

FIGURA 6
Separazione delle
zone di decisione
operata dalle
Support Vector
Machines



supervisionato. Le SVM affrontano direttamente il problema della costruzione delle zone di decisione nello spazio degli attributi. La figura 6 mostra, nel caso bidimensionale di due attributi, i valori (punti) delle istanze presenti nella sequenza di addestramento. Sono presenti due classi (applicazioni) distinte da un cerchio vuoto e un cerchio nero. Nella figura 6 A si mostra il caso in cui le istanze delle due classi si raggruppano in due distinte zone, separabili da una linea. Questa linea in generale è denominata iperpiano di separazione (la denominazione “iperpiano” deriva dal fatto che nel generale caso di N attributi la linea di separazione è in realtà un piano in $N-1$ dimensioni). L’attribuzione di una nuova istanza ad una classe, quando il motore di classificazione è in esercizio, risulta molto semplice se la linea di separazione è un iperpiano (e quindi lineare). L’operazione è più complessa nel caso generale in cui la linea di separazione non è più un iperpiano, ma una superficie più generica, come mostrato nella figura 6 B. In questo caso, le SVM operano una trasformazione dello spazio degli attributi mediante la quale la linea di separazione diventa un iperpiano e quindi operano nello spazio trasformato.

3.5. Tecniche di clustering

Le tecniche di classificazione supervisionate sono basate sulla conoscenza a priori di quali siano le classi (applicazioni) da discriminare. Le tecniche di *clustering*, invece, non sono provviste di questa informazione e cercano

autonomamente correlazioni e somiglianze/differenze negli attributi dei flussi osservati, rintracciando schemi ricorrenti. In tal modo, un algoritmo di *clustering* raggruppa autonomamente le istanze osservate in regioni nello spazio degli attributi, andando ad individuare quei gruppi di istanze che formano nello spazio degli attributi “costellazioni” abbastanza raggruppate (si veda la Figura 3 A). I gruppi così creati possono essere *hard* (le regioni sono completamente separate) o *soft* (le regioni sono parzialmente sovrapposte). Un buon algoritmo di *clustering* tende a produrre *cluster* che sono *dominati* da un’applicazione (cioè, una percentuale elevata dei flussi concentrati nel cluster appartengono ad una singola applicazione, che domina il *cluster*) e nei quali la presenza di flussi appartenenti ad altre applicazioni è marginale. Ciò garantisce che, quando il motore di classificazione utilizzerà i *cluster* per classificare i flussi, si avrà una buona precisione, cioè, pochi falsi positivi e pochi falsi negativi.

Un esempio di algoritmo di *clustering* è presentato in [14, 17] (si veda la Figura 3 A), dove il grado di somiglianza tra due flussi è quantificato dalla distanza euclidea dei rispettivi punti nel piano che rappresenta le due caratteristiche misurate. Una volta definiti i *cluster*, un flusso ignoto in osservazione è assegnato al *cluster* il cui centro, nel piano di Figura 3 A, è più vicino al punto associato al nuovo flusso.

Una volta creati i *cluster*, è necessario assegnare ad ogni *cluster* l’applicazione che lo

domina, in modo tale da abilitare la classificazione di nuove istanze quando il classificatore è messo in esercizio. È a questo punto che anche un algoritmo di *clustering* deve essere addestrato con una sequenza per la quale le applicazioni che hanno generato le istanze sono note. L'algoritmo di *clustering* associerà ad ogni *cluster* l'applicazione che più lo rappresenta. In tal senso, anche un classificatore che adotta una tecnica di *clustering* prevede una fase supervisionata, denominata anche *fase di labeling*. In [14], si osserva che utilizzando i primi 5 pacchetti di un flusso si ottiene una buona separazione tra i *cluster* e un'efficiente classificazione.

3.5.1. PRESTAZIONI DEGLI ALGORITMI DI MACHINE LEARNING

Non esiste in letteratura uno studio completo che analizzi in condizioni omogenee le prestazioni di tutti gli algoritmi di *machine learning* esistenti. Esistono comunque alcuni lavori che esaminano e raffrontano alcuni casi significativi, per esempio [18], nel quale si comparano le prestazioni di un algoritmo di classificazione supervisionato basato sulle Bayesian Networks [16], tecniche di *clustering* basate sull'algoritmo delle K-means [17], algoritmi Bayesiani (basati su tecniche derivate dal *Naïve Bayes*, ma dotate di soluzioni aggiuntive avanzate per la riduzione degli errori di classificazione) che includono i tempi di interarrivo dei pacchetti negli attributi esaminati [19] e, infine, algoritmi supervisionati, che applicano C.45 [18].

Gli algoritmi sono addestrati e testati fornendo in input tre diverse tracce pubblicamente disponibili: le tracce auckland-vi-20010611 e auckland-vi-20010612, che sono state raccolte sul medesimo link di rete e la traccia nzix-

ii-20000706, raccolta su un link diverso. Gli algoritmi sono addestrati con la traccia auckland-vi-20010611 (il *campione A-addestramento*), sono poi verificati tramite la traccia auckland-vi-20010612 (il *campione A-test*) e con la traccia nzix-ii-20000706 (il *campione B-test*). Lo scopo di verificare gli algoritmi in due casi, con una traccia raccolta sullo stesso link utilizzato per l'addestramento e su di un link diverso, è quello di analizzare la "portabilità" dell'algoritmo di classificazione, cioè, la possibilità di eseguire un unico addestramento e poi di utilizzare l'algoritmo in punti della rete diversi da quello sul quale l'addestramento è stato svolto. I vantaggi di un motore di classificazione "portabile" sono evidenti: si possono ridurre notevolmente i costi di addestramento in quanto un solo ciclo di addestramento è sufficiente per operare su tutti i link della rete.

Le metriche utilizzate per quantificare le prestazioni dei diversi algoritmi sono il *True Positive Rate* (TPR) e il *False Positive Rate* (FPR). Le categorie di applicazioni esaminate sono HTTP, SMTP, POP3, FTP, DNS, Telnet.

La tabella 4 riporta il tasso di veri positivi (TPR) dei quattro algoritmi esaminati, addestrati con il *campione A-addestramento* e testati con il *campione A-test*, utilizzando i gruppi di attributi 1 e 4 (Tabella 1). Nella tabella 5 si riporta il tasso di falsi positivi, nella stessa situazione sperimentale. Si nota che non esiste un algoritmo migliore degli altri in tutti i casi. D'altra parte, l'algoritmo basato su C.45 è in molti casi il migliore, con poche eccezioni. Si riscontra inoltre che la precisione degli algoritmi esaminati è abbastanza buona (alto tasso di veri positivi e basso tasso di falsi positivi).

Una quantificazione della portabilità dell'algo-

Protocollo	Bayesian Networks [16]	Clustering [17]	Tecniche Bayesiane avanzate [19]	C.45 [18]
HTTP	89.2%	96.2%	91.8%	99.7%
SMTP	97.2%	90.1%	94.5%	98.6%
POP3	97.2%	93.4%	94.6%	-
FTP	97.9%	92.4%	-	94.8%

TABELLA 4

Tasso di veri positivi (TPR) nel caso di algoritmi addestrati con il *campione A-addestramento* e testati con il *campione A-test*; sono stati utilizzati gli attributi dei gruppi 1 e 4 (Tabella 1)

Protocollo	Bayesian Networks [16]	Clustering [17]	Tecniche Bayesiane avanzate [19]	C.45 [18]
HTTP	10.4%	1.3%	6.4%	0.1%
SMTP	2.3%	0.1%	3.1%	1.4%
POP3	2.3%	0.7%	3.1%	-
FTP	1.8%	0.4%	-	0.5%

TABELLA 5

Tasso di falsi (FPR) positivi nel caso di algoritmi addestrati con il campione A-addestramento e testati con il campione A-test; sono stati utilizzati gli attributi dei gruppi 1 e 4 (Tabella 1)

	Campione A-test		Campione B-test	
	TPR	FPR	TPR	FPR
DNS	100%	0.4%	99%	0.1%
FTP	95%	0.5%	77%	1.4%
Telnet	92%	0.1%	84%	0.4%
SMTP	98%	1.4%	95%	3%
HTTP	99%	0.1%	99%	0.2%

TABELLA 6

Per l'algoritmo basato su C.45, tasso di veri e falsi positivi con addestramento con il campione A-addestramento e test con il campione A-test e il campione B-test; sono stati utilizzati gli attributi dei gruppi 1 e 4 (Tabella 1)

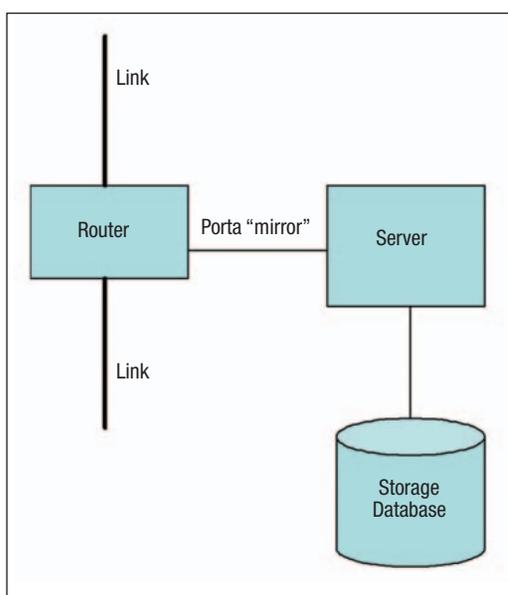


FIGURA 7
Allestimento per la misurazione dei pacchetti e dei flussi di traffico

ritmo basato su C.45 è riportata nella tabella 6, dove si considerano le applicazioni DNS, FTP, Telnet, SMTP e HTTP e il test è effettuato sia con il campione A-test sia con il campione B-test. Le buone prestazioni dell'algoritmo che agisce sullo stesso link di rete sul quale è stato

addestrato non si riscontrano in ugual misura quando il classificatore è messo in esercizio su di un link diverso. In questo caso, le prestazioni non sono pessime (a parte il tasso di veri positivi per l'applicazione FTP). In ogni caso, sono necessari ulteriori miglioramenti per poter ottenere un algoritmo veramente portabile.

3.6. Misurazione e classificazione del traffico in pratica

Sia che si desideri misurare le caratteristiche aggregate dei flussi di traffico, sia che si voglia operare un'identificazione delle applicazioni, è necessario prelevare il traffico dalla rete per poi analizzarlo. Nella figura 7 è mostrato un tipico allestimento per la raccolta e l'analisi dei dati di traffico. Per misurare il traffico su di un link, si deve dotare il router che lo gestisce di una *mirror port* la quale replica il traffico presente sul link selezionato. La *mirror port* è collegata direttamente ad un server che memorizza il traffico su un database. Un tale allestimento presenta diverse criticità tecniche e ha costi non trascurabili. Infatti, se il link da esaminare ha grande capacità e i flussi pre-

senti sono numerosi, il numero di pacchetti da memorizzare è enorme (può essere dell'ordine delle centinaia di migliaia di pacchetti al secondo per un link a più di un Gbit/s) e questo richiede ingenti capacità di memorizzazione (per esempio circa 1 Tbyte/h di registrazione su un link da 1 Gbit/s). Inoltre, la velocità di accesso del database deve essere estremamente elevata, in quanto si deve poter lavorare alla velocità di linea del link. Similmente, il server deve essere adeguatamente attrezzato per lavorare a velocità così elevate. Esistono hardware specializzati disponibili sul mercato nella forma di schede da installare nei server, in grado di collegarsi alla *mirror port* e di lavorare alle velocità richieste, sollevando così la CPU del server dal compito di lavorare a velocità estremamente elevate. Naturalmente, il costo di questo tipo di schede è alto.

Per la misurazione dei flussi lo strumento più adottato è NetFlow della Cisco, applicazione inclusa nel sistema operativo IOS dei router Cisco. Tramite NetFlow è possibile registrare e analizzare le caratteristiche dei flussi di traffico che attraversano un router sotto osservazione, seguendo per esempio lo schema applicativo mostrato nella figura 7.

Per la classificazione delle applicazioni uno strumento largamente utilizzato è NBAR (*Network Based Application Recognition*), sempre della Cisco, in grado di integrarsi con NetFlow. La classificazione delle applicazioni da parte di NBAR è eseguita essenzialmente sull'osservazione dei numeri di porta e sulla packet inspection. Esistono anche strumenti forniti da altre case, come il NetFlow Analyzer [20], che sono in grado di acquisire le misurazioni di traffico effettuate da NetFlow e di elaborare analisi dei dati, offrendo viste personalizzabili (tramite interfaccia web) sulla tipologia e caratteristiche del traffico in rete. Per esempio, si possono ottenere rapporti tabellari e grafici sulle applicazioni utilizzati, gli utenti e più in generale sul traffico presente nei diversi punti della rete (velocità, volume, numero di pacchetti, utilizzazione dei link), il tutto anche in tempo reale.

Per quanto riguarda l'applicazione pratica delle tecniche di *machine learning* per la classificazione del traffico si è ancora in uno stadio iniziale (la ricerca in questo settore è ancora molto intensa). Alcuni operatori di telecomuni-

cazioni stanno effettuando sperimentazioni, in alcuni casi, in collaborazione con gruppi di ricerca universitari.

4. CONCLUSIONI

L'analisi del traffico trasportato dai flussi e l'identificazione delle applicazioni che generano i flussi di traffico sono due attività di importanza strategica che permettono all'ISP di gestire al meglio le risorse della sua rete e la qualità del servizio di trasporto del traffico generato dalle applicazioni in rete. L'analisi del traffico fornisce utili informazioni su quali sono i flussi critici (che trasportano un volume di traffico maggiore e che presentano velocità di trasmissione molto elevate) che vanno gestiti con particolare attenzione. L'identificazione del traffico permette all'ISP di conoscere le applicazioni che generano i flussi di traffico e dà quindi indicazioni su quale sia il miglior modo di trattare i flussi di traffico cercando di fornire i livelli di qualità del servizio richiesti dalle applicazioni e, quindi, dagli utenti.

Entrambe le attività sono complesse e richiedono un costoso dispiegamento di mezzi hardware/software. Inoltre, la difficoltà dell'identificazione del traffico comporta l'utilizzo di algoritmi complessi ed è impossibile ottenere un'identificazione sicura al 100% dell'applicazione che ha generato un dato flusso di traffico. Si è visto come le più avanzate tecniche di riconoscimento delle applicazioni implementano metodologie di *machine learning*, nelle quali l'algoritmo di identificazione è in primo luogo addestrato con sequenze di traffico pre-identificate, tramite le quali l'algoritmo costruisce autonomamente una serie di regole di decisione. Le migliori tecniche di *machine learning* portano a percentuali di errore in generale contenute (qualche punto percentuale, o frazioni di per cento per alcune applicazioni facilmente riconoscibili). D'altra parte, se si mette in esercizio un identificatore di traffico su un link diverso dal quale si è prelevata la sequenza di traffico utilizzata per il suo addestramento, si nota che la precisione dell'identificazione tende a degradare significativamente. La concezione di un algoritmo di *machine learning* "portatile" che possa funzionare correttamente anche su link sui quali non è stato addestrato è attualmente oggetto di ricerca.

Bibliografia

- [1] Callado A., Kamienski C., Szabó G., Geró B.P., Kelner J., Member S.F., Sadok D.: A Survey on Internet Traffic Identification. *IEEE Communications Surveys & Tutorials*, Vol. 11, n. 3, Third Quarter 2009, p. 37-52.
- [2] DecinaM., Giacomazzi P.: Il futuro del protocollo IP. *Mondo Digitale*, n.2, giugno 2007, p. 17-29.
- [3] http://mediablog.corriere.it/2010/04/-da_corte_usa_colpo_alla_neutra.html
- [4] Brownlee N., Claffy K.C.: Understanding Internet traffic streams: dragonflies and tortoises. *IEEE Communications Magazine*, Vol. 40, n. 10, October 2002, p. 110-117.
- [5] Wallerich J., Dreger H., Feldmann A., Krishnamurthy B., Willinger W.: A methodology for studying persistency aspects of internet flows. *SIGCOMM Comput. Commun. Rev.*, Vol. 35, n. 2, April 2005, p. 23-36.
- [6] <http://www.iana.org/assignments/port-numbers>
- [7] Karagiannis T., Broido A., Brownlee N., Claffy K.C., Faloutsos M.: *Is P2P dying or just hiding?* IEEE Global Telecommunications Conference, November 2004.
- [8] <http://www.snort.org/>
- [9] Paxson V.: Empirically derived analytic models of wide-area TCP connections. *IEEE/ACM Trans. Networking*, Vol. 2, n. 4, 1994, p. 316-336.
- [10] Dewes C., Wichmann A., Feldmann A.: *An analysis of Internet chat systems*. In: ACM/SIGCOMM Internet Measurement Conference, 2003, Miami, Florida, USA, October 2003.
- [11] Claffy K.: *Internet traffic characterisation*. PhD Thesis, University of California, San Diego, 1994.
- [12] Lang T., Armitage G., Branch P., Choo H.-Y.: *A synthetic traffic model for Half-life*. In: Proc. Australian Telecommunications Networks and Applications Conference, ATNAC2003, Melbourne, Australia, December 2003.
- [13] Lang T., Branch P., Armitage G.: *A synthetic traffic model for Quake 3*. In: Proc. ACM SIGCHI International Conference on Advances in computer entertainment technology (ACE2004), Singapore, June 2004.
- [14] Bernaille L., Teixeira R., Akodkenou I., Soule A., Salamatian K.: Traffic classification on the fly. *SIGCOMM Comput. Commun. Rev.*, Vol. 36, n. 2, 2006, p. 23-26.
- [15] Kohavi R., Quinlan J.R., (Will Klossgen, Jan M. Zytkow, editors): *Decision-tree discovery*. Handbook of Data Mining and Knowledge Discovery. Oxford University Press, 2002, p. 267-276.
- [16] Auld T., Moore A.W., Gull S.F.: Bayesian neural networks for internet traffic classification. *IEEE Transactions on Neural Networks*, Vol. 18, n. 1, Jan. 2007, p. 223-239.
- [17] Bernaille L., Teixeira R., Salamatian K.: *Early application identification*. In: The 2-nd ADET-TI/ISCTE CoNEXT Conference, Dec. 2006.
- [18] Verticale G., Giacomazzi P.: *Performance Evaluation of a Machine Learning Algorithm for Early Application Identification*. Proc. of the International Multiconference on Computer Science and Information Technology, 2008, p. 845-849.
- [19] Crotti M., Dusi M., Gringoli F., Salgarelli L.: Traffic classification through simple statistical fingerprinting. *SIGCOMM Comput. Commun. Rev.*, Vol. 37, n. 1, 2007, p. 5-16.
- [20] <http://www.manageengine.com/products/-netflow/netflow-traffic-analysis.html>

PAOLO GIACOMAZZI si è laureato in Ingegneria Elettronica presso il Politecnico di Milano nel 1990 ed ha conseguito il Master in tecnologia dell'informazione al CEFRIEL. Dal 1992 al 1998 è stato ricercatore con il Politecnico di Milano dove ora è professore associato di telecomunicazioni. L'attività didattica e la ricerca riguardano la qualità del servizio nella rete Internet multimediale, le reti radiomobili B3G e la sicurezza nelle reti di telecomunicazioni. È editor del *IEEE Network Magazine* ed è editor della *Book Reviewing Feature* del *IEEE Network Magazine*. E-mail: giacomaz@elet.polimi.it

BUON COMPLEANNO Mr. HOLLERITH

Centocinquant'anni fa nasceva Herman Hollerith, l'ideatore delle macchine a schede perforate che, fino all'avvento dei moderni computer, rimasero gli unici strumenti per l'elaborazione automatica dei dati. Per quasi settant'anni tale tecnologia costituì il "cervello" delle aziende produttrici, degli enti e dei servizi pubblici, delle biblioteche, delle banche e delle assicurazioni. Il loro uso si estese anche alla ricerca scientifica e alla crittoanalisi. Un rilevante sostegno all'inventore americano venne da un nostro connazionale, il celebre statistico italiano Luigi Bodio.

1. HERMAN HOLLERITH

Herman Hollerith (Figura 1) nacque a Buffalo, nello stato di New York il 29 febbraio 1860¹, figlio di due immigranti tedeschi fuggiti dalla loro patria a seguito della fallita Rivoluzione di Marzo del 1848. Il padre era insegnante di greco e latino, ma morì presto, lasciando Herman e la madre in condizioni tutt'altro che agiate. L'aiuto dei parenti materni permise al giovane Herman di continuare gli studi. L'istruzione primaria non fu facile per Hollerith che soffriva di una lieve dislessia, e si dovette ricorrere ad insegnanti privati. Più tardi però frequentò con profitto la *Columbia School of Mines*, una scuola di formazione secondaria per ingegneri, diplomandosi con distinzione nel 1879. Subito dopo il diploma, un suo insegnante lo aiutò a trovare impiego presso il Census Bureau di Washington, l'organismo federale incaricato della raccolta ed elaborazione dei dati del censimento, con la carica di Ca-

po Agente Speciale. In questa veste collaborò con il dottor John Shaw Billings all'elaborazione delle statistiche sanitarie del censimento del 1880 [1, 2, 3]. Ventidue anni più vecchio di Hollerith, Billings era un'importante figura nel

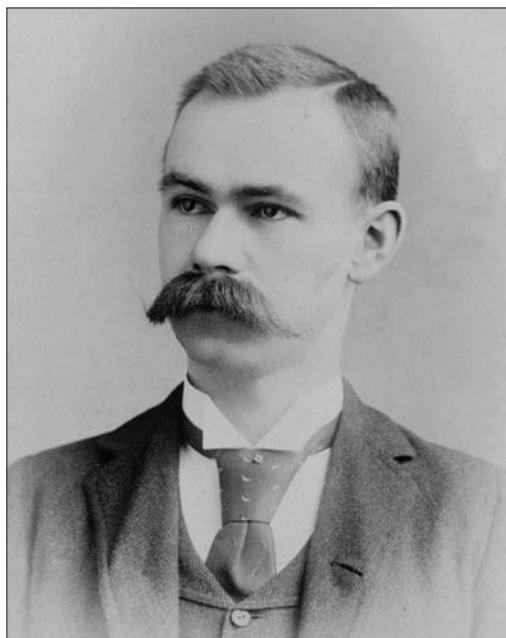
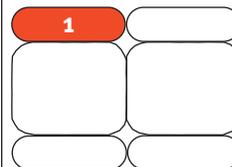


FIGURA 1
Herman Hollerith
(1860 - 1929)

¹ Hollerith usava festeggiare il suo genetliaco ogni quattro anni [1].



Silvio Hénin



mondo della sanità americana, fondatore della *National Library of Medicine* ed esperto di organizzazione sanitaria, progettò fra l'altro l'Ospedale John Hopkins di Baltimora, ancora oggi uno dei più rinomati nosocomi del mondo [4]. L'influenza di Billings sul giovane Hollerith fu, come vedremo, fondamentale.

All'epoca il Census Bureau non era ancora un'istituzione stabile, ma si costituiva solo in occasione delle decennali rilevazioni demografiche dell'Unione, per poi essere sciolto una volta terminato il compito [1, 5]. Hollerith si trovò presto senza impiego e, dopo un breve periodo come istruttore al *Massachusetts Institute of Technology* (MIT), lavorò per un anno all'Ufficio Brevetti di Washington, esperienza che gli sarà molto utile per il successo delle sue future invenzioni [1, 2].

Gli ultimi decenni dell'Ottocento e i primi del Novecento furono, negli Stati Uniti, un periodo di impressionante espansione tecnologica ed industriale – il numero di brevetti depositati salì da 6700 nel 1860 a quasi 82.000 nel 1920 [6] – e quindi un terreno ricco di opportunità per un giovane dotato di inventiva e determinato ad affermarsi. Hollerith cercò dapprima di approfittare delle opportunità offerte da questo clima di ottimistico sviluppo, brevettando un sistema elettro-pneumatico per la frenatura dei treni. Purtroppo il suo concorrente era il colosso industriale Westinghouse e Hollerith, nonostante la superiorità tecnica dei suoi freni, perse la partita [1].

Nel frattempo si avvicinava la data del nuovo censimento del 1890, e Hollerith tornò a concentrarsi sulle idee elaborate assieme a Billings, quelle cioè di creare un sistema meccanico per il conteggio dei dati. Il bisogno era pressante, la crescita della popolazione rendeva sempre più difficile il computo; nel 1840 gli americani erano 17 milioni, ma nel 1880 erano già triplicati. Anche il ventaglio di informazioni raccolte diventava sempre più ampio (educazione scolastica, immigrazione, lavoro, stato di salute ecc.) acuendo ulteriormente il problema. La crisi esplose col censimento del 1880, la cui elaborazione durò più di sette anni, arrivando quasi a sovrapporsi

alla rilevazione successiva² [1, 5]. Era tempo che la meccanizzazione, che tanto aveva contribuito alla rivoluzione industriale, si estendesse anche alle attività amministrative e contabili.

2. L'IDEA DI HOLLERITH

Per rispondere a questi bisogni Hollerith ideò un sistema elettromeccanico per facilitare sia il conteggio di ogni singolo dato raccolto dalla popolazione sia tutte le possibili combinazioni di questi. Il sistema si basava sulla trascrizione delle informazioni sotto forma di fori praticati su schede di cartoncino, una per ogni cittadino. Le schede potevano essere poi elaborate tramite il passaggio della corrente elettrica attraverso i fori e l'azionamento di un contatore elettromeccanico. Il maggiore vantaggio che ne derivava era che, una volta perforata la scheda, la stessa poteva essere oggetto di diverse elaborazioni, in modo da produrre le più svariate tabelle che incrociavano le variabili. Per esempio, si potevano contare quanti individui maschi bianchi erano in possesso di un titolo di studio, poi con le stesse schede stabilire quanti di questi avevano una certa età ed erano nati all'estero, e così via.

Non è mai semplice cercare di ricostruire le origini di un'idea innovativa; nel caso di Hollerith possiamo provare a cercare indizi su ciò che lui stesso e i suoi collaboratori rammentarono molti anni dopo. Nel 1925 il dott. Walter F. Wilcox del Census Bureau ricordava: *“Durante il conteggio del censimento del 1880 (...) Billings passò, assieme ad un visitatore [Hollerith], tra le centinaia di impiegati che trasferivano i dati dalle schede ai tabulati (...) egli disse al suo accompagnatore ‘Dovrebbe esserci un sistema meccanico per fare questo lavoro, qualcosa sul principio del telaio di Jacquard, in cui i fori su un cartoncino determinano il disegno del tessuto’”* [5].

L'invenzione del francese Joseph Marie Jacquard, un telaio in cui i movimenti dei fili della trama sono comandati da schede di cartone perforate, risale al 1801 ed era ormai una

² I risultati del censimento avevano notevoli ripercussioni sociali e politiche. L'Articolo 1 della Costituzione americana infatti recita: *I Rappresentanti e le imposte dirette saranno suddivisi fra i vari Stati (...) in base ai rispettivi numeri (...) di liberi Cittadini.*

tecnologia consolidata³. Hollerith poteva esserne a conoscenza poiché suo cognato lavorava nel settore tessile [1].

Nel 1919 Hollerith ricordava il suggerimento di Billings, ma in modo diverso: *“Una domenica sera, alla tavola del Dott. Billings, questi mi disse che avrebbe dovuto esserci una macchina per svolgere il lavoro puramente meccanico di tabulare la popolazione ed altre statistiche (...) pensava di utilizzare schede di cartoncino in cui la descrizione dell’individuo sarebbe stata registrata mediante incisioni lungo il bordo”* [1, 5]. In un’altra occasione però Hollerith forniva una versione differente: durante un viaggio in treno nell’Ovest, aveva osservato che *“Il controllore perforava nel biglietto una descrizione del viaggiatore, come capelli biondi, occhi scuri, naso grande ecc.. Così, vedete, ebbi l’idea di perforare in una scheda la ‘fotografia’ di ogni persona”* [1, 5]. Questa usanza può sembrarci strana, ma nell’Ottocento il biglietto era un vero e proprio contratto che veniva firmato dal passeggero ed era assolutamente personale, come si può evincere dai regolamenti ferroviari dell’epoca⁴.

All’epoca il telaio di Jacquard e il punzone del controllore non erano gli unici sistemi esistenti che registravano le informazioni tramite fori in un supporto; almeno altri due settori avevano già sfruttato questo sistema, quello della telegrafia⁵ e quello della musica meccanica.

Nel 1857 l’inglese Charles Wheatstone aveva brevettato un trasmettitore telegrafico che faceva uso di una striscia di carta perforata⁶. Ogni foro, passando tra due contatti elettrici, permetteva la chiusura del circuito, inviando automaticamente il codice lungo la linea [7]. L’invenzione di Wheatstone fu brevettata an-

che negli USA (US Patent 166.168, 1875). In quei decenni almeno una decina di altri brevetti simili furono depositati negli Stati Uniti, uno dei quali a nome di Thomas A. Edison (US patent 200.994, 1876). Quanto alle pianole meccaniche, nel 1842 il francese Claude Seytre inventò un sistema di registrazione della musica che faceva uso di un largo nastro di carta perforato [9] e più tardi i fratelli Schmoele brevettarono in America un piano meccanico comandato da un nastro perforato che veniva letto tramite circuiti elettrici (US Patent 145.532, 1873), come nel telegrafo di Wheatstone. Non bisogna infine dimenticare che nel 1837 Charles Babbage, nel progetto della sua Macchina Analitica – un antesignano meccanico del computer, mai costruito – proponeva di utilizzare schede perforate per registrare i dati e i programmi di elaborazione.

Avendo lavorato all’Ufficio Brevetti nel 1884, Hollerith era forse venuto a conoscenza delle invenzioni di Wheatstone, di Edison o di Schmoele, anche se sembra avesse già iniziato a sviluppare le sue macchine nell’anno precedente, quando lavorava come istruttore al MIT [1, 2]. È impossibile quindi stabilire quanto dell’idea originale sia dovuto a Hollerith e quanto ad influenze di Billings o di altri. Gli indubitabili meriti di Hollerith non devono però essere cercati in questo o quel particolare tecnico della meccanografia⁷, ma sono piuttosto da ascrivere al *sistema* coerente che credò, composto da diversi elementi complementari e perfettamente adatto allo scopo a cui era destinato, l’elaborazione statistica del censimento, ma che poi si dimostrò trasferibile con successo ad altri compiti contabili e gestionali.

Nel 1884 Hollerith depositò la sua domanda di

³ Prima di Jacquard tentarono lo stesso approccio i suoi connazionali Basile Bouchon (1725), Jean Falcon (1728) e Jacques Vaucanson (1740).

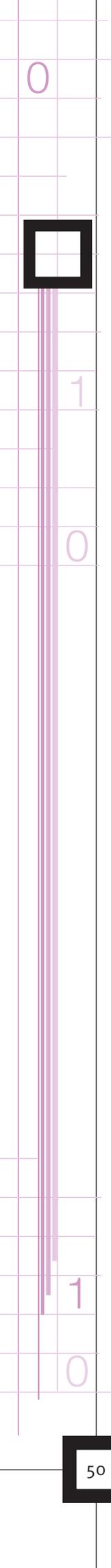
⁴ Per esempio, su F. L. Meyer, *Twentieth Century Manual of Railway Station Services*, del 1902 (p. 190), si legge che sul biglietto si deve *perforare* una descrizione delle sembianze del passeggero e che questa deve essere controllata accuratamente prima della perforazione per evitare errori o contestazioni.

⁵ La telegrafia aveva bisogno di un sistema per velocizzare la trasmissione di lunghi messaggi, soprattutto nelle connessioni transoceaniche in cui l’occupazione della linea costava parecchio, ma anche per trasmettere automaticamente messaggi ripetitivi.

⁶ Sembra che anche Alexander Bain avesse già brevettato un sistema simile nel 1846 [8].

⁷ La parola *meccanografia*, deriva probabilmente dal francese *“mécanographie”*, già usata nell’Ottocento con un significato diverso, più vicino alla dattilografia che all’elaborazione dei dati. Solo dopo il 1947, i dizionari la definirono *“impiego delle macchine calcolatrici, o contabili, e delle macchine selezionatrici per classificare i documenti”*.

0



brevetto, seguita da molte altre negli anni successivi. Tra il 1885 e il 1890 sperimentò i suoi primi prototipi in alcune rilevazioni di piccole dimensioni, come le statistiche sanitarie di Baltimora, del New Jersey e di New York, ricavando dall'esperienza molti spunti per i successivi perfezionamenti delle sue macchine. Arrivò il tempo di predisporre il censimento del 1890 e le macchine a schede perforate furono messe alla prova in confronto con altri due sistemi più semplici, puramente manuali [1, 2, 5]. Quello di Hollerith si dimostrò nettamente più veloce e affidabile dei concorrenti e l'inventore si aggiudicò la fornitura al Census Bureau di quasi un centinaio di tabulatori elettrici. La decisione si provò corretta, in soli sei mesi il governo poteva comunicare al pubblico che la popolazione degli Stati Uniti aveva raggiunto la cifra di 66.622.250 abitanti. Secondo Hollerith, le sue macchine avevano permesso un risparmio di quasi 5 milioni di dollari [1, 5].

3. L'IMPRESA DI HOLLERITH CRESCE

Nei primi anni l'impresa di Hollerith faticò a decollare. Due furono i problemi di crescita: Il primo fu la decisione dell'inventore di non vendere le sue macchine, ma di noleggiarle. La seconda fu l'eccessiva attenzione all'applicazione nei censimenti; queste rilevazioni avvenivano in media ogni dieci anni e, una volta finito il computo, le macchine venivano restituite, azzerando gli introiti del produttore. Hollerith cercò allora altri sbocchi tentando nuove applicazioni in altri settori, prima nelle ferrovie, poi nelle assicurazioni, nelle banche, nell'industria e nella grande distribuzione. Nonostante tutto il suo impegno creativo per adattare le schede e le tabulatrici ai nuovi compiti, l'accettazione da parte di questi nuovi mercati fu però lenta e irta di ostacoli. A peggiorare la situazione contribuì la personalità stessa di Hollerith: uomo di grande generosità e fedeltà alla famiglia e agli amici, era però un accentratore, incapace di delegare ad altri almeno una parte dell'enorme lavoro necessario a sviluppare,

produrre e vendere i suoi prodotti. La stessa idea di impiegare forza-vendite per tenere i contatti con i clienti era assolutamente aliena al suo pensiero⁸. Così si sobbarcava frequenti viaggi transatlantici non solo per tenere i contatti commerciali in America e in Europa, ma perfino per installare le macchine e fornire assistenza tecnica ai clienti. Fu solo grazie al sostegno della moglie e della suocera⁹, oltre alla sua infaticabilità, che Hollerith riuscì a superare i momenti peggiori, quando riusciva a malapena a nutrire la famiglia [1].

I due eventi che più lo colpirono furono la perdita del ricco contratto per il censimento USA del 1910 e la contemporanea nascita di potenziali concorrenti (riquadro 1 a p. 55). A queste difficoltà Hollerith seppe reagire, assumendo validi collaboratori e, soprattutto, unendosi ad alcuni soci per trasformare quella che era un'attività quasi artigianale in una società per azioni. Nel 1896 nacque così la *Tabulating Machines Company* (TMC), e gli affari cominciarono a decollare [1, 2, 3]. Il successo della TMC attirò presto l'attenzione degli investitori, tra i quali un certo Charles R. Flint, famoso per la sua capacità di realizzare fusioni tra aziende. Nel 1911 Flint propose agli amministratori della TMC e ad Hollerith, detentore del 52% delle azioni, la fusione con altre tre aziende che operavano nel mercato delle macchine per ufficio: *l'International Time Recording Company* e *la Bundy Manufacturing* che costruivano orologi marcatempo, e la *Computing Scale Corporation*, che produceva bilance automatiche e tritacarne (sic). L'agglomerato prese il nome di *Computing-Tabulating-Recording Company* (C-T-R) [1, 2]. L'operazione permise di riorganizzare l'azienda dotandola delle necessarie strutture amministrative, commerciali e produttive, oltre che di un centro di ricerca e sviluppo adeguato alle dimensioni economiche raggiunte. Ma il vero impulso innovativo che porterà la ancor piccola C-T-R a trasformarsi in una grande industria, che sarà leader di mercato con livelli di crescita mai visti, fu l'ingresso ai suoi vertici di un ventisettenne di grande carisma e notevoli capacità imprenditoriali,

⁸ La sua filosofia era quella di creare prodotti così perfetti che i clienti si sarebbero messi in coda pur di ottenerli.

⁹ È grazie ai diari della suocera, Theodosia Talcott, che sappiamo qualcosa della vita privata di Hollerith.

Thomas Watson Sr. Nel 1924 Watson ribattezzò la società International Business Machines, IBM [1, 2, 3].

Herman Hollerith, ormai cinquantunenne al tempo della fusione, e abbastanza ricco da permettersi un'agiata e tranquilla vecchiaia, non accettò di ritirarsi e rimase per alcuni anni alla C-T-R con funzioni consultive. L'accordo gli permise di conservare la funzione di guida tecnico-scientifica, con libero accesso ai laboratori di ricerca, ma non si trattò di un titolo puramente onorifico, infatti nessun progetto di sviluppo poteva proseguire senza il suo benessere. I suoi rapporti con Watson non erano facili – i due avevano idee spesso diametralmente opposte – ma restarono sempre cordiali e di reciproco rispetto. La salute dell'inventore era ormai minata dai trascorsi periodi di stress, di superlavoro, di viaggi continui e faticosi e di incertezze economiche. Per una vita più sana e rilassata, acquistò per se e per la famiglia una fattoria vicino a Georgetown, nel verde che circonda la capitale federale. Incapace di vero riposo, presto vi costruì una seconda abitazione e iniziò, con alterni risultati, allevamenti di animali e coltivazioni di ortaggi, di cui faceva spesso dono ai suoi amici. Otto anni dopo il suo definitivo ritiro dalla vita professionale Herman Hollerith morì di attacco cardiaco all'età di 69 anni [1]:

4. HOLLERITH IN ITALIA

Fin dagli inizi della sua attività imprenditoriale, Hollerith pensò di estendere il mercato dei suoi sistemi a schede perforate anche all'estero, fra l'altro presentandoli alle Esposizioni Universali di Parigi (1889) e di Berlino (1896). Grazie a quest'opera di propaganda, le sue macchine furono adottate per i censimenti dell'Impero Austro-Ungarico, del Canada e della Norvegia, ma il maggior successo fu l'impiego delle schede perforate nel primo censimento dell'Impero Russo (1897), che contava quasi 126 milioni di abitanti. [1, 2].

Precoce fu anche l'incontro di Herman Hollerith con l'Italia e di particolare importanza furono i contatti che l'americano ebbe con lo sta-

tistico italiano prof. Luigi Bodio. Bodio nacque a Milano nel 1840. Dopo gli studi a Milano e a Pisa, nel 1864 divenne professore di economia a Livorno, poi a Milano, infine occupò la cattedra di statistica e geografia economica a Venezia. Nel 1872, venne chiamato a dirigere l'Ufficio di Statistica del Ministero dell'Agricoltura, Industria e Commercio, ufficio che poi divenne la Direzione Generale di Statistica. Fu tra i promotori dell'Istituto Internazionale di Statistica, di cui fu segretario generale fino al 1905 e poi presidente dal 1909 fino al 1920, anno della sua morte. Le capacità scientifiche ed organizzative di Bodio godevano di riconoscimento internazionale e gli anni in cui fu a capo della Direzione furono definiti il "periodo d'oro" della statistica italiana. Non solo seppe riorganizzare la struttura e ridefinirne i compiti, ma riuscì a circondarsi di un gruppo di giovani valenti, stimolandone la formazione e la crescita professionale, portando la statistica del nostro Paese tra le più avanzate del mondo [10].

Nel 1891 Bodio si trovava a Vienna in occasione della III Sessione dell'Istituto Internazionale di Statistica; all'Assemblea Generale del convegno l'austriaco Henri Rauchberg presentò la macchina di Hollerith e descrisse l'uso che se ne stava facendo per il censimento della popolazione dell'Impero Austro-Ungarico¹⁰ [11]. Lo statistico italiano capì immediatamente le potenzialità dell'invenzione e plaudì ad Hollerith, suggerendogli altri possibili campi d'utilizzo. Così, infatti, preconizzava Bodio: "*Verrà il tempo in cui le ferrovie, le grandi industrie, le case commerciali e tutti i settori della vita industriale e commerciale useranno le macchine di Hollerith per motivi non solo di economia, ma di necessità*" [1].

In Italia si era alle porte il IV censimento della popolazione del 1891 e Bodio propose l'acquisizione di alcune macchine di Hollerith per l'elaborazione statistica. Purtroppo non poté ottenere i finanziamenti necessari; anzi, la mancanza di fondi costrinse addirittura il governo a cancellare la rilevazione e a rinviarla al 1901 [10, 13].

In occasione di un suo viaggio negli Stati Uniti per i festeggiamenti Colombiani di Chicago del

¹⁰ Gli austriaci, per motivi di protezionismo, evitavano di riferirsi ad Hollerith, chiamando la sua invenzione *machine électrique*. In effetti, le macchine usate nel 1890 erano state noleggiate dal concessionario austriaco Theodor Schäffler.

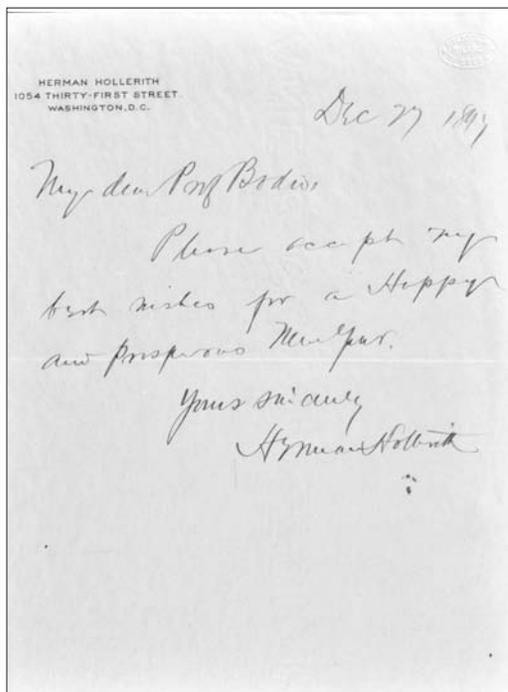


FIGURA 2

Lettera di Herman Hollerith a Luigi Bodio. (Gentile concessione della Biblioteca Braidense, Milano).
 “22 dicembre 1893. Mio caro prof. Bodio, la prego di accettare i miei migliori auguri per un felice e prospero anno nuovo. Sinceramente vostro, Herman Hollerith”
 (Traduzione dell'autore)

1893, Bodio fece visita a Hollerith. L'evento è ricordato nei diari della suocera dell'inventore americano, la signora Talcott: “Domenica Hollerith cenò con tre italiani. Uno parlava poco inglese, un altro poco tedesco, l'ultimo solo italiano e francese.” Un quotidiano locale ne diede notizia: “Le logore monarchie del vecchio mondo devono spesso ricorrere a questa giovane e progressista repubblica [gli Stati Uniti]. Anche nei dettagli dell'esecuzione dei compiti governativi siamo più avanti del resto del mondo. Il caso più recente è la visita del Professor Bodio e Boseo¹¹ Bonelli¹² in questa città [Washington] con lo scopo di scoprire come lo Zio Sam fa i suoi conti”. Come riconosce il biografo di Hollerith, Geoffrey Austrian, l'af-

fermazione roboante del giornalista non era giustificata: alcuni paesi europei erano decisamente più avanti degli Stati Uniti nel campo della statistica [1].

Fu proprio in occasione di quella visita che Bodio contrattò con l'inventore americano l'acquisizione di due tabulatrici e delle necessarie perforatrici, che furono sperimentate tra il 1894 e il 1895 per le statistiche giudiziarie [1]. Questa esperienza italiana fu spunto per la prima descrizione delle macchine di Hollerith pubblicata nel nostro paese [12]. L'incontro personale con Bodio fu molto gratificante per Hollerith che lo ricordò in due lettere inviate allo statista italiano. Le missive sono molto stringate, in consonanza con il carattere di Hollerith, sempre impegnato e frettoloso (Figure 2 e 3).

Hollerith ricambiò la visita nel novembre 1894, quando, durante uno dei suoi molti viaggi in Europa, trascorse un breve periodo a Roma per curare l'installazione e l'avvio delle due macchine. I contatti con gli italiani non furono facili; così, infatti, descrive i nostri connazionali: “Questo è un Paese paurosamente lento. Essi [gli italiani] vivono di quello che successe migliaia di anni fa. Li ho visti tagliare legna sulla strada da Napoli a Pompei, e, quando sono arrivato a Pompei, ho visto sulle pareti dipinte che mostravano esattamente lo stesso metodo di tagliare il legname”. Persino i cavalli in Italia gli sembravano lenti “come mucche.” Non dobbiamo adombrarci per questi giudizi negativi, anche la Francia era per Hollerith “un manicomio a porte aperte”. Ciò che lo disturbava era l'eccessiva formalità negli incontri con gli europei; le convenzioni sociali e il rispetto delle differenze di classe che, allora come oggi, sembravano incomprensibili perdite di tempo ad un americano. Durante la sua visita, Hollerith incontrò anche l'ing. Mengarini¹³, i Ministri della Guerra, Stanislao Mocenni, del Commercio, Paolo Boselli, e il sottosegretario di stato, Roberto Galli. Ormai saturo di ruderi antichi – “non ho più alcun interesse per le antichità romane e non mi interessa quando que-

¹¹ Augusto Bosco, un giovane della piccola aristocrazia piemontese che fece parte degli “ufficiali di statistica” reclutati da Bodio. Fu anche libero docente di statistica all'Università di Roma [10].

¹² Paolo Boselli redattore, con Cesare Correnti e Luigi Bodio, dello “Archivio di statistica”. Divenne Ministro dell'agricoltura nel terzo governo Crispi e Presidente del Consiglio dei Ministri nel 1916-17.

¹³ Guglielmo Mengarini. Fu responsabile della realizzazione di un impianto idroelettrico in corrente alternata tra Tivoli e Roma nel 1892, uno dei primi al mondo.

sto vecchio muro o quella antica porta furono fatti. Vorrei solo rivedere i tre radiosi visini [dei miei figli] e quello della loro madre” - Hollerith soffriva di nostalgia per la sua casa, la sua famiglia e il suo laboratorio, unici ambienti in cui si sentisse veramente a proprio agio [1].

Nello stesso anno Hollerith fu nominato membro della prestigiosa *Royal Society of Statistics* britannica. La nomina fu certo facilitata dall'intervento di Bodio che, nella sua lettera di raccomandazione, scrisse: “Sono convinto che il futuro della statistica risieda nell'uso delle macchine di Hollerith, perché queste permettono l'uso di materiale statistico in un modo molto più completo che i metodi ordinari (...). Le macchine permettono numerose combinazioni tra diversi elementi e non solo semplici addizioni di elementi separati” concludendo che “Il principio della macchina è così flessibile che può essere adattata alle applicazioni più svariate (...). La mia esperienza già conferma l'impressione che la macchina sia eccellente” [1]. Anche il vantaggioso contratto per il censimento russo deve qualcosa all'entusiasmo di Bodio. Alla fine della V Sessione dell'Istituto Internazionale di Statistica tenuta a Berna nel 1895, in cui fu nuovamente presentata l'esperienza austriaca, Bodio e Hollerith si fermarono nella città svizzera due giorni in più per discutere delle applicazioni e dei vantaggi delle tabulatrici a schede perforate assieme al delegato russo Nicholas Trointsky [1]. Nonostante l'interesse di Bodio e la sua precoce sperimentazione, proprio in Italia l'accoglienza del sistema di Hollerith fu piuttosto fredda, anche a causa delle difficoltà finanziarie e del progressivo deterioramento della situazione della Direzione di Statistica - il periodo d'oro della statistica italiana di Bodio stava per finire¹⁴ [10]. Come detto, il censimento del 1891, per il quale Bodio aveva proposto l'uso di macchine, non fu effettuato e il successivo del 1901 fu seguito con strumenti di tipo diverso, i *classi-compteurs* del francese Lucien March¹⁵, macchine che non facevano uso di schede perforate, limitando le possibilità di ottenere tante diverse tabulazioni incrociate [14].

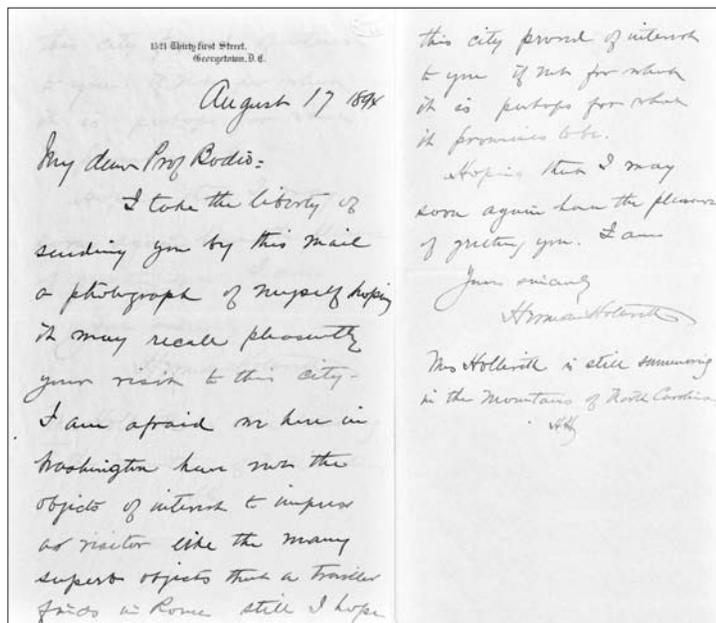


FIGURA 3

Lettera di Herman Hollerith a Luigi Bodio (Gentile concessione della Biblioteca Braidense, Milano). “17 agosto 1894. Mio caro Prof Bodio: mi prendo la libertà di inviarle, con questa lettera, una mia fotografia, sperando le possa ricordare piacevolmente la sua visita in questa città. Temo la mia casa in Washington non abbia gli oggetti interessanti che impressionano il visitatore come quelli superbi che un viaggiatore trova a Roma, ma spero ugualmente che questa città si sia dimostrata interessante per lei, non per ciò che è, ma per ciò che promette di diventare. Spero di avere ancora una volta il piacere di salutarla. Sinceramente suo, Herman Hollerith. – La sig.ra Hollerith è ancora in villeggiatura sui monti del North Carolina. H.H.” (Traduzione dell'autore)

La meccanografia a schede perforate fece la sua seconda apparizione in Italia nel 1914, quando fu aperta la rappresentanza del principale concorrente di Hollerith, la Powers Tabulating Machines (riquadro 2 a p. 56), col nome di SIMCA (*Società Italiana Macchine Classificatrici Addizionate*), costituita con capitali Powers e Pirelli. I primi due clienti della SIMCA furono la stessa Pirelli e l'Istituto Nazionale delle Assicurazioni (INA) [26]. Nel 1924 la SIMCA vendette 29 perforatrici e 2 selezionatrici alla Direzione di Statistica per l'elaborazione dei dati censuari del 1921 [15, 16]. Due anni più tardi la Direzione fu sciolta e sostituita da un nuovo ente, l'Istituto Nazionale di Statistica (ISTAT), che rispondeva direttamente al capo

¹⁴ Bodio si dimise dalla Direzione di Statistica nel maggio 1898 [10].

¹⁵ Una soluzione tecnica simile al tabulatore di March era stata già proposta nel 1880 da Luigi Perozzo, ingegnere impiegato presso la Direzione di Statistica. La macchina di Perozzo fu forse usata sperimentalmente per il censimento del 1881 [11, 25].

del governo e che venne dotato di maggiori risorse. I generosi fondi disponibili permisero di incrementare il parco macchine che, nel 1936, era ormai di 90 perforatrici e 25 selezionatrici della SIMCA-Powers¹⁶. Restavano ancora in uso 141 classi-compteurs di March [17].

L'impresa fondata da Hollerith, ormai nota come IBM, aprì la sua prima rappresentanza nel 1927, la SIMC (*Società Internazionale Macchine Commerciali*) con sede a Milano. Salvo sporadiche applicazioni precedenti, le prime installazioni di sistemi Hollerith furono implementate alle Ferrovie dello Stato nel 1928 e alla assicuratrice RAS tre anni più tardi. Nel 1934 la SIMC cambierà nome in Hollerith Italiana S.A., poi nel 1939 in Watson Italiana S. A. Nazionale Macchine Aziendali. Gli aggettivi "nazionale" e "italiana" servivano probabilmente a far accettare all'autarchico governo Mussolini le forniture di un operatore straniero.

Fino al 1940 però le imprese private e gli enti pubblici italiani che facevano uso della meccanografia a schede erano ancora pochi, non più di una sessantina; la vera esplosione della meccanografia avverrà dopo la II Guerra Mondiale, grazie al grande sviluppo industriale e commerciale che conosciamo col nome di "boom economico". Nel 1949 entrerà nel nostro Paese anche il terzo produttore di meccanografia, la francese Bull, grazie ad un accordo con Olivetti [18]¹⁷.

Nonostante la tardiva adozione, un uso veramente originale della meccanografia fu fatto proprio nel nostro Paese grazie al gesuita Padre Roberto Busa. Lavorando alla sua tesi di laurea su Tommaso d'Aquino, il linguista si convinse che sarebbe stato possibile studiare il lessico degli scritti di Tommaso facendo uso di macchine a schede perforate. Ottenuto l'aiuto dal presidente di IBM, Thomas Watson Sr., nel 1949 Padre Busa iniziò il suo progetto trasferendo i testi dell'Aquinate su schede perforate, che poi sottopose ad analisi automatica delle ricorrenze [19]. Il progetto durerà ben trent'anni risultando nel prezioso *Index Thomisticus*, oggi consultabile sul web.

¹⁶ Alcune tabulatrici IBM erano presenti anche all'ISTAT nel 1931 [31].

¹⁷ Qualche installazione di macchine Bull era già presente in Italia prima della guerra, una proprio alla Olivetti.

Bibliografia

- [1] Austrian G.D.: *Herman Hollerith. Forgotten Giant of Information Processing*, New York, Columbia University Press, 1982.
- [2] Pugh E.W.: *Building IBM. Shaping an Industry and Its Technology*, Cambridge (MA), MIT Press, 1995.
- [3] Cortada J.W.: *Before the Computer. IBM, NCR, Burroughs, & Remington Rand & the Industry They Created. 1865-1956*, Princeton, Princeton University Press, 1993.
- [4] Miller Lydenberg H., John Shaw Billings: *Creator Of The National Medical Library And Its Catalogue, First Director Of The New York Public Library*. American Library Association, Chicago, 1924 (ripubblicato: Kessinger Publishing, 2007).
- [5] Truesdell L.E.: *The Development of Punch Card Tabulation in the Bureau of the Census. 1890-1940*, Washington, U.S. Department of Commerce, Bureau of Census, 1965.
- [6] United States Patents and Trademark Office, U.S. Patent Activity, Calendar Years 1790 to the Present: *Table of Annual U.S. Patent Activity Since 1790*. <http://www.uspto.gov/web/offices/ac/ido/oeip/taf/tafp.html>.
- [7] Bowers B.: *Sir Charles Wheatstone FRS: 1802-1875*. IEE, London 2001, p. 181-182.
- [8] Hurdeman A.A.: *The worldwide history of telecommunications*. John Wiley, 2003, p. 72.
- [9] Dolge A.: *Pianos and their makers: a comprehensive history of the development of the piano*. Dover Publ., 1972, p. 131-162.
- [10] Soresina M.: *Conoscere per amministrare. Luigi Bodio: statistica, economia e pubblica amministrazione*. Milano, Franco Angeli, 2001.
- [11] Rauchberg M.H.: Description de la machine électrique servant au dépouillement du recensement autrichien de 1890. *Bulletin de l'Institut international de statistique*, Vol. VI, n. 2, 1892, p. 19-26.
- [12] G.R.: Sul sistema meccanico Hollerith per lo spoglio delle notizie contenute nelle schede di un censimento di una popolazione o di altri documenti statistici. *Giornale degli Economisti*, maggio 1894, p. 504-511.
- [13] Favero G.: *Statistica ufficiale e politica nel secondo Ottocento*. SIS Magazine, 25 febbraio 2010.
- [14] Ligonnière R.: *Préhistoire et histoire des ordinateurs*. Paris, Laffont, 1987.
- [15] Calamani G.: Relazione sui lavori della Direzione Generale di Statistica, Atti del Consiglio Superiore di Statistica, Sessione 9-13 giugno 1924. *Annali di Statistica*, Serie VI, Vol. 6, 1931, All. 8°, pg. 140-141, All. 13, 161-165.
- [16] Marucco D.: *L'amministrazione della statistica nell'Italia unita*. Bari, Laterza, 1996, p. 106-108.

- [17] ISTAT: *Decennale, I Servizi Centrali*. Roma, Poligrafico dello Stato, 1936, p. 46-60
- [18] Sacerdoti G., Ranci F.: *Aspetti industriali dell'informatica in Italia*. In: Cuzzer A. (a cura di), *La cultura informatica in Italia: Riflessioni e testimonianze sulle origini 1950-1970*. Bollati Boringhieri, 1993, p. 125.
- [19] Busa R.: *Cinquant'anni a... "bitizzar" parole*. In: Convegno Internazionale sulla storia e preistoria del calcolo automatico e dell'informatica, Siena, 10-12 settembre 1991, Milano, AICA, 1991, p. 71-82
- [20] Heide L.: *Punched Card Systems and the Early Information Explosion*. John Hopkins University Press, 2009.
- [21] Kistermann F.W.: The Invention and Development of the Hollerith Punched Card. *Annals Hist. Comput.*, Vol. 13, n. 3, 1991, p. 245-259.
- [22] Billings. J.S.: *On Some Forms of Tables of Vital Statistics with Special Reference to the Needs of the Health Department of the City*. Public Health Papers and Reports, 1887, 13, p. 203-221.
- [23] Martin T.C.: Counting a Nation by Electricity. *The Electrical Engineer*, Vol. 2, 11 novembre 1891, p. 521-530.
- [24] Kahn D.: *The Code-breakers. A Comprehensive History of Secret Communication from Ancient Times to the Internet*. New York, Scribner, 1996, p. 562-595.
- [25] Pietra G.: *La prima classificatrice meccanica è stata ideata da un italiano*. Barometro Economico Italiano, 10 agosto 1934.
- [26] Zane M.: Il percorso italiano verso l'informatizzazione: dal centro meccanografico al primo elaboratore elettronico (1914-1954). *Altronovecento*, n. 3, luglio 2000.
- [27] Bonfanti C.: *Appunti per un corso di storia dell'informatica*. Lezioni 5-6, Università di Udine, 2009-2010, <http://nid.dimi.uniud.it/history/history.html>.

Riquadro 1 - I concorrenti di Hollerith

Il primo concorrente di Hollerith fu proprio il suo primo cliente, il *Census Bureau* degli Stati Uniti. In previsione del censimento del 1910, l'Ufficio divenne un'istituzione stabile al cui vertice fu posto il direttore Simon North. Questi reputò eccessivi i costi sostenuti per il noleggio delle macchine di Hollerith nei censimenti del 1890 e 1900 e progettò di slegarsi dal monopolio istituendo un laboratorio interno che avrebbe dovuto sviluppare le proprie macchine. A colmare la misura, North riuscì a sottrarre a Hollerith alcuni brillanti collaboratori. La situazione, oltre ad irritare non poco l'inventore, divenne un problema politico: in un Paese liberale come gli USA era impensabile che il governo facesse concorrenza agli imprenditori privati. North fu costretto a dimettersi, il suo laboratorio non riuscì a produrre macchine affidabili e il censimento del 1910 fu effettuato in parte con le macchine Hollerith del 1900 [1, 2, 3].

Il tentativo di North ebbe però un'importante conseguenza. Tra i tecnici assunti dal *Census Bureau* vi era un immigrante russo di nome James Powers. Powers era un abile ingegnere e cercò di superare alcuni dei maggiori difetti delle macchine di Hollerith inventando, tra l'altro, una tabulatrice scrivente. Le macchine di Powers non facevano uso dell'elettricità per la lettura dei fori, ma erano puramente meccaniche¹. Nel 1911 Powers lasciò il *Census Bureau* e aprì una sua impresa, la Powers Accounting Machines Co., in diretta concorrenza con la TMC di Hollerith. Presto espanse la sua attività anche in Inghilterra, in Francia (col nome di Powers-SAMAS) e in Germania. Nel 1920 la Powers Accounting Machines fu assorbita dalla ben più grande Remington-Rand, produttrice di calcolatrici e macchine da scrivere. Pur disponendo di prodotti più avanzati dei contemporanei della TMC ed ottenendo successi in alcuni settori e aree geografiche, la Remington-Rand rimase sempre in seconda posizione nel mercato della meccanografia a schede [1, 2, 3]².

Vi furono anche altri più limitati tentativi, come quello dell'americano John K. Gore che nel 1894 brevettò e costruì perforatrici e tabulatrici per la sua azienda, la Prudential Insurance Co., dove vennero usati fino agli anni '30. Sempre negli Stati Uniti, nel 1911, J. Roydon Peirce brevettò alcune interessanti macchine a schede perforate note come *Royden System of Perforated Cards*, usate dalla Metropolitan Life Insurance, ma poi vendette i brevetti alla C-T-R [1, 3]. Nel 1889 anche Alexander Graham Bell progettò una selezionatrice a schede perforate [20].

Anche in Europa ci fu chi pensò di approfittare del potenziale mercato, e questo avvenne nella piccola e periferica Norvegia, dove l'ingegnere Fredrik Rosing Bull progettò macchine contabili a schede che brevettò nel 1919 e che introdusse nella società per cui lavorava, l'assicuratrice Storebrand. Nel 1923 finì il suo primo prototipo di tabulatrice, migliore e più economica di quelle dei due concorrenti IBM e Remington-Rand. La prematura morte di Bull, nel 1925, lasciò l'impresa in condizioni difficili e nel 1928 la produzione passò alla società svizzera Egli, industria di calcolatrici. Nuovi investitori furono poi trovati in Francia dove, nel 1931, fu fondata la

¹ Più tardi le macchine di Powers adottarono motori elettrici per la movimentazione delle schede, ma il meccanismo di lettura delle perforazioni rimase meccanico.

² Nel 1930 la IBM aveva lo 88% del mercato della meccanografia, contro il 12% della Remington Rand.

segue

Egli-Bull che divenne la Compagnie des Machines Bull. La nuova azienda, che per effetto di acquisizioni e fusioni cambiò spesso nome³ mantenne un buon posizionamento nel mercato europeo raggiungendo a volte il secondo posto [3, 20]⁴.

Oltre a Bull, vi furono in Europa altri tentativi di imitare i successi di IBM e Remington-Rand, soprattutto in Germania per effetto della politica protezionistica del IV Reich. Nel 1934 la tedesca Siemens & Halske depositò alcuni brevetti di macchine a schede perforate, ma poi si accordò con la Remington per la loro commercializzazione, dedicandosi solo alla produzione. Negli stessi anni l'ingegnere Gustav Tauschek della Rheinmetall sviluppò alcune tabulatrici a schede da interfacciare alle calcolatrici della sua azienda. L'attività fu poi venduta alla IBM e Tauschek passò alla compagnia americana, per la quale ottenne ben 169 brevetti [20].

Nel secondo dopoguerra IBM, Remington-Rand e Bull - con le loro consociate e licenziatarie nel mondo - erano rimasti gli unici produttori, in un mercato che nel 1951, al suo apice, raggiunse un fatturato di 440 milioni di dollari. Può sembrare curioso che, diversamente dal mercato delle calcolatrici, delle macchine da scrivere e, più tardi, dei computer, quello della meccanografia sia rimasto sempre nelle mani di un piccolissimo gruppo di operatori. La spiegazione è però semplice: solo le grandi aziende e le amministrazioni statali che trattavano grandi quantità di dati potevano permettersi i rilevanti costi, quindi il numero di potenziali clienti rimaneva limitato [1, 2, 3, 20].

³ Bull General Electric, Honeywell Bull, CII Honeywell Bull, e Bull HN.

⁴ Nel 1949 la Bull concluse un accordo con l'italiana Olivetti, per commercializzare macchine a schede perforate col nome Olivetti-Bull.

Riquadro 2 - Gli elementi del sistema e la loro evoluzione

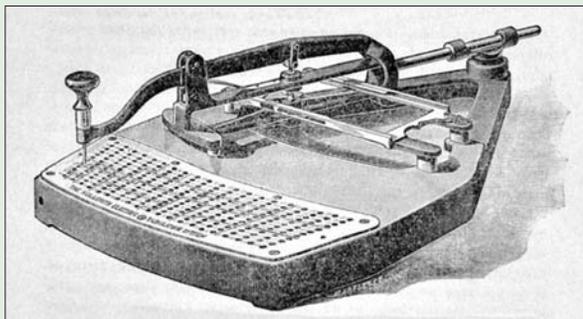
Inizialmente, l'invenzione di Hollerith si basava su pochi componenti. L'elemento centrale del sistema era la scheda perforata, un cartoncino di dimensioni ridotte, suddiviso in un certo numero di zone che potevano essere forate mediante una *perforatrice* a mano (Figura A). La macchina che eseguiva il conteggio delle schede, la *tabulatrice* (Figura B), era la parte più complessa del sistema; una testa di lettura con numerosi aghi veniva abbassata sulla scheda; in presenza di un foro l'ago corrispondente chiudeva il circuito elettrico che azionava un contatore. Alla tabulatrice poteva essere collegata una *selezionatrice* o *ordinatrice* che, in base alle perforazioni, apriva un cassetto tra molti in cui riporre la scheda appena letta. In questo modo le schede lette venivano separate in gruppi, che poi potevano subire ulteriori elaborazioni. La tabulatrice poteva anche eseguire, oltre al semplice conteggio, anche la congiunzione logica (a AND b AND c); grazie ad appositi circuiti a *relè* era possibile contare o selezionare in un solo passaggio le schede che appartenevano, per esempio, a cittadine di sesso femminile, nubili e di origine cinese.

Col crescere delle applicazioni in settori diversi dai censimenti, fu necessario non solo rendere sempre più veloci ed efficienti questi primi componenti, ma anche creare attorno alla scheda perforata una popolazione di nuove macchine, ognuna atta ad eseguire una determinata operazione. Il successo delle macchine a schede nelle applicazioni contabili richiedeva almeno due importanti migliorie:

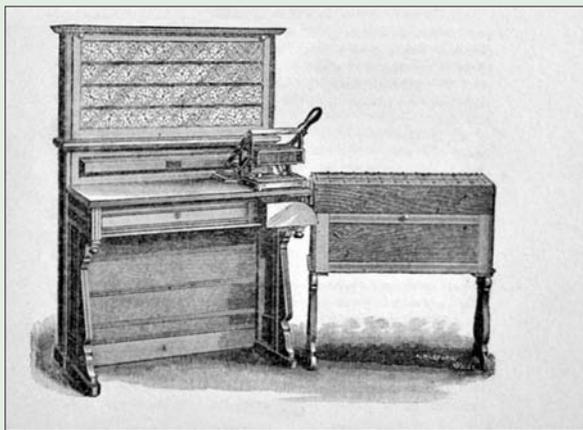
1. tabulatrici che stampassero i risultati intermedi e finali, evitando di dover annotare manualmente le letture dei singoli contatori;
2. la sostituzione dei contatori con addizionatrici capaci di operare aritmeticamente su quantità numeriche rappresentate sulla scheda con opportuni codici di perforazione.

Questo perfezionamento fu introdotto da Hollerith già nel 1896, mentre la stampa dovrà attendere il 1914 ed è dovuta a Powers.

Nei decenni successivi furono inventate le macchine per il controllo dei dati (*verificatrici*), altre che riunivano due gruppi di schede in uno solo e in un certo ordine (*inseritrici*), le *traduttrici* che traducevano il codice perforato della scheda e lo stampavano in chiaro sulla scheda stessa. Vi erano macchine che creavano copie delle schede perforate, le *riproduttrici*. Le nuove perforatrici e selezionatrici alimentavano automaticamente le schede, raggiungendo velocità di elaborazione impensabili con i primi modelli della fine

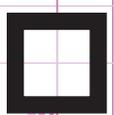


A - Perforatrice a pantografo di Hollerith (1890) [12]

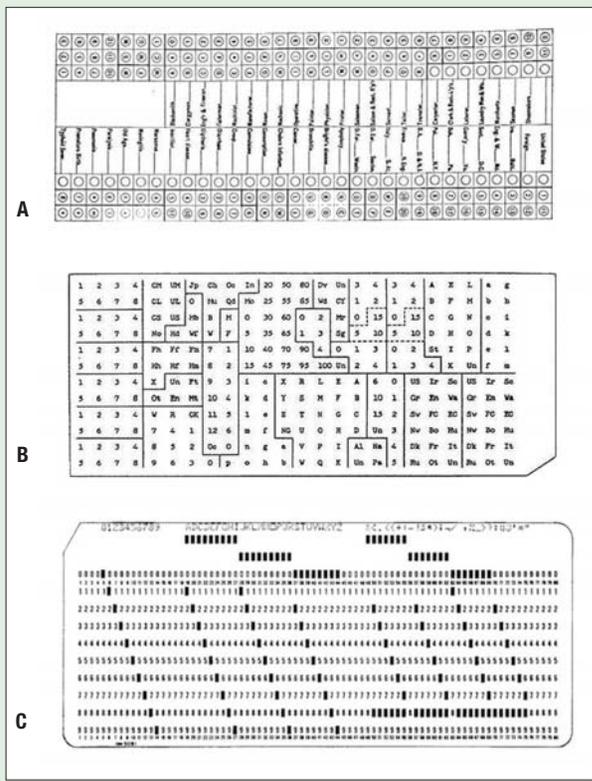


B - Tabulatrice e selezionatrice di Hollerith (1890) [12]

segue



dell'Ottocento, che erano alimentati manualmente una scheda alla volta. Anche le elaborazioni eseguibili sui dati numerici arrivarono a comprendere non solo le quattro operazioni aritmetiche, ma perfino brevi sequenze programmabili di calcoli. Nel contempo anche le schede perforate subirono una lunga evoluzione; dalle 12 righe per 24 colonne del 1890 si arrivò prima a 45 colonne, poi la IBM standardizzò le schede a 80 colonne (Figura C) con fori rettangolari, mentre la Powers giunse a 90 colonne, su due sezioni di 45, mantenendo i fori tondi [1, 2, 20, 21].



Capaci di trattare velocemente enormi quantità di dati e ormai dotate di una seppur limitata capacità di programmazione, le macchine di Hollerith attirarono l'attenzione del mondo scientifico, particolarmente quello dell'astronomia e quello della fisica, impegnati nella soluzione di problemi che richiedevano grandi quantità di calcoli numerici. Nel 1928 Leslie Comrie, che lavorava al Nautical Almanac Office di Greenwich (UK), usò macchine a schede perforate per il calcolo del moto lunare, producendo tavole astronomiche per il periodo 1935-2000. Negli USA, all'Università di Columbia, l'astronomo Wallace Eckert pubblicò il libro *Punched Card Methods in Scientific Computation*, in cui trattava della soluzione di calcoli astronomici usando la meccanografia di IBM. Lo stesso presidente Watson fu talmente interessato alle applicazioni scientifiche, che finanzia l'istituzione dello Astronomical Computing Bureau. Un'altra esperienza fu quella fatta a Los Alamos nell'ambito del Progetto Manhattan con i calcoli necessari per le prime bombe nucleari [2, 3]. La meccanografia entrò in guerra anche su un altro fronte: la guerra dei codici segreti. Già dal 1932 la Marina degli Stati Uniti sperimentò l'uso di perforatrici e tabulatrici IBM per la decodifica dei messaggi giapponesi; l'esperimento fu affidato al capitano Thomas Dyer, criptoanalista della marina, e continuò per tutta la durata della seconda guerra [24].

Le fotografie dell'epoca che ritraggono i centri meccanografici mostrano grandi locali dove numerose perforatrici sono allineate su lunghi tavoli; tabulatrici, selezionatrici e altre macchine, in numero minore, sono vicino alle pareti; vi si vede sempre un discreto numero di operatori, nella maggioranza donne. Ogni elemento del centro meccanografico era un'isola indipendente e la comunicazione tra di essi era garantita da esseri umani che trasferivano pacchi di schede da una macchina all'altra. La "memoria di massa" del centro era costituita da grandi scaffalature a cassette zeppate di schede perforate, il cui numero si contava spesso in milioni. Se paragonato ad un moderno computer, il centro meccanografico era caratterizzato quindi da un certo grado di parallelismo; mentre le perforatrici registravano i dati in ingresso, le selezionatrici raggruppavano altre schede, su altre ancora le tabulatrici eseguivano

C - Schede perforate di Hollerith. **A** - Statistiche sanitarie di Baltimora 1886 (82,6 x 219,1 mm); veniva perforata solo lungo i bordi con una comune pinza oblitteratrice per biglietti ferroviari [22]. **B** - Censimento USA 1890 (82,6 x 168,3 mm); non era sovrastampata come nella figura; si usava la perforatrice a pantografo di figura **A** [23]. **C** - La definitiva scheda IBM a 80 colonne alfanumeriche del 1920 (82,6 x 187,3 mm)

calcoli e ne stampavano i risultati. Il parallelismo compensava in parte il difetto intrinseco del centro, la sua lentezza di elaborazione, dovuta sia alla tecnologia elettromeccanica che ne era alla base, sia alla necessità di trasferire e archiviare le schede manualmente [27]. Negli anni '50, comparvero sul mercato i primi elaboratori elettronici di uso generale. I neonati computer permettevano velocità di elaborazione impensabili con la meccanografia - oltre che una molto maggiore flessibilità d'uso (programmabilità) - per adeguarsi alla quale l'archiviazione dei dati fu presto affidata ai più rapidi nastri magnetici. Le schede perforate sopravvissero però per più di vent'anni; lettrici e perforatrici di schede erano, infatti, collegabili ai calcolatori elettronici come unità di ingresso e di uscita. Ciò permetteva agli utenti non solo di continuare ad utilizzare gli archivi preesistenti, ma anche di mantenere metodi e organizzazione consolidati. La vendita delle schede costituiva fino al 30-40% degli introiti di IBM e Remington-Rand [3] e sarebbe stato economicamente avventato disincentivarne l'uso. Le ultime macchine a schede perforate furono prodotte dalla IBM nei primi anni '80, quasi un secolo dopo i primi esperimenti di Hollerith. A questo punto le schede perforate entrarono definitivamente nel cimitero dei supporti obsoleti. La parola *meccanografia* e i suoi derivati sopravvissero più a lungo del mondo che stavano ad indicare, soprattutto negli uffici pubblici dove *centro meccanografico*, *operatore meccanografico* e *codice meccanografico* si trovano ancora oggi, quando ormai di *meccanico* non vi è quasi più nulla.

SILVIO HÉNIN, Diplomato in Elettronica e laureato in Scienze Biologiche, ha lavorato per un decennio all'Università di Milano per poi passare alla farmaceutica Roche dove ha implementato sistemi computerizzati per l'information management e retrieval. Fondatore di un'associazione di information scientists (GIDIF-RBM), è stato membro del direttivo dell'Ass. Italiana Documentazione Avanzata (AIDA) e del Comitato per la Documentazione di Federchimica. Cultore della storia della tecnologia, particolarmente di quella del calcolo automatico, è consulente del Museo Nazionale della Scienza e della Tecnologia Leonardo da Vinci di Milano e collabora con riviste del settore. E-mail: silvio.henin@fastwebnet.it

QUANDO L'UTENTE GUIDA L'INNOVAZIONE IL WEB MASHUP

Una pratica sempre più emergente nello scenario del Web 2.0 è lo sviluppo di mashup, applicazioni Web basate sulla composizione di contenuti e servizi aperti. In questo articolo illustreremo come, tramite strumenti e processi di sviluppo leggeri, anche gli utenti meno esperti possano assumere il ruolo di sviluppatori di mashup. Evidenzieremo la capacità dei mashup di far leva sul potenziale di innovazione degli utenti del Web e la necessità in questo contesto di metodi di sviluppo rinnovati. Traceremo quindi le maggiori problematiche che dovranno essere affrontate affinché i nuovi metodi trovino applicabilità.

1. INTRODUZIONE

La tendenza corrente nello sviluppo delle moderne applicazioni Web, e in particolare delle applicazioni del Web 2.0, punta chiaramente verso il coinvolgimento sempre più marcato dell'utente. Le cosiddette applicazioni sociali sono la prova del valore, inizialmente inaspettato, dell'integrazione degli utenti finali nel processo di creazione dei contenuti. Un'altra pratica emersa recentemente è lo sviluppo di *Web mashup*, applicazioni Web ottenute dalla combinazione di contenuti e servizi disponibili sul Web sotto forma di API (*Application Programming Interface*, cioè interfacce software di programmazione) "aperte" o, più in generale, di servizi riusabili. L'esempio più affermato di mashup è www.housingmaps.com, nato dall'integrazione tra gli annunci di affitto pubblicati su www.craigslist.com e le mappe di Google. Il risultato è un'applicazione Web innovativa, che semplifica la localizzazione degli appartamenti offerti in affitto.

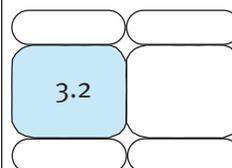
Il primo e fondamentale passo nello sviluppo dei mashup è la produzione di *servizi aperti*, pubblicati sul Web e pertanto facilmente ac-

cessibili e riutilizzabili. L'uso di servizi aperti è una caratteristica unica, che distingue lo sviluppo dei mashup da altri paradigmi di sviluppo basati su componenti (riquadro a p. 30 per un approfondimento sui mashup e sulle tecnologie abilitanti). Tali servizi sono eterogenei, e possono essere API remote, servizi basati sullo scambio di messaggi (per esempio, servizi Web), API basate sull'integrazione programmatica di codice (come accade per le API di Google Maps e di Twitter), o anche feed RSS/Atom (per esempio, notizie sui titoli di borsa), o contenuti estratti dai siti Web più disparati (per esempio, i prezzi di certi prodotti). Tali servizi possono inoltre essere dotati di un'interfaccia utente propria (si parla in tal caso di servizi UI), che poi diventa parte integrante dell'interfaccia del mashup finale, possono offrire supporto alla computazione di certe funzionalità, o possono agire come semplici sorgenti di dati.

I mashup più popolari integrano API pubbliche che permettono a utenti e sviluppatori generici di invocare servizi accessibili tramite Web. Tuttavia, una prospettiva promettente è



Florian Daniel
Maristella Matera



Web mashup

I mashup sono applicazioni web – nella maggior parte dei casi molto semplici, per esempio composti da una sola pagina Web – sviluppate a partire da componenti software accessibili tramite il Web, come per esempio contenuti e dati, servizi che forniscono logica applicativa oppure elementi di presentazione (pezzi di interfaccia utente, solitamente in formato HTML) [4]. Casi tipici possono essere trovati facilmente, per esempio, sul sito programmableweb.com diventato nel corso degli ultimi anni uno dei riferimenti principali per sviluppatori di mashup, specialmente per quanto riguarda la ricerca di componenti base. I mashup sono quindi una pratica di composizione e di riuso di componenti pubblicati sul Web, che aggiunge valore agli stessi in quanto trova forme di utilizzo nuove e creative che spesso vanno ben oltre lo scopo originario dei singoli componenti. Basti pensare che i primi mashup si basavano soprattutto sul riuso di contenuti estratti da siti Web convenzionali senza che il proprietario dei contenuti ne fosse a conoscenza.

La composizione di applicazioni o di processi a partire da servizi esistenti non è una disciplina nuova. Sono ormai anni che la comunità scientifica studia metodi e linguaggi di composizione per i cosiddetti *Web service* con risultati molto buoni (pensiamo per esempio a BPEL o ai vari approcci di *Enterprise Application Integration*), e il campo dell'integrazione di dati è ormai un'area consolidata, nonostante le nuove sfide poste da dati sempre meno strutturati. L'aspetto innovativo introdotto dai mashup nel mondo della composizione è che, diversamente dagli approcci d'integrazione a livello di dati o di logica applicativa, integrano anche elementi di presentazione, un aspetto facilitato dall'uso di tecnologie web standardizzate.

I componenti usati nello sviluppo di mashup sono quindi di tre tipi:

□ Servizi di dati come i feed RSS (*Really Simple Syndication*) o Atom, contenuti formattati in JSON (*JavaScript Object Notation*) o XML oppure semplici file di testo. Per esempio, quasi tutti i giornali pubblicano ormai i titoli delle loro notizie tramite feed RSS che possono essere letti da un cosiddetto RSS reader e permettono all'utente di saltare facilmente dettaglio delle varie notizie.

□ Servizi Web o API (*Interfacce Applicative Programmabili*) accessibili tramite il Web come i servizi SOAP (*Simple Object Access Protocol*) o REST (*REpresentational State Transfer*). Questi servizi tipicamente non forniscono semplici dati, ma permettono il riuso di logica applicativa come, per esempio, il calcolo del nome di una città a partire dalle sue coordinate GPS.

□ Componenti UI (cioè dotati di interfaccia utente) come pezzi di codice HTML o interfacce programmabili in JavaScript (per esempio, i cosiddetti widget – <http://www.w3.org/TR/widgets-apis/>). Il tipico esempio di componente UI è Google Maps, che fornisce non solo dati in forma di carte geografiche ma anche un'interfaccia utente facilmente integrabile in una pagina Web che permette all'utente di navigare le mappe. Comunque, anche l'estrazione di contenuti da pagine Web tradizionali è ancora una prassi molto diffusa, specialmente in assenza di servizi equivalenti già disponibili e pronti per l'uso.

Data questa eterogeneità di componenti – e quindi di tecnologie e di logiche applicative diverse da componente a componente – lo sviluppo di un mashup è tutt'altro che facile e spesso ancora una prerogativa di sviluppatori esperti. Sono loro che conoscono i linguaggi di programmazione, i protocolli di comunicazione, e gli standard per implementare la logica di integrazione necessaria per mettere in comunicazione i vari componenti e l'impaginazione grafica dei componenti all'interno del layout del mashup.

La situazione è ancora più complessa qualora i componenti necessari non fossero già disponibili online. In questo caso è lo sviluppatore che deve prima implementare il componente per poi poterlo comporre. Nella maggior parte delle volte questo vuol dire estrarre dati da una o più pagine Web, una pratica che tipicamente richiede l'uso di espressioni regolari complesse e sforzi significativi di programmazione. Per alleviare tali sforzi, di recente sono emersi strumenti online, che sono in grado di assistere lo sviluppatore nell'estrazione di dati da pagine Web tramite interfacce grafiche e interattive e che permettono di selezionare i contenuti desiderati e di ottenere automaticamente un componente riusabile. Esempi di questi strumenti sono Dapper o Openkapow, tutti e due accessibili online e gratuiti.

Comunque, come anche proposto in questo articolo, la visione della comunità scientifica e dell'industria del software è quella di mettere a disposizione dell'utente Web medio degli strumenti di sviluppo e ambienti di esecuzione che siano semplici ed intuitivi (per esempio basati su linguaggi di composizione grafici e paradigmi *drag-and-drop*) e che permettano all'utente stesso di sviluppare applicazioni personali a partire da componenti esistenti in maniera assistita e senza avere bisogno dell'aiuto da parte di programmatori esperti. Strumenti come Yahoo! Pipes, Intel Mash Maker, IBM Lotus Mashups o JackBe Presto rappresentano un primo passo in quella direzione, ma ancora non possiamo parlare di utenti veramente autosufficienti. Spesso tali strumenti forniscono semplificazioni tecnologiche rilevanti, ma quello che manca all'utente Web medio sono le nozioni base dello sviluppo di software. In questo contesto, il design di meccanismi che assistano l'utente tramite raccomandazioni di sviluppo interattive è una delle sfide di ricerca del futuro.

data dai *mashup aziendali* [7], un'estensione dei mashup verso le intranet, per permettere ai membri di un'azienda di utilizzare servizi interni per l'accesso alle risorse informative dell'azienda stessa e “mescolarli” in modi innovativi, per esempio per automatizzare una procedura burocratica ricorrente.

Entrambi i tipi di servizi, sia le API pubbliche, sia i servizi interni all'azienda, possono incoraggiare gli sviluppatori e gli utenti a scoprire e

promuovere usi innovativi. Se coadiuvati da strumenti di sviluppo e metodologie adeguate, anche gli utenti Web meno esperti sul piano delle tecnologie potrebbero far uso di tali servizi per creare applicazioni nuove, trasformandosi così da ricettori passivi d'innovazione ad attori direttamente coinvolti nella creazione di innovazione. L'aggregazione di questo processo su tutti gli utenti che in parallelo conducono esperimenti con gli stessi servizi accelera

enormemente l'innovazione, e copre uno spazio di progettazione più ampio rispetto a quello che i fornitori originari riuscirebbero a concepire se non aprissero i loro servizi vero parti esterne. Lo sforzo che molti dei maggiori produttori dell'economia di internet odierna (per esempio, IBM, Intel, Yahoo!, SAP) stanno investendo nella ricerca sui mashup, testimonia che qualcosa di nuovo sta accadendo, che va ben oltre le soluzioni basate sulla programmazione spinta a cui attualmente ricorrono i programmatori di mashup esperti.

In questo articolo, esploreremo il mondo dei mashup e il loro potenziale come strumenti tramite cui gli utenti del Web, non necessariamente esperti di tecnologie, possono creare innovazione. Spiegheremo perché gli utenti del Web possono trovare interesse nello sviluppare in prima persona le loro applicazioni, e quali altre figure possono trarne benefici. Discuteremo quindi il processo di sviluppo dei mashup e gli strumenti e i metodi che possono supportare anche gli utenti più inesperti nell'innovare attraverso i mashup. Questa discussione ci permetterà di dare uno sguardo critico riguarda come i mashup sono sviluppati oggi e di individuare quindi cosa manca per realizzare i mashup del futuro.

2. INNOVAZIONE CREATA DAGLI UTENTI E STRUMENTI PER L'INNOVAZIONE

Cosa spinge gli utenti a sviluppare le proprie applicazioni? Esiste un fattore specifico che guida il fenomeno dei mashup: *la possibilità, offerta agli utenti, di innovare*, cioè il desiderio e la capacità degli utenti di sviluppare "prodotti" propri e di realizzare le proprie idee, esprimendo così le proprie capacità personali. Per comprendere meglio questo fattore trainante, consideriamo il fenomeno dilagante di Twitter, un servizio di microblogging attraverso cui amici e collaboratori possono rimanere in contatto e scambiarsi opinioni. Circa l'80% del traffico di Twitter parte attraverso la sua API pubblica; ciò vuol dire che il traffico è generato principalmente da applicazioni web che al loro interno integrano tale servizio. I fondatori di Twitter considerano l'API pubblica la loro più importante

scelta di innovazione¹: "Ci ha permesso in primo luogo di mantenere il servizio semplice e di creare uno strumento di facile utilizzo tramite cui gli sviluppatori possono costruire applicazioni nuove sulla base della nostra piattaforma e venir fuori con idee [come per esempio *Twitterific*] che sono decisamente migliori [grazie all'innovazione creata dagli utenti] delle nostre". È così che Twitter è diventata un'infrastruttura per applicazioni Web create dagli utenti del Web, secondo un paradigma che l'economista Thomas Hughes ha definito *invenzione conservativa* [4]: l'innovazione generata da terze parti contribuisce alla crescita di un sistema, mentre il sistema stesso continua a rimanere sotto il controllo del suo produttore originale.

In un ciclo tradizionale di progettazione-sviluppo-valutazione, il riscontro del cliente che commissiona l'applicazione diventa disponibile ai progettisti solo dopo il rilascio di un prototipo, quando ogni modifica può comportare costi considerevoli. In un approccio basato sull'innovazione guidata dagli utenti, un'azienda offre un *pacchetto di innovazione* attraverso cui i clienti stessi possono costruire il loro prodotti [12]. Questo pacchetto fornisce un'interfaccia limitata sulle potenzialità della piattaforma dell'azienda, così da assicurare che i nuovi prodotti siano costruiti in modo corretto e appropriato (per esempio senza violazioni d'uso). L'idea alla base è quindi che la sperimentazione iterativa, necessaria per sviluppare un nuovo prodotto, può essere portata avanti interamente dall'utente stesso del prodotto. Diversi utenti possono lavorare in parallelo alla soluzione di un problema, ognuno concentrandosi su una propria versione. Essi possono creare soluzioni ad hoc per i loro requisiti e possono velocemente verificarne la fattibilità tramite i loro esperimenti di sviluppo. Allo stesso tempo, l'azienda che fornisce il pacchetto di innovazione non deve sostenere il costo di produzioni senza successo. Inoltre, quando un esperimento produce valore nuovo e rilevante, l'azienda ha il diritto di integrare l'innovazione nel suo prodotto originario. Questo è quanto successo nel Web quando alcuni

¹ <http://readwritetalk.com/2007/09/05/biz-stone-co-founder-twitter>

sviluppatori integrarono in un mashup il servizio di Flickr con un servizio di mappe: Flickr ha successivamente incorporato una funzionalità di mappa sia nella sua piattaforma Web, sia nel suo servizio pubblico. Anche Google monitora costantemente l'uso delle sue API pubbliche (come per esempio Google Maps e Google Search) per raffinarle "appropriarsi" degli usi innovativi migliori [6].

L'apertura dei servizi affinché terze parti possano utilizzarli nei propri mashup è quindi una scelta strategica, che rivoluziona il modello di business che per anni ha caratterizzato il Web e le sue applicazioni. Piuttosto che continuare a essere ricettori passivi d'innovazione, gli utenti del Web diventano attori attivi nella creazione dell'innovazione. Il controllo sempre

più crescente sulla proprietà intellettuale della soluzione creata fornisce un incentivo per gli utenti affinché essi si assumano le responsabilità dello sviluppo del software. Questo significa anche che il valore creato dall'innovazione è ora condiviso con gli utenti. Il valore globale è con buona probabilità più grande, così che la condivisione con gli utenti rimanga comunque vantaggiosa.

In tale contesto, l'innovazione del software è divenuta sempre più sistemica [8]. Un'azienda dipende sempre più da fornitori esterni per la produzione di servizi che estendono un prodotto base. I fornitori esterni non sono sotto il controllo diretto dell'azienda. Quindi, per poter contare su di essi, l'azienda deve dimostrare un impegno credibile nell'innovazione sistemica. I fornitori di servizi aperti, per esempio, cercano di attrarre fornitori esterni offrendo libero accesso ai loro dati e riducendo il controllo sulla proprietà dei dati e dei formati di scambio usati dalle API. I progressi dei mashup sono quindi il risultato logico di una tendenza generale a incrementare la complessità dei sistemi e allo stesso tempo la loro specificità. La complessità è legata al numero di tecnologie e di componenti incorporati in un prodotto. La specificità si riferisce invece al fatto che ogni azienda coinvolta nello sviluppo ha la possibilità di concentrarsi sulle competenze in cui eccelle, aumentando così l'efficienza del processo di produzione.

3. LO SCENARIO DI SVILUPPO DEI MASHUP

Il modo in cui i mashup sono sviluppati dipende dal tipo di mashup. Attualmente, i mashup destinati ai "consumatori" (per esempio, tutti i numerosi mashup che integrano mappe) sono perlopiù il risultato di attività di programmazione da parte di sviluppatori esperti. I mashup "aziendali" evidenziano invece diversi scenari di sviluppo, che vedono anche il coinvolgimento di utenti meno esperti. Prenderemo quindi spunto da studi recenti sui mashup aziendali [10, 9] e cercheremo di evidenziare i contributi che diversi attori, con diversi livelli di esperienza tecnica, possono apportare al loro sviluppo.

La figura 1 illustra tre diversi scenari, che differiscono per l'eterogeneità dei servizi integrati nei mashup, la diversità dei bisogni de-

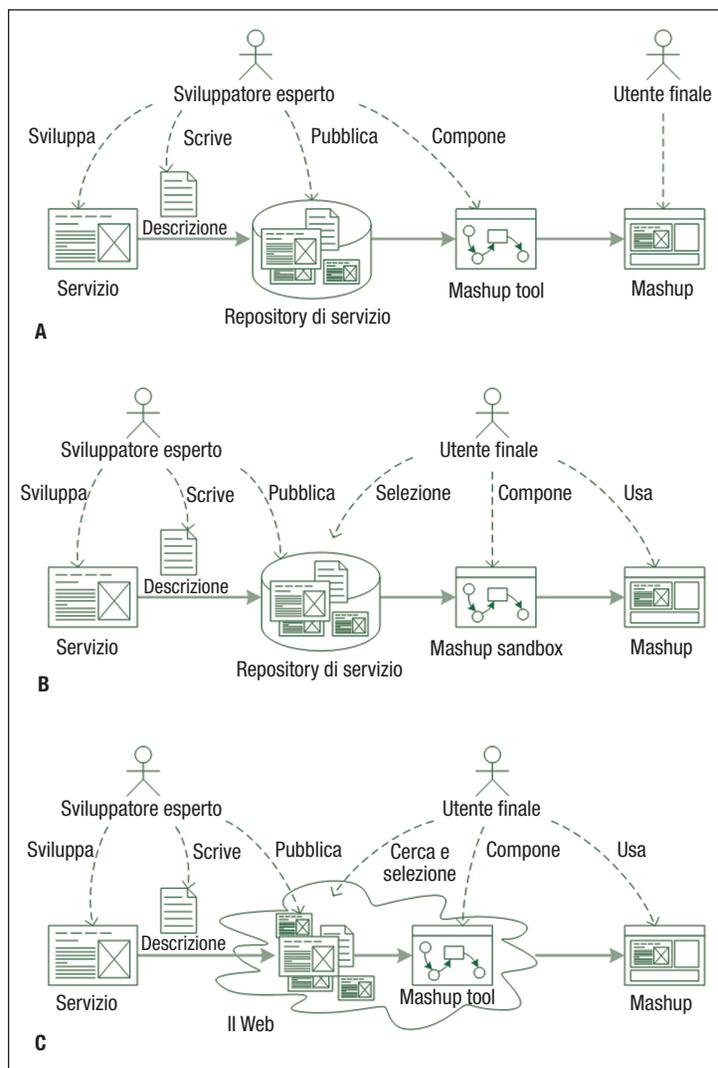


FIGURA 1
Scenari per l'uso dei mashup come tecnica di sviluppo software

gli utenti, e il grado di sofisticazione degli utenti e degli strumenti che supportano il loro lavoro:

A. I mashup possono essere creati da sviluppatori esperti (per esempio sviluppatori di un dipartimento IT o fornitori di servizio) in grado di rilasciare applicazioni in breve tempo. Gli utenti finali non sono coinvolti nella produzione dei mashup ma traggono in ogni caso benefici dai tempi di sviluppo ridotti.

B. Gli sviluppatori esperti creano servizi adatti a essere composti in un mashup e forniscono allo stesso tempo un ambiente ben delimitato (sandbox) dove gli utenti finali possono combinare i servizi. Il ruolo degli sviluppatori è quindi creare servizi, mentre i mashup sono costruiti dagli utenti, i quali sono maggiormente a conoscenza del dominio applicativo in cui i mashup saranno utilizzati.

C. Gli sviluppatori esperti rilasciano un ambiente che permette a ognuno di creare i propri mashup. Questo è analogo al modo in cui i fogli di calcolo sono usati oggi nelle organizzazioni: gli utenti finali, per esempio gli analisti finanziari di un'azienda, creano i propri fogli di calcolo senza dover necessariamente coinvolgere un programmatore esperto. I mashup sono spesso creati per un singolo scopo e per un unico utente e sono legati a una situazione specifica. Queste applicazioni sono, infatti, note come *situational applications* [1].

Quando i mashup sono sviluppati in modo centralizzato (scenario A), lo sviluppo è affidato ai soli sviluppatori esperti del dipartimento IT. Date le risorse limitate di un dipartimento IT, potranno essere sviluppate solo le applicazioni più richieste. Tuttavia, gli sviluppatori esperti hanno l'esperienza e le capacità necessarie a integrare servizi largamente eterogenei a un basso livello di astrazione, e possono quindi essenzialmente produrre applicazioni di ogni tipo.

Quando il dipartimento IT concentra le sue risorse sullo sviluppo di componenti e sul supporto necessario agli utenti per usare tali componenti (scenario B), esporrà certi tipi di dati e servizi in un formato facile da usare da utenti non sviluppatori. Rispetto ai programmatori esperti, questi utenti avranno una migliore conoscenza del dominio applicativo. Poiché gli sforzi per lo sviluppo dei mashup non sono totalmente a carico dello staff IT, è possibile sod-

disfare un maggior numero di richieste degli utenti. Tuttavia, i mashup che possono essere costruiti sono esclusivamente limitati ai componenti forniti dallo staff IT: la composizione del mashup consisterà in generale nella parametrizzazione dei componenti, poiché gli utenti non hanno la capacità di eseguire un'integrazione più "profonda" e articolata.

La produzione di uno strumento per lo sviluppo di mashup (scenario C) è sicuramente la soluzione più impegnativa da realizzare, ma allo stesso tempo la più proficua per i vari attori. Attraverso tale strumento, gli utenti possono integrare servizi e dati per creare i propri mashup. Rispetto ai mashup programmati "a mano", lo strumento limita le opzioni di integrazione al fine di garantire la correttezza della composizione, ma allo stesso tempo concede agli utenti la libertà di integrare qualsiasi tipo di componente disponibile nel Web. Tale strumento fa cioè in modo che gli utenti possano creare le proprie applicazioni, che possono essere di diversi tipi così da assecondare un'ampia diversità di requisiti [12].

Un'altra differenza tra i tre diversi scenari riguarda il grado di controllo sulla qualità dei mashup creati. Nello scenario A, lo sviluppo da parte dello staff IT è garanzia di qualità. Tuttavia, non tutti i mashup hanno requisiti stringenti in termini di sicurezza, prestazioni, o affidabilità, soprattutto se destinati a essere usati per uno scopo ben specifico, per il quale una soluzione complessa sviluppata da un dipartimento IT sarebbe troppo costosa. Nello scenario B, il dipartimento IT seleziona i componenti che possono essere integrati nei mashup degli utenti e fornisce inoltre una piattaforma per comporre ed eseguire i mashup. Gli utenti sono quindi abilitati a creare mashup per soddisfare bisogni specifici che difficilmente potrebbero essere previsti e assecondati dal dipartimento IT. I mashup sviluppati dagli utenti nello scenario C possono in seguito essere utilizzati come prototipi per rendere più solide le applicazioni sviluppate dal dipartimento IT, per esempio nel caso in cui dovesse emergere il bisogno di rendere disponibili quei mashup a diversi utenti all'interno dell'azienda o a clienti esterni.

Secondo von Hippel [13], i mashup rendono l'innovazione "democratica", permettendo agli utenti finali di soddisfare quei bisogni cui

un dipartimento IT centralizzato, o più in generale un fornitore di servizio, non è sempre in grado di rispondere. Il loro uso riduce i tempi necessari a implementare una certa funzionalità. Uno dei possibili usi dei mashup è, infatti, la prototipazione di soluzioni a problemi di utenti specifici, che possono in seguito essere generalizzate per soddisfare i bisogni di una comunità più ampia. Lo sviluppo dei mashup è quindi simile allo sviluppo *open source* (che ha ispirato la metafora di von Hippel) secondo due diverse sfaccettature:

- coloro che contribuiscono allo sviluppo di un progetto open source sono anche utenti del software prodotto;
- i progetti open source forniscono un meccanismo attraverso il quale, chi contribuisce attivamente al progetto passa dall'essere un utente passivo al fornire utili commenti, richieste di funzionalità aggiuntive e in alcuni casi pezzi di codice.

In modo simile, gli sviluppatori di mashup sono spesso anche gli utenti dei mashup (vedi scenari B e C nella Figura 1); non tutti gli utenti dei mashup rivestono necessariamente il ruolo di sviluppatori, ma possono comunque contribuire allo sviluppo fornendo commenti e richiedendo nuove funzionalità durante lo sviluppo dei prototipi.

4. PROCESSI DI SVILUPPO LEGGERI

A partire dagli scenari di sviluppo tipici dei mashup, è possibile fare qualche considerazione su come i processi di sviluppo per il Web debbano modificarsi per soddisfare le esigenze di questa nuova classe di applicazioni. È noto come il ciclo di vita delle applicazioni

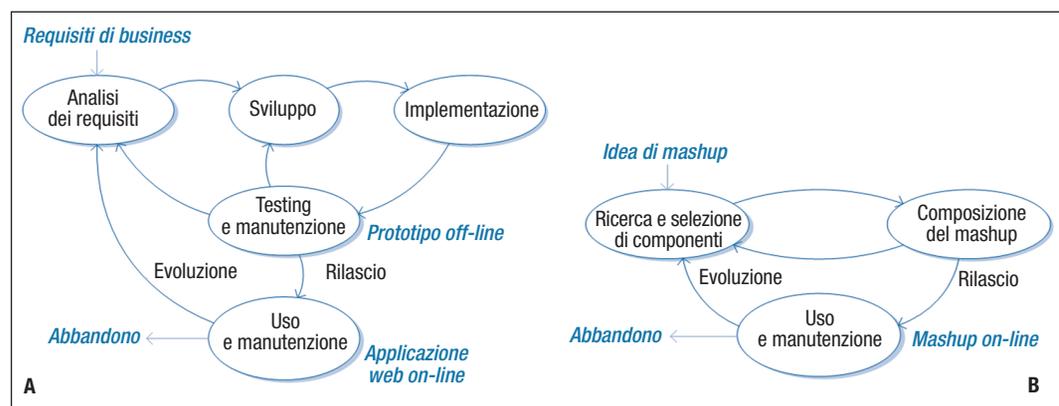
Web sia già molto più dinamico rispetto ad altre classi di applicazioni software:

1. lo sviluppo per il Web deve essere veloce: i prototipi e le applicazioni finali devono essere sviluppate nei "tempi di Internet", e cioè in giorni o settimane, invece che in mesi o anni;
2. la possibilità di registrare i dati di utilizzo di una grande mole di utenti facilita e snellisce l'attività di testing e le analisi di usabilità;
3. dopo il rilascio di un'applicazione, eventuali evoluzioni e raffinamenti possono essere realizzati mentre l'applicazione è ancora online; in altre parole, un'applicazione Web è potenzialmente soggetta a continue evoluzioni.

Lo sviluppo per il Web si articola quindi in due fasi principali: *lo sviluppo incrementale* della versione base dell'applicazione e *l'evoluzione incrementale* successiva al rilascio. Se pur dinamico e veloce, questo processo va ben oltre i requisiti di sviluppo dei mashup. Sposa bene le caratteristiche e le capacità dei programmatori esperti che sviluppano applicazioni Web tradizionali - si veda la figura 2 A - ma non incontra le aspettative e le capacità degli utenti che vogliono prender parte allo sviluppo; quindi, non promuove l'innovazione.

Il processo di sviluppo ideale per i mashup dovrebbe riflettere il potenziale di tali applicazioni: poter *comporre* un'applicazione partendo da *contenuti* e funzionalità che rispondono a bisogni personali, e poterla *eseguire* con facilità, senza doversi preoccupare di ciò che succede dietro la scena. Lo sviluppo iterativo, centrato sul rilascio di prototipi, è ancor più accentuato: il compositore di mashup combina servizi aperti ed esegue il risultato per controllare che tutto funzioni correttamente e soddisfi i suoi bisogni. In caso di risultati insoddisfacenti, il compositore stesso risolve i

FIGURA 2
A - Modello del ciclo di vita delle applicazioni web attuali, B - Modello del ciclo di vita dei mashup



problemi e quindi esegue immediatamente il nuovo mashup per provare l'efficacia delle modifiche apportate. Data la natura del mashup, strettamente legata alla specifica situazione di utilizzo, il ruolo del committente dell'applicazione deve essere reinterpretato: i requisiti iniziali, di fatto, corrispondono a quei bisogni "di breve termine" del compositore del mashup, dai quali ha origine l'idea di sviluppare il mashup. Inoltre, le fasi di *progettazione, implementazione, testing e valutazione* assumono un'importanza diversa, e dovrebbero essere in qualche modo semplificate (o addirittura nascoste) attraverso l'uso di strumenti adeguati.

Così come mostrato nella figura 2 B, queste considerazioni possono essere riassunte in un modello che promuova un *processo di sviluppo leggero*, caratterizzato da tre attività principali:

□ *scoperta e selezione*: partendo da un'idea iniziale di mashup, che generalmente riflette bisogni e preferenze personali, il compositore seleziona servizi in grado di fornire i dati, la logica applicativa e l'interfaccia di cui ha bisogno – per la maggior parte servizi aperti, reperibili nel Web.

Questa è un'attività nuova, specifica dello sviluppo di mashup, che precede la creazione del mashup e assorbe implicitamente l'analisi e la specifica dei requisiti. L'idea iniziale può essere, infatti, vista come un'espressione informale dei requisiti dell'applicazione. I servizi selezionati provano la fattibilità dell'idea e allo stesso tempo esprimono un primo abbozzo di quella che sarà l'organizzazione finale del mashup.

□ *Creazione del mashup*: strumenti "ad-hoc" per lo sviluppo dei mashup devono permettere anche agli utenti meno esperti di combinare alcuni componenti e definire la logica di integrazione necessaria e gli aspetti di presentazione dell'applicazione composita. La logica d'integrazione deve basarsi su formalismi e su modelli intuitivi. I modelli d'integrazione e di esecuzione del mashup possono essere mascherati da notazioni visuali, per semplificare così ancor di più agli utenti meno esperti l'attività di composizione.

Questa attività dà un senso diverso alle tradizionali fasi di *progetto e implementazione*, perché ridimensiona la necessità di eseguire attività tipiche nel mondo del Web, come la

progettazione dell'ipertesto. Il rilascio dell'applicazione consisterà semplicemente nel salvare il mashup e renderlo disponibile su una piattaforma di esecuzione (un'attività che dovrebbe richiedere un solo click).

□ *Uso e manutenzione*: una volta rilasciato sulla piattaforma di esecuzione, il mashup sarà immediatamente pronto per essere utilizzato. La manutenzione dell'applicazione sarà condivisa tra il compositore del mashup (che per esempio risolverà problemi nella logica di composizione) e il gestore della piattaforma di esecuzione (che per esempio risolverà i problemi legati all'ambiente di esecuzione).

Questa attività incorpora le tradizionali fasi di *test e valutazione* e *uso e manutenzione*. Eseguendo facilmente il mashup, il compositore può controllare con altrettanta facilità che l'applicazione funzioni correttamente e che risponda ai suoi bisogni, o addirittura testarla con altri utenti. Eventuali evoluzioni successive richiederanno semplicemente di ripetere il processo di sviluppo, ripartendo dalla prima fase di scoperta e selezione.

L'applicabilità di un tale processo di sviluppo dipende dalla disponibilità di un insieme di strumenti adeguati, tra i quali una piattaforma dedicata per l'esecuzione dei mashup e un insieme di servizi aperti, disponibili nel Web, in grado di fornire nuovo valore attraverso un ricco insieme di funzionalità e di dati. Il processo di sviluppo può essere ulteriormente semplificato nella fase di rilascio e di pubblicazione del mashup se si fa uso di piattaforme in grado di supportare sia la composizione del mashup sia la sua esecuzione (questo in parte è già possibile). In tale contesto di sviluppo, è quindi possibile affermare che il ciclo di vita dei mashup parte dal punto di rilascio nella figura 2 A e procede attraverso evoluzioni incrementali. Infatti, una volta creato, il mashup è immediatamente in esecuzione, senza la necessità di cicli di sviluppo incrementali.

5. STRUMENTI DI SVILUPPO PER I MASHUP

Quali strumenti di sviluppo possono abilitare anche gli utenti meno esperti a sviluppare i propri mashup e a innovare?

L'integrazione di componenti in un mashup può essere realizzata tramite un qualsiasi lin-

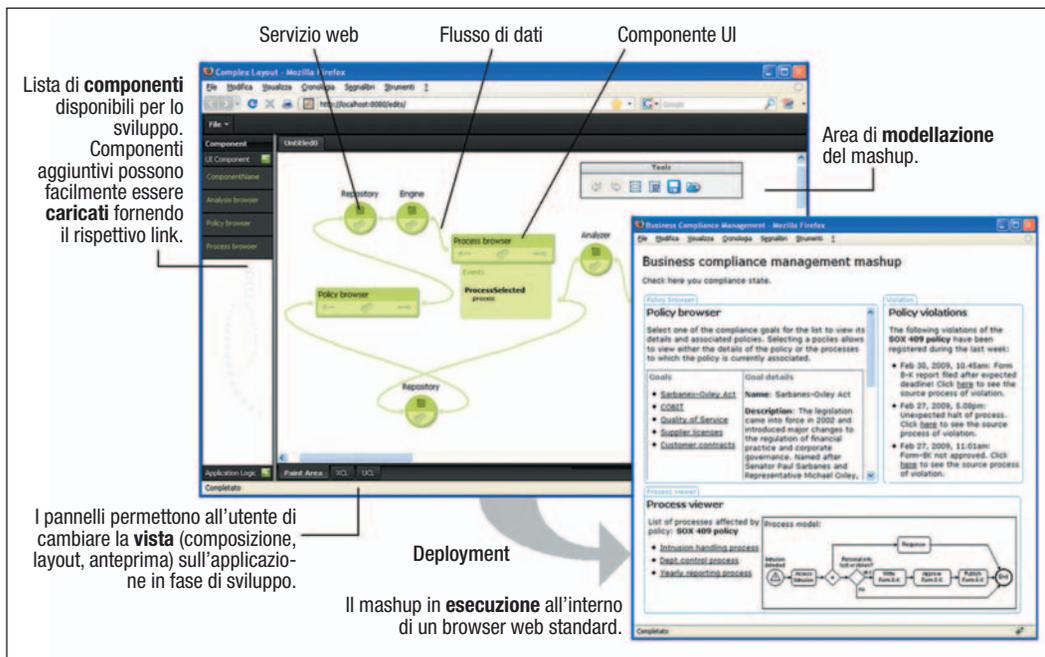


FIGURA 3
L'editor mashArt per lo sviluppo visuale di "composizioni universali". Le due schermate mostrano la composizione visuale e il mashup finale per un'applicazione di vigilanza sulla conformità alle norme (la cosiddetta "compliance")

guaggio di programmazione, per esempio PHP o JavaScript, tanto per citare i linguaggi più utilizzati [11]. Vista l'eterogeneità dei componenti, dei linguaggi di programmazione, dei protocolli di interazione, della complessità della logica di integrazione, lo sviluppo manuale dei mashup è tuttavia praticabile solo da parte di programmatori esperti ed è sicuramente fuori della portata degli utenti finali del Web. Gli approcci ai BPEL (*Business Process Execution Language*) proposti per la composizione dei servizi non sono in grado di far fronte in modo adeguato all'eterogeneità delle tecnologie che caratterizza i mashup e sono inoltre troppo complessi da utilizzare.

Permettere a un'ampia classe di utenti (non solo agli sviluppatori esperti) di comporre mashup, e quindi di innovare, richiede la disponibilità di strumenti di sviluppo intuitivi e facili da usare [2]. Esistono già in letteratura diverse proposte di strumenti per lo sviluppo dei mashup (i cosiddetti "mashup makers"), che in genere mettono a disposizione ambienti di sviluppo visuali per poter facilmente comporre i mashup a partire da una selezione di servizi. Tuttavia, nessuno di questi strumenti offre un ambiente integrato di sviluppo in grado di supportare appieno la composizione di servizi eterogenei. Ognuno di essi offre supporto per specifiche attività di sviluppo. Per esempio, Yahoo Pipes (<http://pipes.yahoo.com>) si concentra sull'in-

tegrazione dei dati (RSS o Atom feed) e offre un linguaggio tramite cui comporre flussi di dati (*pipes*), mentre non fornisce alcun supporto per l'integrazione a livello di presentazione. JackBe Presto (<http://www.jackbe.com>) adotta un approccio per l'integrazione dei dati simile a quello delle pipes di Yahoo, permettendo in più l'aggregazione di componenti visuali (*mashlets*) per la resa dell'output del mashup finale.

Nessuno di questi strumenti permette l'integrazione a tutti e tre i livelli che caratterizzano le applicazioni Web, e cioè dati, logica applicativa e presentazione.

Lo sviluppo di una piattaforma che copra i tre strati è oggetto della nostra ricerca corrente, il cui risultato è un ambiente di sviluppo e di esecuzione dei mashup, *mashArt* (<http://mashart.org>)², che ambisce a supportare una *integrazione universale*, cioè un approccio di integrazione in grado di gestire in modo trasparente una varietà di tipi di componenti e di tecnologie, dai semplici RSS feed ai più complessi servizi SOAP e RESTful e ai componenti UI. Come mostrato nella figura 3, in *mashArt* la composizione dei mashup è eseguita per mezzo di un editor visuale, sviluppato in AJAX. Gli utenti

² *mashArt* è l'evoluzione di *Mixup* [15], uno strumento precedentemente sviluppato per l'integrazione al solo livello di presentazione.

possono tracciare visualmente la logica di composizione dei loro mashup e definire il layout dell'applicazione finale. Possono quindi salvare e caricare la loro applicazione sulla piattaforma, ed eseguirla istantaneamente col supporto della piattaforma in modalità "hosted"; tutto senza dover scrivere una sola riga di codice. La logica di composizione è basata su eventi (tramite cui è gestita la sincronizzazione delle interfacce utente di ogni componente), e sul flusso dei dati (in base al quale è possibile orchestrare l'esecuzione dei servizi)³.

6. SVILUPPI FUTURI

In questo articolo, abbiamo proposto una visione sui mashup, sugli strumenti per il loro sviluppo, e sul loro ciclo di vita, nel tentativo di contribuire alla definizione di una base comune di concetti e di metodi per lo sviluppo di tali applicazioni. È nostra opinione che una comprensione sistematica di come le pratiche correnti per lo sviluppo di applicazioni Web debbano modificarsi per rispondere alle nuove esigenze dei mashup sia necessaria, soprattutto considerando che il fenomeno dei mashup è destinato a crescere sia nel contesto aziendale [7], sia in quello più ampio degli utenti del Web [15].

Abbiamo mostrato come abilitare ogni classe di utenti a sviluppare le proprie applicazioni richieda la definizione di metodi facili da comprendere e gestire. Al fine di rendere applicabile il modello di sviluppo leggero discusso in questo articolo, la ricerca sui mashup deve necessariamente concentrarsi sui seguenti aspetti:

□ *servizi facili da usare*: è necessario definire modelli e linguaggi di descrizione intuitivi ed espressivi per i dati, la logica applicativa e le interfacce utente dei servizi, in modo da rendere semplice la scoperta e l'integrazione dei servizi. Funzionalità di ricerca e di raccomandazione per la selezione dei servizi sono altrettanto necessarie.

□ *Strumenti per la composizione visuale*: è auspicabile definire notazioni visuali intuitive per la modellazione dei mashup, tali da poter mascherare l'eventuale complessità del linguaggio di composizione con cui è realmente

gestita l'esecuzione dei mashup, accoppiati a meccanismi per la trasformazione automatica delle notazioni visuali in codice eseguibile dell'applicazione.

□ *Piattaforme di esecuzione adeguate*: vista la natura situazionale dei mashup, il rilascio e la pubblicazione dei mashup devono essere facilitati, così da poter eseguire e testare in tempi brevi le applicazioni create, per esempio tramite un interprete online del linguaggio di composizione. Un contributo in tale direzione può derivare dall'adozione di piattaforme web dedicate all'esecuzione "hosted" di mashup.

□ *Progettazione orientata all'interoperabilità*: i servizi, in primo luogo, e anche i mashup devono essere interoperabili, cioè riusabili su più piattaforme. È perciò auspicabile definire standard specifici per i mashup.

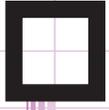
□ *Robustezza dei mashup*: sebbene gli sforzi della ricerca corrente sui mashup si concentrino soprattutto sugli aspetti metodologici e sulle tecnologie abilitanti, sarà necessario affrontare problematiche legate all'affidabilità e alla sicurezza dei mashup – soprattutto per i mashup utilizzati in contesti critici, quali quelli di business.

In fine, è importante sottolineare che, se da un lato i processi di sviluppo leggeri sono necessari per alleviare gli sforzi degli sviluppatori di mashup, e in particolare quelli degli utenti meno esperti, lo sviluppo dei servizi da integrare nei mashup rimane un'attività che richiede competenze tecniche specifiche e che deve essere eseguita secondo le pratiche di sviluppo tradizionali. Infatti, se da un lato il successo di un mashup è influenzato dal valore aggiunto dato dalla combinazione di servizi diversi, dall'altro la qualità della composizione finale dipende fortemente dalla qualità di ogni singolo servizio. La definizione di modelli e di tecniche per lo sviluppo di servizi di qualità e per la valutazione della loro qualità è quindi un'altra promettente direzione verso lo sviluppo di mashup di qualità [3], che può ancor più promuovere l'innovazione guidata dagli utenti.

Bibliografia

- [1] Balasubramaniam S., Lewis G.A., Simanta S., Smith D.B.: *Situated Software: Concepts, Motivation, Technology, and the Future*. IEEE Software, Nov-Dec, 2008, p. 50-55.

³ Una dimostrazione di mashArt è disponibile all'indirizzo <http://mashart.org/mashArt.wmv>.



- [2] Burnett M., Cook C., Rothermel C.: End-User Software Engineering. *Communications of the ACM*, Vol. 47, n. 9, 2004, p. 53-58.
- [3] Cappiello C., Daniel F., Matera M.: *A Quality Model for Mashup Components*. ICWE'09, Springer LNCS, p. 235-249.
- [4] Daniel F., Yu J., Benatallah B., Casati F., Matera M., Saint-Paul R.: Understanding UI Integration: A Survey of Problems, Technologies, and Opportunities. *IEEE Internet Computing*, Vol. 11, n. 3, 2007, p. 59-66.
- [5] Hughes T.P.: *The evolution of large technological systems*. In: The social construction of technology systems: New directions in the sociology and history of technology. Bijker W.E., Hughes T.P., Pinch T.J., eds. Cambridge, Mass., MIT Press, p. 51-82.
- [6] Iyer B., Davenport T.H.: Reverse Engineering Google's Innovation Machine. *Harvard Business Review*, Vol. 86, n. 4, p. 58-69.
- [7] Jhingran A.: *Enterprise information mashups: integrating information, simply*. VLDB'06, p. 3-4.
- [8] Maula M., Keil T., Salmenkaita J.-P.: *Open Innovation in System Innovation Contexts*. Open Innovation: Researching a New Paradigm, Chapter 12, 2006, p. 249-257.
- [9] Obrenovic Z., Gasevic D.: *Mashing Up Oil and Water: Combining Heterogeneous Service for Diverse Users*. *IEEE Internet Computing*, Nov/Dec, 2009, p. 56-64.
- [10] Ogrinz M.: *Mashup Patterns: Designs and Examples for the Modern Enterprise*. Addison-Wesley, 2009.
- [11] Sabbouh M., Higginson J., Semy S., Gagne D.: *Web mashup scripting language*. WWW'07, p. 1305-1306.
- [12] Thomke S., von Hippel E.: Customers as Innovators: A New Way to Create Value. *Harvard Business Review*, Vol. 80, n. 4, 2002, p. 74-81.
- [13] von Hippel E.: *Democratizing Innovation*. MIT Press, 2005.
- [14] Weiss M., Gangadharan G.R.: *Modeling the Mashup Ecosystem: Structure and Growth*. R&D Management, 2009 (accepted for publication).
- [15] Yu J., Benatallah B., Saint-Paul R., Casati F., Daniel F., Matera M.: *A Framework for Rapid Integration of Presentation Components*. WWW'07, p. 923 - 932.

FLORIAN DANIEL è ricercatore post-doc al Dipartimento di Ingegneria e Scienza dell'Informazione dell'Università degli Studi di Trento. Ha conseguito il titolo di dottore di ricerca in Ingegneria dell'Informazione presso il Politecnico di Milano nel 2007 e ha lavorato come visiting researcher presso gli HP Labs di Palo Alto, California, nel 2006. I suoi interessi di ricerca comprendono i mashup e, più in generale, i metodi di sviluppo di applicazioni Web basate su componenti (Web services o interfacce utenti), lo sviluppo di applicazioni e servizi Web complessi, la compliance nelle applicazioni basate su Web services, e la qualità e la privacy nelle applicazioni di business intelligence. Florian è co-autore del libro "Engineering Web Applications" (Springer, 2009) e ha pubblicato più di 50 articoli in riviste, conferenze e workshop internazionali.
E-mail: daniel@disi.unitn.it

MARISTELLA MATERA è ricercatrice di ruolo presso il Politecnico di Milano, dove nel 2000 ha conseguito il dottorato in Ingegneria dell'Informazione e dell'automazione. La sua attività di ricerca si concentra su modelli e strumenti per lo sviluppo dei mashup e per la valutazione della loro qualità, sulla modellazione e lo sviluppo di applicazioni web, con particolare enfasi sull'adattività e la context-awareness e su processi cooperativi, sulla qualità delle applicazioni Web. Su tali argomenti, ha scritto più di cento articoli scientifici, pubblicati su riviste e atti di conferenze internazionali. È inoltre coautrice del libro "Designing Data-Intensive Web Applications" (Morgan Kaufman, 2000) e del libro "Engineering Web Applications" (Springer, 2009).
E-mail: maristella.matera@polimi.it

ICT E INNOVAZIONE D'IMPRESA

Casi di successo

Rubrica a cura di

Roberto Bellini, Chiara Francalanci

La rubrica *ICT e Innovazione d'Impresa* vuole promuovere la diffusione di una maggiore sensibilità sul contributo che le tecnologie ICT possono fornire a livello di innovazione di prodotto, di innovazione di processo e di innovazione di management. La rubrica è dedicata all'analisi e all'approfondimento sistematico di singoli casi in cui l'innovazione ICT ha avuto un ruolo critico rispetto al successo nel business, se si tratta di un'impresa, o al miglioramento radicale del livello di servizio e di diffusione di servizi, se si tratta di una organizzazione pubblica.



Il ruolo dell'IT nell'ottimizzazione dei consumi energetici: il caso Stamperia Olonia

Eugenio Capra, Daniele Zagordi, Alex Zazzera

1. INTRODUZIONE

La sensibilità per i problemi legati alla sostenibilità ambientale e al consumo energetico è fortemente aumentata negli ultimi anni, sia per l'instabilità e l'aumento dei costi unitari dell'energia [1] che per l'attenzione che le istituzioni, specialmente europee, hanno dimostrato verso queste tematiche (si pensi alla riduzione del 20% delle emissioni che l'Europa si è imposta per il 2020 [2]).

Secondo quanto riportato dall'economista Nicholas Stern nel rapporto *Stern Review* prodotto per il governo inglese [3], ignorare quanto sta accadendo in tema di cambiamenti climatici causerà un enorme danno all'economia mondiale nel prossimo futuro; infatti, se non verrà predisposta alcuna azione, i danni ammonteranno a circa il 5% del Prodotto Interno Lordo mondiale [3].

Anche per questo motivo, trentaquattro Paesi di tutto il mondo hanno sottoscritto il protocollo di Kyoto, un accordo promosso nell'ambito della *Convenzione quadro delle Nazioni Unite sui cambiamenti climatici* (UNFCCC), che pone come traguardo la riduzione da parte di ciascuna nazione partecipante delle proprie emissioni nocive per

una quota pari al 5,4% di quelle relative all'anno 1990, entro il 2012.

Questo trend ha investito anche l'IT, che gioca un ruolo non trascurabile per quanto riguarda l'impatto sull'ambiente, come è stato dimostrato da ricerche ormai molto diffuse e affermate: una ricerca Gartner del 2007 ripresa da Intel afferma che l'IT consuma il 2% dell'energia mondiale e che una maggiore efficienza dell'IT stesso porterebbe a grossi risparmi. Tuttavia concentrandosi solo sull'IT si rischierebbe di escludere il restante 98%, settori in cui l'IT potrebbe diventare fattore abilitante al cambiamento e all'efficienza; una percentuale decisamente maggiore e da non sottovalutare [4]. Questo ha fatto nascere il fenomeno del "Green IT", cioè dello studio dell'impatto ambientale delle tecnologie IT, di cui tanto si parla ultimamente.

Se da un lato dunque le tecnologie ICT sono responsabili di una quota rilevante dei consumi energetici mondiali e di conseguenza delle emissioni nocive ad essi correlati, dall'altro possono ricoprire un ruolo strategico nel monitoraggio e nell'ottimizzazione di diversi processi industriali e aziendali [5]. L'impatto potenziale di questo approccio, che ci piace chiamare "IT for a greener

business” è di gran lunga più ampio di quello del Green IT. Secondo uno studio di McKinsey [6] l’IT entro il 2020 contribuirà ad abbattere 7,8 miliardi di tonnellate equivalenti di CO₂, pari a circa il 15% delle emissioni complessive di oggi.

Questo articolo ha l’obiettivo di dimostrare attraverso un caso concreto come soluzioni IT implementate con tecnologie già mature e ampiamente disponibili sul mercato, e quindi a fronte di investimenti modesti, possano far diminuire notevolmente i consumi di un processo industriale. In particolare, il caso di studio qui presentato riguarda l’ottimizzazione del processo di asciugatura dei tessuti della Stamperia Olenia, un’azienda operante nella provincia di Varese, da sempre attenta ad un uso strategico delle tecnologie ICT.

L’implementazione di un sistema informativo basato su speciali sonde di temperatura, un comune controllore PID, e un algoritmo di ottimizzazione integrato, ha permesso una riduzione del 10% dei costi legati ai consumi energetici (elettricità e gas). Se una soluzione di questo tipo fosse adottata in tutte le industrie tessili italiane, si potrebbero risparmiare circa 2.788 TWh/anno, equivalenti a 2 milioni di tonnellate di CO₂eq, pari a circa 1 milione di alberi.

2. COME L’IT PUÒ OTTIMIZZARE I CONSUMI ENERGETICI DEI PROCESSI INDUSTRIALI

L’IT ricopre un ruolo strategico per la riduzione delle emissioni nocive, innanzi tutto perché permette di raccogliere e analizzare un’enorme quantità di informazione sull’impatto ambientale di tutti i processi, che possono così essere monitorati tramite opportuni sistemi informativi direzionali basati su *Green Performance Indicator* (GPI). Solo conoscendo i consumi e le inefficienze è infatti possibile attuare le strategie di miglioramento più opportune.

Le tecnologie IT possono però anche favorire l’ottimizzazione dei processi, generando una riduzione delle emissioni nocive pari a circa cinque volte le emissioni di cui è causa [7].

Questo può avvenire secondo diverse modalità, che raggruppiamo in cinque principali leve:

□ **Dematerializzazione** – sostituire prodotti e servizi ad alto *carbon footprint*, con soluzioni “digitali” equivalenti; per esempio l’utilizzo di sistemi di videoconferenza al posto delle confe-

renze in prima persona oppure l’introduzione della documentazione elettronica.

□ **Ottimizzazione dei processi produttivi** – introdurre sistemi di controllo che riescano ad ottimizzare i consumi energetici dei motori elettrici utilizzati nel settore industriale.

□ **Smart logistics** – ovvero migliorare i mezzi di trasporto e ottimizzarne i percorsi così da ridurre il numero di km percorsi e la quantità di combustibile utilizzato.

□ **Smart buildings** – dotare gli edifici residenziali di sistemi di controllo intelligenti in grado di ottimizzarne i consumi di energia.

□ **Smart grids** – introdurre sistemi di monitoraggio sulle reti di trasmissione elettrica così da ottimizzarne la distribuzione.

Risulta evidente come nei diversi contesti di ottimizzazione l’obiettivo sia sempre la riduzione dei consumi energetici in virtù dello stretto legame che c’è tra questi e le emissioni nocive.

Il caso presentato di seguito riguarda un esempio di ottimizzazione dei processi produttivi, in cui l’applicazione di opportune tecnologie ICT a supporto di un processo industriale del settore tessile ha portato miglioramenti non solo all’efficienza energetica, ma anche alle prestazioni aziendali.

3. I CONSUMI ENERGETICI DEL SETTORE TESSILE ITALIANO

Il settore tessile italiano è responsabile di circa il 7% dei consumi energetici nazionali [8]. In realtà sarebbe più corretto parlare di “filiera tessile”, intendendo l’insieme dei settori industriali coinvolti nella realizzazione e nella vendita di un manufatto tessile stampato. L’utilizzo del termine “filiera” è dovuto al fatto che il processo che la materia prima attraversa per giungere al prodotto finito, comprende una serie di macrofasi, ciascuna realizzata da piccole imprese altamente specializzate in una o più fasi della produzione. Ciascuna azienda lavora per conto terzi il lavoro o semilavorato ricevuto dall’azienda che la precede favorendo così un’organizzazione delle diverse imprese in specifici *distretti industriali*, che ad oggi risultano sparsi sul territorio italiano in cinque differenti regioni, a seconda del tipo di fibra lavorata. All’interno della filiera tessile è possibile identificare le principali fasi:

- produzione delle materie prime;
- filatura;

- tessitura;
- tintura;
- stampa;
- finissaggio;
- confezionamento e distribuzione.

Alcune analisi sui consumi energetici richiesti da ciascuna di queste fasi hanno mostrato che circa il 77% di questi consumi sono generati dai processi di tintura, stampa e finissaggio in quanto tutti richiedono a valle una fase di asciugatura del tessuto. Infatti, il colore che viene utilizzato per la stampa di quella che in gergo viene chiamata "pezza", è composto al 90% da acqua e richiede così un processo in grado di asciugare il tessuto stampato. Ciò avviene trasportando il tessuto tramite un nastro trasportatore all'interno di vere e proprie caldaie, le cui temperature variano da 90 a 190 gradi centigradi e che permettono di eseguire l'asciugatura. All'interno di queste caldaie sono tipicamente presenti alcuni bruciatori (per esempio a metano) e opportuni ventilatori in grado di "soffiare" l'aria calda sul tessuto permettendo così lo scambio termico. All'uscita il tessuto risulterà così asciutto e pronto per la successiva fase della filiera.

Nella maggior parte dei casi i parametri di funzionamento del processo di asciugatura vengono stabiliti in base all'esperienza empirica e non a precisi algoritmi. Inoltre, l'attuazione delle regolazioni è affidata agli operai a bordo macchina. È abbastanza evidente un disallineamento di obiettivi fra l'addetto a bordo macchina e la strategia aziendale: l'addetto, che non è responsabile dei costi di bolletta, tenderà infatti a regolare la temperatura e gli altri parametri al livello più alto possibile nei limiti della sicurezza, per essere certo che il tessuto sia effettivamente asciutto e non crei problemi (fermo macchina, necessità di pulire i rulli ecc.). La strategia aziendale invece prevede l'esecuzione dell'asciugatura a costo minimo.

4. LA STAMPERIA OLONIA

La Stamperia Olonia è un'azienda facente parte del distretto di Varese con sede a Gorla Minore, che opera nel settore della stampa a partire dal 1969. Negli anni '90 ha cominciato a espandersi anche sui mercati internazionali e la sua qualità l'ha spinto a diventare partner di importanti firme. Si occupa di candeggio, stampa e finissaggio raggiungendo una produzione annua di

circa 30 milioni di m². L'azienda conta 80 dipendenti e un fatturato annuo di circa 10 milioni di euro. Occupandosi di stampa, l'azienda lavora esclusivamente per conto terzi, ricevendo i tessuti parzialmente trattati e restituendoli opportunamente lavorati.

La spesa energetica della Stamperia Olonia è tra le più basse dell'intero settore (mediamente il costo energetico raggiunge il 10% del fatturato) grazie ai numerosi investimenti fatti per migliorare i controlli e l'efficienza. Nel 2008 il costo energetico si assestava intorno al 7% del fatturato mentre nel 2009 è salito all'8% a causa degli incrementi del prezzo del petrolio. Gli investimenti in questo campo sono cominciati attorno agli anni 2001-2002, dettati esclusivamente dal puro spirito di innovazione della proprietà e da una spiccata propensione verso la tecnologia. Conclusi nel 2007, con alcune migliorie ancora in via di sviluppo, questi progetti di innovazione si sono rivelati presto notevolmente convenienti generando un risparmio di circa il 15% sulle spese energetiche con un tempo di recupero degli investimenti di circa due anni. Con una forte propensione all'innovazione e anticipando i tempi, la Stamperia Olonia risulta ancora oggi all'avanguardia e tra le più importanti realtà del settore tessile.

5. UN SISTEMA INFORMATIVO A SUPPORTO DEL PROCESSO DI ASCIUGATURA

5.1. Il processo di asciugatura

Il processo di asciugatura della Stamperia Olonia viene attualmente svolto all'interno di un asciugatoio composto da 10 caldaie, avente ciascuna un bruciatore e un ventilatore, della potenza di circa 5 kW.

L'oggetto principale del processo di asciugatura è la *lavorazione* che risulta definita da alcuni fattori/componenti, tra cui:

- tipo di tessuto (cotone, tela, lycra ecc.);
- spessore del tessuto;
- tipo di colore utilizzato (pigmento, reattivo);
- disegno da applicare (in termini di quantità d'acqua che sarà immessa: un disegno molto coprente immetterà una grande quantità d'acqua nelle caldaie).

È evidente come ciascuna di queste caratteristiche incida sul processo di asciugatura del tessuto richiedendo una differente quantità di energia.

Nell'asciugatoio sono state rilevate quattro variabili da tenere in considerazione:

- temperatura dei bruciatori;
- velocità dei ventilatori;
- livello di umidità;
- temperatura del tessuto al secondo giro;

La temperatura dei bruciatori viene regolata tramite un controllo digitale a bordo macchina e, allo stato attuale delle cose, viene impostata inizialmente sulla base dell'esperienza pregressa. La velocità dei ventilatori è responsabile dello scambio termico che avviene sulla superficie del tessuto. Una velocità più elevata permetterà di alzare la temperatura sul tessuto e quindi di migliorarne l'asciugatura. È importante notare che, per ragioni fisiche, il tempo per aumentare la velocità dei ventilatori è decisamente minore rispetto a quello che si impiega per aumentare la temperatura dei bruciatori delle caldaie.

L'acqua che entra insieme al tessuto viene trasformata, nel processo di asciugatura, in vapore. L'umidità presente nell'asciugatoio è un parametro molto importante da tenere sotto controllo poiché un livello eccessivo di umidità nelle camere rischierebbe di creare della condensa e quindi di bagnare nuovamente il tessuto.

L'ultimo parametro da tenere in considerazione è la temperatura del tessuto al secondo giro dentro la caldaia, poiché il tessuto, prima di uscire dall'asciugatoio, viene ripiegato su se stesso: è importante che il tessuto sia asciutto al fine di evitare che il rullo si bagni e sporchi le successive lavorazioni.

In seguito ad alcune analisi, l'umidità e la temperatura al secondo giro sono state rivalutate in vincoli e si è quindi deciso che le possibili ottimizzazioni del processo consistevano nell'individuare per ciascuna lavorazione la migliore combinazione di intensità della fiamma dei bruciatori/velocità delle ventole (quindi consumo di gas/elettricità) in grado di asciugare il tessuto. Tuttavia, le due variabili in questione non sono completamente indipendenti tra loro come si potrebbe ipotizzare. I bruciatori hanno, per ragioni fisiche, un tempo di risposta maggiore rispetto ai ventilatori e, inoltre, la modifica di entrambe le variabili nello stesso momento potrebbe far saltare la condizione di temperatura al secondo giro, unico vero vincolo del problema. La soluzione migliore è dunque quella di far mutare una delle due variabili e attendere che un PID installato nella macchina modifichi l'altra per evitare che il

tessuto subisca danni (non asciugatura o essiccazione eccessiva).

Il problema quindi può essere modellato come ricerca di un minimo, all'interno di una funzione di costo bidimensionale, in cui il costo è in funzione di una delle due variabili da modificare.

La velocità dei ventilatori ha un tempo di reazione molto inferiore rispetto alla temperatura dei bruciatori, per cui si è optato per modificare quest'ultima e lasciare che il sistema si adatti alle nuove condizioni.

5.2. Il Sistema informativo proposto

Il primo intervento effettuato presso la Stameria Olonia è stato l'*inverterizzazione* di tutti i motori dell'asciugatoio, sia quelli che regolano la velocità e l'utilizzo dei ventilatori, sia quelli che regolano le cappe di aspirazione dell'umidità al fine di permetterne il controllo via software. Tuttavia l'inserimento degli inverter ha comportato costosi effetti collaterali in quanto è stata necessaria una re-implementazione del quadro elettrico per far sì che questo fosse in grado di gestirli. Successivamente sono state introdotte nel macchinario due tipi di sonde, una di temperatura e una di umidità. Queste sonde forniscono dati al sistema informativo, che regola così l'azione sui bruciatori e sui ventilatori grazie ad appositi attuatori.

Infine, un controllore PID monitora la situazione all'interno della caldaia e modifica opportunamente la velocità dei ventilatori rispettando i vincoli sopra citati.

È interessante sottolineare che inizialmente l'azienda utilizzava una politica ben precisa circa le azioni che il controllore doveva effettuare: partendo dalla convinzione che l'elettricità avesse un costo maggiore rispetto al gas faceva in modo che il PID tendesse ad abbassare il più possibile il livello di ventilazione a discapito dei bruciatori. Questa ipotesi si è rivelata in realtà non corretta.

5.3. L'algoritmo di ottimizzazione

Per poter trovare la combinazione di gas/energia elettrica in grado di ridurre al minimo i costi del processo di asciugatura è stato applicato un algoritmo di ottimizzazione di tipo *Hill-Climbing*, focalizzato sulla ricerca del minimo all'interno della funzione di costo. Le valutazioni effettuate su alcune lavorazioni di prova hanno portato alla conclusione che la funzione di costo, ottenuta

dalla somma dei costi spesi in gas ed energia elettrica, risulta essere una funzione ad un solo minimo, rendendo quindi la scelta dell'algoritmo di discesa la migliore anche in termini prestazionali. Tale algoritmo è infatti caratterizzato da una complessità non particolarmente elevata, che lo rende adatto ad un utilizzo in real-time grazie ai *rapidi tempi* di risposta. Nella figura 1 sono mostrati i risultati di una lavorazione:

□ la figura 1 A mostra il variare del costo di gas ed elettricità in corrispondenza delle diverse combinazioni temperature bruciatori/ velocità ventilatori;

□ la figura 1 B mostra il costo totale rispetto alle diverse combinazioni;

□ la figura 1 C mostra, infine, la differenza di costo percentuale tra le varie combinazioni e quella a costo minimo trovata tramite l'algoritmo. È interessante notare come il minimo non si trovi in corrispondenza del minor utilizzo possibile di elettricità e che presenti invece un'opportunità di risparmio del 10% rispetto a quella combinazione.

Il funzionamento dell'algoritmo di *Hill-Climbing* è semplice, ma efficace. Il sistema parte misurando il costo per una metratura predefinita e, successivamente, decrementa la temperatura dei bruciatori di 5° lasciando scorrere alcuni metri di assestamento. Dopodiché inizia a misurare il costo per un'altra metratura predefinita: se il costo è minore, prosegue e diminuisce la temperatura di ulteriori 5°, altrimenti la rialza di 2° e ripete il processo di attesa e misurazione. Se il costo è diminuito prosegue, altrimenti inverte di nuovo la direzione. L'algoritmo termina quando effettua 2 volte il passaggio sopra uno stesso valore. Dai test eseguiti si è rilevato che l'esecuzione termina mediamente in 6 passi, una soluzione che permette forti miglioramenti e risparmi in termini energetici.

5.4. L'inserimento dell'ottimizzatore

Il software di ottimizzazione è stato inserito a monte del PID. La scelta è dovuta al fatto che le due variabili (temperatura dei bruciatori e velocità di ventilazione), benché controllabili separatamente, non sono completamente indipendenti tra loro. La velocità d'entrata del tessuto nella caldaia e la temperatura superficiale che questo deve rispettare per poter essere considerato asciutto, impongono grossi limiti. La soluzione, come già precedentemente spiegato, è stata quella di modificare una sola delle due va-

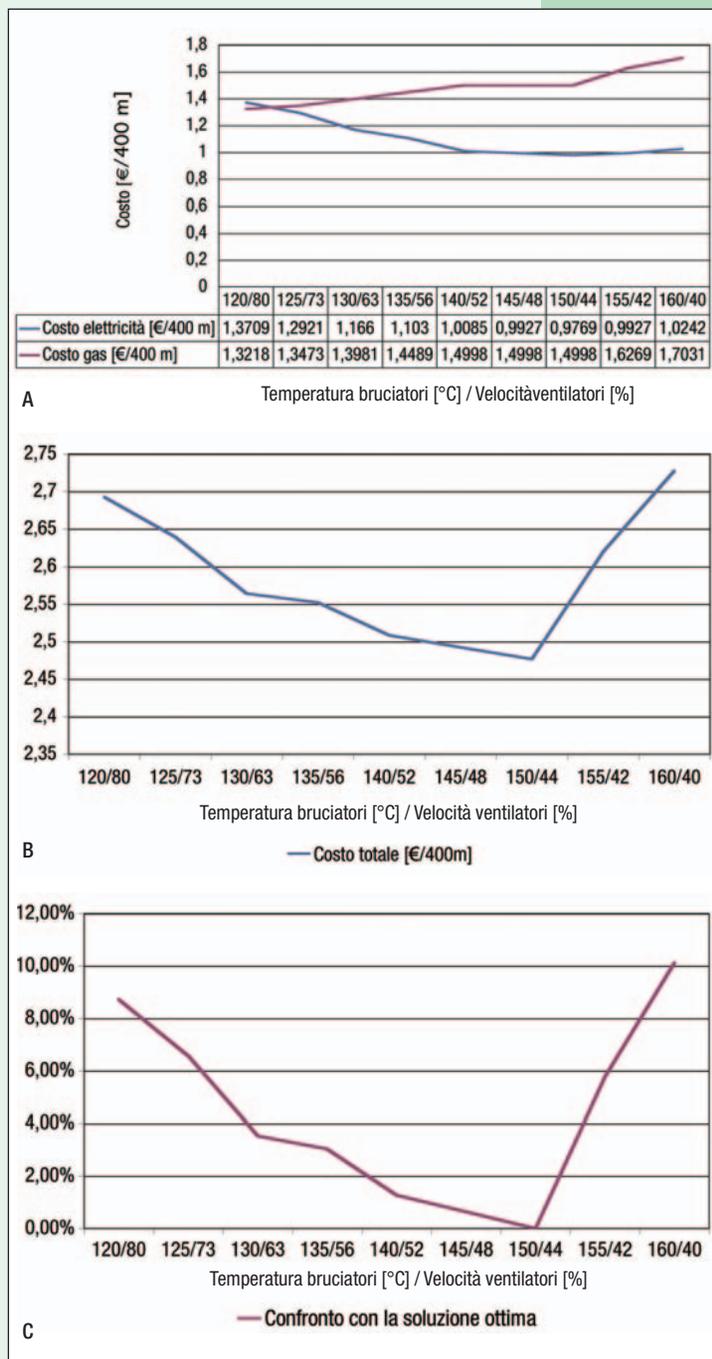


FIGURA 1
La funzione di costo di una lavorazione, A - Costo di gas ed elettricità; B - Costo totale; C - Confronto con la soluzione ottima

riabili e lasciare che il PID controlli l'altra al fine di evitare danni alla lavorazione. A tal proposito il software di ottimizzazione manda direttamente dei segnali agli attuatori delle varie caldaie che, quindi, modificano la temperatura. La figura 2 mostra l'architettura complessiva del sistema di controllo.

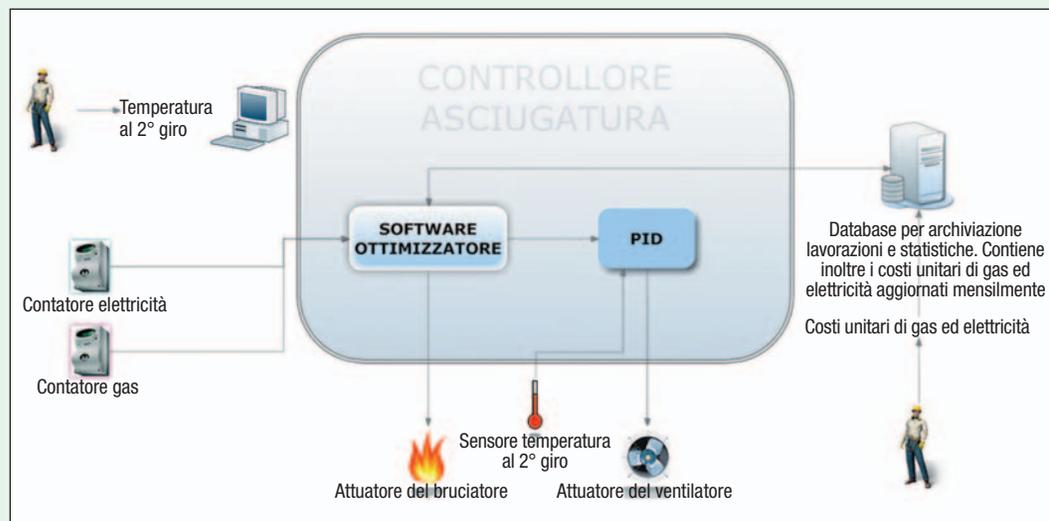


FIGURA 2

Lo schema del sistema di controllo

6. CONCLUSIONI

L'utilizzo di tecnologie IT per ridurre i consumi energetici e l'impatto ambientale dei processi industriali è molto probabilmente una strada che sarà sempre più percorsa in futuro. Il caso della Stamperia Olonia, facilmente replicabile in tante altre realtà simili del settore tessile italiano, dimostra come la progettazione, hardware e software, di un sistema informativo ad hoc orientato all'efficienza energetica sia in grado di portare a risultati significativi.

La Stamperia Olonia già aveva ottenuto risparmi energetici dell'ordine del 15% grazie all'impiego di sistemi a bordo macchina innovativi. L'applicazione di un algoritmo di ottimizzazione ben noto in letteratura e l'implementazione di piccole modifiche al sistema già esistente hanno portato ad un ulteriore risparmio del 10%, quantificabile in circa l'1% del fatturato annuo.

È importante ricordare che la spesa energetica è una voce assolutamente rilevante sul fatturato di molte imprese, come affermato da diversi manager del settore, in quanto rappresenta mediamente tra il 7% e il 10% del fatturato. Nel 2003 i sottosettori di tessitura e finissaggio avevano un fabbisogno energetico annuo di 2,788 TWh [8], che comportava una produzione di circa 2 milioni di tonnellate di CO₂. Pertanto un risparmio del 10% esteso a questi due soli settori comporterebbe una riduzione delle emissioni di CO₂ di circa 209 mila tonnellate annue, pari a circa 1 milione di alberi. Questo ri-

sultato deriva dall'ottimizzazione di un unico specifico processo industriale di due soli sottosettori dell'ampia filiera tessile; è facile e interessante immaginare quante altre applicazioni dell'"IT for a greener business" sarebbe possibile trovare in altri settori.

Ringraziamenti

Un sentito ringraziamento va a Mauro Miele della Stamperia Olonia per la sua disponibilità e per il suo continuo supporto.

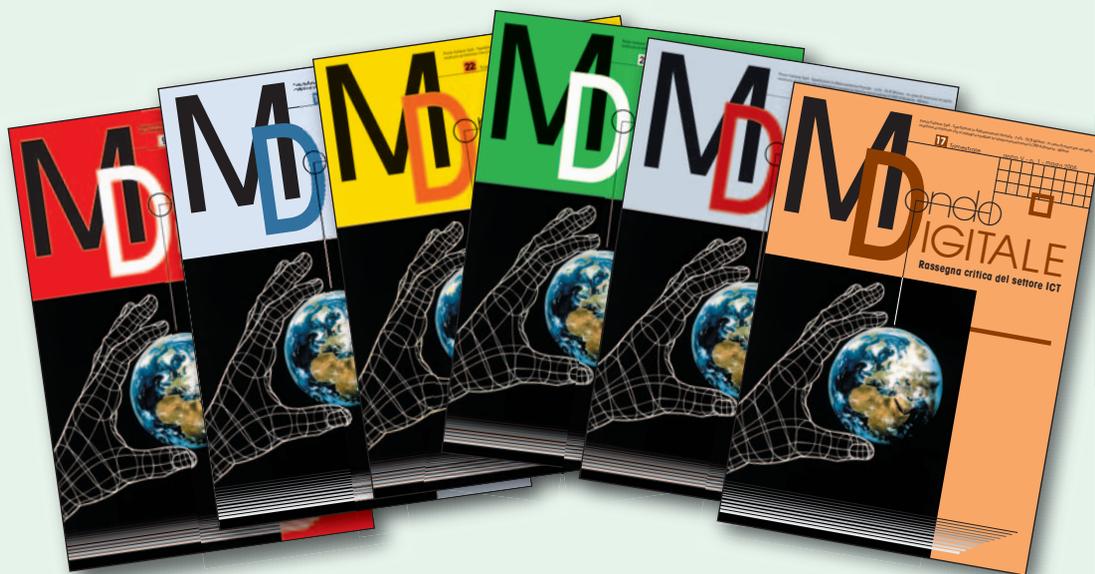
Bibliografia

- [1] CIA: *World factbook website*. 2007.
- [2] European Commission: *Eu spring summit*. Brussels, 2007.
- [3] Stern N.: *Executive summary, Stern Review on the Economics of ClimateChange*. HM Treasury, 2006.
- [4] Gartner: *Green it: The new industry shockwave*. Presentation at Symposium/ITXPO Conference, 2007.
- [5] McKinsey Global Institute: *The carbon productivity challenge: curbing climate change and sustaining economic growth*. June 2008
- [6] McKinsey quarterly: *How IT can cut carbon emissions*. October 2008.
- [7] The Climate Group: *Smart 2020: Enabling the low carbon economy in the information age*. Paper from <http://www.theclimategroup.org/>, 2009.
- [8] ISTAT: *I consumi energetici delle imprese industriali*. Informazioni, n. 5, 2004.

EUGENIO CAPRA è docente a contratto di Sistemi Informativi e ricercatore PostDoc presso il Politecnico di Milano, dove ha conseguito il Dottorato di Ricerca in Ingegneria dell'Informazione nel 2008 e la laurea in Ingegneria Elettronica nel 2003. Come parte dei suoi studi di dottorato è stato Visiting Researcher presso la Carnegie Mellon West University (NASA Ames Research Park, CA) da settembre 2006 a marzo 2007. Ha lavorato come business analyst per McKinsey & Co. dal 2004 fino al 2005, svolge servizi di consulenza in ambito di gestione e innovazione dei processi IT. Le sue attività di ricerca principali riguardano il Green ICT, i modelli manageriali in ambiente open source e l'impatto dell'IT sui processi di business. Su questi temi ha scritto diversi articoli a livello sia nazionale che internazionale.
E-mail: eugenio.capra@polimi.it

DANIELE ZAGORDI ha conseguito nel 2009 la Laurea Specialistica in Ingegneria Informatica presso il Politecnico di Milano, dopo aver ottenuto nel 2006 la laurea in Ingegneria Informatica presso l'Università degli Studi di Napoli Federico II. Attualmente svolge attività di consulenza nell'ambito del Program e Demand Management nel settore finanziario.
E-mail: daniele.zagordi@gmail.com

ALEX ZAZZERA ha conseguito la Laurea Specialistica in Ingegneria Informatica presso il Politecnico di Milano nel 2009 e la Laurea in Ingegneria Informatica sempre presso il Politecnico di Milano nel 2007. Ha svolto attività di analista presso Deloitte Enterprise Risk Services dal 2009 al 2010. Attualmente svolge attività di consulenza informatica nel settore Energy&Utilities per Power Reply. E-mail: zazzeraalex@gmail.com



ICT E DIRITTO

Rubrica a cura di

Antonio Piva, David D'Agostini

Scopo di questa rubrica è di illustrare al lettore, in brevi articoli, le tematiche giuridiche più significative del settore ICT: dalla tutela del *domain name* al *copyright* nella rete, dalle licenze software alla *privacy* nell'era digitale. Ogni numero tratterà un argomento, inquadrandolo nel contesto normativo e focalizzandone gli aspetti di informatica giuridica.



La sicurezza delle informazioni in ambito sanitario

David D'Agostini, Antonio Piva, Attilio Rampazzo

1. INTRODUZIONE

L'Italia sta assistendo a un incremento dell'*Information and Communication Technology* (ICT) nel settore sanitario. Si va verso una nuova era, fondata sull'integrazione dei processi amministrativi, organizzativi e clinici tra le diverse strutture sanitarie e sull'avvio di reti regionali sanitarie, a supporto di modelli organizzativi innovativi, che promuovono la continuità delle cure e la centralità del servizio al cittadino. Si diffondono sempre con maggiore incidenza sistemi di supporto all'ospedalizzazione domiciliare, reti per patologia, portali istituzionali e telemedicina. Soprattutto si va verso un'integrazione complessiva di prodotti e servizi ICT, che porterà ad un impatto sistemico globale sul "Sistema Sanitario".

Le informazioni sanitarie, che sono principalmente dati sensibili, stanno mutando la loro memorizzazione da supporti cartacei a supporti digitali, un passaggio obbligato per le strutture sanitarie sia pubbliche che private.

Si rende necessario parlare di garanzie di sicurezza per gli utenti e anche per chi gestisce questa enorme mole di dati personali.

Risulta, pertanto, auspicabile che le strutture sanitarie applichino la normativa sulla tutela dei dati personali meglio conosciuta come Codice della privacy (d.lgs. 196/2003) e il relativo allegato B "Disciplinare Tecnico in materia di misure minime di sicurezza" dove sono previsti diversi

tipi di prescrizioni per il trattamento dei dati anche in relazione ai dati sensibili¹.

Già nell'ottobre 2008 sono state sollevate questioni di sicurezza sui dati sanitari italiani da una ricerca condotta dall'Osservatorio Nazionale per la Sicurezza Informatica:

(www.osservatoriosicurezzainformatica.org). È emerso, infatti, un elevato rischio per la privacy dei pazienti considerando che il 60% delle ASL coinvolte nell'indagine dichiara di non disporre di strumenti adeguati per la protezione dei dati sensibili; un numero che indubbiamente fa riflettere ed è innegabile che per la tutela dei dati personali vi siano ancora ampi margini di miglioramento. È sempre più diffuso l'utilizzo della rete internet per migliorare il servizio ai cittadini, per esempio prevedendo la possibilità di ricevere i referti o le cartelle cliniche attraverso il personal computer. Un servizio importante, che però espone le aziende sanitarie/ospedaliere a possibili problematiche di security.

Ecco i risultati della ricerca condotta su un campione di 50 ASL, dove emerge che:

- il 100% delle aziende ospedaliere usa internet;
- l'82% delle ASL dichiara di essere a norma con la legge sulla privacy, sia dal punto di vista formale che tecnico;

¹ Per una completa trattazione sulla Privacy si rinvia al numero 2 di giugno 2008, al numero 2 di giugno 2007 e al numero 1 di marzo 2004, all'interno della presente rubrica ICT e Diritto di Mondo Digitale.

0

□ il 60% delle ASL ritiene comunque di non avere strumenti adeguati per la protezione dei dati sensibili;

□ l'85% delle aziende ospedaliere hanno importanti problemi di budget da destinarsi all'IT.

La tecnologia oggi è in continua evoluzione e apre scenari sempre nuovi come la fruibilità del dato, nella sanità online. Il Garante della Privacy ha dimostrato grande attenzione al tema e nel 2009 gli interventi sulla cartella clinica, sul fascicolo sanitario elettronico, sui certificati medici hanno delineato delle linee guida comuni per gli operatori sanitari nella tutela della privacy dei cittadini.

2. I DATI SANITARI NEL CODICE DELLA PRIVACY

I dati personali idonei a rivelare lo stato di salute di un individuo rientrano nel novero dei cosiddetti "dati sensibili" rispetto ai quali, sin dalla legge n.675/96, l'attenzione del legislatore e la tutela approntata sono state maggiori.

All'interno dei dati sensibili si può addirittura sostenere che i dati sanitari siano "super-sensibili" tali sono le garanzie normative e le misure tecnologiche previste nel caso di loro trattamento.

Con l'approvazione del Codice della privacy, al trattamento di dati personali in ambito sanitario è stato dedicato un intero titolo (il titolo V della parte II contenente le disposizioni relative a specifici settori, dall'art. 75 all'art. 94).

Il principio cardine sancito in tali norme è che gli esercenti le professioni sanitarie e gli organismi sanitari pubblici trattano i dati personali idonei a rivelare lo stato di salute:

a. con il consenso dell'interessato e anche senza l'autorizzazione del Garante, se il trattamento riguarda dati e operazioni indispensabili per perseguire una finalità di tutela della salute o dell'incolumità fisica dell'interessato;

b. anche senza il consenso dell'interessato, previa autorizzazione rilasciata del Garante, sentito il Consiglio superiore di sanità, se tale finalità riguarda un terzo o la collettività.

In questi casi il consenso può essere prestato con modalità semplificate, vale a dire anziché con atto scritto dell'interessato, con un'unica dichiarazione, anche orale, che viene annotata dell'esercente la professione sanitaria o dell'organismo sanitario pubblico.

Naturalmente nel caso di emergenze sanitarie

l'informativa e il consenso al trattamento dei dati personali possono intervenire successivamente alla prestazione, così pure nelle ipotesi in cui la prestazione medica possa essere pregiudicata dall'acquisizione preventiva del consenso, in termini di tempestività o efficacia e in particolare nei casi di tutela urgente della salute (bene primario costituzionalmente garantito e tutelato), ossia di:

a. impossibilità fisica, incapacità di agire o incapacità di intendere o di volere dell'interessato, quando non è possibile acquisire il consenso da chi esercita legalmente la potestà, ovvero da un prossimo congiunto, da un familiare, da un convivente o, in loro assenza, dal responsabile della struttura presso cui dimora l'interessato;

b. rischio grave, imminente e irreparabile per la salute o l'incolumità fisica dell'interessato.

Il Codice della privacy dispone, altresì, che in ambito sanitario vengano adottate idonee misure per garantire, nell'organizzazione delle prestazioni e dei servizi, il rispetto dei diritti, delle libertà fondamentali e della dignità degli interessati, nonché del segreto professionale, fermo restando quanto previsto dalle leggi e dai regolamenti in materia di modalità di trattamento dei dati sensibili e di misure minime di sicurezza.

In particolare vengono previste le seguenti misure:

a. soluzioni volte a rispettare, in relazione a prestazioni sanitarie o ad adempimenti amministrativi preceduti da un periodo di attesa all'interno di strutture, un ordine di precedenza e di chiamata degli interessati prescindendo dalla loro individuazione nominativa;

b. l'istituzione di appropriate distanze di cortesia, tenendo conto dell'eventuale uso di apparati vocali o di barriere;

c. soluzioni tali da prevenire, durante colloqui, l'indebita conoscenza da parte di terzi di informazioni idonee a rivelare lo stato di salute;

d. cautele volte ad evitare che le prestazioni sanitarie, ivi compresa l'eventuale documentazione di anamnesi, avvenga in situazioni di promiscuità derivanti dalle modalità o dai locali prescelti;

e. il rispetto della dignità dell'interessato in occasione della prestazione medica e in ogni operazione di trattamento dei dati;

f. la previsione di opportuni accorgimenti volti ad assicurare che, ove necessario, possa essere data correttamente notizia o conferma anche telefonica, ai soli terzi legittimati, di una prestazione di pronto soccorso;

g. la formale previsione, in conformità agli ordinamenti interni delle strutture ospedaliere e territoriali, di adeguate modalità per informare i terzi legittimati in occasione di visite sulla dislocazione degli interessati nell'ambito dei reparti, informandone previamente gli interessati e rispettando eventuali loro contrarie manifestazioni legittime di volontà;

h. la messa in atto di procedure, anche di formazione del personale, dirette a prevenire nei confronti di estranei un'esplicita correlazione tra l'interessato e reparti o strutture, indicativa dell'esistenza di un particolare stato di salute;

i. la sottoposizione degli incaricati che non sono tenuti per legge al segreto professionale a regole di condotta analoghe al segreto professionale.

Si ricorda, inoltre, che per determinati trattamenti di dati idonei a rivelare lo stato di salute effettuati da organismi sanitari, l'art. 34 del Codice della privacy impone l'adozione di tecniche di cifratura o di codici identificativi.

Particolare attenzione viene, poi, dedicata alle cartelle cliniche (documenti che per antonomasia contengono dati sanitari), per la cui redazione e conservazione gli organismi sanitari pubblici e privati devono adottare opportuni accorgimenti per assicurare la comprensibilità dei dati e per distinguere i dati relativi al paziente da quelli eventualmente riguardanti altri interessati.

Eventuali richieste di presa visione o di rilascio di copia della cartella e dell'acclusa scheda di dimissione ospedaliera da parte di soggetti diversi dall'interessato possono essere accolte, in tutto o in parte, solo se la richiesta è giustificata dalla documentata necessità:

a. di far valere o difendere un diritto in sede giudiziaria di rango pari a quello dell'interessato, ovvero consistente in un diritto della personalità o in un altro diritto o libertà fondamentale e inviolabile;

b. di tutelare, in conformità alla disciplina sull'accesso ai documenti amministrativi, una situazione giuridicamente rilevante di rango pari a quella dell'interessato, ovvero consistente in un diritto della personalità o in un altro diritto o libertà fondamentale e inviolabile.

In questo ambito, come previsto dallo stesso Codice, è già intervenuto il Garante per la protezione dei dati personali, con provvedimenti a tutela dei diritti degli interessati.

3. IL GARANTE DELLA PRIVACY E IL FASCICOLO SANITARIO ELETTRONICO

Nel documento di lavoro sul trattamento dei dati personali relativi alla salute contenuti nelle cartelle cliniche elettroniche (CCE)², adottato il 15 febbraio 2007, vengono fornite linee guida sul trattamento dei dati personali relativi alla salute contenuti nelle cartelle cliniche elettroniche; esso fornisce degli orientamenti sull'interpretazione del quadro giuridico in materia di protezione dei dati applicabile alle CCE, e spiega alcuni principi generali; fornisce altresì indicazioni sui requisiti relativi alla protezione dei dati richiesti per la costituzione delle CCE e sulle garanzie necessarie³.

Di conseguenza il Garante della privacy ha pubblicato il 16 luglio 2009 (G.U. n. 178 del 3 agosto 2009) le "Linee guida in tema di Fascicolo sanitario elettronico e di dossier sanitario"⁴.

Questo documento si inquadra all'interno del processo di ammodernamento della sanità pubblica e privata per il quale sono in atto numerose iniziative volte a migliorare l'efficienza del servizio sanitario attraverso un ulteriore sviluppo delle reti e una più ampia gestione informatica e telematica di atti, documenti e procedure. In tale contesto si collocano diverse iniziative volte ad archiviare, mediante nuove tecniche, la svariata documentazione di cui gli organismi sanitari si avvalgono a diverso titolo nei processi di cura dei pazienti come, per esempio, le più recenti esperienze di informatizzazione della cartella clinica e documento/dossier sanitario.

Le citate linee guida del Garante indicano diverse garanzie per l'interessato.

² http://ec.europa.eu/justice_home/fsj/privacy/docs/-wp-docs/2007/wp131_it.pdf

³ Il documento del Gruppo europeo dei Garanti (cd. Gruppo Art. 29), emesso nel 2007, ha sicuramente il pregio di dare delle indicazioni di massima per una disciplina legislativa di diritto interno; si tratta in sostanza di linee guida valide per 27 Stati Membri contenenti perciò i principi generali e le soluzioni conformi, lasciando contempo agli Stati Membri un discreto margine di manovra per adottare una disciplina interna. Il Gruppo di lavoro formula undici raccomandazioni per gli ambiti in cui speciali garanzie risultano particolarmente necessarie per tutelare i diritti alla protezione dei dati dei pazienti e delle persone in genere.

⁴ Il Fascicolo Sanitario Elettronico trae spunto dalla definizione di CCE e viene inteso come un documento formato con riferimento a dati sanitari originati da diversi titolari del trattamento e operanti di solito nella stessa regione o medesimo ambito territoriale.

Il *dossier* sanitario viene inteso come uno strumento costituito presso un organismo sanitario in qualità di unico titolare del trattamento (per esempio, ospedale o clinica privata) all'interno del quale operino più professionisti.

3.1. Diritto alla costituzione di un Fascicolo sanitario elettronico (Fse) o di un dossier sanitario

In base alle disposizioni contenute nel Codice dell'amministrazione digitale - d.lgs. 7 marzo 2005, n. 82⁵ -, deve essere assicurata la disponibilità, la gestione, l'accesso, la trasmissione, la conservazione e la fruibilità dell'informazione in modalità digitale utilizzando le tecnologie dell'informazione e della comunicazione nel rispetto della disciplina rilevante in materia di trattamento dei dati personali. Il diritto alla costituzione o meno del Fse/dossier si deve, quindi, tradurre nella garanzia di decidere liberamente, sulla base del consenso, se acconsentire o meno alla costituzione di un documento che, come si è detto, raccoglie un'ampia storia sanitaria. Il consenso, anche se manifestato unitamente a quello previsto per il trattamento dei dati a fini di cura, deve essere autonomo e specifico.

3.2. Individuazione dei soggetti che possono trattare i dati

Il trattamento di dati personali effettuato attraverso il Fse/dossier, perseguendo esclusivamente fini di prevenzione, diagnosi e cura dell'interessato, deve essere realizzato esclusivamente da parte di soggetti operanti in ambito sanitario, con esclusione di periti, compagnie di assicurazione, datori di lavoro, associazioni o organizzazioni scientifiche e organismi amministrativi anche operanti in ambito sanitario. Inoltre, le persone fisiche legittimate a consultare il Fse/dossier devono essere adeguatamente edotte delle particolari modalità di creazione e utilizzazione di tali strumenti.

3.3. Accesso ai dati personali contenuti nel Fascicolo sanitario elettronico e nel Dossier sanitario

3.3.1. DIRITTI DELL'INTERESSATO SUI PROPRI DATI PERSONALI

Deve essere garantita la possibilità di esercitare in ogni momento i diritti previsti all'art. 7 del Codice della Privacy ed all'interessato devono es-

sere garantite facili modalità di consultazione del proprio Fse/dossier.

3.4. Informativa e consenso

Per consentire all'interessato di esprimere scelte consapevoli, il titolare del trattamento deve fornire previamente un'ideale e chiara informativa, inoltre viene previsto lo specifico consenso.

3.5. Misure di sicurezza

Vengono imposti l'adozione di specifici accorgimenti tecnici per assicurare idonei livelli di sicurezza, ferme restando le misure minime che ciascun titolare del trattamento deve comunque adottare ai sensi del Codice della Privacy.

In particolare nell'utilizzo di sistemi di memorizzazione dei dati devono essere utilizzati idonei accorgimenti per la protezione dei dati registrati rispetto ai rischi di accesso abusivo, furto o smarrimento parziali o integrali dei supporti di memorizzazione o dei sistemi di elaborazione portatili o fissi (per esempio, attraverso l'applicazione anche parziale di tecnologie crittografiche a *file system* o *database*, oppure tramite l'adozione di altre misure di protezione che rendano i dati inintelligibili ai soggetti non legittimati).

Devono essere, inoltre, assicurati:

- idonei sistemi di autenticazione e di autorizzazione per gli incaricati in funzione dei ruoli e delle esigenze di accesso e trattamento (per esempio, in relazione alla possibilità di consultazione, modifica e integrazione dei dati);
- procedure per la verifica periodica della qualità e coerenza delle credenziali di autenticazione e dei profili di autorizzazione assegnati agli incaricati;
- individuazione di criteri per la cifratura o per la separazione dei dati idonei a rivelare lo stato di salute e la vita sessuale dagli altri dati personali;
- tracciabilità degli accessi e delle operazioni effettuate;
- sistemi di *audit log* per il controllo degli accessi al *database* e per il rilevamento di eventuali anomalie.

Infine, per il Fascicolo Sanitario Elettronico, devono essere garantiti protocolli di comunicazione sicuri basati sull'utilizzo di standard crittografici per la comunicazione elettronica dei dati tra i diversi titolari coinvolti.

Per quanto riguarda i Referti on-line, Il Garante della Privacy ha pubblicato un apposito provve-

⁵ Per una completa trattazione sul Codice dell'Amministrazione digitale si rinvia al numero di dicembre 2005, all'interno della presente rubrica ICT di Diritto di Mondo Digitale.

dimento il 19 novembre 2009 denominato “Linee guida in tema di referti on-line”⁶. Le Linee guida in tema di referti on-line fissano rigorose misure a protezione dei dati sanitari dei pazienti che intendono ricevere referti medici e risultati di esami clinici (RX, analisi del sangue ecc.) in modalità telematica (via e-mail, e-mail certificata, o attraverso il collegamento con il sito web della struttura sanitaria).

I punti principali si possono riassumere:

□ per fornire il servizio, le strutture sanitarie pubbliche e private dovranno adottare elevate misure di sicurezza tecnologica come per esempio utilizzo di standard crittografici, sistemi di autenticazione forte, convalida degli indirizzi e-mail con verifica on-line, uso di password per l’apertura del file⁷;

□ il referto può rimanere a disposizione on-line per un massimo di 45 giorni e dovrà essere accompagnato da un giudizio scritto e dalla disponibilità del medico a fornire ulteriori indicazioni su richiesta dell’interessato;

□ l’adesione al servizio è facoltativa e il referto elettronico non sostituirà quello cartaceo che rimarrà comunque disponibile;

□ l’Interessato dovrà dare il suo consenso sulla base di un’informativa chiara e trasparente che spieghi tutte le caratteristiche del servizio di “refertazione on-line”;

□ nel caso le strutture sanitarie offrano la possibilità di archiviare e continuare a consultare via web i referti, dovranno anche sottoporre ai pazienti un’ulteriore specifica informativa e acquisire un autonomo consenso.

A completamento, il Garante per la protezione dei dati personali ha licenziato lo schema di decreto sulle modalità di assorbimento della tessera sanitaria nella carta nazionale dei servizi,

predisposto dal Ministro per la Pubblica Amministrazione e l’Innovazione, ma ha chiesto alcune garanzie per rafforzare la tutela dei dati dei cittadini: in particolare, misure di sicurezza e procedure uniformi per l’attivazione e la gestione della carta⁸.

Il decreto consentirà alle Regioni, mediante il ricorso a un’unica tessera con microprocessore (*Smart Card*) che riunisce le funzioni di tessera sanitaria e carta nazionale dei servizi, di diffondere in maniera omogenea uno strumento sicuro per l’accesso ai servizi in rete. Sino a oggi tale strumento elettronico “multiuso” è stato adottato in forma sperimentale e con modalità diverse solo in alcuni Comuni e Regioni, permettendo ai cittadini, come previsto dal già citato Codice dell’Amministrazione digitale, l’accesso telematico ai differenti servizi di volta in volta offerti dalla pubblica amministrazione (prenotazione di prestazioni specialistiche, pagamenti di ticket sanitari on-line, accesso ai servizi dei centri per l’impiego, visualizzazione dei propri dati fiscali, verifica delle pratiche edilizie ecc.).

4. LA GESTIONE ORGANIZZATIVA DELLA SICUREZZA DELLE INFORMAZIONI

Il problema della sicurezza delle informazioni digitali, anche in ambito sanitario, sembra essere più di ordine organizzativo, che di tipo meramente legale o tecnologico.

Sebbene ogni anno negli Stati Uniti si stimi che siano alcune centinaia di migliaia le vittime di furto di identità, è bene ricordare che nella maggior parte dei casi la minaccia viene dall’interno delle strutture (*insider* e personale scontento). Purtroppo è la componente umana e non tecno-

⁶ Il documento tiene conto di osservazioni e commenti formulati con una consultazione pubblica da organismi e professionisti sanitari pubblici e privati, medici di base, pediatri, organismi rappresentativi, associazioni di pazienti. Come è avvenuto in altri casi (esempio, per il Fascicolo sanitario elettronico) il Garante per la Protezione dei dati personali assolve con le Linee guida un ruolo di supplenza normativa in attesa di una legislazione che disciplini sulla base di regole chiare e uniformi un importante e innovativo processo di ammodernamento tecnologico.

⁷ Protocolli di comunicazione sicuri, basati sull’utilizzo di standard crittografici per la comunicazione elettronica dei dati, con la certificazione digitale dell’identità dei sistemi che erogano il servizio in rete (protocolli https ssl – *Secure Socket Layer*);

- tecniche idonee ad evitare la possibile acquisizione delle informazioni contenute nel file elettronico nel caso di sua memorizzazione intermedia in sistemi di *caching*, locali o centralizzati, a seguito della sua consultazione on-line;
- utilizzo di idonei sistemi di autenticazione dell’interessato attraverso ordinarie credenziali o, preferibilmente, tramite procedure di *strong authentication*;
- possibilità da parte dell’utente di sottrarre alla visibilità in modalità on-line o di cancellare dal sistema di consultazione, in modo complessivo o selettivo, i referti che lo riguardano.

⁸ Fonte: newsletter del Garante n. 336 del 18 marzo 2010.

logica a mostrare “punti deboli”. L’adeguatezza dei sistemi di sicurezza deve ad ogni modo essere valutata e migliorata, tra l’altro gli strumenti per farlo già esistono. Il *security assessment*, la *vulnerability assessment* e la *risk analysis* sono metodologie ormai di dominio pubblico tra i gestori dei sistemi informativi che possono dimostrare la loro validità anche in ambito sanitario. Riguardo alla componente umana sarebbero sicuramente opportune attività di sensibilizzazione e di formazione interne alle strutture sanitarie, poiché gli operatori sanitari che accedono alle informazioni devono essere consapevoli delle norme di sicurezza e del corretto comportamento da tenere. A tale proposito è auspicabile la corretta definizione di policy interne con il duplice obiettivo di comunicare e di regolamentare l’uso delle tecnologie informatiche. Inoltre, sarebbe molto utile formare il personale utilizzando gli standard della certificazione ECDL Health, che assicura la competenza del personale nel gestire i dati sanitari e in generale i processi automatizzati attraverso l’uso di sistemi⁹.

Già nell’articolo “Sicurezza delle Informazioni e norme ISO 27000” pubblicato sul n. 3/2008 di “Mondo Digitale” è stato trattato come attuare

un corretto Sistema di Sicurezza delle Informazioni. Nel caso della Sanità, l’adozione delle norme della famiglia ISO 27000 è una delle più efficaci soluzioni per poter garantire riservatezza, integrità e disponibilità della molteplicità di informazioni trattate. Le stesse guide di CNIPA (ora DigitPA) consigliano l’adozione di queste norme internazionali al fine di attuare una corretta *governance* della Sicurezza delle Informazioni.

La famiglia di Norme ISO/IEC 27000, riferita alla Gestione alla Sicurezza delle Informazioni, comprende ad oggi 7 pubblicazioni (Tabella 1).

Queste norme sono il nocciolo di base per l’implementazione del sistema di gestione della sicurezza delle informazioni in un’organizzazione (sia essa impresa commerciale, organizzazione governativa, non profit o, appunto, un’organizzazione sanitaria privata/pubblica).

In particolare la ISO/IEC 27001 specifica i requisiti per la definizione, l’attuazione, il funzionamento, il monitoraggio, la verifica, il mantenimento e il miglioramento del sistema di gestione della sicurezza delle informazioni documentato nel contesto complessivo dei rischi aziendali. Sono inoltre specificati i requisiti per l’attuazione dei controlli di sicurezza su misura ri-

TABELLA 1
Norme della
famiglia ISO 27000
ad oggi pubblicate

ISO/IEC 27000:2009	Information technology -- Security techniques -- Information security management systems -- Overview and vocabulary
ISO/IEC 27001:2005	Information technology -- Security techniques -- Information security management systems -- Requirements
ISO/IEC 27002:2005	Information technology -- Security techniques -- Code of practice for information security management
ISO/IEC 27003:2010	Information technology -- Security techniques -- Information security management system implementation guidance
ISO/IEC 27004:2009	Information technology -- Security techniques -- Information security management -- Measurement
ISO/IEC 27005:2008	Information technology -- Security techniques -- Information security risk management
ISO/IEC 27006:2007	Information technology -- Security techniques -- Requirements for bodies providing audit and certification of information security management systems

⁹ ECDL Health è una certificazione indirizzata agli utenti dei Sistemi Informativi Sanitari, comprendendo ruolo sanitario, tecnico, professionale e amministrativo e a studenti universitari di Facoltà di Scienze Mediche. La certificazione è motivata dalla crescente importanza che sta assumendo, a livello internazionale, l’informatica in ambito sanitario, e di conseguenza vengono riconosciuti anche i crediti ECM. Basandosi su un approccio centrato sul paziente una componente importante della certificazione Health è dedicata alla sicurezza, alla Privacy, alla riservatezza e alle autorizzazioni di accesso. (<http://aiconet.net/certificazioni/ecdl/specialised-level/health/presentazione>).

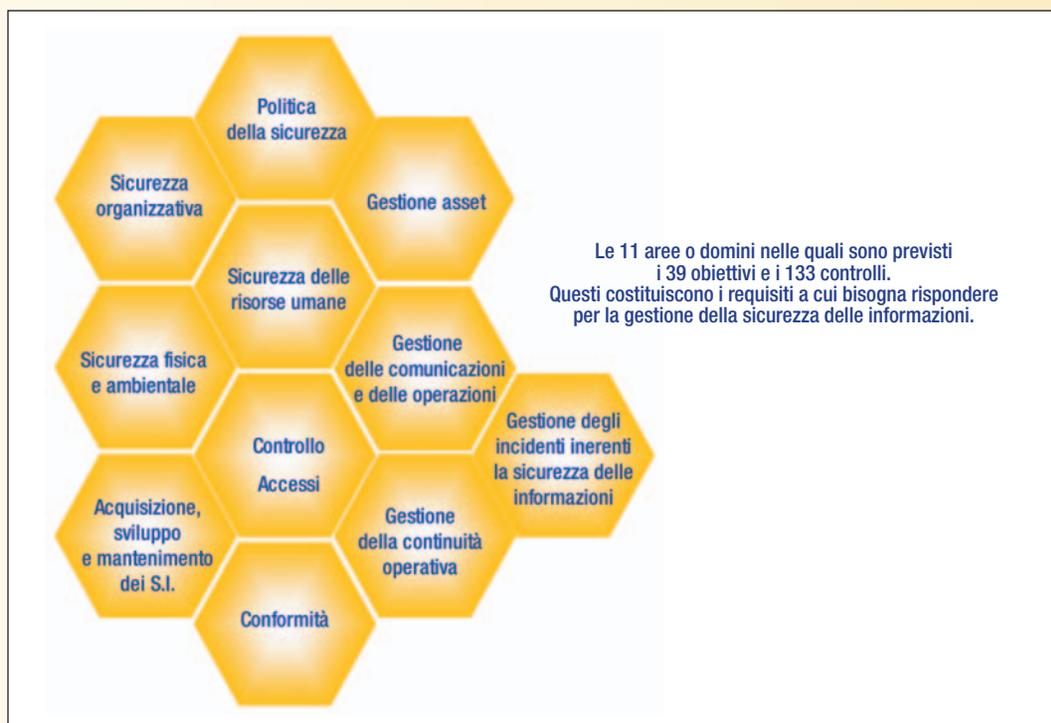


FIGURA 1
Le 11 aree di intervento

spetto alle esigenze delle singole organizzazioni o parti di esse.

L'ISO, seguendo una politica che si va diffondendo tra gli enti di normazione che è quella di emettere standard "general purpose" e poi far seguire documenti di approfondimento per i singoli settori applicativi, nel 2008 ha pubblicato la ISO 27799 (*Health Informatics – Information security management in health using ISO/IEC 27002*) con l'intento di guidare per mano all'applicazione in ambito sanitario della norma ISO/IEC 27002 per la sicurezza delle informazioni.

Come la ISO/IEC 27002:2005, la ISO 27799:2008 riporta quindi le raccomandazioni per la gestione della sicurezza in ambito sanitario, descrivendo undici aree di intervento e relativi "controlli", come indicazioni specifiche per la gestione della sicurezza.

Resta immutato il metodo per affrontare i rischi per la sicurezza delle informazioni che viene stabilito dalla ISO/IEC 27005:2008 prevedendo la netta distinzione tra la fase dell'analisi e quella della gestione. L'analisi può dirsi completa solo se esplora tutte le 11 aree di intervento previste (Figura 1) per l'individuazione delle contromisure.

Il capitolo 5 della Norma ISO 27799 è dedicato a tutti coloro che non hanno familiarità con la sicurezza delle informazioni sanitarie e i suoi obiettivi.

Nel capitolo 6 si può invece acquisire le linee guida su come implementare la Norma ISO/IEC 27001 nel settore sanitario tramite un piano di azione pratico. Nel capitolo non sono contenuti requisiti obbligatori ma, vengono forniti consigli e linee guida generali su come meglio implementare la norma 27002 nel settore sanitario. Il capitolo è organizzato secondo il ciclo di attività (Plan/Do/Check/Act) descritto nella Norma ISO/IEC 27001 che, se seguito, condurrà ad una robusta implementazione del Sistema di Gestione della Sicurezza delle Informazioni (SGSI).

Nel capitolo 7 si possono trovare validi consigli specifici sulle 11 aree di controllo e sui 39 obiettivi di controllo descritti nella Norma ISO/IEC 27002. Il capitolo descrive ciascuna delle 11 aree di controllo della ISO/IEC 27002 dove vengono specificati i requisiti minimi da applicare quando opportuno e, in alcuni casi, vengono fornite le linee guida sulla corretta attuazione di alcuni controlli di sicurezza della ISO/IEC 27002 per proteggere le informazioni sanitarie.

La norma conclude con tre allegati di carattere informativo:

- l'allegato A descrive le minacce generali alle informazioni sanitarie;
- l'allegato B descrive brevemente gli obiettivi e i documenti correlati del sistema di gestione della sicurezza delle informazioni;

- l'allegato C discute sui vantaggi dell'utilizzo delle strumentazioni di supporto come aiuto all'implementazione.

5. CONCLUSIONI

Rispettare le normative vigenti e tutelare le informazioni personali dei cittadini/utenti del servizio oggi significa gestire il settore sanitario curando l'informatizzazione con particolare riguardo alla gestione della sicurezza delle informazioni trattate. Informatizzare un'organizzazione sanitaria e curarne gli aspetti legati alla sicurezza delle informazioni implica il pieno coinvolgimento e la sensibilizzazione di tutto il personale delle strut-

ture sanitarie e in particolare del management. È quindi necessario uno sforzo eccezionale prima di tutto da parte del management per avvicinare tutti i soggetti coinvolti, specialisti, tecnici, medici, stakeholders, amministratori ecc. al concetto di conoscenza, consapevolezza e considerazione dell'importanza da dare al trattamento delle informazioni all'interno delle organizzazioni sanitarie. Non ultimo è da considerare che le organizzazioni che processano informazioni sanitarie, comprese quelle personali, devono avere una politica di gestione delle informazioni scritta, che sia approvata dalla direzione, pubblicata e successivamente comunicata a tutti i dipendenti e terze parti coinvolte.

DAVID D'AGOSTINI, avvocato, master in informatica giuridica e diritto delle nuove tecnologie, docente all'Università degli studi di Udine. Presiede la Commissione informatica dell'Ordine degli avvocati di Udine, è responsabile dell'area "Diritto& informatica" della rivista "Il foro friulano". Presiede l'Organismo di vigilanza di Autovie Venete SpA.

E-mail: studio@avvocatodagostini.it

ANTONIO PIVA, laureato in Scienze dell'Informazione, *Vice Presidente dell'ALSI* (Associazione Nazionale Laureati in Scienze dell'Informazione ed Informatica) e Presidente della commissione di informatica giuridica. Docente a contratto di *diritto dell'ICT e qualità* all'Università di Udine. Consulente sistemi informatici e Governo Elettronico nella PA locale, valutatore di sistemi di qualità ISO9000 ed ispettore AICA.

E-mail: antonio@piva.mobi

ATTILIO RAMPAZZO, consulente di Sistemi Informativi e Sicurezza delle Informazioni in AlmavivA Finance Spa. Ha maturato un'esperienza più che trentennale nello sviluppo e conduzione di progetti informatici in ambito bancario e finanziario, nei quali la qualità e la sicurezza hanno ricoperto un ruolo determinante. È Vice Presidente del Comitato AICQ "Qualità del Software e dei Servizi IT", Valutatore Sistemi di Sicurezza delle Informazioni R.G.V.I. (AICQ_SICEV cert. n.3), certificato CISA, Auditor ISO/IEC 20000, LoCSI e ITIL v.3 foundation. È socio AIEA-Associazione Italiana Information Systems Auditor e AIPSI-Associazione Italiana Professionisti Sicurezza Informatica.

E-mail: attilio@rampazzo.it



PROFESSIONE ICT

Competenze e professionalità per l'innovazione digitale

Rubrica a cura di

Roberto Bellini, Federico Butera, Alfonso Fuggetta

Il tema dell'innovazione e della competitività del sistema Italia è all'ordine del giorno della discussione economica e di quella sulle politiche industriali; sono promosse iniziative istituzionali a supporto dell'innovazione e si auspica un maggiore contributo della ricerca a livello universitario e privato. Anche l'Unione Europea spinge sul tema dell'innovazione, in particolare sul ruolo che le tecnologie ICT possono svolgere sia nei sistemi industriali che nei sistemi di governo e sull'importanza che può avere la definizione di un *framework* comune delle competenze ICT, compatibile con quanto previsto dall'EQF - *European Qualification Framework* - recentemente approvato dall'Unione Europea (2006).

Mondo Digitale vuole sostenere la diffusione di una maggiore sensibilità sul contributo che le competenze e le professionalità relative alle tecnologie digitali possono fornire in termini di innovazione dei servizi e del business dell'Impresa e di servizi per la cittadinanza erogati dagli enti della Pubblica Amministrazione. Questa nuova rubrica è dedicata appunto all'approfondimento sistematico di tutti gli aspetti che riguardano i progetti di analisi e di miglioramento delle competenze per l'innovazione digitale, il monitoraggio dei bisogni di competenza richiesti dal mercato e la valutazione delle offerte di qualificazione e aggiornamento delle competenze proposte dalle istituzioni educative di base e dagli operatori della formazione professionale e permanente.

La rubrica analizzerà l'andamento del mercato del lavoro delle professionalità ICT, i casi di successo nella crescita di competenze del personale dei fornitori di tecnologie e servizi e degli specialisti ICT, sia delle imprese manifatturiere e di servizio che degli enti della Pubblica Amministrazione, nonché l'andamento delle retribuzioni a livello nazionale e internazionale, usando come riferimento i profili e le competenze dello Standard EUCIP che AICA promuove in Italia.

Pianificare la formazione nelle imprese a rete

Roberto Bellini, Pierfranco Ravotto

1. INTRODUZIONE

Su questa rivista e in questa rubrica si è già ripetutamente parlato di EUCIP, lo Standard delle Competenze e dei Profili ICT promosso, a livello europeo, da CEPIS e, in Italia, con lusinghieri risultati, da AICA. Questo ulteriore articolo spiega come il Sistema EUCIP possa essere utilizzato anche per progettare percorsi formativi che permettano di coprire le competenze mancanti, necessarie per migliorare il livello complessivo di professionalità del profilo rispetto a cui il candidato si misura. Si evidenzia infine che il Modello EUCIP aiuta nella progettazione di percorsi di apprendimento continuo per il miglioramento delle competenze nelle im-

prese a rete che comprendono sia reti di aziende, sia imprese con strutture reticolari.

2. IL MODELLO EUCIP

EUCIP è un modello europeo delle competenze e dei profili richiesti dal Mercato del Lavoro nel settore delle Tecnologie Digitali. Il Modello è costituito da quattro componenti fondamentali:

1. il Syllabus strutturato a partire da 3.000 unità elementari di conoscenza – “conoscenze e abilità”, secondo la terminologia proposta da EQF ed e-CF – articolate nelle tre aree: *Plan, Build e Operate*, e raggruppate in macrocategorie e categorie di competenza;

2. il **Sistema dei 21 + 1 fra Profili Elettivi e IT Administrator** che, basati ciascuno su una selezione definita di categorie di competenze, rappresentano quanto richiesto dal mercato del lavoro in termini condivisibili da una molteplicità di *stakeholder*; ciascun profilo è univocamente definito dalle competenze tecniche che ne identificano l'area di professionalità specialistica e comprende comunque una base comune, almeno per tutti i profili elettivi, denominata "core";

3. le metriche e le regole che permettono di:

a. misurare il livello di profondità che le competenze devono avere per ciascun profilo di specializzazione EUCIP (*deep* o *incisive*);

b. sviluppare i percorsi di Certificazione EUCIP Elective che valorizzano sia i titoli di educazione e formazione che l'esperienza di lavoro;

c. integrare come contributo alla Certificazione EUCIP quello che ogni altra certificazione esterna può dare alla Certificazione EUCIP Elective;

4. l'insieme dei **Servizi EUCIP** che supportano la gestione e lo sviluppo professionale delle risorse umane specialistiche.

Il Modello EUCIP rappresenta un importante riferimento per chi voglia progettare e gestire formazione iniziale o continua in ambito informatico, dunque sia per gli Istituti Tecnici e Professionali che per le Università, sia per i Centri di Formazione Professionale che per le aziende [1].

La strumentazione sviluppata nell'ambito del Modello EUCIP può essere utilizzata sia a livello di organizzazione pubblica o impresa che per un singolo specialista interessato - in una logica di *Lifelong Learning* - a costruirsi un proprio percorso formativo.

3. PROFILO DI PROSSIMITÀ E PROFILO OBIETTIVO

Per poter disegnare un percorso formativo abbiamo bisogno di uno strumento che permetta di verificare le competenze attuali (punto di partenza del percorso formativo) di uno specialista e quelle che dovrebbe avere in coerenza con il tipo di specializzazione che vuole raggiungere (punto di arrivo): il servizio di *assessment* delle competenze possedute, basato sul *tool* ECCO-EUCIP, è quello che permette di ottenere il Profilo di Prossimità, come punto di partenza di un percorso formativo, e il Profilo Obiettivo come punto di arrivo che lo specialista (o l'organizza-

zione per cui lavora) vuole ottenere al termine o di un percorso formativo o di una serie di esperienze lavorative.

In entrambi i casi il professionista attraverso l'accesso al servizio di *assessment*, secondo una delle varie modalità previste a seconda che lo faccia a titolo individuale o per conto di un'organizzazione, ottiene come risultato dell'*assessment* stesso un **Rapporto Personale** che evidenzia il Profilo di Prossimità, ovvero quello dei 22 profili cui il candidato più si avvicina. Nel Rapporto Personale viene esplicitato, in sintesi, quanto il candidato ha imparato e sa fare in base agli studi e alle esperienze di lavoro pregresse e viene indicato quali sono le principali aree di competenza ancora da sviluppare, per migliorare il proprio livello di competenza complessivo rispetto allo stesso profilo. Il candidato può anche decidere di raggiungere una più alta specializzazione per un profilo diverso da quello di prossimità per cui cambia Profilo Obiettivo, o sulla spinta del suo orientamento personale oppure perché l'organizzazione per cui lavora glielo chiede: anche in questo caso, dal Rapporto Personale potrà individuare quali sono le competenze da acquisire per raggiungere un buon livello della nuova specializzazione.

La figura 1 mostra un esempio della prima pagina di un Rapporto Personale: questo è costituito da una componente grafica, il "radar", in cui sono sintetizzati i risultati dell'*assessment* e un testo che indica al massimo livello di dettaglio quali sono le competenze da rinforzare e quelle eventualmente in eccesso rispetto sia al Profilo di Prossimità che al Profilo Obiettivo. Nel Report di figura 1 viene rappresentato il Profilo di un candidato che è risultato un ottimo *IT Trainer*, ma che ha deciso di volersi specializzare come *IS Project Manager* e che quindi sceglie questo come Profilo Obiettivo.

Il cerchio interno evidenzia graficamente il livello di possesso delle competenze di base, *core*. Oltre tale cerchio è evidenziato il possesso delle altre competenze necessarie per la configurazione del profilo rappresentato:

- in verde quelle richieste dal profilo;
- in rosso quelle mancanti;
- in azzurro quelle presenti anche se non richieste dal profilo in esame.

(Il rosso sulla circonferenza esterna segnala i settori in cui si manifestano mancanze).

Il Rapporto segnala nel testo quali sono le com-

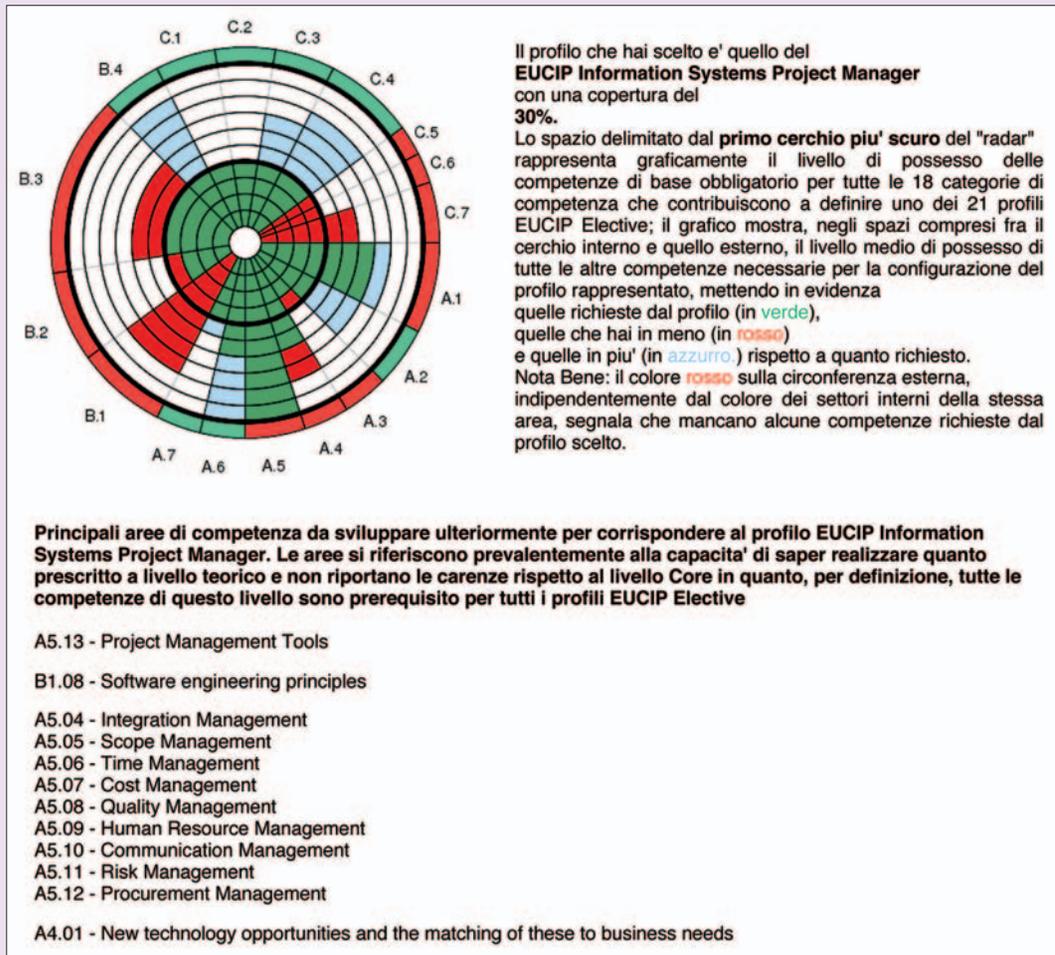


FIGURA 1
Il Rapporto
Personale prodotto
dal Servizio di
Assessment

petenze insufficienti rispetto al Profilo Obiettivo di "IS Project Manager". In particolare risulta carente nelle categorie di competenza da A5.04 ad A5.13.

4. PROGETTAZIONE DI UN PERCORSO FORMATIVO BASATO SULLO STANDARD EUCIP

Come passare dalla rilevazione del gap alla progettazione del percorso formativo? Per rispondere è necessaria un'ulteriore specificazione del Modello EUCIP: nell'ambito dei Servizi EUCIP è anche disponibile un Data Base dell'Offerta Formativa congruente con EUCIP, in cui sono disponibili informazioni strutturate sui "moduli didattici segnalati e gli esami riconosciuti" disponibili sul mercato della formazione IT e pesati nella loro rilevanza rispetto al Syllabus EUCIP.

Un aspetto essenziale del Modello EUCIP è, infatti, quello di riconoscere certificazioni "altre", sia quelle "indipendenti" (ITIL, PMI, ISACA, ISO ecc.)

che quelle dei *vendor* (Microsoft, Cisco, IBM ecc.) [2]; AICA aggiorna e arricchisce sistematicamente questo archivio attraverso la Mappatura, cioè l'analisi e la pesatura delle varie certificazioni sul mercato; analisi e pesatura sono effettuate con uno strumento spinto al livello delle 3.000 unità di conoscenza e secondo una procedura che si basa sul coinvolgimento di chi è esperto della specifica certificazione. Il punteggio assegnato è diverso da profilo a profilo: un certificato da programmatore Java, per esempio, darà punteggio per i profili *Software developer* e *Web e multimedia master*, non per il profilo *Database manager*; tale punteggio è riconosciuto – in sede di esame *elective* – a chi sia in possesso di quella certificazione (si noti che sono accreditati gli esami, non semplicemente i corsi).

Nella figura 2 è riportato l'esempio di Mappatura che ha permesso di costruire il percorso formativo dell'aspirante IS Project Manager (si veda il Rapporto Personale di figura 1): le colonne riportano, per ogni categoria di com-

Moduli Accreditati	Esami															
	Codice	A5											A6			
		A 5.01	A 5.02	A 5.03	A 5.04	A 5.05	A 5.06	A 5.07	A 5.08	A 5.09	A 5.10	A 5.11	A 5.12	A 5.13	A 5.14	A 6.01
COMPTIA																
A+																
CDIA+																
Project+		1	0,5	1	1	1	1	1	1	1	1	1	1			
RFID+																
Security+																
Server+																
IEEE																
CSDA Certified Software Development Associate		0,5	0,5	1												0,5
CSDP Certified Software Development Professional		1	1	1	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5			1
PMI																
CAPM		1	0,5	0,5	1	1	1	1	1	1	1	1	1			
PMP		1	0,5	0,5	1	1	1	1	1	1	1	1	1			0,5
PRMP		1	0,5	0,5	1	1	1	1	1	1	1	1	1			0,5
RMP		0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	1	0,5	
SP		0,5	0,5	0,5	0,5	0,5	1	0,5	0,5	0,5	0,5	0,5	0,5	0,5		
IPMA																
IPMA A		1	0,5	0,5	1	1	1	1	1	1	1	1	1			0,5
IPMA B		0,5	0,5	0,5	1	1	1	1	1	1	1	1	1			0,5
IPMA C		1	0,5	0,5	1	1	1	1	1	1	1	1	1			0,5
IPMA D		1	0,5	0,5	1	1	1	1	1	1	1	1	1			
Adfor/Università Cattolica																
Alta Formazione in Project Management	AFPM	1	0,5	0,5	1	1	1	1	1	1	1	1	1			0,5

FIGURA 2
 Individuazione
 dei corsi/esami
 che coprono
 le categorie
 di interesse

petenza, i moduli di esame accreditati per quella competenza; per il caso in questione si nota come numerose certificazioni coprono le categorie A5.04-A5.12: in particolare le certificazioni COMPTIA Project+, PMI CAPM, PMP e PRMP, le certificazioni IPMA e quella del corso Adfor/Università Cattolica a conclusione del master “Alta formazione in Project management”. Resta scoperta la categoria A5.13 per cui al momento sembra non esista copertura. L'esperto di un Centro di Competenza EUCIP, conoscendo il Profilo Obiettivo di un professionista, il suo Rapporto Personale con la segnalazione dei gap derivante dall'assessment e la mappatura degli esami (e relativi corsi segnalati) disponibili, può agevolmente suggerire un percorso formativo [3].

5. INTERVENTO AZIENDALE DI QUALIFICAZIONE DEL PERSONALE

Vediamo ora un esempio, sviluppato per un caso specifico.

L'azienda è interessata a disegnare dei percorsi formativi personalizzati per un sottoinsieme di propri dipendenti, distribuiti su più sedi di cui una dislocata in un altro Paese (impresa a rete).

Il responsabile del progetto articola il lavoro nelle seguenti fasi:

1. individua, con il supporto della Direzione del Personale, un gruppo di propri *professional* su cui disegnare percorsi formativi di ulteriore specializzazione: i profili professionali di specifico interesse aziendale sono quelli dell'*IT System Architect* e dell'*IS Analyst*;
2. i professionisti sono distribuiti su due sedi in particolare, una in Italia (sede 1) e una in un altro Paese (sede 2); il problema da affrontare è doppio: da una parte rilevare che cosa i *professional* sanno fare effettivamente e dall'altra la confrontabilità di profili denominati in modo simile, anche se hanno preso forma su culture e titoli di studio diversi. In entrambi i casi i *professional* delle due sedi considerate effettuano, in forma assistita, il proprio *assessment* in modo da rilevare il loro portfolio di competenze, indipendentemente dal ruolo ricoperto in azienda e dalla sede in cui lavorano; si tratta di un'autodichiarazione guidata dall'esperto di EUCIP che aiuta a comprendere il Syllabus e le domande proposte dal *tool*;
3. un Consulente EUCIP procede, tramite colloquio con i *professional* che hanno completato la propria valutazione, ad una “validazione” dell'autodichiarazione: discutendo con il professio-

nista della sua esperienza professionale, l'esperto lo aiuta ad individuare dove possa aver sovrastimato o sottostimato le proprie competenze, lo aiuta a modificare eventualmente le valutazioni precedenti, lo sollecita a confrontarsi rispetto al Profilo Obiettivo richiesto dall'azienda;

4. per ogni specialista viene quindi condiviso e accettato sia il Profilo di Prossimità che il Profilo Obiettivo relativamente a *IT System Architect* e *IS Analyst*; il Report Personale di ciascun dipendente indica le competenze carenti rispetto al profilo obiettivo fissato e quelle possedute in eccesso rispetto a quel profilo;

5. al Committente Aziendale viene presentato un Report conclusivo che evidenzia:

- come per la maggior parte degli specialisti sottoposti ad *assessment* il profilo più prossi-

mo risulti quello di *software developer* (67%) contro un 17% di Profili di *IS Analyst* e un 17% di *IT Architect*;

□ la corrispondenza, per ogni specialista, con il profilo di prossimità (sempre superiore al 50%, ma in un terzo dei casi superiore al 90%);

□ il livello di corrispondenza con i due profili obiettivi chiesti dall'azienda, segmentati anche per sede.

Il responsabile del progetto aziendale possiede a questo punto gli elementi per decidere su come disegnare, per ogni singolo specialista, il percorso formativo più adatto che lo porti a coprire i gap di competenza rispetto ai due Profili Obiettivo dell'*IT System Architect* e dell'*IS Analyst*.

Le figure 3 e 4 mostrano, per il Profilo Obiettivo dell'*IT System Architect*, i percorsi formativi per-

FIGURA 3
Disegno dei Percorsi
Formativi per gli IT
System Architect-
sede 2 estera

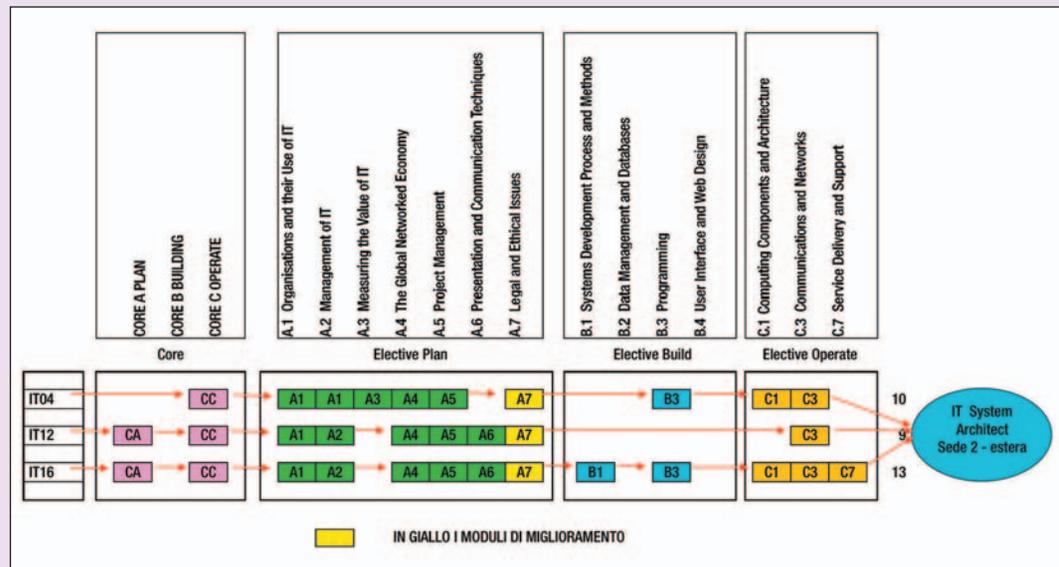
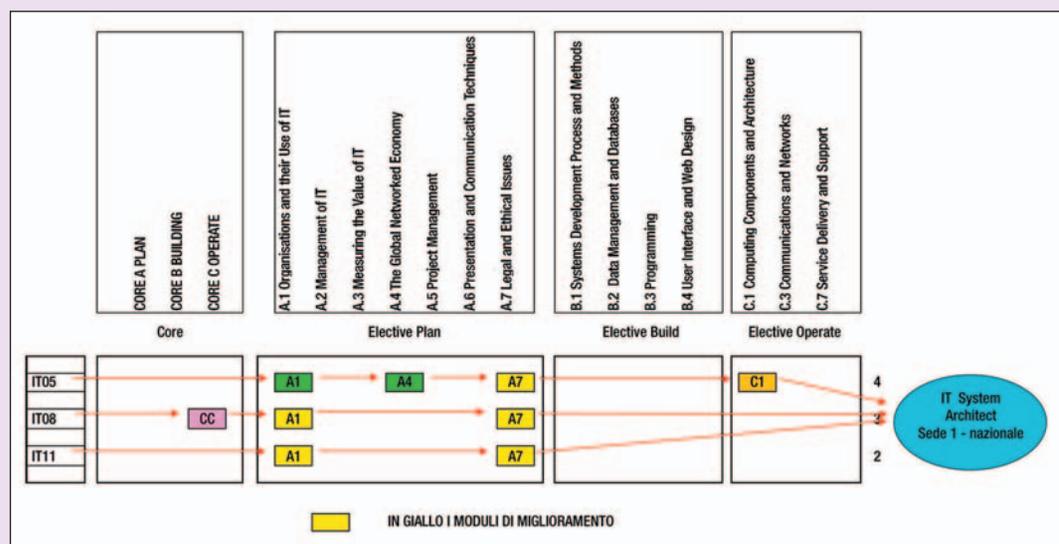


FIGURA 4
Disegno dei Percorsi
Formativi per gli IT
System Architect-
sede 1 nazionale



sonalizzati suggeriti all'azienda per ciascuno specialista, tenendo in considerazione la necessità di ottimizzare l'impegno formativo, i costi di formazione e la sede di lavoro dello specialista; i "blocchi" in giallo si riferiscono a moduli che, per gli specialisti in questione, rappresentano un miglioramento, rispetto a competenze già possedute a livello buono ma che sono richieste per un più alto livello di competenza. La figura 3 si riferisce ai *professional* della sede 2-estera.

I 3 *professional* della sede estera mostrano un **livello di scopertura** piuttosto importante rispetto al profilo obiettivo richiesto: da 9 a 13 moduli didattici per arrivare a coprire le competenze richieste.

Viceversa i 3 *professional* della sede nazionale mostrano un livello di scopertura molto contenuto: nella figura 4 emerge come siano necessari da 3 a nessun modulo didattico per un buon livello di copertura delle competenze per il profilo richiesto. È consigliato qualche modulo didattico di miglioramento per competenze che si possono considerare comunque sufficienti.

Sulla base delle indicazioni formulate, il responsabile del progetto formativo è in grado ora di andare ad individuare la soluzione formativa più conveniente sia in termini di costo che dal punto di vista logistico, attraverso il ricorso a società di formazione specializzata che abbiano in portafoglio un'offerta formativa accreditata in grado di coprire le richieste.

La proposta formativa può naturalmente prevedere soluzioni diverse:

□ corsi tradizionali in aula o, in alternativa in corsi di E-learning/autoistruzione (sia per competenze *CORE* che per competenze *ELECTIVE*), realizzati direttamente dai Centri di Competenza EUCIP;

□ moduli elettivi accreditati di *Vendor* indipendenti, a cui corrispondono corsi accreditati (cioè seguiti da esami la cui mappatura ha fornito i punteggi riconosciuti da EUCIP);

□ Stage di laboratorio, per carenze che richiedano il miglioramento di competenze in parte possedute.

6. CONCLUSIONI

Il Modello EUCIP si sta imponendo all'attenzione delle Imprese, delle organizzazioni pubbliche e dei singoli professionisti per la ricchezza degli strumenti di gestione e del miglioramento della

professionalità, anche in assenza di un'eventuale certificazione. La granularità del Syllabus EUCIP e le metriche adottate nel Modello aiutano, infatti, le imprese a selezionare le risorse per il loro migliore impiego nei progetti, a disegnare e attuare percorsi formativi mirati e con costi controllati, a definire percorsi di carriera professionale coerenti con gli obiettivi strategici delle Tecnologie Digitali applicate per soddisfare i bisogni formativi sia delle imprese dell'offerta che per le organizzazioni della domanda. AICA rispetto alle necessità di formazione che stanno emergendo, è impegnata a favorire, presso tutti i Centri di Competenza accreditati, la costruzione di un portafoglio di offerta formativa secondo il Modello EUCIP che permetta di coprire i bisogni sia di singole imprese che di intere filiere.

Bibliografia

- [1] Bellini R. e Ravotto P.: *Il contributo di EUCIP per una scuola delle competenze*. In: Atti di Didattica 2009, Trento.
- [2] Schgor P., Mockler F., Farre N.: Exploring the EUCIP Certification Range and Progression Options. In: *UPGRADE – The European Journal for the Informatis Professional*, Vol. IX "EUCIP: A Model for Definition and measurement of ICT Skills", 2008.
- [3] Vighi E.: Caso Vittoria Assicurazioni. *Mondo Digitale*, VII-4, dicembre 2008.
Trimarchi P.P.: La formazione informatica per specialisti nella PA: il caso RGS. *Mondo Digitale*, VII-3, settembre 2008.

PIERFRANCO RAVOTTO, laureato in Ingegneria Elettronica al Politecnico di Milano nel 1974, è stato docente di Elettrotecnica, Elettronica, Automazione e Sistemi organizzativi in istituti tecnici occupandosi di corsi post-diploma, alternanza scuola-lavoro, scambi di giovani con alternanza all'estero ed e-learning. Ha coordinato diversi progetti europei nel programma Leonardo da Vinci ed ha partecipato alla progettazione e realizzazione di corsi di aggiornamento per docenti e dirigenti scolastici. Dal 2008 collabora con AICA per il framework EUCIP. È Project Manager del progetto Ensemble per il Dipartimento di Scienze dell'Educazione dell'Università di Firenze. È membro del direttivo nazionale della Sie-L, Società Italiana e-Learning, e del direttivo milanese di AICA. È certificato EUCIP IT Trainer.

E-mail: p.ravotto@aicanet.it

COMPUTER SIMULATIONS ED ESPERIMENTI IN VIRTUO



Viola Schiaffonati

L'impresa scientifica contemporanea è caratterizzata dal diffuso utilizzo di simulazioni condotte al computer. In questo lavoro ci proponiamo di riflettere sulle potenzialità, sui limiti e sui problemi di queste simulazioni usate come esperimenti. Il lavoro analizza, in particolare, le ragioni epistemologiche sottostanti a tale uso e le strategie di validazione dei risultati delle simulazioni. Sono avanzati alcuni spunti di riflessione, a partire da una prospettiva filosofica, sulla natura dell'informatica stessa e la sua legittimità come "infra-science", ossia come scienza al servizio delle altre scienze.

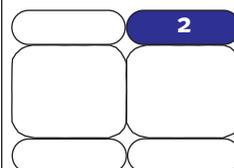
1. INTRODUZIONE

Il modo in cui le sostanze chimiche si distribuiscono attraverso le correnti oceaniche e i sistemi idrici, la resistenza degli edifici ai terremoti, il comportamento degli aerei in condizioni climatiche avverse e la diffusione delle infezioni nel mondo sono solo alcuni esempi di sistemi complessi analizzati mediante l'uso di simulazioni condotte al computer. Si tratta di progetti che hanno valenze scientifiche e anche un forte impatto sui singoli individui e sull'intera popolazione umana. Ma quali ragioni abbiamo per fidarci di queste simulazioni e per giustificare il loro uso in indagini scientifiche che hanno ricadute così importanti sulle nostre vite? In questo lavoro ci proponiamo di riflettere sulle potenzialità, sui limiti e sui problemi delle simulazioni a partire dal riconoscimento che l'impresa scientifica contemporanea è caratterizzata dal diffuso utilizzo di simulazioni condotte al computer per fini sperimentali. Il lavoro analizza, in particolare, le ragioni epistemologiche sottostanti a tale uso e le strategie di validazione dei risultati delle simulazioni stesse. In questo mo-

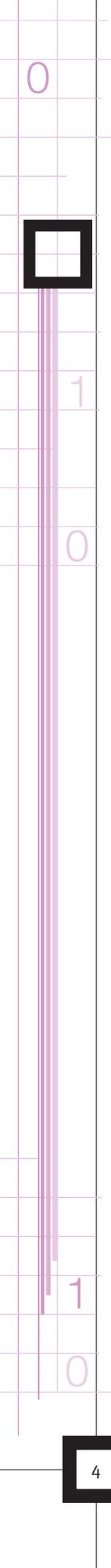
do intendiamo ragionare criticamente su una delle realizzazioni dell'informatica (le simulazioni) e avanzare alcuni spunti di riflessione, a partire da una prospettiva filosofica, sulla natura dell'informatica stessa e la sua legittimità come "infra-science", ossia come scienza al servizio delle altre scienze.

Per riflettere sulla legittimità, sulle potenzialità e sui limiti delle simulazioni impiegate come esperimenti occorre anzitutto riflettere sul concetto di simulazione, partendo dal quale forniremo una definizione di *computer simulation*¹ e una schematica tassonomia delle simulazioni (paragrafo 2). Successivamente presenteremo le ragioni teoriche e le ragioni pratiche per usare le *computer simu-*

1 Utilizziamo la locuzione inglese *computer simulation* per mancanza di un'adeguata traduzione italiana. Infatti la traduzione "simulazione eseguita al computer", come si vedrà nel seguito, non è del tutto esatta. Una *computer simulation* non solo è eseguita da un computer, ma è basata su un modello computazionale che rappresenta il comportamento di un sistema il cui stato cambia nel tempo.



0



lazioni come esperimenti (paragrafo 3), corredando la trattazione con alcuni esempi significativi. Discuteremo poi il problema della validazione dei risultati delle simulazioni, domandandoci se abbiamo ragioni sufficienti per fidarci di tali risultati (paragrafo 4). Infine, nel trarre le nostre conclusioni sull'uso e sulla validità delle simulazioni come strumenti di indagine scientifica, presenteremo alcune riflessioni sulla natura dell'informatica e sulla sua legittimità come scienza al servizio delle altre scienze (paragrafo 5).

2. SIMULAZIONI E COMPUTER SIMULATIONS

Sebbene il termine "simulazione" sia oggi ampiamente utilizzato in diversi ambiti, il suo significato non è sempre chiaro. In questo paragrafo, quindi, ci proponiamo di fare chiarezza su questo concetto per poi introdurre, in accordo con quanto stabilito in generale per le simulazioni, il concetto di *computer simulations*.

2.1. Modelli e simulazioni

Non è affatto semplice trovare una definizione generale di simulazione in grado di includere le molteplici declinazioni di questo termine. Secondo il vocabolario Treccani una simulazione indica "ogni procedimento atto a studiare il comportamento di un sistema in determinate condizioni che si basi sulla riproduzione del sistema o dell'ambiente in cui esso deve operare attraverso modelli". Stephan Hartmann, uno dei primi filosofi della scienza a ragionare sul ruolo delle simulazioni in ambito scientifico (Hartmann 1996), considera una simulazione come ciò che consente agli scienziati di *imitare un processo per mezzo di un altro processo*, dove il termine processo in questo contesto indica la sequenza temporale degli stati di un sistema. In altre parole, una simulazione *riproduce* il comportamento di un sistema usando un altro sistema e fornisce così una rappresentazione *dinamica* di una porzione della realtà. Il termine dinamico sembra qui essere centrale e ci serve per mettere in luce un aspetto importante delle simulazioni: le simulazioni sono basate su modelli dinamici, ossia modelli che contengono assunzioni

sull'evoluzione temporale di un sistema. Modelli e simulazioni condividono la capacità di rappresentare, anche se rappresentano la realtà in modo differente. Cerchiamo di capire meglio questo punto che è un aspetto importante, considerando che un *modello* può essere visto o come rappresentazione di una parte selezionata della realtà oppure come rappresentazione di una teoria (nel senso logico per cui il modello è ciò che è in grado di interpretare le leggi della teoria) (Frigg, Hartmann 2009).

D'ora in avanti ci concentreremo solo sulla prima concezione di rappresentazione, ossia la rappresentazione di una parte della realtà che, solitamente, viene indicata come *sistema target*. Proviamo per esempio a pensare a un modellino in scala di un ponte che deve essere costruito. In questo caso il modellino, nel rappresentare un ponte, replica alcune caratteristiche dell'oggetto reale (il ponte che verrà costruito), ma astrae da tutti i dettagli ritenuti inessenziali per concentrarsi esclusivamente sugli aspetti rilevanti allo scopo che si vuole perseguire. Se il modellino viene costruito per illustrare la forma definitiva del ponte, non sarà importante il materiale di cui è fatto e neppure il suo colore o la sua scala. Ciò che è rappresentato in un modello dipende dagli obiettivi per cui è stato concepito.

2.2. Eseguire un modello

Consideriamo ora il caso in cui il modellino in scala di un ponte sia stato costruito per testare la resistenza di alcuni materiali, che saranno usati nella costruzione effettiva del ponte, agli agenti atmosferici. Il puro e semplice modellino non è sufficiente da solo per testare la resistenza di questi materiali; il modellino deve essere posto in un ambiente fisico controllato per poter essere soggetto all'azione delle condizioni atmosferiche. In questo caso non si tratta di una semplice rappresentazione, ma di un'*esecuzione* del modello nella realtà per mezzo dell'azione svolta dall'ambiente stesso. L'esecuzione di tale modello è ciò che chiamiamo "simulazione". Si noti che, secondo questa prospettiva, il modello è fatto evolvere per imitare l'evoluzione corrispondente che ha luogo nella realtà.

In una simulazione è possibile distinguere due elementi: il *tipo* di modello e l'*agente* che esegue tale modello (Tabella 1). Nel caso di un modello fisico - inteso come rappresentazione fisica di un sistema target - l'agente che esegue tale modello è la "natura" stessa; nel caso di un modello matematico (come per esempio un sistema di equazioni), l'agente può essere sia un essere umano, sia un computer; nel caso di un modello computazionale, l'agente è generalmente un computer. Concentriamoci ora proprio su quest'ultimo caso, quello delle simulazioni basate su modelli computazionali e chiamate *computer simulations*. I modelli computazionali sono meccanismi formali in grado di manipolare stringhe di simboli, ossia di computare funzioni. Un modello computazionale completamente specificato definisce un programma (inteso come una sequenza di operazioni), mentre il processo che risulta dall'esecuzione di un modello computazionale, e che rappresenta il comportamento di un sistema, è una simulazione eseguita da un computer. Non ogni esecuzione di un modello computazionale è una *computer simulation*: affinché si possa parlare di *computer simulation* il modello computazionale deve rappresentare il comportamento di un sistema il cui stato cambia nel tempo. Una *computer simulation* necessita sia della presenza di un modello computazionale che rappresenti un sistema in evoluzione, sia del fatto che tale modello sia eseguito da un computer.

2.3. Controllo e gradi di rappresentazione

La distinzione che abbiamo introdotto fra i tipi di modello sottostanti alle simulazioni, ci consente di evidenziare un altro aspetto interessante per distinguere fra le simulazioni: il loro *grado* di rappresentazione. Si tratta in sostanza di capire che cosa deve essere rappresentato affinché la simulazione possa aver luogo. Se consideriamo la simulazione basata su un modello fisico, è facile osservare che in questa situazione il modello è eseguito dalla realtà fisica che lavora - per così dire - da sé, e che la rappresentazione di questa realtà non è necessaria perché possa esserci una simulazione. In questo caso, piuttosto, l'obiettivo è di isolare nel modello fisico

Modello / Agente	Fisico	Matematico	Computazionale
Ambiente naturale	x		
Essere umano		x	
Computer		x	x

TABELLA 1

I modelli e gli agenti che eseguono tali modelli nelle diverse tipologie di simulazione

sottostante gli aspetti di questa realtà che ci interessano valutare, come è evidente nell'esempio del modellino in scala di un ponte discusso precedentemente. Al contrario, una *computer simulation* che è basata su un modello computazionale (per esempio la simulazione del processo di ripiegamento di una proteina, ossia il cosiddetto ripiegamento molecolare attraverso il quale le proteine ottengono la loro struttura tridimensionale e possono quindi assumere la loro funzione fisiologica) necessita di un grado di rappresentazione maggiore. Non solo è necessario un modello del fenomeno che si sta investigando (il ripiegamento della proteina), ma anche un modello dell'ambiente in cui questo fenomeno ha luogo (per esempio le condizioni alle quali le proteine si ripiegano nelle loro forme biochimiche funzionali).

Nel rappresentare questo ambiente, lo scopo non è quello di isolare alcuni aspetti di interesse (come nel caso della rappresentazione fornita dal modello), ma di rendere questa rappresentazione più precisa possibile per ciò che si vuole ottenere. Si potrebbe, in un certo senso, affermare che il grado di rappresentazione è direttamente proporzionale al grado di controllo. Quando occorre un alto grado di rappresentazione - come nel caso delle *computer simulations* che hanno luogo in una realtà rappresentata - il grado di controllo che possiamo avere è maggiore dato che si tratta di rappresentazioni semplificate che non considerano molti degli elementi presenti invece nella realtà. Quando, al contrario, le simulazioni hanno luogo nella realtà, e non in sue rappresentazioni, il controllo è inferiore dato che la realtà è complessa e ricca di elementi, la maggior parte dei quali non sono neppure conosciuti.

3. ESPLORARE IL MONDO: SIMULAZIONI COME ESPERIMENTI

Come si è visto, le simulazioni possono essere impiegate per diversi scopi. In questo paragrafo discutiamo i casi in cui le simulazioni possono essere usate come esperimenti, presentando l'idea di esperimento esplorativo e fornendo le ragioni epistemologiche che possono giustificare tale uso.

3.1. Simulazioni ed esperimenti

L'uso per scopi sperimentali di *computer simulations*, che ha caratterizzato la pratica scientifica in questi ultimi anni, porta a domandarsi quale relazione possa esistere fra simulazioni ed esperimenti. Un esperimento è un'esperienza controllata, ossia un insieme di osservazioni e azioni, che hanno luogo in un contesto controllato e che servono per indagare una certa ipotesi. Generalmente, per spiegare che cos'è un esperimento, più che dare una definizione, è preferibile fornire una lista non esaustiva di proprietà. Gli esperimenti fanno uso di misure precise per descrivere quantitativamente i fenomeni sotto indagine; devono essere ripetibili in tempi e luoghi differenti per valutare la validità e l'universalità dei risultati; devono essere riproducibili per confermare l'indipendenza dei risultati dallo sperimentatore; devono essere descritti con un linguaggio preciso che dia rigore ai dati sperimentali; i loro risultati devono essere confrontabili. Sebbene oggi le discipline scientifiche sperimentali abbiano raggiunto alti gradi di specializzazione, e sia quindi impossibile offrire una definizione di esperimento che riesca a restituire la specificità di ognuna di esse, queste proprietà generali costituiscono un nucleo comune in grado di dare un'idea delle caratteristiche che un esperimento deve avere in generale.

Inizialmente le simulazioni sono state semplicemente considerate alla stregua di tecniche per condurre esperimenti su computer digitali (Naylor 1966); solo in seguito è stata avanzata l'ipotesi che le *computer simulations* possano essere viste come strumenti intermedi, concettualmente situati fra le teorie e i metodi empirici tradizionali (Rohrlich 1991). Le *computer simulations*, inoltre, sono state considerate come sostituti per esperimenti impossibili da

svolgere nella realtà (Hartmann 1996) e, in generale, come nuovi strumenti sperimentali (Humphreys 2004), arrivando a vederle come tipi speciali di esperimenti (Simpson 2006). Riteniamo che un modo chiaro per affrontare questa relazione sia di ammettere che le *computer simulations* possono essere usate come esperimenti, nel caso in cui l'obiettivo per cui una simulazione viene eseguita coincide con l'obiettivo per cui un esperimento viene eseguito, come ad esempio per scoprire nuove ipotesi esplicative, per confermare o per refutare teorie, per scegliere fra ipotesi rivali. Al contrario, riteniamo che accomunare completamente simulazioni ed esperimenti sia fuorviante, dato che è perfettamente plausibile che ci siano simulazioni che non sono esperimenti, ossia simulazioni che sono condotte senza avere alcuno scopo sperimentale in mente. Si pensi, per esempio, al caso di un programma per la simulazione del processo di ripiegamento delle proteine. Gli scopi per cui una tale simulazione è eseguita possono essere molteplici, comprendendo motivazioni didattiche (per illustrare il funzionamento del processo a degli studenti) fino a includere motivazioni scientifico-sperimentali (per ispirare altri esperimenti, per trovare valori significativi per parametri importanti, per sviluppare nuove ipotesi e modelli).

Secondo l'idea che abbiamo appena avanzato, le *computer simulations* costituiscono, negli ambiti che abbiamo citato, una parte degli esperimenti tradizionali, con gradi di estensione diversi a seconda dei casi. Può, infatti, esservi il caso estremo in cui una simulazione sia così estesa da rappresentare l'intero esperimento in modo che quest'ultimo sia "ridotto" a un certo numero di simulazioni e, simulazione ed esperimento, praticamente coincidano; è il caso degli esperimenti impossibili da fare nella realtà in cui le simulazioni sostituiscono (sperimentalmente parlando) gli esperimenti. Si pensi a questo proposito all'analisi delle conseguenze a lungo termine dell'accrescimento della tassazione di un punto percentuale. In altri casi, più comuni, le simulazioni possono "partecipare" a un esperimento in modi differenti. Possono essere usate come tecniche nei casi in cui è impossibile derivare le soluzioni analitiche di sistemi di equazioni differenziali che descrivono determinati fenomeni.

Se tale uso è piuttosto comune in fisica, dove le teorie sono spesso espresse in termini matematici, non altrettanto comune è in biologia, dove teorie eleganti e compatte come quel-

le della fisica sono rare. Le spiegazioni in biologia sono spesso espresse mediante il linguaggio naturale e può capitare che non siano sempre basate su paradigmi scientifici stabili e definitivi. Ma è proprio per questo motivo che le simulazioni in biologia possono contribuire alla definizione del quadro concettuale e alla costruzione della conoscenza teorica, ricoprendo un ruolo importante nello sviluppo di nuove ipotesi, come nel caso in cui i risultati di una simulazione suggeriscono nuove regolarità che non avrebbero potuto essere estratte dal modello sottostante in alcun altro modo.

3.2. Esperimenti esplorativi

Per i casi in cui i risultati delle simulazioni contribuiscono alla definizione di modelli teorici, introduciamo il termine “esperimenti esplorativi”. Naturalmente ogni esperimento è, per sua natura, esplorativo; in questo caso però adottiamo questo aggettivo proprio per sottolineare il fatto che la correttezza della congettura avanzata non è garantita, anche se l’esperimento può essere di aiuto nel costruirla. In altre parole, gli esperimenti esplorativi sono in grado di raccogliere nuovi elementi da trasformare in conoscenza che deve poi essere ulteriormente verificata. Come vedremo, si tratta di una classe di esperimenti i cui risultati, in un certo senso, offrono meno garanzie di affidabilità ma hanno la fondamentale funzione di aumentare lo spazio concettuale delle esplorazioni e, proprio per questo motivo, richiedono un’attenzione maggiore nei confronti del problema della validazione dei loro risultati (cfr. paragrafo 4).

In questo ambito, è stata proposta la locuzione *esperimento in virtuo* (Desmuelles *et al.* 2005) per indicare i casi in cui le *computer simulations* vengono utilizzate come esperimenti esplorativi. L’idea è quella di sottolineare la possibilità di intervenire virtualmente

La computer simulation costituisce una nuova via per l’indagine dei fenomeni. Rispetto alla sperimentazione tradizionale presenta sostanziali vantaggi di flessibilità, di tempo e di costo. Al limite, può sostituire esperimenti altrimenti impossibili per ragioni economiche o temporali.

sulle procedure di simulazione in modo da esplorare le diverse possibilità dei fenomeni indagati. Il concetto di esperimento *in virtuo* può essere interpretato in due modi, uno più ristretto e uno più ampio. Nel primo caso si tratta della possibilità tecnica (tipica dei sistemi multiagente) di intervenire a *run-time* nel processo di simulazione grazie agli specifici linguaggi di programmazione adottati (Querrec *et al.* 2003); nel secondo caso si tratta, invece, di una caratteristica più generale degli approcci computazionali che non implica l’adozione di nessuna tecnica particolare ed è tipica di quella parte della biologia che, accanto ai metodi sperimentali tradizionali, utilizza la modellazione computazionale (Holcombe 2007). La sperimentazione *in virtuo* promuove una nuova modalità investigativa che consente un’estensione dell’ambito di indagine lungo tre dimensioni: *spaziale* (accesso a parti della realtà altrimenti inaccessibili), *temporale* (accelerazione della scala temporale reale), e *concettuale*. In quest’ultimo caso gli spazi esplorativi possono essere potenziati mediante la manipolazione virtuale della realtà simulata. Si pensi, ad esempio, al caso della modifica virtuale delle proprietà di certi fenomeni biologici per indagare la loro influenza su altri processi più complessi (Sun *et al.* 2008). Vale la pena notare che, in questi casi, le simulazioni non sono più semplicemente dedicate a rendere esplicite le implicazioni di modelli teorici ben formulati dato che, generalmente, non vi sono sistemi di equazioni che rappresentino in maniera definitiva tali processi biologici. Al contrario, le simulazioni sono dirette ad aumentare le possibilità esplorative fornendo agli utenti delle modalità per controllare la reattività e l’adattabilità dei modelli in via di definizione in modo che questi ultimi prendano progressivamente forma sulla base dei risultati delle simulazioni. Tale indagine trova dei punti di contatto significativi con l’idea di “modellazione dall’alto” (*modeling from above*) avanzata qualche anno fa da una nota filosofa della biologia come Evelyn Fox Keller (2003).

3.3. Quali ragioni?

L'uso sempre più esteso delle *computer simulations* come esperimenti non illumina, tuttavia, le ragioni più profonde di tale pratica: quali sono effettivamente le ragioni *perché* le simulazioni possano essere usate come esperimenti? Riteniamo che a questo proposito possano essere identificati due differenti ordini di ragioni (teoriche e pratiche). In primo luogo, esistono ragioni teoriche che sono connesse alla somiglianza esistente fra tecniche di sperimentazione e tecniche di simulazione (analisi di dati, problematiche relative a incertezza ed errore). Sebbene queste ragioni siano importanti, riteniamo tuttavia che non siano sufficienti per giustificare da un punto di vista epistemologico l'uso delle simulazioni come esperimenti. Infatti, oltre a condividere tecniche e strumenti, esperimenti e simulazioni condividono una necessità, che è quella di *controllare* le proprietà oggetto di indagine. Il concetto di controllo è strettamente connesso all'idea che gli esperimenti consistano, fra le altre cose, nella produzione di circostanze controllate, sotto le quali si può assumere che una relazione non dipende da cambiamenti in fattori non controllati. In questo senso il controllo è uno dei termini chiave nella definizione di esperimento come *esperienza controllata*. Non è difficile riconoscere che la stessa idea di controllo si applica bene anche alle simulazioni quando usate per scopi sperimentali; anche in questo caso occorre selezionare e controllare i fattori sperimentali all'interno di un contesto artificiale. Il processo di preparazione di una simulazione, inoltre, è molto simile a quello di un esperimento: ipotesi da testare, circostanze da controllare, parametri da stabilire, tutte attività che, in entrambi i casi, sono compiute da esseri umani che stabiliscono intenzionalmente le pre-condizioni più adeguate. Anche l'esecuzione di una simulazione condivide con l'esecuzione di un esperimento alcuni aspetti. Entrambe richiedono la presenza di un agente che, nel caso di una *computer simulation*, può essere un computer, mentre nel caso di un esperimento è un essere umano. In entrambe le situazioni non è necessario che tale agente sia costantemente coinvolto nel processo di esecuzione: così come in una simulazione l'agente agisce

da sostrato nel quale l'esecuzione può avvenire, così in un esperimento non occorre che lo sperimentatore, pur avendo messo in atto il processo di esecuzione, sia costantemente e attivamente presente affinché l'esecuzione dell'esperimento possa procedere.

Come già anticipato, esistono anche delle ragioni pratiche per usare le simulazioni come esperimenti. In primo luogo, le simulazioni possono essere usate per fare esperimenti numerosi e accelerati, dato che tali esperimenti possono essere ripetuti esattamente, tutte le volte che si desidera, con la garanzia di un grado di precisione non sempre possibile nei casi empirici. In secondo luogo, le simulazioni possono essere usate per fare degli esperimenti che sono difficili da realizzare nella realtà, essendo le simulazioni svincolate dalle molte limitazioni pratiche degli esperimenti reali, come ad esempio la possibilità di cambiare le condizioni al contorno e le condizioni iniziali. In terzo luogo, le simulazioni possono essere usate per fare esperimenti che sono impossibili da eseguire nella realtà, come ad esempio lo studio di porzioni di realtà non accessibili fisicamente (esempio, i processi coinvolti nella formazione delle galassie).

4. BUONE RAGIONI PER FIDARSI DELLE SIMULAZIONI?

Dopo aver discusso l'uso delle *computer simulations* come esperimenti, e le ragioni epistemologiche sottostanti a questo uso, passiamo ora a considerare il problema della validazione dei risultati di queste simulazioni, soprattutto nei casi in cui i modelli su cui le simulazioni si basano non sono fondati in teorie stabili o non possono essere facilmente confrontati con i dati a disposizione.

4.1. Validare i risultati

Quando le simulazioni, e in particolare le *computer simulations*, sono utilizzate come esperimenti, si possono avere due ragioni per fidarsi dei risultati prodotti dalle simulazioni: o i modelli su cui le simulazioni si basano sono radicati in teorie ben fondate indipendentemente (come nei già citati casi in cui i fenomeni in oggetto sono adeguatamente modellati da equazioni), oppure sono presenti dati sperimentali con i quali è possibile confrontare i risultati del-

la simulazione (come nei casi in cui i fenomeni riprodotti artificialmente nelle simulazioni sono comparabili con processi reali). Come abbiamo visto, ciò non è sempre possibile, soprattutto nelle situazioni in cui le simulazioni sono usate per esplorare la realtà senza essere completamente radicate in processi fisici. Come è possibile in questi casi fidarsi dei risultati delle simulazioni? Per rispondere a questa domanda, che è – ricordiamolo – centrale non solo per motivi filosofici ma anche pratici, occorre riflettere ulteriormente sul concetto di simulazione. Una direzione promettente è indicata dalla proposta avanzata in (Winsberg 1999) di considerare le simulazioni basate su gerarchie di modelli. Sarebbe, infatti, semplicistico pensare che i modelli su cui le simulazioni si basano derivino direttamente da modelli teorici (Lenhard 2007), dato che non sempre questi modelli teorici sono chiari, ben fondati e definitivi, e tali quindi da consentire di derivare direttamente i modelli che sono eseguiti per avere le simulazioni. Quando si analizzano i diversi livelli da cui una *computer simulations* è costituita, occorre prendere in considerazione il fatto che una componente decisiva per valutare la “bontà” di una simulazione è il suo basarsi su un buon modello del sistema target che si desidera simulare. Tale modello deve essere trasformato in algoritmi che, alla fine, devono restituirci l’informazione che si desidera sul sistema target (Barberousse *et al.* 2009). Per trasformare i modelli in algoritmi e far sì che gli algoritmi funzionino è necessario gestire una gerarchia di modelli, che vanno dalla conoscenza teorica generale (che in molti casi non è neppure espressa sotto forma di modelli), ai modelli meccanici che servono per applicare la conoscenza teorica idealizzata ai sistemi del mondo reale, ai modelli computazionali che devono rendere trattabili i modelli meccanici, fino ai modelli simulativi che semplificano le assunzioni di fondo, rimuovono gradi di libertà, ottimizzano il codice e così via. Vi sono quindi diversi passaggi critici per muovere dalla conoscenza teorica alla definizione di un modello simulativo, con il rischio che la simulazione perda così il suo “ancoraggio” ai processi fisici reali che dovrebbe riprodurre. Ciò è evidente nei casi in cui i risultati del-

La capacità di simulazione in ogni campo riflette il ruolo trasversale dell'informatica come scienza al servizio delle altre scienze.

le simulazioni non possono essere messi a confronto con dati reali, come per esempio nel caso in cui sia si-

mulata in maniera accelerata l’evoluzione temporale di un organismo.

4.2. Dalla verificabilità all’affidabilità

Supponiamo di aver simulato l’evoluzione di un organismo condensando migliaia di anni e di esperienze in pochi giorni di tempo macchina. Questa simulazione ha prodotto una serie di risultati che ora non possiamo confrontare con nessun dato reale (dovremmo attendere i migliaia di anni che la simulazione ci ha permesso di superare) e che ci appaiono problematici perché risultato di ipotesi e congetture che sono ancora in via di definizione. Come possiamo fidarci di questi risultati?

Innanzitutto riteniamo che il problema della validazione debba essere affrontato sostituendo al più tradizionale e diffuso concetto di verifica quello di *affidabilità*. Ciò significa rinunciare alla verifica intesa come operazione di confronto fra dei risultati e una supposta realtà esterna che può coincidere o meno con tali risultati, in favore di una valutazione sull’affidabilità di questi ultimi che, come vedremo, deve essere affrontata con strategie differenti a seconda dei casi. Si tratta di mettere sullo sfondo il tema della verità, non perché non lo si ritenga importante, ma in quanto il concetto di “affidabilità senza verità” (Winsberg 2006) sembra essere più funzionale a un’operazione di valutazione come quella di cui stiamo parlando. Ciò significa non tanto indebolire lo scenario di riflessione, quanto piuttosto modificare radicalmente la prospettiva in cui ragionare: superare l’idea che alla domanda sulla validità dei risultati possa essere data una risposta univoca e accettare che tale risposta implichi necessariamente dei gradi.

Per stabilire l’affidabilità dei risultati delle simulazioni è possibile considerare diverse strategie. Queste, naturalmente, dipendono dai singoli casi, ma possono essere suddivise all’incirca in tre gruppi. Nel primo gruppo si trovano tutti i casi per cui non vi è bisogno di alcuna strategia particolare perché la validazione avviene rispetto a un insieme ben radicato di conoscenze precedenti o perché il ruolo speri-

mentale delle simulazioni è minimo e si rivela unicamente un completamento all'analisi sperimentale tradizionale (Fisher, Bodenstern 2006). Nel secondo gruppo troviamo i casi che potremmo chiamare ibridi e che richiedono una *strategia minimale*: si tratta dei casi in cui la simulazione è esplorativa - che nei casi di *computer simulations* è classificabile come esperimento *in virtuo* (cfr. paragrafo 3) - ma la validazione avviene *in vitro* o *in vivo*, ossia secondo le metodologie sperimentali più consuete e stabili. Le simulazioni sono impiegate da un punto di vista teorico per esplorare nuove ipotesi conoscitive, funzionando come delle "teorie di prova" (Fox Keller 2003), i cui risultati devono essere poi validati, laddove è possibile, secondo metodi tradizionali. Nel terzo gruppo troviamo infine tutti i casi in cui o i dati empirici sono incompleti e difficili da recuperare oppure lo scenario teorico è ancora congetturale ed è quindi richiesto un *insieme di strategie* che possano fornire una ragionevole credenza nei risultati delle simulazioni. Si tratta di strategie *locali*, che devono essere definite e valutate caso per caso, ma che sono ispirate ad alcuni principi regolativi sottostanti di carattere generale.

In quest'ultimo caso, la credibilità dei risultati delle simulazioni, e quindi della loro affidabilità, proviene da diverse fonti che, singolarmente o combinate, forniscono buone ragioni per fidarsi dei risultati prodotti. In primo luogo il fatto che le tecniche adottate per la realizzazione del modello sottostante la simulazione siano state già impiegate con successo, nonché il fatto che i risultati di una simulazione siano in accordo con dati, osservazioni e intuizioni precedentemente condivise e accettate. Inoltre, la capacità che tali risultati hanno di fare delle previsioni di successo e, infine, la loro capacità di produrre risultati pratici significativi. Questo insieme di strategie non ci obbliga a prendere alcuna posizione relativamente alla "verità" dei risultati: piuttosto che di inferenza dal "successo alla verità" (Kitcher 2001) è più opportuno parlare di un'inferenza dal "successo all'affidabilità". Non si tratta infatti di una singola strategia ma di un insieme di strategie che - come il filosofo della scienza Ian Hacking (1983) ha fatto notare a proposito degli esperimenti in generale - sono fallibili anche se offrono buone ragioni perché possiamo fidarci

delle procedure di simulazione e dei loro risultati. In sostanza il fallimento di alcuni casi non implica il rifiuto dell'impiego delle simulazioni come strumenti di indagine. Si pensi alle previsioni meteorologiche ottenute in simulazione e non sempre attendibili o alle simulazioni basate su modelli matematici per lo studio di fenomeni finanziari non sempre in grado di fare previsioni efficaci, come la recente crisi finanziaria ha dimostrato.

Ragionare in una prospettiva fallibilista non significa rinunciare a domandarsi dell'efficacia di tali procedure, ma riconoscere che non ci può essere alcuna garanzia definitiva relativamente alla correttezza di questi risultati; anche quando tali strategie siano correttamente applicate, i risultati delle simulazioni potrebbero in seguito rivelarsi scorretti. Sarebbe semplicistico pensare che nella scienza si possano sempre avere risposte chiare e distinte, e soprattutto definitive. Pensiamo alle simulazioni e al problema della validazione dei loro risultati: in alcuni casi possono esserci dati empirici, ma non essere completi; in altri casi gli scenari teorici possono essere in via di definizione e i risultati delle simulazioni possono contribuire a dargli una forma più definitiva. Alla domanda che dà il titolo a questo paragrafo "Abbiamo delle buone ragioni per fidarci delle simulazioni?" possiamo rispondere "Dipende", intendendo con ciò affermare che risposte generali e universalmente valide non possono essere date e che, invece, è necessario valutare caso per caso, sulla base delle strategie che abbiamo delineato, le risposte più opportune a questa domanda.

5. DISCUSSIONE

La trattazione che abbiamo proposto è ben lungi dall'aver esaurito l'argomento dell'uso delle simulazioni come esperimenti, così come la discussione sulle strategie più consone alla validazione dei loro risultati. Riteniamo, tuttavia, di aver fatto chiarezza su alcuni termini del dibattito, lasciando il lettore non con una soluzione definitiva, ma con una serie di spunti per la riflessione. Per concludere la nostra analisi proponiamo ora qualche considerazione sull'informatica come scienza al servizio delle altre scienze.

Da un punto di vista filosofico, l'informatica è diventata un oggetto di indagine solo negli ultimi anni, quando si è cominciato a pensare sistematicamente al suo status disciplinare (scienza/ingegneria) nel contesto di un nuovo ambito di ricerca denominato *filosofia dell'informatica*. All'interno del dibattito sulla scientificità dell'informatica, un punto focale sembra oggi essere rappresentato dall'analisi del ruolo degli esperimenti in questa disciplina, dato che è proprio l'aggettivo sperimentale a definire in molti casi l'attività scientifica stessa. Il dibattito sul ruolo degli esperimenti nell'informatica può essere inteso secondo due prospettive differenti. La prima è una prospettiva metodologica che ha a che fare con il modo in cui la pratica sperimentale è condotta nell'informatica. Qui possiamo osservare diversi modi in cui il concetto di esperimento è inteso e applicato: dall'ambito degli algoritmi in cui il termine esperimento si ispira al modo in cui esso è concepito nelle scienze naturali tradizionali (Johnson 2002), fino a quello della robotica mobile autonoma che solo recentemente ha riconosciuto la necessità di un dibattito sistematico su tali temi allo scopo di muoversi verso una metodologia sperimentale più rigorosa (Amigoni *et al.* 2009). La seconda prospettiva è epistemologica e concerne il modo in cui strumenti prodotti in seno all'informatica possono essere legittimamente impiegati per fare esperimenti in altri ambiti scientifici (per esempio in fisica, chimica, biologia).

Sebbene sia la prima delle due prospettive a coinvolgere in maniera più diretta il dibattito sulla natura dell'informatica, ci sembra tuttavia che, anche a partire dalla seconda delle due prospettive che è quella che abbiamo affrontato in questo lavoro, possano emergere alcune riflessioni interessanti che illuminano un aspetto dell'informatica oggi molto diffuso ma poco teorizzato: il suo ruolo di scienza al servizio delle altre scienze. Concepire strumenti, come ad esempio le *computer simulations*, adatti a condurre esperimenti in altre discipline scientifiche non è infatti solo un modo per ampliare costantemente gli ambiti di applicazione dell'informatica, ma costringe a riflettere criticamente sui suoi metodi, i suoi usi e le sue potenzialità trasformando problemi pratici in riflessioni teoriche che portano ad ampliare e rinnovare la visione

tradizionale dell'informatica, nonché a riconoscere i suoi intimi legami, da un lato, con diverse discipline scientifiche e, dall'altro, con la riflessione filosofica.

Ringraziamenti

Ringrazio Francesco Amigoni e Davide Mazza per la discussione di molte delle idee contenute in questo lavoro. Desidero, inoltre, ringraziare Chiara Francalanci che, con grande apertura mentale e curiosità, mi ha invitato a scriverlo, nonché l'anonimo revisore per i preziosi consigli che mi ha fornito.

Bibliografia

- [1] Amigoni F., Reggiani M., Schiaffonati V.: An insightful comparison between experiments in mobile robotics and in science. *Autonomous Robots*, Vol. 27, 2009, p. 313-325.
- [2] Barberousse A., Franceschelli S., Imbert C.: Computer simulations as experiments. *Synthese*, Vol. 169, 2009, p. 557-574.
- [3] Desmeulles G. et al.: The virtual reality applied to biology understanding: The in vitro experimentation. *Expert Systems with Applications*, Vol. 30, n. 1, 2005, p. 82-92.
- [4] Fisher J., Bodenstern L.: Computer simulation analysis of normal and abnormal development of the mammalian diaphragm. *Theoretical Biology and Medical Modelling*, 3, 9, 2006.
- [5] Fox Keller E.: Models, Simulation, and 'Computer Experiments'. In: Radder H. (a cura di) *The Philosophy of Scientific Experimentation*. Pittsburgh University Press, 2003, p. 198-215.
- [6] Frigg R., Hartmann S.: *Models in science*. In: Zalta, E. N. (a cura di) *The Stanford Encyclopedia of Philosophy*, edizione estate 2009.
- [7] Hacking I.: *Representing and Intervening*. Cambridge University Press, 1983.
- [8] Hartmann S.: *The world as a process: simulations in the natural and social sciences*. In: Hegselmann, R. et al. (a cura di) *Simulation and Modelling in the Social Sciences from the Philosophy of Science point of view*. Theory and Decision Library, Kluwer, 1996, p. 77-100.
- [9] Holcombe M.: Delving beneath the skin. *Ercim News*, Vol. 67, 2007, p. 48-49.
- [10] Humphreys P.: *Extending Ourselves. Computational Science, Empiricism, and Scientific Method*. Oxford University Press, 2004.
- [11] Kitcher P.: Real Realism: The Galileian Strategy. *Philosophical Review*, Vol. 110, 2001, p. 151-197.

- 0
-
- [12] Johnson D.: *A theoretician's guide to experimental analysis of algorithms*. In: Goldwasser M.H., Johnson D., McGeoch C. (a cura di): *Data Structures, Near Neighbor Searches, and Methodology: Fifth and Sixth DIMACS Implementation Challenges*. American Mathematical Society, 2002, p. 215-250.
- [13] Lenhard J.: Computer simulation: the cooperation between experimenting and modeling. *Philosophy of Science*, Vol. 74, 2007, p. 176-194.
- [14] Naylor T. H.: *Computer Simulation Techniques*. John Wiley, 1966.
- [15] Querrec G., Rodin V., Abgrall J.F., Kerdelo S., Tisseau J.: *Uses of multi-agents systems for simulation of mapk pathway*. Proceedings of the Third IEEE Symposium on Bioinformatics and Bioengineering (BIBE03), 2003, p. 421-425.
- [16] Rohrllich F.: *Computer simulation in the physical sciences*. In: Fine A., Forbes M., Wessels L. (a cura di), Proceedings of the 1990 Biennial Meeting of the Philosophy of Science Association, 1991, p. 145-163.
- [17] Simpson J.: *Simulations are not models*. Presentato a Models and Simulations Conference 1, 2006.
- [18] Sun T., et al.: Agent based modeling helps in understanding the rules by which fibroblasts support keratinocyte colony formation. *PLoS ONE*, Vol. 3, n. 5, 2008.
- [19] Winsberg E.: Sanctioning models: the epistemology of simulation. *Science in Context*, Vol. 12, 1999, p. 275-292.
- [20] Winsberg E.: Models of success vs. success of models: Reliability without truth. *Synthese*, Vol. 152, n. 1, 2006, p. 1-19.

Questo articolo costituisce una introduzione epistemologica alla computer simulation.

Il tema verrà trattato in dettaglio, nei suoi aspetti tecnici e applicativi, nei prossimi numeri della rivista.

VIOLA SCHIAFFONATI si è laureata in Filosofia presso l'Università degli Studi di Milano e ha conseguito il dottorato di ricerca in Filosofia della Scienza presso l'Università di Genova. Visiting scholar presso il Department of Philosophy della University of California at Berkeley durante l'anno accademico 2000/2001 e visiting researcher presso il Suppes Center for the Interdisciplinary Study of Science and Technology della Stanford University nel 2005. Attualmente è assegnista di ricerca presso il Dipartimento di Elettronica e Informazione del Politecnico di Milano e docente a contratto della Facoltà di Ingegneria dei Sistemi e della Facoltà di Ingegneria dell'Informazione del Politecnico di Milano. Dal 1999 è membro del laboratorio di Intelligenza Artificiale e Robotica del Politecnico di Milano. I suoi interessi di ricerca riguardano i fondamenti dell'intelligenza artificiale e la filosofia dell'informatica, con particolare attenzione alle problematiche filosofiche della computational science e all'epistemologia dell'esperimento.
E-mail: schiaffo@elet.polimi.it