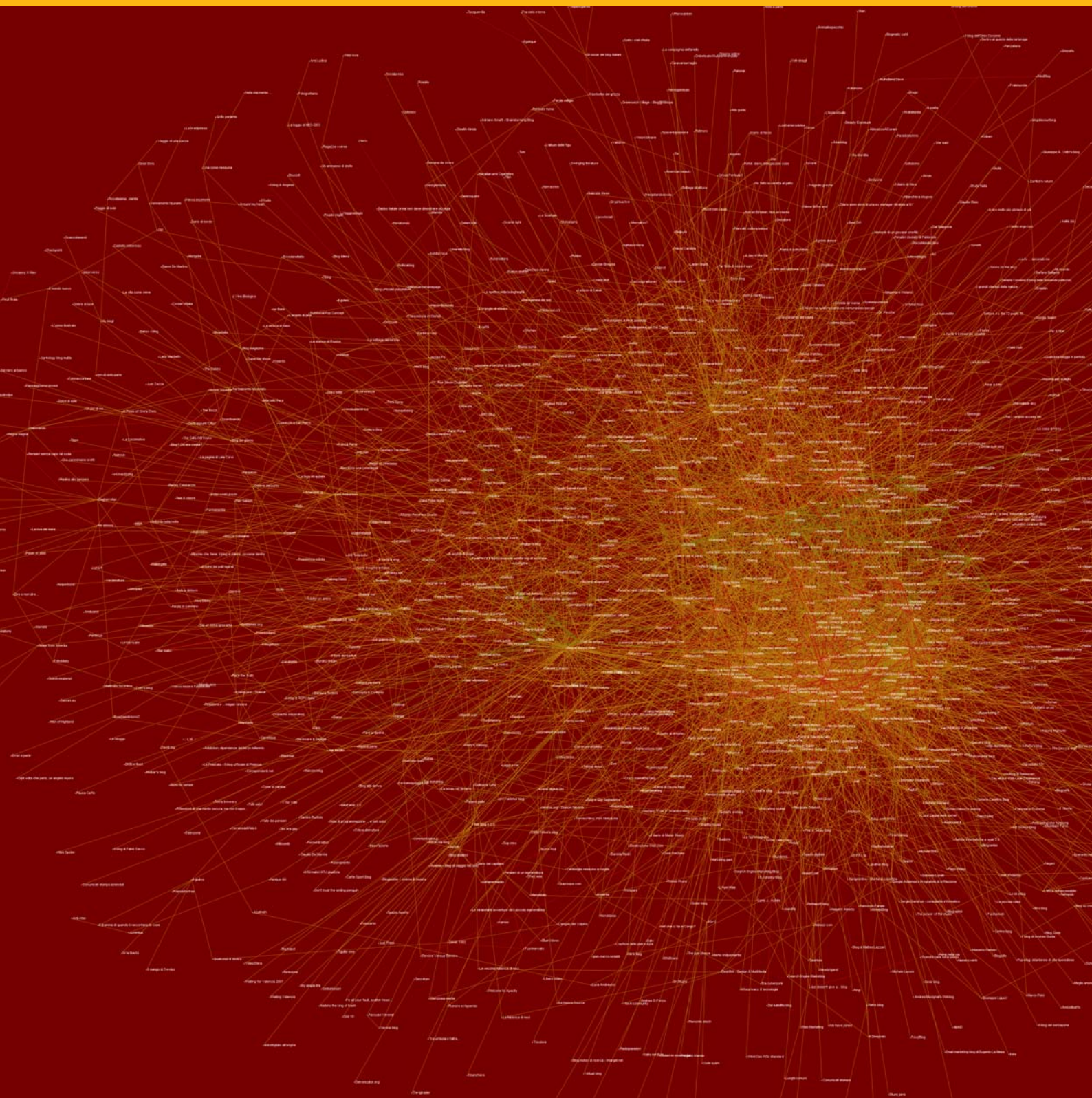


DAL WEB 2.0 AI MEDIA SOCIALI

Tracce e percorsi della partecipazione in rete

A cura di CSP s.ca.rl. - Lara Marcellin, Fabiana Vernero, Ferdinando Ricchiuti, Enrico Sola, Andrea Toso, Giovanni Calia



DAL WEB 2.0 AI MEDIA SOCIALI

Tracce e percorsi della partecipazione in rete

A cura di CSP s.ca.rl. - Lara Marcellin, Fabiana Venero, Ferdinando Ricchiuti, Enrico Sola, Andrea Toso, Giovanni Calia

© 2007 - CSP Innovazione nelle ICT S.c.a.r.l., via Livorno n. 60, 10144 TORINO - Tutti i diritti riservati.

La presente pubblicazione è distribuita da CSP con la licenza Creative Commons "Attribuzione - Non commerciale - Condividi allo stesso modo - 2.5 Italia", reperibile presso il seguente sito Internet: <http://creativecommons.org/licenses/by-nc-sa/2.5/it/>.

Qui di seguito si riporta un sintetico riassunto della licenza ("Commons Deed") e delle facoltà concesse da CSP attraverso essa. Il Commons Deed ha valore puramente informativo, non ha valore legale e il suo testo non compare nella licenza vera e propria, alla quale si rinvia. Qualunque altro diritto o attività non espressamente concessi in licenza sono da ritenersi riservati ai sensi della legge sul diritto d'autore.



Attribuzione - Non commerciale - Condividi allo stesso modo 2.5 Italia

Tu sei libero:

- di riprodurre, distribuire, comunicare al pubblico, esporre in pubblico, rappresentare, eseguire e recitare quest'opera;
- di modificare quest'opera.

Alle seguenti condizioni:



Attribuzione. Devi attribuire la paternità dell'opera nei modi indicati dall'autore o da chi ti ha dato l'opera in licenza.



Non commerciale. Non puoi usare quest'opera per fini commerciali.



Condividi allo stesso modo. Se alteri o trasformi quest'opera, o se la usi per crearne un'altra, puoi distribuire l'opera risultante solo con una licenza identica a questa.

- Ogni volta che usi o distribuischi quest'opera, devi farlo secondo i termini di questa licenza, che va comunicata con chiarezza.
- In ogni caso, puoi concordare col titolare dei diritti d'autore utilizzi di quest'opera non consentiti da questa licenza.

Le utilizzazioni consentite dalla legge sul diritto d'autore e gli altri diritti non sono in alcun modo limitati da quanto sopra.

CSP INNOVAZIONE NELLE ICT S.C.A.R.L. OFFRE LA PUBBLICAZIONE IN LICENZA "COSÌ COM'È" E NON FORNISCE ALCUNA DICHIARAZIONE O GARANZIA DI QUALSIASI TIPO CON RIGUARDO ALL'OPERA, SIA ESSA ESPRESSA OD IMPLICITA, DI FONTE LEGALE O DI ALTRO TIPO, ESSENDO QUINDI ESCLUSE, FRA LE ALTRE, LE GARANZIE RELATIVE AL TITOLO, ALLA COMMERCIALITÀ, ALL'IDONEITÀ PER UN FINE SPECIFICO E ALLA NON VIOLAZIONE DI DIRITTI DI TERZI O ALLA MANCANZA DI DIFETTI LATENTI O DI ALTRO TIPO, ALL'ESATTEZZA OD ALLA PRESENZA DI ERRORI, SIANO ESSI ACCERTABILI O MENO. ALCUNE GIURISDIZIONI NON CONSENTONO L'ESCLUSIONE DI GARANZIE IMPLICITE E QUINDI TALE ESCLUSIONE POTREBBE NON ESSERE APPLICABILE IN ALCUNI CASI.

SALVI I LIMITI STABILITI DALLA LEGGE APPLICABILE, IL LICENZIANTE NON SARÀ IN ALCUN CASO RESPONSABILE NEI CONFRONTI DEL LICENZIATARIO A QUALUNQUE TITOLO PER ALCUN TIPO DI DANNO, SIA ESSO SPECIALE, INCIDENTALE, CONSEGUENZIALE, PUNITIVO OD ESEMPLARE, DERIVANTE DALLA PRESENTE LICENZA O DALL'USO DELL'OPERA, ANCHE NEL CASO IN CUI IL LICENZIANTE SIA STATO EDOTTO SULLA POSSIBILITÀ DI TALI DANNI. NESSUNA CLAUSOLA DI QUESTA LICENZA ESCLUDE O LIMITA LA RESPONSABILITÀ NEL CASO IN CUI QUESTA DIPENDA DA DOLO O COLPA GRAVE.

Immagine di copertina: Ludovico Magnocavallo - BlogBabel - <http://qix.it/archive/20061214/ecco-le-immagini>.
Tutte le immagini ove non sia specificato altrimenti, sono tratte da Wikipedia.

INDICE

INTRODUZIONE	6	CONCLUSIONI	49
1.1 INTRODUZIONE	6	ALLEGATO 1 - APPLICAZIONI WEB 2.0 DI SUCCESSO	53
INSIDE WEB 2.0	11	6.1 DEL.ICIO.US	55
2.1 ASPETTI SOCIALI	11	6.2 NETVIBES	56
2.1.1 Usi sociali del software	11	6.3 WRITELY	57
2.1.2 Il valore dei contenuti delle applicazioni Web 2.0	12	6.4 GOOGLE MAPS	58
2.1.3 La partecipazione	12	6.5 FLICKR	59
2.1.4 Principi fondanti: fiducia, credibilità e controllo sociale	14	6.6 PANDORA	60
2.1.5 Nuove dimensioni del diritto d'autore: Copyright e Copyleft	15	6.7 WORDPRESS E TECHNORATI	61
2.2 ASPETTI CONTENUTISTICI	16	6.8 YOUTUBE	62
2.2.1 Riutilizzo dei contenuti e aggregazione	16	6.9 TAG CLOUD	62
2.2.2 Folksonomy - Tag e metadati	18	6.10 LINKEDIN	63
2.2.3 La geolocalizzazione	18	ALLEGATO 2 - ANALISI COMPARATIVA DI ALCUNI FRAMEWORK AJAX MATURI	65
2.2.4 Redazioni distribuite	19	7.1 INTRODUZIONE	67
2.3 ASPETTI TECNOLOGICI E ARCHITETTURALI: LE TECNOLOGIE ABILITANO LE FUNZIONI SOCIALI DEL SOFTWARE	20	7.1.1 Breve storia	67
2.3.1 Il valore dell'interfaccia	20	7.1.2 Funzionamento di AJAX	67
2.3.2 Il modello GUI (Graphical User Interface)	20	7.2 ELEMENTI DI AJAX	68
2.3.3 Usabilità	23	7.2.1 XmlHttpRequest	68
2.3.4 AJAX e Piattaforme di pubblicazione (Ruby on Rail – Monorail)	24	7.2.2 XML e altri standard di interscambio	69
2.3.5 Formato dati: ATOM e RSS	25	7.2.2.1 XML	69
2.3.6 API: integrazione e interazione	26	7.2.2.2 JSON	70
2.3.7 MASH-UP: modularità e interoperabilità tra servizi	26	7.2.2.3 WDDX	71
2.3.8 Time to market: la versione beta	27	7.2.3 Presentazione e interazione	71
SOCIAL NETWORK: APERTURA ALL'UTENTE E ALLE SUE RELAZIONI	32	7.3 FRAMEWORK, TOOLKIT A AMBIENTI DI SVILUPPO	72
3.1 UNA RETE DI IDENTITÀ	32	7.3.1 Framework Javascript	72
3.1.1 Linee guida	32	7.3.1.1 Prototype	72
3.2 TAGGING E FOLKSONOMY	33	7.3.1.2 Script.aculo.us	73
3.2.1 Usare la Folksonomy	34	7.3.1.3 DOJO Toolkit	73
3.2.2 Linee guida	34	7.3.1.4 Qooxdoo	74
3.3 SISTEMI DI GEOREFERENZIAZIONE E POSSIBILI APPLICAZIONI	35	7.3.1.5 Google Web Toolkit (GWT)	74
3.3.1 Linee guida	35	7.3.2 Framework ibridi	74
3.4 VALUTARE LA USER EXPERIENCE	36	7.3.2.1 Echo2	75
3.4.1 Linee guida	36	7.3.2.2 ZK	75
3.5 REPUTAZIONE, CREDIBILITÀ E CONSENSO	36	ALLEGATO 3 - DESCRIZIONE DEL PROGETTO WAI-ARIA (W3C)	77
3.5.1 Linee guida	37	8.1 RUOLO (ROLE)	79
ACCESSIBILITÀ E WEB 2.0	39	8.2 STATO (STATE)	79
4.1 STATO DELL'ARTE SULLE ARIA (ACCESSIBILE RICH INTERNET APPLICATIONS)	42	8.3 FOCUS	80
4.1.1 Definizione di RIA	42	8.4 ACCESSO ALLE SEZIONI STRUTTURALI DI UN DOCUMENTO	80
4.1.2 Accessibilità nelle applicazioni RIA	43	8.5 GESTIONE EVENTI	80
4.1.3 Definizione di scenari nell'ambito accessibilità AJAX	43	8.6 XFORMS	80
4.1.3.1 Un caso di studio: la gestione delle form	44	WEBGRAFIA E BIBLIOGRAFIA	81
4.1.3.2 AJAX e accessibilità: i possibili approcci alla progettazione	45		
4.1.3.2.1 Interfaccia alternativa senza scripting	46		
4.1.3.2.2 Funzioni alternative senza scripting	46		
4.1.3.2.3 Progressive enhancement	47		
4.1.3.2.4 Rendere gli script accessibili	47		
4.1.4 Riflessioni conclusive: WEB 2.0 e accessibilità	47		

1. INTRODUZIONE

1

*La pubblicazione “Dal Web 2.0 ai Media sociali”
prende spunto dai documenti
oggetto delle attività di ricerca svolte da
CSP - Innovazione nelle ICT per CSI Piemonte
nel corso del 2006.*

*Si ringraziano per la collaborazione
l'ing. Alessandro Sala, Responsabile
del coordinamento della ricerca e sviluppo,
e Flavio Bianchi di CSI Piemonte
e tutte le altre persone
che hanno fornito spunti interessanti
e consigli preziosi ai fini della ricerca
e della realizzazione della pubblicazione.*

1. INTRODUZIONE



Il mondo della comunicazione digitale è in profondo cambiamento, una rivoluzione nella rivoluzione che, dopo molto fermento, molta attività di riorganizzazione e molti clamorosi insuccessi, sta aprendo nuovi fronti creativi, alcuni dei quali già rappresentati da realtà di grandissimo interesse, non solo per gli esperti, ma anche e soprattutto per l'utente finale.

Sentiamo ripetere da tempo che le applicazioni nei prossimi anni saranno sempre meno di tipo top-down e che gli utenti saranno chiamati a ricoprire un ruolo di partecipazione attiva per la condivisione della conoscenza e la valorizzazione dei contenuti sia di tipo professionale, sia di tipo culturale, sia legati alla vita di tutti i giorni e al tempo libero. Questa affermazione è vera, ma va consolidata e rafforzata.

Le applicazioni top-down continueranno a esistere, ma saranno più "tagliate" sull'utente. Sembra un paradosso, ma per molti anni la comunicazione digitale non ha tenuto in considerazione il destinatario: gli utenti target sui quali ci siamo basati per progettare le nostre attività di comunicazione non sono mai esistiti, o per meglio dire, non esistevano nel mondo reale. Il mondo reale non è infatti suddiviso in stereotipi. Le nuove applicazioni "sociali" consentono di abbattere molte delle categorizzazioni che hanno limitato fortemente l'uso della Rete da parte di utenti non informatizzati. È l'utente che lo chiede, è l'utente che crea nuove opportunità partendo magari da un semplice blog o condividendo una foto, un link, una tag con il resto del mondo.

Le dinamiche che nascono intorno ai nuovi Social Media, tecnologie online e pratiche utilizzate dagli utenti per condividere opinioni, esperienze e punti di vista, vedono la tecnologia come elemento trasparente di un processo innovativo di generazione, riuso, e aggregazione dei contenuti, il tutto con una fortissima partecipazione dal basso.

I servizi di nuova generazione presenti nel web, che consentono alle persone di collaborare e condividere le informazioni on line, vengono definiti in modo generico e spesso un po' approssimativo: "Web 2.0". Un'etichetta facile per definire un fenomeno che vedremo essere connotato da notevole complessità.

Questa pubblicazione¹ si pone l'obiettivo di fotografare una realtà in continuo cambiamento, ponendo in rilievo gli

elementi di stacco rispetto alla fase precedente di evoluzione del Web ed evidenziandone gli elementi di continuità. La rivoluzione che sta vivendo il mondo digitale è "fluida", non è caratterizzata da scontri o da fratture di nessun tipo. Siamo immersi in un processo di evoluzione inarrestabile del web. Alle spalle abbiamo più di dieci anni di storia, fatta di errori, grandi scoperte e persone. Oggi abbiamo un media maturo, con numerose sfumature e con la capacità di aggregare informazioni, processi e persone.

Il web 2.0, da semplice etichetta a nuovo sistema di organizzazione delle forme di comunicazione e dell'informazione presenta alcuni elementi distintivi rispetto al web 1.0, che sono rilevabili e analizzabili: possiamo infatti indagare meglio nel DNA di questa evoluzione, con spirito critico, ma anche con molta curiosità.

Il web 2.0 verrà analizzato in modo dettagliato all'interno del primo capitolo che abbiamo chiamato "Inside Web 2.0". dove cercheremo di fornirne un'organizzazione logica funzionale alle linee evolutive e di sviluppo, trattate poi in dettaglio nel secondo capitolo "Social Network: apertura all'utente e alle sue relazioni". Un'analisi di questo genere non può prescindere da una riflessione su quello che è a oggi l'aspetto più critico delle tecnologie utilizzate, l'accessibilità, svolta nel terzo capitolo "Accessibilità e Web 2.0". Per supportare la lettura e fornire un ulteriore approfondimento, viene quindi riportata in conclusione un'analisi di dieci applicativi di successo, dei framework di sviluppo basati su AJAX e delle principali linee guida sull'accessibilità delle nuove applicazioni, elaborate dal W3C.

Vi lasciamo alla lettura della pubblicazione, ricordandovi che come ogni "fotografia" questa riprende e rappresenta un momento ben identificato. Con estrema difficoltà abbiamo provato a fotografare un momento che possa durare oltre il tempo "naturale" di posa. Siamo coscienti che questo sia impossibile, ma fiduciosi che gli elementi fondamentali dell'evoluzione che vi descriveremo siano facilmente assimilabili, identificabili e, ci auguriamo, utili.

¹ Questa pubblicazione nasce da un lavoro di analisi svolto da febbraio 2006 a febbraio 2007 su commessa del CSI Piemonte che ha generato tre report. In fase di redazione di questa pubblicazione il materiale prodotto in precedenza è stato sottoposto a revisione e aggiornamento.

2. INSIDE WEB 2.0



*A fronte di una difficoltà condivisa
all'interno dello stesso mondo Web 2.0,
la ricerca di una definizione di questo concetto
è partita dall'analisi di un insieme di quelli
che sono stati comunemente indicati
come applicativi di successo caratterizzanti
questo stadio evolutivo.
A partire da questi è stata individuata
una serie di tratti comuni,
successivamente categorizzati in tre dimensioni:
sociale, contenutistica e tecnologica,
analizzate qui di seguito.*

2. INSIDE WEB 2.0



2.1 ASPETTI SOCIALI

All'interno di questo capitolo verranno analizzati alcuni dei tratti disitintivi delle applicazioni Web 2.0 rispetto alla dimensione sociale. Il ruolo centrale assunto dall'utente può infatti declinarsi in diverse forme: influenza l'uso che viene fatto dei software, determina il valore dei contenuti, risulta in uno scenario di partecipazione attiva e modifica il modo di intendere alcuni aspetti da sempre centrali nel web, quali il diritto d'autore e i concetti di fiducia e credibilità.

2.1.1 USI SOCIALI DEL SOFTWARE

Con l'aggettivo "sociale" intendiamo, in questo caso, un software che sappia offrire agli utenti uno strumento di relazione attraverso la condivisione di contenuti. A seconda della natura di questa relazione, è possibile distinguere e valutare la "socialità" di un servizio.

Un'applicazione Web 2.0, infatti, può essere:

- **collettiva**: i contenuti prodotti dai singoli utenti sono raccolti e messi a disposizione online. In sostanza, servizi simili si configurano come archivi remoti e organizzati di contenuti terzi;
- **puramente sociale**: i contenuti prodotti dai singoli utenti sono raccolti, messi a disposizione online ed è concesso agli utenti stessi di sviluppare relazioni

Figura 1: mappa concettuale del Web 2.0 creata da Marcus Angermeier nel novembre del 2005
<http://kosmar.de/wp-content/web20map.png>





legate a essi, per esempio commentando, apponendo tag², valutando i commenti stessi. Un fotoblog come Flickr³, per esempio è un classico esempio di applicazione puramente sociale online. I contenuti degli utenti, infatti, sviluppano un valore ulteriore grazie ai commenti e alle note che altri utenti possono apporre alle fotografie presentate da ciascuno nelle proprie gallerie personali.

Entrambe le tipologie possono essere caratterizzate dai seguenti attributi:

- **condivisione di valore:** in questo caso si tratta di applicazioni puramente sociali in cui a una maggiore partecipazione degli utenti corrisponde un aumento della qualità e quantità dei contenuti e di conseguenza un maggior valore del servizio. La dimensione collettiva, quindi, è la garanzia stessa dell'esistenza e della qualità del servizio. Wikipedia⁴, per esempio, è un classico caso di applicazione collettiva in cui il valore per tutti i visitatori del sito è costruito interamente dai contenuti degli utenti.
- **condivisione di un valore e cooperazione:** in questo caso, la cooperazione sincronica o diacronica su contenuti che non sono solo condivisi ma diventano co-creati diventa un ulteriore elemento di valore. La possibilità, insomma, di poter "creare insieme" contenuti aumenta di un ordine di grandezza il grado di relazione tra utenti e favorisce la socialità fin dalla fase produttiva. Sistemi di scrittura condivisa come Writely⁵, per esempio, rispondono pienamente all'esigenza – emersa prepotentemente in ambito business nel corso degli anni – di avere un servizio che consenta la scrittura a più mani online, facilitando per esempio la creazione di contenuti in ambito lavorativo e di fatto annullando i tempi di attesa e logistici tipici delle catene gerarchiche.

2.1.2 Il valore dei contenuti delle applicazioni Web 2.0

L'ottica ispiratrice per cui un servizio acquisisce contenuti (e valore) grazie alla partecipazione diretta degli utenti e dei visitatori, infatti, rende spesso le applicazioni Web 2.0 "ostaggio" di chi lo utilizza.

Alcuni esempi in negativo chiariscono immediatamente lo scenario: un'enciclopedia aggiornata in tempo reale e condivisa come Wikipedia non avrebbe modo di esistere se non vi fossero utenti che regolarmente ne aggiornano le definizioni e aggiungono nuovi lemmi. Allo stesso

modo, servizi online di social tagging, come Del.icio.us⁶, o sistemi di fotoblog non avrebbero ragione di essere senza la costante immissione di contenuti e tag da parte dell'utenza.

Anche applicazioni con una ridotta quantità di contenuti, come i social-network⁷, non possono prescindere da un'alta e costante partecipazione da parte della propria base utenti. Il sistema di social-networking a uso professionale – LinkedIn⁸ – infatti non avrebbe senso se gli utenti fossero pochi. In questo caso il valore, infatti, non è dato dai contenuti (che si limitano alle biografie professionali dei singoli partecipanti), ma dalle relazioni che gli utenti riportano e sviluppano online.

L'impressione è che la socialità (e relativa popolarità) di un servizio Web 2.0 sia un elemento così critico da sovrastare talvolta la qualità stessa. Il fatto che sistemi centralizzati di blogging come Splinder⁹ continuino a godere di un'altissima partecipazione, pur non avendo qualità tecniche particolarmente rilevanti e offrendo funzionalità ridotte rispetto a numerosi concorrenti, testimonia l'importanza del time-to-market in un'ottica sociale in cui il valore condiviso si "sedimenta" e "fa sistema": Splinder, infatti, è stata la prima piattaforma di blogging in ordine di tempo a rivolgersi all'utenza italiana.

2.1.3 La partecipazione

Nel panorama Web 2.0 creare, stimolare, mantenere la partecipazione significa garantire il successo per un servizio online.

Nel paragrafo precedente si è evidenziata l'essenzialità della dimensione sociale, condivisa ed eventualmente collettiva delle applicazioni Web 2.0. Le applicazioni online, tuttavia, possono essere valutate a seconda del grado di partecipazione che consentono agli utenti. La libertà di intervento sul contenuto, sulle relazioni e sui tag ad esso apposti, infatti, è un elemento critico per comprendere il livello di valorizzazione dell'utente, dei suoi contenuti e delle sue relazioni.

I servizi Web 2.0 online, infatti, a grandi linee possono garantire:

- **accesso limitato ai soli utenti registrati e coinvolti nella condivisione:** è il caso di social network come LinkedIn, che sono fruibili solo da utenti registrati che a loro volta condividono i propri curricula e il proprio sistema di relazioni. Gli utenti, quindi, possono usufruire del servizio – che è gratuito – solo se lo alimentano a loro volta;

2_ Un tag è una parola chiave che può essere attribuita ad una risorsa presente in rete (sia essa un documento, un'immagine o un filmato), per individuarla e classificarla. Per un'analisi più approfondita su questo argomento, si veda il capitolo 2.2.2

3_ Applicazione Web 2.0 per la raccolta e la condivisione di materiale fotografico. Per una descrizione più dettagliata si faccia riferimento all'allegato 1 (cap. 6.5).

4_ Si faccia riferimento alla tavola 4

5_ Sistema dinamico per la creazione e condivisione online di documenti. Per una descrizione più dettagliata, si faccia riferimento all'allegato 1 (cap 6.3).

6_ Sistema di social bookmarking dedicato all'archiviazione, condivisione e ricerca di segnalibri. Per una descrizione più dettagliata, si faccia riferimento all'allegato 1 (cap 6.1).

7_ Si faccia riferimento alla tavola 1.

8_ Sistema di social networking finalizzato al business. Per una descrizione più dettagliata, si faccia riferimento all'allegato 1 (cap 6.10).

9_ <http://www.splinder.com>



- **accesso limitato ai soli utenti registrati:** è il caso di sistemi come Flickr, che consentono il commento e l'annotazione delle fotografie solo ad utenti registrati, ma non prevedono l'obbligo di creazione di gallerie. Un utente, quindi, può entrare a fare parte del sistema di relazione/valutazione del servizio senza alimentarlo ulteriormente;
- **accesso relazionale aperto a tutti gli utenti:** i sistemi di blogging, pur in presenza di eventuali strumenti di moderazione, tendenzialmente consentono il commento da parte di utenti non registrati e talvolta anonimi. In questo caso, quindi, agli utenti è concessa una limitata capacità d'azione sui contenuti condivisi (commento, valutazione, feedback), senza che debbano contribuire con contenuti di altro tipo o debbano sottostare a procedure di registrazione;
- **accesso completamente aperto a tutti gli utenti:** in questo caso, gli utenti possono liberamente modificare i contenuti condivisi, senza dover sottostare a registrazioni. Alcuni Wiki, per esempio, consentono a chiunque la modifica e l'aggiunta di lemmi e definizioni, senza richiedere dati sensibili agli utenti. In questi casi, solitamente sono presenti sistemi remoti di moderazione.

Lo scenario, tuttavia, non è così definito come evidenziato nell'elenco appena riportato. Molti servizi, infatti, adottano politiche miste di promozione della partecipazione. Un servizio come LinkedIn, per esempio, da poco si è aperto all'esterno e consente agli utenti di pubblicare su una pagina Web un proprio curricu-

lum visibile a tutti (anche a chi non è registrato al servizio).

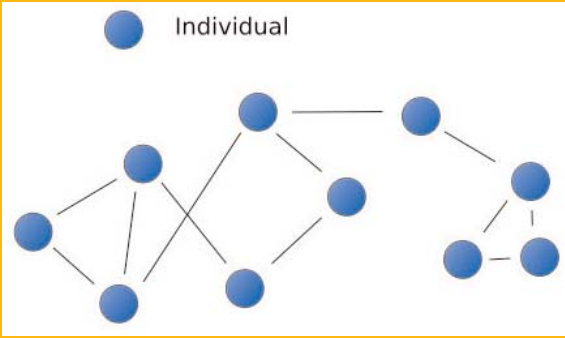
Un altro elemento di cui tenere conto è il grado di personalizzazione della partecipazione consentito agli utenti. Sistemi come WordPress¹⁰, per esempio, permettono ai titolari dei blog di impostare diversi livelli di accesso alla produzione dei contenuti e consentono di abilitare i commenti a tutti o solo a utenti registrati.

Allo stesso modo, un servizio come Google Calendar permette di stabilire se un calendario è pubblico, privato o condiviso. In quest'ultimo caso, consente di personalizzare la condivisione in modo estremamente raffinato, includendo singoli utenti, gruppi di utenti o addirittura consentendo la libera condivisione in scrittura e lettura.

Resta, infine, un ultimo aspetto da valutare. Molti servizi Web 2.0 fanno concorrere un utilizzo gratuito e un utilizzo a pagamento. Ovviamente agli utenti paganti sono garantiti servizi di maggiore qualità, funzionalità aggiuntive, mentre i non paganti sono esclusi.

Le implicazioni di questa struttura – peraltro assolutamente naturale in un'ottica di mercato – sono tutte da valutare: servizi a pagamento di grande qualità ma non di massa potrebbero paradossalmente avere meno valore di quelli gratuiti ma partecipati collettivamente da molti utenti. Per una valutazione di questi aspetti, tuttavia, è ragionevole pensare che sia necessario attendere un più completo sviluppo del panorama Web 2.0, la commercializzazione definitiva di prodotti ora gratuiti e in beta e un assestamento dello scenario, che è ancora in fase pionieristica.

Tavola 1



Individual

SOCIAL NETWORK

Una rete sociale (spesso si usa il termine inglese social network) consiste di un qualsiasi gruppo di persone connesse tra loro da diversi legami sociali, che vanno dalla conoscenza casuale ai vincoli familiari.

La regola dei 150 afferma che le dimensioni di una vera rete sociale sono limitate a circa 150 membri. Questo numero è stato calcolato da studi di sociologia e soprattutto di antropologia, sulla dimensione massima di un villaggio (in termini più attuali meglio definibile come un ecovillaggio).

[...] La versione di Internet delle reti sociali è una delle forme più evolute di comunicazione in rete, ed è anche un tentativo di violare la “regola dei 150”. La rete delle relazioni sociali che ciascuno di noi tesse ogni giorno, in maniera più o meno casuale, nei vari ambiti della nostra vita, si può così “materializzare”, organizzare in una “mappa” consultabile, e arricchire di nuovi contatti.

Il fenomeno delle social network nasce negli Stati Uniti e si è sviluppato attorno a tre grandi filoni tematici: l'ambito professionale, quello dell'amicizia e quello delle relazioni.

Le social network online ebbero un'esplosione nel 2003, grazie alla popolarità di siti web come Friendster [www.friendster.com], Tribe.net [tribe.net] e LinkedIn [www.linkedin.com]. Il motore di ricerca Google, ha lanciato Orkut [www.orkut.com] il 22 gennaio 2004. Kibop [kibop.com], una social network in spagnolo e portoghese, ha debuttato anch'essa nel 2004. In Italia il primo dei grandi portali passati verso questo tipo di social network è stato superava [it.superava.com], ma sono comunque vivissime le comunità di italiani su Orkut e LinkedIn.

(parzialmente tratto da Wikipedia)

¹⁰ Piattaforma di blogging opensource. Per una descrizione più dettagliata, si faccia riferimento all'allegato 1 (cap 6.7).



2.1.4 Principi fondanti: fiducia, credibilità e controllo sociale

Poiché le persone tendono a rispondere socialmente alle tecnologie¹¹, concetti propri del dominio delle relazioni interpersonali, quali fiducia e credibilità, assumono rilevanza anche come chiavi per spiegare il successo o l'insuccesso di determinate applicazioni.

Secondo la definizione classica di Fogg¹², la *credibilità* è una qualità percepita, una valutazione che si basa su molteplici dimensioni, tra cui le principali sono **l'affidabilità e l'esperienza**. L'affidabilità si riferisce all'etica dell'oggetto che viene valutato, al fatto che questo venga o meno percepito come benintenzionato e sincero. Al contrario, l'*esperienza* si riferisce alle conoscenze e competenze possedute. Perché una persona o una tecnologia sia considerata credibile, deve ottenere una valutazione positiva rispetto a entrambe le dimensioni. Nell'interazione con una tecnologia, la credibilità è rilevante quando questa:

- è una fonte di conoscenze;
- consiglia o fornisce istruzioni agli utenti;
- riporta misure;
- riferisce i risultati delle operazioni svolte;
- comunica il proprio stato;
- offre delle simulazioni;
- ricrea ambienti virtuali.

Nel caso del Web 2.0, la rilevanza della credibilità dovrebbe essere valutata a seconda dei casi e del tipo di servizi offerti da ciascuna applicazione.

In generale, tuttavia, è significativo che molte applicazioni Web 2.0 contino tra le proprie principali caratteristiche il riuso di informazioni provenienti da fonti terze: in tal caso, la credibilità del servizio viene influenzata, in positivo o in negativo, dalla credibilità stessa delle informazioni pubblicate. Tale relazione non è vera se la scelta delle informazioni da pubblicare dipende del tutto dall'utente (ad esempio, in un aggregatore¹³ che non fornisce contenuti di default).

Molte applicazioni Web 2.0 ricadono inoltre nella seconda categoria introdotta, in quanto offrono all'utente (più o meno esplicitamente) consigli e istruzioni. In particolare, spesso vengono proposti contenuti "personalizzati" che l'utente potrebbe trovare interessanti, sulla base dei propri gusti, dei comportamenti mostrati in passato e delle scelte di altri utenti.

La prospettiva di accedere a contenuti personalizzati suscita generalmente maggiore attenzione e interesse¹⁴ ma, qualora i contenuti proposti non soddisfino l'utente, il servizio perde credibilità, in quanto i suoi consigli vengono percepiti come "errati". La possibilità offerta

da molti servizi Web 2.0 di abbinare ai contenuti proposti un feedback proveniente da altri utenti (commenti, valutazioni, o anche semplici "tag", nel caso, ad esempio, di del.icio.us, in cui il fatto stesso di inserire un sito tra i propri bookmark implica una valutazione positiva) contribuisce a rinforzare la credibilità del servizio, agendo, in particolare, sulle seguenti dimensioni:

- **Credibilità "di superficie"** (il fatto che altri utenti abbiano usato/valutato positivamente un servizio contribuisce a creare una prima impressione positiva);
- **Reputazione** (livello di credibilità attribuito ad un certo oggetto sulla base delle valutazioni di attori terzi ritenuti "autorevoli": può essere sfruttato se l'utente conosce le persone che "commentano" un certo contenuto, ad esempio, perché sono suoi amici o perché ha già avuto modo di apprezzare la bontà dei loro consigli).

Tali dinamiche non sono tuttavia sufficienti nel determinare una valutazione stabile della credibilità, che si può ottenere solo in seguito all'esperienza diretta dell'utente. In tal caso, le impressioni formate sulla base di valutazioni "di superficie" e legate alla reputazione possono venire confermate, ma anche sovvertite, nel qual caso anche l'autorità dei personaggi che hanno valutato positivamente un certo contenuto può risentirne.

Per le applicazioni che propongono attivamente contenuti all'utente, la dimensione della credibilità risulta dunque un elemento "critico" nel determinare il successo del servizio.

Qui di seguito vengono riportati degli esempi significativi riscontrati in alcune applicazioni Web 2.0.

- **del.icio.us**: tutti i contenuti sono inseriti in seguito ad una scelta dell'utente; la credibilità di del.icio.us in quanto servizio non dipende dalla credibilità dei contenuti. Tuttavia, sono ben presenti i meccanismi di credibilità e autorità legati alle valutazioni degli utenti sulle singole risorse ed è anche possibile che il servizio in generale perda credibilità se la maggior parte delle esperienze dell'utente risultano poco soddisfacenti (anche se in seguito a scelte poco accurate da parte di altri utenti e non a causa di malfunzionamenti del servizio di per sé): ad esempio, se un utente constata spesso che i "related tags" risultano fuorvianti, o che i commenti che possono accompagnare i bookmarks sono generici e poco utili;
- **Pandora**¹⁵: i contenuti proposti dovrebbero essere personalizzati per incontrare i gusti del singolo utente: se la maggior parte della musica non risulta gradevole, o se l'utente non concorda con le scelte del servizio (ad esempio, perché non ritiene che due

11 Reeves B. - Nass C, The Media Equation, Stanford (CA), CSLI, 1996

12 <http://credibility.stanford.edu/guidelines/index.html>

13 Un aggregatore (o feed reader) è un software che ricerca automaticamente e raccoglie contenuti strutturati in un particolare formato detto "feed". Gli aggregatori possono presentarsi sia come pagine Web, sia come programmi indipendenti, accessibili da desktop.

14 Hovland C.L. - Janis, I.L. - Kelley, H.H., Communication and Persuasion. Psychological studies on opinion change, New Haven, CO, Yale University Press, 1953.

15 Web-radio che permette di creare stazioni radio a partire da un brano o da un artista scelto dall'utente. Per una descrizione più dettagliata, si faccia riferimento all'allegato 1 (cap 6.6).



artisti possano essere considerati simili, o che un brano possa essere descritto in un certo modo), questo perde credibilità e, di conseguenza, risulta inadatto a raggiungere il proprio obiettivo.

La *fiducia* è un concetto contiguo a quello della credibilità, ma ad esso non del tutto sovrapponibile: secondo Lee e See¹⁶, si può definire come “fiducia” la credenza di un individuo che un agente (sia esso una persona o una tecnologia) lo aiuterà a raggiungere i propri obiettivi, in una situazione caratterizzata da incertezza e vulnerabilità. Nel contesto del Web 2.0, l’utente si può considerare “vulnerabile” in rapporto alla diffusione dei propri dati, siano essi “sociodemografici” o “contenutistici”, intendendo con questo termine le risorse che una persona sceglie di condividere con altri utenti.

In particolare, è necessario che l’utente abbia la sicurezza che i propri dati sensibili siano adeguatamente protetti e che i contenuti che pubblica siano effettivamente accessibili solo dai gruppi di persone specificati, nel caso sia possibile compiere una scelta in questo senso.

Si può invece definire il contesto del Web 2.0 “incerto” rispetto alla disponibilità nel tempo di un determinato servizio/applicazione (di fatto, non viene data alcuna garanzia esplicita rispetto al fatto che certe applicazioni continueranno ad esistere anche in futuro) e alla possibilità di accedere “per sempre” ai contenuti depositati presso una certa applicazione che risiede su una macchina remota ed è accessibile solo tramite collegamento ad Internet.

Tra gli elementi che influiscono sulla scelta di attribuire o meno fiducia ad una certa applicazione risultano particolarmente importanti nel panorama del Web 2.0:

- la possibilità di confrontarsi con le scelte di altri utenti (ad esempio, si attribuisce fiducia ad un determinato servizio perché è stato scelto da molte altre persone, eventualmente conosciute, e quindi costituisce una sorta di “standard di fatto” e sembra offrire maggiori garanzie di stabilità);
- il fatto che organizzazioni relativamente consolidate decidano di acquisire determinati servizi fornisce una sorta di garanzia sulla stabilità e sull’affidabilità di questi ultimi.

Così come per la credibilità, tuttavia, la fiducia che un utente attribuisce a un servizio dipende in larga parte dalla sua esperienza diretta e dal fatto di aver verificato personalmente la capacità di un sito di svolgere adeguatamente i servizi promessi e di tutelare i dati dell’utente stesso.

Ad esempio, in Flickr, quando l’utente carica sul sito le proprie foto, deve avere la ragionevole certezza che il servizio sarà accessibile anche in futuro e che le proprie risorse

non andranno perse. Inoltre, l’utente deve essere sicuro che le restrizioni all’accesso che sceglie per ciascuna delle sue foto siano applicate efficacemente. L’utente può arrivare ad una valutazione positiva del servizio sia basandosi su prove dirette (ad esempio, verificando personalmente che le foto definite “private” non siano fruibili senza effettuare il login), sia valutando la popolarità del servizio all’interno della comunità degli utenti, sia assumendo la proprietà da parte di Yahoo come una garanzia della stabilità dell’applicazione.

2.1.5 Nuove dimensioni del diritto d’autore: Copyright e Copyleft

Produzione, condivisione, valutazione e immissione in un circuito di relazioni di un contenuto aprono numerosi problemi legati al copyright. Anzi, in alcuni casi mettono in crisi il concetto di copyright stesso.

Un contenuto condiviso e commentato, infatti, a chi appartiene? È interamente di proprietà di chi lo ha scritto o è un’opera attribuibile ad autore materiale e commentatori? E se sì, in quali misure? Sono sufficienti e adeguati i soli criteri quantitativi, per stabilire il grado di autorialità?

Il tutto, per di più, si complica nel caso di contenuti collettivi o co-creati in redazioni distribuite¹⁷. Ad esempio, chi è l’autore e il titolare dei diritti editoriali delle definizioni di Wikipedia?

Tutti questi interrogativi di fatto aprono il “problema-copyright”. L’attribuzione del diritto di autore e il concetto di proprietà intellettuale, infatti, paiono superati dallo sviluppo tecnologico, contenutistico e relazionale tipico del Web 2.0.

In gran parte la risoluzione del problema spetterà agli esperti di materie giuridiche. Dal punto di vista dell’analisi dei nuovi media, tuttavia, è possibile sviluppare alcune osservazioni che consentano di inquadrare meglio il problema e sviluppare una forma iniziale di analisi delle applicazioni Web 2.0 dal punto di vista del diritto d’autore.

In particolare, è necessario tenere conto di questi elementi:

- alcuni servizi online di fatto si appropriano per contratto dei contenuti condivisi dagli utenti: è il caso di alcune piattaforme di blogging, che – almeno tecnicamente – diventano proprietarie di quanto prodotto e condiviso dai propri utenti. Spesso si tratta di servizi gratuiti;
- alcuni servizi impongono agli utenti determinate licenze copyleft¹⁸, che stabiliscono modalità e caratteristiche di impiego, riuso e modifica dei contenuti prodotti e condivisi;
- alcuni servizi online permettono agli utenti di stabilire la licenza copyleft da utilizzare sui propri contenuti.

¹⁶ Lee, J.D. - See, K.A., Trust in automation: designing for appropriate reliance, in “Human Factors”, vol. 46, no. 1, Iowa, University of Iowa, 2004, pp. 50–80.

¹⁷ Si definiscono “distribuite” le redazioni formate da gruppi di persone fisicamente distanti che collaborano alla stesura di progetti editoriali comuni. Per un’analisi più approfondita su questo argomento, si veda il capitolo 2.2.4.

¹⁸ L’espressione copyleft nasce da un gioco di parole su copyright e si riferisce a un modello alternativo di gestione dei diritti d’autore. Con le licenze copyleft, è l’autore dell’opera a indicare ai fruitori se e in che modo questa può essere utilizzata, diffusa e modificata legalmente. Sono licenze copyleft ad esempio GNU GPL (per il software) e Creative Commons.



Flickr, per esempio, permette agli utenti di impostare una macro-licenza per l'intero corpus di fotografie condivise.

Lo scenario, in questo caso, è ancora di difficile interpretazione. In particolare, il caso di Wikipedia ha suscitato numerosi dibattiti legati in primo luogo al copyright e – in secondo luogo – al suo derivato: la responsabilità editoriale. La responsabilità editoriale, di fatto, è tuttora un problema aperto: è difficile, per un'opera collettiva e condivisa online, magari da un pubblico internazionale, stabilire chi è responsabile dei contenuti.

Tornando a Wikipedia, è capitato più volte che alcune persone si sentissero diffamate o danneggiate da alcune definizioni presenti sul servizio. Chi è responsabile legalmente di quanto scritto? I titolari di Wikimedia, azienda che patrocina Wikipedia? L'autore materiale della definizione (se identificabile)? E se si individua un colpevole, qual è il dominio legale di competenza? Lo Stato di residenza del presunto danneggiato? Lo Stato di residenza del presunto danneggiatore? Lo Stato di residenza dei server su cui risiede l'informazione?

Anche in questo caso, gran parte degli interrogativi sono in attesa di risposta. Ciò che si può osservare è che gran parte dei termini d'uso delle applicazioni Web 2.0 contengono clausole di manleva, in cui gli utenti si fanno carico della responsabilità di quanto prodotto e condiviso.

2.2 ASPETTI CONTENUTISTICI

Sul fronte dei contenuti che popolano gli applicativi Web 2.0 è possibile riscontrare alcune costanti: le informazioni vengono spesso sottoposte a un processo di riuso e aggregazione, arricchite e classificate attraverso l'utilizzo di tag – anche riferiti alla posizione geografica – e generalmente sono il prodotto di redazioni distribuite.

2.2.1 Riuso dei contenuti e aggregazione

L'aspetto collettivo e condiviso delle applicazioni Web 2.0 sviluppa naturalmente dinamiche in cui i contenuti e il valore prodotto dai singoli utenti sono messi in comune, sono oggetto di lettura e revisione collettiva e – in molti casi – sono modificati, etichettati e a loro volta ri-condivisi da terzi.

Questo è un chiaro scenario di riuso dell'informazione, che non è più una materia fissa e intoccabile, ma contenuto fluido e aperto, che si arricchisce in virtù della sua condivisione e modificazione costante.

In alcuni casi perfino oggetti che sembrano naturalmente immutabili come le fotografie subiscono un'evoluzione contenutistica collettiva. In Flickr, per esempio, la possibilità per tutti gli utenti di aggiungere commenti,

note (direttamente sull'immagine) e giudizi ad ogni immagine condivisa da terzi, di fatto aumenta il valore comunicativo di ciascuna fotografia.

Senza eccedere in apodissi, è facile immaginare che i servizi Web 2.0 in gran parte agevolino la condivisione e il riuso delle informazioni, dei dati, delle relazioni e delle esperienze. Il loro obiettivo primario, in genere, è basato sulla costruzione di valore a partire dalla condivisione e dalla cooperazione nella produzione di contenuti. Il riuso può quindi configurarsi come un fattore solitamente necessario.

D'altronde una delle definizioni più felici dell'universo Web 2.0 è *"The power of us"* (il potere di "noi"), uno scenario in cui la dimensione collettiva, dal punto di vista della produzione di valore, è polidirezionale: tutti sono o possono essere autori, co-autori, revisori e fruitori allo stesso tempo. Uno sguardo panoramico alle principali applicazioni Web 2.0 chiarisce subito le dimensioni e la forza del riuso: servizi come Wikipedia non avrebbero alcun vantaggio competitivo sulle enciclopedie tradizionali, se non godessero di una costante revisione e modifica dei propri contenuti.

Con il boom del Web 2.0 si è evidenziato il successo – più fattuale che mediatico – del cosiddetto UGC (*User Generated Content*)¹⁹, cioè i contenuti creati dagli utenti. Per questo fenomeno, che vede i navigatori della Rete protagonisti e produttori di contenuto e non solo fruitori passivi, è stata coniata l'espressione "read-write Web": uno spazio universale in cui non c'è più distinzione di ruolo tra chi legge e chi scrive e in cui chiunque, attraverso numerosi mezzi, può partecipare con i propri contenuti alla "grande conversazione" online.

Grazie alla diffusione di tecnologie come i feed RSS²⁰, i contenuti online non sono più dipendenti dall'interfaccia grafica con cui sono inizialmente presentati e, anzi, si prestano ad essere condivisi, riutilizzati e impiegati in modo diverso. È il fenomeno del **riuso**, cioè la riproposizione con nuove forme di contenuti online. I contenuti, quindi, non esauriscono la loro funzione e il loro ciclo vitale nel momento della loro prima pubblicazione online ma, grazie al riuso, possono essere impiegati per servizi terzi, associati a contenuti simili, proposti per la discussione o la valutazione, etichettati e socialmente condivisi, ecc.

Il principale riuso dei contenuti online è l'**aggregazione**: associare più contenuti da fonti diverse in base a parametri specificati in precedenza. Il proliferare dell'UGC di fatto ha ingigantito a dismisura la quantità dei contenuti presenti online. Per poter rendere efficace la loro fruizione è necessario l'utilizzo di servizi in grado di filtrare il più possibile gli elementi non pertinenti agli interessi dell'utente e associare quelli più interessanti e vicini alle preferenze espresse dall'utente. Il fenomeno dell'aggregazione risponde proprio a questa necessità,

¹⁹ Si faccia riferimento alla tavola 2.

²⁰ RSS è un formato per la condivisione di informazione pubblicata su siti web con altre applicazioni. Per un'analisi più approfondita su questo argomento, si veda il capitolo 2.3.5.



consentendo agli utenti di associare (eventualmente filtrandoli attraverso l'uso di keyword specifiche) contenuti simili ma provenienti da fonti diverse. In questo modo una verticalità tematica si associa un'orizzontalità delle fonti, garantendo un'informazione migliore e una maggiore circolazione dei contenuti.

Oltre ai tradizionali aggregatori online, che altro non sono che versioni su Internet dei tradizionali reader RSS software, il panorama Web 2.0 presenta numerosi aggregatori tematici, cioè servizi online come Digg²¹ in cui gli utenti possono specificare alcune chiavi di ricerca e ottenere un elenco di contenuti a tema, ordinati per pertinenza. Non solo, gli utenti possono anche segnalare nuovi contenuti, associarli al tag e condividerli socialmente.

La moltiplicazione degli autori, dei contenuti e delle loro versioni fa sì che per i fruitori dei servizi Web 2.0 l'ordine sia un criterio fondamentale per poter operare con efficacia. Allo stesso tempo, la necessità di ordine si sposa con l'esigenza di associare contenuti simili, di correlarli e in generale di operare su di essi con criteri semantici. La risposta a queste esigenze è lo strumento dell'aggregazione, cioè la raccolta dei contenuti attraverso particolari servizi – gli “aggregatori” – in grado di assemblare in uno spazio unico materiale proveniente da fonti diverse.

La tecnologia alla base degli aggregatori è la *syndication*, cioè la fornitura distribuita dei contenuti da parte di siti Web e servizi online. Il principale formato di syn-


dication è la Really Simple Syndication, cioè il formato RSS. Si tratta, nei fatti, di un sistema di distribuzione dei contenuti attraverso specifici file XML²², che consentono di aggiornare costantemente i fruitori di un servizio ogni volta che i contenuti di questo sono aggiornati.

Usare un aggregatore, presente sia come software sia come servizio online, consente di “abbonarsi” a più fonti di contenuto (ad esempio a numerosi blog) e ricevere gli aggiornamenti di contenuto ogni volta che questi si aggiornano. Il risultato è una tecnologia in grado di offrire informazioni pertinenti – perché la fonte è stabilita dal fruitore – in grado di giungere automaticamente a chi le consuma, senza che questi debba visitare le fonti per ottenerle.

Già in passato vi erano stati tentativi di creazione di un sistema di aggregazione – per esempio l'Active Desktop di Microsoft – ma erano troppo in anticipo sui tempi. L'aggregazione, infatti, ha senso in un'epoca in cui la cultura dell'*always one*, la connettività permanente, ben sviluppata e i fruitori sono costantemente – o per lunghissimi periodi – online.

Un vantaggio ulteriore dell'aggregazione è la sua sinergia coi tag: grazie a servizi come Technorati²³ è possibile “abbonarsi” ad uno o più tag e ricevere i contenuti pertinenti più aggiornati man mano che vengono pubblicati. In questo caso l'aggregazione assembla più fonti sulla base di una categoria o di una parola chiave e produce immediatamente un insieme di contenuti che hanno il vantaggio della pertinenza e della novità.

Tavola 2



USER GENERATED CONTENT (UGC)

La definizione “contenuto generato dagli utenti” (User-Generated Content o UGC in inglese) è nata nel 2005 negli ambienti del web publishing e dei new media per indicare il materiale disponibile sul web prodotto da utenti invece che da società specializzate. Essa è indice della democratizzazione della produzione di contenuti multimediali reso possibile dalla diffusione di soluzioni hardware e software semplici e a basso costo. [...]

L'avvento del contenuto generato dagli utenti segna una svolta nelle imprese che si occupano di media, dalla creazione di contenuti in linea alla creazione di ambienti e strumenti per permettere a non professionisti (ovvero la “gente comune”) di pubblicare i propri contenuti in luoghi importanti.

Il network britannico BBC, ad esempio, organizzò un gruppo di prova “User-Generated Content” (UGC) composto da tre persone, nell'aprile del 2005. A seguito dell'attacco terroristico di Londra del 7 luglio 2005 e dell'incendio del deposito petrolifero dell'Hertfordshire nello stesso anno, il gruppo divenne permanente e venne ampliato, riflettendo l'arrivo del concetto di citizen journalism, il cittadino giornalista nei media principali. Durante il disastro di Buncefield, la BBC ricevette oltre 5.000 foto dai suoi spettatori. Il dibattito sulle vignette di Maometto (gennaio 2006) portò a oltre 22.000 email inviate da parte degli spettatori nelle prime due settimane. Di norma la BBC non remunera i contenuti generati dai suoi spettatori.

In anni recenti i contenuti generati dagli utenti sono stati per alcuni di essi una strada verso il successo. La canzone umoristica *Because I Got High*, di Afroman, divenne la più richiesta nelle radio statunitensi nel luglio 2002, grazie al suo successo su Napster. Nell'estate del 2006, Brooke Brodack siglò un contratto per lo sviluppo di talenti, dopo la comparsa delle sue brevi scenette su YouTube. Le aziende si stanno sempre più interessando allo sfruttamento di spazi intesi in origine per i contenuti generati dagli utenti, come evidenziato dallo scandalo Lonelygirl15 su YouTube, dove un finto videoblog venne usato per promuovere una aspirante attrice. (Parzialmente estratto da Wikipedia)

21_ <http://www.digg.com>

22_ L'XML, acronimo di eXtensible Markup Language, è un linguaggio per la descrizione della struttura e del significato dei dati contenuti in un documento.

23_ Motore di ricerca online basato sui feed RSS creati dai blog. Per una descrizione più dettagliata, si faccia riferimento all'allegato 1 (cap 6.7).

2.2.2 Folksonomy - Tag e metadati

Nel panorama del Web 2.0, per tag si intendono delle etichette che vengono apposte ai contenuti, caratterizzandoli per categorie e parole chiave. L'idea alla base dei tag è semplice: fare sì che i contenuti siano ricercabili, correlabili e utilizzabili in base a parametri semantici qualitativi e non quantitativi.

Se i motori di ricerca tradizionali – tipo Google – di fatto basano le proprie ricerche (tra i tanti parametri) sul numero di volte che le parole cercate compaiono in una pagina e sulla “popolarità” di quest’ultima (misurata con un complesso algoritmo in cui prevalgono i siti più linkati da altri siti), la filosofia dei tag adotta una prospettiva completamente diversa: gli autori – e in alcuni casi i lettori – etichettano con keyword i contenuti e il valore che stabilisce un ranking tra contenuti non è più la popolarità, ma l’aggiornamento: i contenuti più recenti hanno la precedenza su quelli più vecchi.

Gran parte delle applicazioni, quindi, permette di affiancare ad ogni contenuto uno o più tag, scelti liberamente. Questo capita per tutti i tipi di contenuti, dal testo (sui blog) alle fotografie, fino ai video su YouTube²⁴.

L’uso massiccio dei tag permette una maggiore agilità nell’uso dei contenuti. Questi, infatti, sono più facilmente ricercabili (raramente un tag genera equivoci, cosa che invece può accadere facilmente con una ricerca full-text) e sono estremamente correlabili. Gli esempi chiariscono i vantaggi del creare un sistema condiviso di contenuti “taggati”. Un servizio come Flickr, per esempio, consente di apporre tag alle immagini: basta ricercare tutte le foto che dispongono di un tag comune, per esempio “Cipro”, per poter visualizzare con facilità tutte le immagini scattate dagli utenti nell’isola mediterranea. La stessa dinamica, applicata al testo, per esempio, consente di ricostruire filoni concettuali, conversazioni, panoramiche di opinioni multiple, ecc.

L’idea di dotare il corpus di contenuti della Rete di un sistema tassonomico condiviso, aperto e generato dal basso è chiaramente in contrasto con i principi del Web semantico, il cui obiettivo è costruire un ordine dall’alto. Il tagging invece produce, in un modo assolutamente ingovernato ma sorprendentemente efficiente, una *folksonomy* (neologismo formato dall’unione di *folk* – compare/amico – e *taxonomy* – tassonomia/sovrastruttura di classificazione), cioè – con una traduzione impropria – una “tassonomia di noi amici”, il cui obiettivo non è produrre l’ordine assoluto, ma il “miglior disordine possibile”, cioè quello che soddisfa le ricerche e sa adattarsi ad un insieme di contenuti in evoluzione, adattando costantemente il proprio sistema classificatorio in base al modello mentale emergente tra gli utenti.

Tavola 3



VISUAL TAGGING

I visual tag sono etichette, al cui interno viene codificata una URL utilizzando una tecnica simile a quella dei codici a barre, che possono essere posizionate sugli oggetti del mondo reale e lette attraverso la telecamera integrata all’interno dei dispositivi mobili. Il telefono può così utilizzare il browser interno per presentare le pagine web che si riferiscono all’oggetto taggato.

Sono due gli esempi più noti. Shotcode [<http://www.shotcode.com>], creato nel 1999 all’Università di Cambridge, è caratteristico per la sua forma circolare ed è stato utilizzato per la prima volta a fini commerciali dalla Heineken. Semacode [<http://semacode.org>] invece è stato realizzato da una compagnia canadese che mette a disposizione il software liberamente sulle pagine del proprio sito oltre a suggerire diverse e interessanti possibili applicazioni. Il 28 giugno del 2005 è stato lanciato Semapedia [<http://www.semapedia.org>], un progetto che utilizza i semacode per connettere oggetti e luoghi del mondo reale ai relativi articoli pubblicati su Wikipedia. A maggio 2007 i tag creati sono stati 6840.

il visual tag qui riportato si riferisce a <http://www.socialmedialab.it>

2.2.3 La geolocalizzazione

La geolocalizzazione può essere intesa come una applicazione particolare del tagging. È possibile infatti categorizzare i contenuti anche da un punto di vista geografico: apporre un tag che contenga informazioni di carattere geografico a un’immagine, un testo o un video è molto facile e può portare a un notevole aumento del valore del contenuto. In precedenza si è fatto l’esempio dei tag geografici di Flickr, che permettono di visualizzare tutte le immagini relative ad un luogo. Il beneficio della pertinenza geografica, tuttavia, si può estendere a ogni tipo di contenuto. Un’evoluzione relativamente recente della geolocalizzazione permette di associare ai contenuti un tag molto più specifico di un’indicazione geografica generica. È facile, infatti, associare a un contenuto informazioni precise sul suo “dove”: basta associarlo a specifiche coordinate su una mappa di riferimento (attraverso, per esempio, l’uso di un GPS) oppure fisicamente puntarlo su una mappa grafica (attraverso l’indicazione manuale o l’immissione – ove possibile – di un indirizzo stradale).

Questa evoluzione permette di sviluppare servizi evoluti, in grado di associare i contenuti a specifiche località. L’insieme di informazioni, quindi, non è più solo fruibile attraverso un sistema cronologico, ma può disporre anche di un sistema geografico. Un semplice esempio chiarisce immediatamente i vantaggi portati dalla geolocalizzazione e dal geo-tagging: un urban-blog (cioè un blog che fa riferimento ad uno specifico contesto metropolitano) può disporre di una mappa cliccabile in cui i singoli post sono graficamente associati ai luoghi a cui fanno riferimento. Questo permette la creazione di mappe interattive in cui i

²⁴ Sistema per l’archiviazione e la condivisione di videoclip. Per una descrizione più dettagliata, si faccia riferimento all’allegato 1 (cap 6.8).



punti di interesse sono di fatto i singoli post prodotti dall'autore. Dal punto di vista dell'utente significa poter produrre contenuti associabili a una mappa condivisa socialmente, cioè la creazione di una mappa commentata personalizzata e condivisibile con terzi. Per gli utenti, quindi, si tratta di una dimensione inedita di relazione, produzione e organizzazione dei contenuti, con scenari d'applicazione innovativi ancora tutti da esplorare.

Il fenomeno della geolocalizzazione è infatti in espansione costante: grazie alla presenza di mappe interattive utilizzabili liberamente e dotate di API²⁵ per l'integrazione in servizi terzi – ad esempio quelle di Google (Google Maps²⁶), di Yahoo o di Windows Live Local – stanno nascendo numerosi servizi in cui il valore collettivo prodotto dagli utenti è map-based. Allo stesso tempo, numerosi plug-in consentono di integrare contenuti di ogni tipo su mappe condivise, dai testi dei blog fino alle immagini e ai video.

Quella geografica sembra proprio essere la nuova dimensione del Web, al punto che la tradizionale conferenza sul Web 2.0 organizzata dalla O'Reilly Inc. è stata intitolata – nell'ultima edizione di giugno 2006 – “Where 2.0”, proprio per sottolineare l'emergere del nuovo paradigma geografico sul Web. L'elemento notevole dei tanti servizi di mapping online, che offrono mappe accurate dell'intero pianeta, spesso abbinata a fotografie satellitari, non è sicuramente la loro qualità intrinseca (anzi, il dibattito è aperto su quali siano le mappe online più fedeli alla realtà e meglio integrate con le immagini satellitari), ma la loro capacità di essere utilizzate da parte degli utenti in modo sociale.


2.2.4 Redazioni distribuite

Il moltiplicarsi degli autori, la disseminazione del sapere e la sua moltiplicazione nella diversità di fatto hanno rafforzato il concetto di “redazione diffusa”. La definizione è già nota dai tempi delle prime redazioni online: sistemi redazionali che vanno “oltre il senso del luogo”, per citare Meyrowitz²⁷, grazie alla Rete e permettono a persone fisicamente distanti di collaborare alla redazione di progetti editoriali comuni, dal semplice testo a strutture più complesse, come una pubblicazione. Grazie al Web 2.0 il concetto di redazione diffusa si sposta oltre: la produzione remota e pluri-fonte di contenuti non è più assoggettata a un'entità coordinatrice centralizzata (solitamente un capo-redattore in grado di tenere le fila dei vari autori), ma è totalmente auto-gestita attraverso servizi online. L'esempio più lampante di redazione globale con decine di migliaia di utenti è Wikipedia, dove i singoli lemmi e le loro definizioni sono frutto del lavoro di condivisione e collaborazione (asincrona e remota) tra soggetti diversi e reciprocamente ignoti.

Chiaramente l'apertura di redazioni collettive e condivise non è indolore: aprire la produzione di contenuti al libero arbitrio

di tanti implica numerosi rischi (dal conformismo alla devianza) tipici di tutte le strutture collettive non controllate. Il caso di Wikipedia, tuttavia, dimostra che una struttura aperta di contenuto di massa – in presenza di strumenti di moderazione non particolarmente forti – riesce ad autogovernarsi, al punto che eventuali “giochi sporchi” praticati con le definizioni di alcuni lemmi (es. alcuni gruppi neonazisti di recente hanno cercato di modificare la definizione di “Shoah”, riempiendola di teorie negazioniste) hanno avuto vita breve, spesso inferiore al minuto: gli utenti stessi provvedono a modificare ulteriormente le definizioni colpite, ristabilendo l'equilibrio informativo. Se da un lato questo è un vantaggio, chiaramente si profila il rischio del conformismo e cioè che i contenuti di redazioni di massa siano di fatto espressione del pensiero della maggioranza, lasciando poco o alcuno spazio al dissenso, alla controinformazione e alle culture emergenti. Si tratta, chiaramente, di un dibattito aperto e – immaginiamo – critico per il futuro dell'informazione condivisa.

Tavola 4



WIKIPEDIA

Wikipedia è un'enciclopedia online, multilingue, a contenuto libero, redatta in modo collaborativo da volontari e sostenuta dalla Wikimedia Foundation, un'organizzazione non-profit.

Attualmente è pubblicata in 250 lingue differenti (di cui circa 180 attive, quella in inglese è attualmente la più sviluppata) e contiene voci sia sugli argomenti propri di una tradizionale enciclopedia che su quelli di almanacchi, dizionari geografici e di attualità. Il suo scopo è quello di creare e distribuire una enciclopedia internazionale libera, nel maggior numero di lingue possibili. Wikipedia è uno dei dieci siti più popolari del web, ricevendo circa 60 milioni di accessi al giorno. [...]

Wikipedia contiene in totale più di 7 milioni di voci, 23 milioni di pagine, con 293 milioni di modifiche alle pagine e 7,3 milioni di utenti registrati. [...]

Wikipedia nacque il 15 gennaio 2001 come progetto complementare di Nupedia (un progetto con scopo analogo ma la cui redazione era affidata ad esperti). Essendo costantemente cresciuta in popolarità, ha dato vita a molti progetti simili come Wiktionary, Wikibooks e Wikinotizie.

La reputazione di Wikipedia come sistema di consultazione è stata soggetta a discussioni: è stata sia elogiata per la sua libera distribuzione, libera compilazione e ampia gamma di argomenti trattati sia criticata per presunti pregiudizi sistemici (per cui sono privilegiate le voci di interesse per gli stessi collaboratori indipendentemente dalla loro oggettiva importanza), lacune su alcuni argomenti e mancanza di responsabilità ed autorità in rapporto alle enciclopedie tradizionali. Le sue voci sono state citate dai mass media e da istituti accademici e sono disponibili sotto la licenza GFDL. (Estratto da Wikipedia)

25_ API è l'acronimo di Application Programming Interface e indica un insieme di procedure che vengono messe a disposizione dei programmatori per far comunicare diversi programmi o per interfacciare livelli di protocollo diversi. Per un'analisi più approfondita su questo argomento, si veda il capitolo 2.3.6.

26_ Sistema di ricerca e localizzazione di contenuti d'interesse su mappe geografiche. Per una descrizione più dettagliata, si faccia riferimento all'allegato 1 (cap 6.4).

27_ Meyrowitz J., *Oltre il senso del luogo. L'impatto dei media elettronici sul comportamento sociale*, Bologna, Baskerville, 1995.



2.3 ASPETTI TECNOLOGICI E ARCHITETTURALI: LE TECNOLOGIE ABILITANO LE FUNZIONI SOCIALI DEL SOFTWARE

Nel quadro del Web 2.0 le tecnologie rappresentano un fattore abilitante consentendo lo sviluppo e l'utilizzo delle nuove applicazioni in modo più semplice e immediato. L'evoluzione si concentra sullo sviluppo di interfacce usabili e sempre più simili a quelle dei programmi desktop, sull'utilizzo innovativo di tecnologie come AJAX ed RSS e sull'impiego massiccio di API di diversa provenienza nello sviluppo di applicativi ibridi e caratterizzati da tempi di mercato estremamente ridotti.

2.3.1 Il valore dell'interfaccia

Il Web 2.0 si può considerare per molti aspetti un cambiamento di paradigma nel modo di intendere le finalità e le modalità di interazione con il Web stesso.

In particolare, si è osservato come il Web stia evolvendo da uno stato di semplice contenitore di informazioni, sostanzialmente “statiche”, a quello di ambiente fortemente dinamico, i cui contenuti tendono a ibridarsi e a mutare continuamente. A questo proposito Steven Johnson ha contrapposto alla metafora della “libreria”, che rappresenta l'approccio 1.0, quella dell'“ecosistema”, evidenziando il modo in cui i dati circolano e alimentano i processi²⁸.

Nel definire tale ambiente dinamico, giocano un ruolo fondamentale le numerose applicazioni che sono state sviluppate avvantaggiandosi delle possibilità di interazione con l'utente offerte dalla tecnologia AJAX²⁹: il risultato è che sono attualmente disponibili *online* strumenti che svolgono funzioni che erano prerogativa delle applicazioni di tipo desktop.

Un'evidente linea evolutiva sul Web è la progressiva liberazione di Internet dai limiti del linguaggio HTML. Al di là delle difficoltà grafiche poste dall'adozione di un linguaggio di mark-up, i veri limiti dell'HTML e in generale del Web 1.0 sono riconducibili all'interazione.

Una pagina Internet tradizionale, infatti, non riconosce alcun movimento dell'utente al di là del click del pulsante sinistro del suo mouse. Questo limita fortemente le capacità di interazione uomo-macchina e, di conseguenza, uomo-contenuti, portando gli sviluppatori ad adottare soluzioni complesse (e, se mal progettate, complicate) per affrontare problemi altrimenti semplici. Al Web tradizionale, insomma, manca l'intuitività d'utilizzo che caratterizza le interfacce grafiche dei sistemi

operativi e questo si traduce in una minore facilità d'uso per gli utenti, che devono confrontarsi con modalità di interazione spesso poco naturali. Grazie alla diffusione di tecnologie come AJAX è finalmente possibile garantire la presenza e il funzionamento sul Web di funzionalità software-like, cioè gestibili con interfacce più intuitive e immediate, simili a quelle che caratterizzano il software offline.

L'integrazione di elementi di AJAX sulle pagine Web permette, quindi, di compiere operazioni precedentemente impossibili, come per esempio trascinare oggetti con il mouse, aggiornare parti di una pagina senza ricaricarla interamente, gestire uno scorrimento continuo, seamless, di una lista, ecc.

Ridimensionare un'immagine online, quindi, con AJAX non è più un'operazione che prevede innumerevoli tentativi manuali, specificando in appositi box le nuove dimensioni desiderate, ma è un'intuitiva operazione grafica di trascinamento dei suoi bordi, come in un qualsiasi software grafico. Un altro grande ambito di rinnovamento ed evoluzione è la sempre maggiore intuitività e “pulizia” delle interfacce. La *user experience* online inizia ad essere valutata in fase progettuale e la navigazione, l'accesso ai servizi e il loro utilizzo diventano sempre più facili e intuitivi.

2.3.2 Il modello GUI (Graphical User Interface)

In questo processo evolutivo è importante sottolineare come le interfacce³⁰ delle principali applicazioni Web 2.0 tendono ad avvicinarsi al modello GUI (*Graphical User Interface*), abbandonando parzialmente le caratteristiche tipiche delle prime interfacce Web (WUI - *Web User Interface*) per offrire agli utenti un ambiente di lavoro più simile a quello delle tradizionali applicazioni di tipo desktop.

Il modello di sviluppo basato sulle GUI permette agli utenti di interagire con una certa applicazione a partire da un'interfaccia grafica, che offre una qualche rappresentazione sia degli oggetti su cui è possibile agire, sia delle funzioni disponibili. Più nello specifico, l'interfaccia dei moderni sistemi operativi si basa sulla metafora di un piano di lavoro, sul quale trovano posto le rappresentazioni dei file (icone) e delle applicazioni (finestre). Tale modello si contrappone nettamente a quello basato su interfacce testuali (CLI - *Command Line Interface*), in cui il peso dell'interazione è sostenuto principalmente dall'utente, che deve imparare una serie di comandi (da impartire attraverso la tastiera) per poter fruire efficacemente l'applicazione stessa.

28_ Nella formulazione originale: “Data circulate like nutrients in a rain forest”. [<http://www.discover.com/issues/oct-05/departments/emerging-technology>]

29_ Ajax è una tecnica di sviluppo per la creazione di applicazioni web interattive. Per un'analisi più approfondita su questo argomento, si veda nei capitoli 2.3.4, 4.1.3 e l'allegato 2.

30_ Nei domini dell'informatica e dell'interazioni uomo-macchina, si definisce interfaccia quella porzione di un artefatto attraverso la quale un utente può interagire con l'artefatto stesso; in particolare, un'interfaccia permette all'utente di governare l'artefatto, comunicargli le proprie intenzioni o comandi e ricevere informazioni circa i risultati delle azioni intraprese. Per essere effettivamente usabile, un'interfaccia deve evidenziare chiaramente quali azioni si possono e quali non si possono compiere, come è necessario procedere per ottenere il risultato desiderato.



Le interfacce di tipo GUI si basano sul cosiddetto modello di interazione “WIMP” (windows, icons, menus and pointers). In particolare, sono caratterizzate dai seguenti elementi³¹:

- **Puntatore:** si tratta di un simbolo che compare sullo schermo e che l'utente può muovere, utilizzando un dispositivo di puntamento. L'utente può selezionare oggetti o comandi posizionando il puntatore su di essi e fornendo un input (ad esempio la pressione di un tasto) tramite il dispositivo di puntamento. Tipicamente, un puntatore ha la forma di una freccia (dispositivo di selezione), di una “i” maiuscola (*i-beam pointer*, utilizzato soprattutto negli editor di testo per evidenziare sezioni o per definire il punto di immissione del testo) oppure di una mano.
- **Strumento di puntamento:** uno strumento che permette di governare il puntatore e, attraverso di esso, interagire con gli oggetti grafici che compaiono sullo schermo. Esempi di strumenti di puntamento sono il mouse e la trackball.
- **Icone:** Sono piccole immagini che compaiono sullo schermo e che possono rappresentare comandi, file o finestre. Le icone permettono due modalità di interazione: una riferita agli elementi che rappresentano e una rivolta alle rappresentazioni di per sé. Nel primo caso, selezionando una certa icona e inviando un segnale di input, ad esempio attraverso un click con il mouse, si possono eseguire comandi specifici o trasformare le icone in finestre. Nel secondo caso, le azioni dell'utente (selezione, trascinamento...) sono finalizzate a modificare l'icona stessa, ad esempio ridimensionandola o spostandola sul piano di lavoro, senza produrre effetti sull'oggetto rappresentato.
- **Desktop:** è il nome che solitamente identifica l'area dello schermo su cui sono visualizzate le icone, a partire dalla metafora secondo cui tale elemento è paragonabile ad un piano di lavoro (*desktop*, appunto) reale, sul quale l'utente può disporre degli oggetti reali (le icone, che rappresentano documenti e strumenti di lavoro).
- **Finestre:** sono le varie porzioni in cui l'utente può suddividere lo schermo. Ciascuna di esse può rappresentare un documento o eseguire un programma diverso. Come le icone, le finestre permettono all'utente di eseguire anche azioni su sé stesse: in particolare, questi può spostarle e modificarne forma e dimensione.
- **Menu:** si tratta di liste delle opzioni e dei comandi tra cui l'utente può scegliere per interagire con una certa applicazione. Tipicamente, l'utente si limita a selezionare con il puntatore l'opzione desiderata e a fornire un segnale di input (pressione del tasto “invio”, click del mouse). I programmi basati su menù si contrappongono nettamente ai sistemi in

cui all'utente è richiesto di digitare il nome del comando che vuole eseguire per la loro facilità d'uso e di apprendimento, benché non siano altrettanto flessibili.

Le GUI offrono inoltre formati standard per rappresentare le componenti testuali e grafiche e si caratterizzano pertanto per la facilità con cui è possibile trasferire dati tra applicazioni diverse, a patto che queste condividano la medesima GUI: ad esempio, diventa possibile copiare un grafico creato con un foglio elettronico in un documento creato con un editor di testo.

Le interfacce basate sul modello WUI condividono con le GUI il principio di affidare l'interazione con l'utente a componenti visuali, oltre che l'adozione della coppia puntatore/dispositivo di puntamento come principale mezzo di input. Tuttavia, il grado di interattività consentito da un'interfaccia di tipo WUI è significativamente ridotto rispetto alle possibilità offerte dalle GUI. Le WUI, infatti, si basano sul modello dell'ipertesto, che è adatto soprattutto per presentare informazioni che saranno semplicemente fruite dall'utente, senza che questi possa intervenire modificando i contenuti proposti o inserendone di propri. Tipicamente, infatti, le WUI non offrono un supporto, paragonabile a quello messo a disposizione dalle GUI tramite le varie icone, per intervenire attivamente sui contenuti.

La caratteristica fondante degli ipertesti è piuttosto rappresentata dalla possibilità di accedere al materiale offerto in modo non sequenziale, ma seguendo percorsi selezionati di volta in volta dall'utente a partire dai collegamenti ipertestuali (*link*) predisposti dagli autori di un certo documento. Per quanto riguarda un confronto tra le interfacce GUI e quelle WUI, è indispensabile sottolineare come entrambe si avvantaggino delle possibilità offerte da una modalità di presentazione grafica. Esse consentono infatti anche agli utenti che non conoscono in modo approfondito il funzionamento del computer di portare a termine con successo le operazioni desiderate, dimostrando generalmente di possedere un buon livello di usabilità. In particolare, entrambi i tipi di interfaccia esaminati sfruttano i seguenti principi del design usabile definiti da Norman³²:

- **Visibilità:** la modalità di presentazione grafica permette di rendere immediatamente visibile ciò che si può fare con un certo artefatto (i comandi disponibili sono quelli presentati) e offre all'utente un “supporto mnemonico” – non è infatti necessario imparare i nomi dei comandi, perché questi vengono “ricordati” dall'interfaccia stessa. L'aspetto di un certo componente può renderne evidente la funzione (ad esempio, l'icona a forma di carrello della spesa si collega naturalmente alla funzione di acquisto).

³¹_ Webopedia [http://www.webopedia.com/TERM/G/Graphical_User_Interface_GUI.html]

³²_ Norman D. A. , La caffettiera del masochista, Firenze, Giunti, 2005



- **Inviti e vincoli all'uso:** gli oggetti grafici possono rendere immediatamente percepibili le proprie modalità di utilizzo: in questo senso, un link o un bottone è un invito a cliccare che favorisce la corretta modalità di accesso ad un certo contenuto; elementi parzialmente nascosti (ad esempio visualizzati in grigio) fanno capire all'utente che non è possibile applicare le funzioni corrispondenti nel contesto di lavoro corrente.
- **Mapping:** un buon mapping si basa su un'associazione esplicita tra i possibili comandi, il modo in cui devono essere eseguiti e i risultati che ne derivano. In tal senso, un'interfaccia grafica costituisce di per sé una soluzione, in quanto permette all'utente di agire direttamente, se non proprio sull'oggetto che vuole modificare, su una sua rappresentazione. Inoltre, l'azione da eseguire è talvolta analoga a quella che si eseguirebbe su un oggetto reale: ad esempio, nel caso in cui l'utente voglia spostare un'icona in un punto più visibile, seleziona l'icona stessa e la trascina dove desidera.

Entrambe le interfacce favoriscono inoltre un approccio di tipo esplorativo che può facilitare l'apprendimento. Al contrario, le WUI si distinguono dalle GUI perché non consentono un elevato livello di interattività; in particolare, non offrono all'utente un supporto specifico nell'inserimento e nella modifica dei contenuti.

La tendenza osservata a questo proposito nel Web 2.0 mostra che le applicazioni fruibili online presentano caratteristiche simili a quelle delle tradizionali applicazioni di tipo "desktop", per quanto riguarda sia le funzionalità offerte, sia il tipo di interfaccia, in una sorta di circolo virtuoso abilitato dalle tecnologie AJAX.

In particolare, molte delle applicazioni più significative dell'approccio "2.0" adottano quegli elementi tipici delle GUI che permettono di gestire efficacemente l'interazione con l'utente, quali la metafora del piano di lavoro, le icone che attivano comandi, i menù.

Alcuni esempi di applicazioni Web 2.0 che integrano elementi del modello delle GUI sono:

- **Pandora:** menù con liste di comandi e bottoni permettono di accedere al proprio profilo, aggiungere musica alla propria stazione radio, aggiungere altre stazioni, fornire un feedback sulla musica proposta, avviare o sospendere la riproduzione di un brano;
- **Ta-da List:** applicativo web che consente di creare e condividere le liste delle "cose da fare" (To-do list). Ta-da list permette inoltre agli utenti di gestire e monitorare le proprie liste attraverso l'uso dei feed RSS (<http://www.tadalist.com>). I comandi principali, che permettono di creare nuove liste, cancellarle, aggiungere elementi, sono accessibili tramite botto-

ni. Gli elementi di una certa lista vengono riconosciuti come "completati" semplicemente cliccando sulla casella che li affianca come se l'utente li spuntasse. Una serie di bottoni contrassegnati da frecce permette di riordinare le liste, spostando gli elementi che le compongono;

- **Flickr:** l'utente può intervenire in vari modi sulle fotografie pubblicate, ad esempio ruotandole o annotandole, attraverso una serie di bottoni. Inoltre, l'interazione è intuitiva e permette all'utente di ottenere i risultati desiderati intervenendo sulle rappresentazioni grafiche degli oggetti: ad esempio, può posizionare una nota in un punto particolare della foto trascinando il campo di testo corrispondente dove desidera, proprio come se stesse utilizzando un normale post-it. Funzioni generali come "save", "cancel" e "delete" sono attivabili tramite bottoni accessibili in modo contestuale. (Ad esempio, i bottoni vengono visualizzati sulla foto, accanto al campo di testo, quando l'utente deve manipolare le proprie note);
- **Writely:** utilizza icone, menù a tendina, a cascata, pop up, e finestre in tutto simili a quelli di un editor di testo tradizionale;
- **Netvibes³³:** riprende esplicitamente la metafora del piano di lavoro, in cui vengono disposti i diversi oggetti in cui l'utente può intervenire. Ciascun contenuto è rappresentato da una finestra, che può essere spostata semplicemente trascinandola. Vengono inoltre utilizzati dei menù a tendina (per mostrare all'utente la lista dei contenuti tra cui può scegliere), vari bottoni per intervenire sulle finestrelle e sul loro contenuto (ad esempio, per cambiare il colore della finestra, per cancellare o visualizzare un certo appunto o per salvare le modifiche apportate); ci sono inoltre aree direttamente editabili in cui inserire le proprie note.

La facilità d'uso di un servizio fin dal primo accesso è diventata un parametro di valutazione della qualità del servizio stesso. Poiché gran parte dei nuovi servizi online, che, ricordiamo, molto spesso sono personalizzati a seconda delle esigenze dell'utente, prevede il riconoscimento di chi li utilizza: la facilità delle procedure di registrazione e login è diventata prioritaria.

L'esempio di Google è estremamente virtuoso: l'azienda, infatti, ha esplicitamente previsto per i propri servizi un sistema di registrazione di nuovi utenti leggero e intuitivo, che non conta più di 3 step e consente l'utilizzo immediato dei servizi in pochi minuti. Parallelamente, tutti i servizi di Google sono accessibili attraverso un login unificato: basta accedere a Google Accounts ed è possibile utilizzare tutte le applicazioni Google online senza doversi far riconoscere ulteriormente.

³³ Applicazione web che, a partire dall'aggregazione di diversi feed, consente di creare la propria «start page» personalizzata. Per una descrizione più dettagliata, si faccia riferimento all'allegato 1 (cap 6.2).



Allargando la prospettiva, sono evidenziabili alcune tendenze di progettazione e design delle interfacce che accomunano molti servizi Web 2.0:

- **Essenzialità grafica:** la lezione di Google, che col suo motore di ricerca “nudo e crudo” ha di fatto dato un esempio virtuoso di design austero, leggero e di grande efficacia online, ha portato gran parte dei servizi di nuova concezione ad adottare una filosofia simile: grafica funzionale, leggera e non gravata da orpelli inutili. Il focus della progettazione e del design delle interfacce, quindi, è prevalentemente sulle funzionalità e sui contenuti.
- **Interfacce più intuitive:** grazie a tecnologie come AJAX l'interazione uomo-Web è più immediata ed è realizzabile attraverso un approccio naturale in cui l'utente interagisce liberamente con gli elementi grafici, senza le limitazioni imposte dall'HTML.
- **Registrazione/login veloci, con un focus sulla soddisfazione immediata dell'utente:** i servizi online tendono sempre più a ridurre il tempo che passa tra l'approccio di un utente a un servizio e la sua soddisfazione. L'idea del “3 clicks and you're in” è stata estesa anche all'accesso a una nuova iscrizione, mentre l'accesso a servizi a cui si è già registrati solitamente prevede un solo click. Molti servizi online, ad esempio quelli offerti da Google, consentono un accesso e un utilizzo immediati: l'immissione di dati aggiuntivi, la creazione di profili-utente ecc., tutte operazioni che in precedenza rendevano lunghissime le procedure di registrazione a un servizio online, sono opzioni secondarie che l'utente può compiere in un secondo momento.

2.3.3 Usabilità

La centralità del ruolo dell'utente nel Web 2.0 si manifesta con una corrispondente evoluzione del design verso l'usabilità: gran parte delle caratteristiche visuali e di interfaccia evidenziate come tipiche delle applicazioni Web 2.0 si giustifica in quanto garantisce una maggiore usabilità. La norma ISO 9241-11 (*Ergonomic requirements for office work with visual display terminals - Guidance on usability*) definisce l'usabilità come l'efficacia, l'efficienza e la soddisfazione con cui un utente esegue un compito tramite un certo artefatto.

L'usabilità abbraccia anche i concetti di facilità nell'apprendimento e nel ricordo³⁴ e richiede che gli artefatti siano progettati tenendo conto dei reali bisogni degli utenti, rispetto a specifiche attività, svolte in particolari contesti³⁵. Nell'analisi svolta si sono considerati come criteri di valutazione i principi del “buon design” eviden-

ziati da Norman, secondo cui un artefatto usabile deve fornire anche visibilità, un buon mapping tra i comandi, il loro azionamento e i risultati che ne derivano, inviti e vincoli all'uso e un buon modello concettuale³⁶.

Per quanto una valutazione di usabilità delle singole applicazioni Web 2.0 non rientri tra gli obiettivi di questo documento, si vogliono qui evidenziare alcune tendenze di progettazione che rivelano una crescita dell'attenzione per l'utente e i suoi bisogni.

Si segnala in primo luogo che le applicazioni più rappresentative dell'approccio Web 2.0 nascono proprio dalla “scoperta” di bisogni e problemi dell'utente ancora insoddisfatti e si propongono come possibili risposte. In altre parole, tali applicazioni aiutano gli utenti a raggiungere obiettivi propri e, pertanto, vengono percepite come “utili” costituendo così un “valore”. Proprio la possibilità di soddisfare un interesse personalmente rilevante costituisce una buona motivazione per utilizzare frequentemente una certa applicazione e accettare le sue richieste, anche quando queste richiedono un piccolo sforzo da parte dell'utente come nel caso in cui sia necessario fornire i propri dati e registrarsi.

Vengono qui di seguito proposti alcuni esempi rilevanti legati ai servizi di carattere innovativo in grado di avvicinarsi a bisogni specifici degli utenti.

- **Pandora:** permette di ascoltare musica di proprio gradimento senza bisogno di disporre di un lettore e di un supporto di memoria. Consente inoltre di scoprire nuovi brani e artisti in grado di avvicinarsi e soddisfare i gusti dell'utente, che si trova così nella possibilità di ascoltare la propria musica preferita da qualunque computer connesso alla rete;
- **Ta-da-list:** consente di organizzare in modo semplice le cose da fare, gestire le proprie liste di cose da fare riunirle su un unico supporto, accedervi da qualunque computer connesso alla rete, e condividerle con altre persone;
- **Flickr:** permette di memorizzare, organizzare, pubblicare e valorizzare le proprie foto. Attraverso un meccanismo di condivisione che consente di accedere a un contenitore di immagini ricco e aggiornato;
- **Writely:** consente la condivisione di documenti di lavoro, permettendo in primo luogo la stesura degli stessi in modalità sincrona e asincrona, senza bisogno quindi di compresenza fisica. Caratterizzato da un editor di testo familiare e ricco di funzioni, è accessibile da qualunque computer connesso;
- **Netvibes:** permette di accedere a molteplici contenuti di interesse da un unico strumento evoluzione della lavagnetta per appunti: posta, bookmarks, news, appunti, liste di cose da fare, motori di ricerca.

³⁴ Nielsen, J., Usability Engineering, Boston, AP Professional, 2003; Nielsen, J., Web Usability, Milano, Apogeo, 2000

³⁵ Rizzo A., Marti P., Bagnara S., “Interazione Uomo-Macchina”, in Burattini E., Cordeschi R., “Intelligenza Artificiale. Manuale per le Discipline della Comunicazione”, Roma, Carocci, 2001

³⁶ Norman D. A., op. cit. par. 2.3



Nel complesso, il fatto stesso di riprendere il modello di interazione basato su GUI, a cui l'utente è avvezzo, anche qualora non avesse molta esperienza in fatto di navigazione, rappresenta una garanzia di usabilità, in quanto si pone nella direzione della standardizzazione e dell'adozione di modelli di interazione ben consolidati. La familiarità dell'interfaccia permette infatti all'utente di evitare completamente (o di svolgere in modo molto veloce e senza sforzo) la fase di apprendimento, rendendosi immediatamente operativo. Come già accennato, la possibilità di manipolare direttamente le rappresentazioni degli oggetti su cui l'utente vuole intervenire aumenta le probabilità di assicurare, da una parte, una buona visibilità delle opzioni disponibili e, dall'altra, un mapping soddisfacente tra i comandi, gli oggetti su cui si applicano e i risultati che si possono ottenere. Ad esempio, nel caso l'utente voglia ridisporre i diversi elementi che vede sullo schermo, la modalità di interazione più "naturale" e intuitiva prevede proprio che gli oggetti da spostare vengano selezionati e trascinati nel punto dello schermo che si desidera.

L'uso di icone a cui sono associati i comandi che l'utente può eseguire permette di assicurare la visibilità e la riconoscibilità delle funzioni, offrendo un supporto mnemonico che facilita sia l'apprendimento, sia il ricordo, senza richiedere grandi sforzi cognitivi da parte dell'utente. Inoltre, le icone, grazie alle loro proprietà grafiche, possono fornire orientamento o vincoli all'uso. Ad esempio, un bottone visualizzato come se fosse in rilievo suggerisce all'utente di cliccarlo piuttosto che di trascinarlo.

I menù condividono con le icone la capacità di garantire sia la visibilità, sia il ricordo delle funzioni disponibili. Inoltre, la possibilità di visualizzare i menù nella specifica porzione di schermo in cui l'utente sta lavorando (sia per scelta degli autori dell'applicazione, sia perché è l'utente stesso a poter spostare il menù, come nel caso di Flickr) permette di contestualizzare chiaramente l'interazione.

La semplicità nel layout permette infine di limitare il sovraccarico informativo e di non superare i limiti percettivi e attentivi dell'utente. Ogni pagina mostra relativamente pochi oggetti, disposti in modo lineare. Viene così garantita un'adeguata visibilità a tutti gli elementi presentati, di modo che l'utente possa rendersi conto facilmente delle funzioni disponibili e/o degli argomenti trattati. La lettura risulta inoltre più lineare, senza che ci siano troppi contenuti in concorrenza tra di loro. Nel caso in cui l'utente non stia semplicemente cercando informazioni, ma stia svolgendo un'operazione (ad esempio, se si sta registrando o se sta caricando delle foto su una pagina personale), la

semplicità del layout permette di supportarlo in modo molto preciso, svolgendo una funzione di guida passo per passo: l'utente riceve, di volta in volta, tutte e soltanto le informazioni necessarie per completare il sottotask che deve svolgere.

L'unione di colori tenui per lo sfondo con elementi attraenti (colori vivaci, icone, effetti tridimensionali) usati con parsimonia, solo per evidenziare i contenuti ritenuti più importanti, permette di guidare l'utente nella fruizione di una certa applicazione. Per l'utente, risulta infatti più facile individuare immediatamente gli elementi centrali: in tal modo, può valutare immediatamente se l'applicazione riveste qualche interesse e quali sono le principali funzioni offerte. Viene così minimizzata la possibilità che l'utente attribuisca per errore eccessiva rilevanza a elementi secondari, con il rischio di vedere deluse le proprie aspettative.

Parimenti, l'abbondante uso di spazi bianchi e di righe vuote per separare i vari elementi e i paragrafi di testo, oltre a rendere le applicazioni Web 2.0 in genere esteticamente gradevoli, ne facilita la fruizione, creando dei "blocchi logici" di contenuto. In particolare, la separazione dei paragrafi attraverso spazi bianchi rende il testo facilmente percorribile con lo sguardo, così da renderne più semplice e immediata la valutazione.

L'uso di caratteri di grandi dimensioni ha due ragioni: da un lato, facilita la lettura, anche nel caso di utenti con lievi disturbi visivi; dall'altro, se applicato solo a specifiche porzioni di testo, permette di evidenziarle, attirando su di esse l'attenzione del lettore e guidandolo nella valutazione del contenuto proposto. In tal modo, dovrebbe essere più facile determinarne l'argomento e il livello di interesse.

Analogamente, l'uso di campi di input sovradimensionati rispetto a quelli tradizionali facilita le operazioni di immissione dei dati³⁷, oltre a rendere più visibile l'input immesso e, quindi, più facili le correzioni nel caso di errori di digitazione.

2.3.4 AJAX e Piattaforme di pubblicazione (Ruby on Rail – Monorail)

Fin dalla definizione del termine da parte di O'Reilly, il Web 2.0 è stato associato alle tecnologie AJAX. Sebbene non tutte le applicazioni caratteristiche dell'approccio "2.0" utilizzino AJAX in modo significativo, tale tecnologia risulta effettivamente utile per garantire all'utente una maggiore interattività e per implementare efficacemente molti degli elementi che caratterizzano le interfacce "Web 2.0".

Più in particolare, Ajax (Asynchronous JavaScript and XML) è una tecnica di sviluppo per creare applicazioni Web interattive. Nelle applicazioni Web tradizionali, ogni

³⁷ Secondo la cosiddetta *legge di Fitts*, il tempo necessario per raggiungere un oggetto target, sia nel mondo reale, sia sullo schermo, è una funzione della distanza e delle dimensioni del target stesso. In particolare, il tempo aumenta al crescere della distanza e al diminuire delle dimensioni. Per un utente risulta quindi più facile raggiungere un oggetto di dimensioni maggiori, in quanto diminuisce la necessità di micro-movimenti di correzione finale, dopo un primo spostamento molto rapido, ma poco preciso.



interazione tra l'utente e la pagina implica una specifica richiesta al Web server, dopodiché la pagina viene caricata ex novo, anche se il cambiamento da effettuare è minimo. In tal modo, i tempi di risposta del servizio risultano relativamente lenti.

L'obiettivo delle tecnologie Ajax consiste nel rendere le pagine più "sensibili" agli input dell'utente permettendo a queste ultime di scambiare piccole quantità di dati con il server, invece di dover ricaricare un'intera pagina ogni volta che l'utente fa qualche richiesta. Infatti, le applicazioni Ajax vengono solitamente eseguite sulla macchina dell'utente (la pagina correntemente visualizzata viene manipolata all'interno del browser utilizzando i metodi del DOM) e le richieste e le risposte scambiate tra client e server coinvolgono quantità di dati relativamente piccole. Tale approccio dovrebbe apportare un miglioramento sia in termini di velocità, sia di usabilità.

Ajax utilizza:

- una combinazione di XHTML e CSS per formattare le informazioni;
- gli oggetti del DOM, manipolato tramite un linguaggio di scripting come Javascript, per permettere di interagire con l'informazione presentata;
- l'oggetto XMLHttpRequest³⁸ per scambiare dati in modo asincrono con il server;
- XML come formato per scambiare dati tra il server e il client.

Ajax può essere combinato con Ruby on Rails (ottenendo il cosiddetto sistema "Ajax on Rails") per rendere più agevole la programmazione. Ruby on Rails è un framework open source per applicazioni Web scritto in linguaggio Ruby e modellato sull'architettura Model-View-Controller (MVC)³⁹. Il suo interesse risiede nella capacità di abilitare lo sviluppo di applicazioni scrivendo relativamente poco codice rispetto ad altri framework e con necessità di configurazione minimale⁴⁰.

2.3.5 Formato dati: ATOM e RSS

RSS ed Atom sono due linguaggi basati su XML nati per la condivisione di informazione pubblicata su siti web con altre applicazioni. Tipicamente, un sito web può mettere a disposizione dei propri utenti un URL che, se inserita in appositi software, permette di ricevere i titoli e gli abstract degli ultimi contenuti pubblicati. Si tratta di tecnologie che sono inizialmente nate con scopi di syndication dei contenuti, ma che grazie alla loro popolarità si stanno diffondendo come *standard de facto* per la condivisione di informazioni sul web.

Rich Site Summary⁴¹ (RSS) è una tecnologia push sviluppata da Netscape. È possibile utilizzare RSS per creare canali (feed) che distribuiscono automaticamente le informazioni agli abbonati. RSS consente agli autori web di creare un link che i visitatori possono selezionare per ricevere un determinato canale.

Atom⁴² è un formato di file basato su XML nato per mettere a disposizione una lista di informazioni (feed) che viene trasmessa da un fornitore di informazione a coloro che hanno sottoscritto tale servizio. È un vocabolario molto simile a RSS per quanto concerne la strutturazione dei dati ma differente quanto a comandi utilizzati. Gli utenti finali possono raccogliere gli URL dei file RSS o Atom attraverso programmi detti feed-reader. Questi strumenti permettono di operare un'operazione di sincronizzazione simultanea di tutti i canali sottoscritti. Inoltre, attraverso una semplice trasformazione XSLT, ne danno una rappresentazione HTML, in modo tale che i contenuti possano essere consultati all'interno di un comune browser web.

L'utilizzo degli RSS, che sono molto diffusi nel mondo dei blog, di fatto mette a disposizione dei milioni di utenti del Web uno strumento facile di condivisione dei contenuti in modo del tutto indipendente dall'interfaccia grafica. L'informazione, quindi, in ambito Web 2.0 transita prevalentemente attraverso i feed RSS o attraverso il formato OPML, che li aggrega gerarchicamente. Per leggere un feed RSS o Atom è necessario avere a disposizione dei lettori *chiamati Feed Reader o aggregatori*. Questi strumenti si occupano dell'interpretazione della struttura del file in XML che caratterizza il feed convertendolo in un formato di più facile lettura. Esistono due diversi generi di lettori e aggregatori di feed: i desktop aggregator e i web aggregator. I primi sono veri e propri programmi che vengono installati direttamente sul device dell'utente e utilizzati per leggere, aggregare e mostrare i contenuti di un feed. Residenti direttamente sul device stesso, sono caratterizzati da un basso livello di portabilità ma da un'elevata accessibilità, dal momento che non richiedono per la consultazione una costante connessione alla rete. I web aggregator, al contrario, sono dei veri e propri servizi on-line capaci di integrare funzioni di social bookmarking e di social networking. Emerge infatti la tendenza ad intendere il reperimento e la catalogazione delle informazioni non più come un'attività individuali ma come un momento di associazione e aggregazione in grado di aumentare il sapere condiviso.

38_ Per una spiegazione più dettagliata, si faccia riferimento all'allegato 2 (cap. 7.2.1).

39_ Si tratta di un'architettura software caratterizzata da una chiara separazione tra il modello dei dati, l'interfaccia utente e la logica di controllo: queste componenti vengono infatti gestite separatamente affinché le modifiche effettuate su di uno specifico modulo abbiano il minimo impatto sull'architettura complessiva.

40_ Alcuni tra i principali applicativi che combinano la tecnologia AJAX con Ruby on Rails sono Basecamp (<http://www.basecamp.com>), Campfire (<http://www.campfirenow.com>), 43things (<http://www.43things.com>), ODEO (<http://www.odeo.com>), Shopify (<http://shopify.com>), Fluxiom (<http://www.fluxiom.com>), Strongspace (<http://www.strongspace.com>), Typo (<http://www.typo3.com>)

41_ <http://blogs.law.harvard.edu/tech/rss>

42_ <http://www.atomenabled.org/developers/syndication/atom-format-spec.php>



2.3.6 API: integrazione e interazione

API è l'acronimo di *Application Programming Interface* e indica un insieme di procedure che vengono messe a disposizione dei programmatori per far comunicare diversi programmi o per interfacciare livelli di protocollo diversi. Le API possono anche interfacciare linguaggi di alto livello con utility di livello più basso: applicando un livello di astrazione al livello sottostante evitano così al programmatore di gestire le operazioni ad un livello più basso.

Esistono molti design model per le API. Le interfacce che consentono la massima velocità di esecuzione sono spesso basate su di un set particolare di funzioni, procedure, variabili e strutture dati. Inoltre sono a disposizione altri modelli come ad esempio gli interpreti applicati nella valutazione delle espressioni come con ECMAScript/Javascript.

Per quanto riguarda la pubblicazione delle API esistono due atteggiamenti dominanti. Le aziende che non traggono profitto dalla vendita dell'implementazione delle API tendenzialmente le conservano gelosamente, fornendole solo a partner commerciali specifici. Altre compagnie distribuiscono invece le proprie API pubblicamente, e spesso il profitto non deriva dalla vendita delle stesse quanto dal loro utilizzo per lo sviluppo di nuovi applicativi, detti *mash-up*.

Ed è proprio questa operazione che sta rivoluzionando profondamente il mondo del web, dando vita ad applicazioni innovative spesso definite *2.0*. Questo processo, permesso dalla liberalizzazione delle API, per molti versi viene considerato come il movimento che ha permesso agli utenti di passare da uno stato di "passività" ad un nuovo modello che prevede un alto grado di partecipazione e interazione non solo per quanto riguarda la produzione di contenuti ma anche per la realizzazione e personalizzazione dei servizi.

2.3.7 MASH-UP: modularità e interoperabilità tra servizi

Mashup, termine derivato dal mondo della musica (che designa canzoni composte a partire da svariate parti di altri pezzi) indica la fusione di diverse fonti informative, fatta con la finalità di creare nuovi servizi partendo da contenuti e informazioni già presenti on line.

Uno degli elementi caratterizzanti del Web 2.0 è proprio la tendenza a favorire l'interoperabilità tra servizi diversi. L'idea alla base di questo processo è considerare la Rete come un universo "conversante" in cui non solo le persone, ma anche i software sono in grado di comunicare reciprocamente condividendo dati e funzionalità.

Per rendere possibile l'interoperabilità tra servizi online di diversa natura sono fondamentali alcuni requisiti:

- la presenza di un formato comune di interscambio di dati;
- la disponibilità tecnologica dei singoli servizi online ad essere "ibridati" con altri.

Nel corso degli anni il formato XML si è affermato (non solo nel Web 2.0) come uno *standard de facto* per la condivisione di dati. La sua struttura modulare completamente aperta e autodescrivente lo rende, estremamente adattabile alle diverse applicazioni, pur consentendogli una compatibilità universale. Tra le tecnologie più interessanti che utilizzano l'XML è necessario segnalare i feed RSS (Really Simple Syndication), una modalità con cui è possibile condividere contenuti testuali e multimediali tra applicazioni.

L'altro grande elemento di apertura del Web 2.0, cioè la **disponibilità all'ibridazione**, parte da un presupposto: considerare migliorabile, diversamente sviluppabile e modificabile liberamente ogni singolo servizio. Questa visione deriva in gran parte dalla filosofia alla base dell'universo opensource, in cui la comunità di sviluppatori ha libero accesso ai codici sorgente del software e ha facoltà di modificarli liberamente.

Nel Web 2.0 coesistono servizi opensource e tradizionali, per questo motivo alla libertà di codice dell'opensource è stata preferita la diffusione di API (*Application Program Interface*), cioè di software che permettono agli sviluppatori di integrare nei propri programmi le funzionalità di alcuni servizi online. La diffusione pubblica delle API di molti servizi Web 2.0 rende possibile la costruzione di nuovi software e nuovi servizi che ne sfruttino le funzionalità, aggiungendone di nuove. In questo modo si crea una vera e propria disseminazione del software, che esce dalla logica della singolarità e si moltiplica.

Uno degli esempi più virtuosi di utilizzo delle API pubbliche di un servizio Web 2.0 è dato da Flickr. Il celebre servizio di photoblogging, infatti, ha tratto beneficio dalla diffusione delle proprie API presso la comunità degli sviluppatori. In pochi mesi sono nati numerosi servizi basati su questo sistema di photoblogging, che ne hanno aumentato notevolmente la popolarità e, soprattutto, hanno fornito agli utenti modalità di utilizzo innovative. Per esempio sono nati numerosi servizi che permettono di utilizzare Flickr con interfacce diverse, ma sono disponibili anche servizi che consentono – grazie all'ibridazione con un software esterno dotato di un algoritmo di identificazione – di trovare fotografie simili tra loro, fino a servizi avanzati in cui è possibile posizionare sulle mappe di Yahoo (utilizzabili anch'esse attraverso API) le fotografie geolocalizzate presenti su Flickr, in modo tale da consentire agli utenti una consultazione "geografica"⁴³.

L'esempio di ibridazione tra Flickr e Yahoo segnalato in precedenza è un caso di mash-up, cioè un software che combina elementi provenienti da sorgenti diverse creando qualcosa di nuovo, nello specifico un servizio geografico di consultazione fotografica, molto utile per chi intende intraprendere un viaggio o conoscere più da vicino una località.

43. Per un elenco costantemente aggiornato di applicazioni e servizi online che utilizzano Flickr, consultare <http://www.flickrbits.com>



Nel Web 2.0 è palese la tendenza a diffondere sempre più la cultura del mash-up, consentita dalla diffusione delle API dei singoli servizi. La combinazione tra software e servizi diversi, infatti, in molti casi garantisce la creazione da parte di terzi di servizi nuovi che beneficiano delle funzionalità dei due sistemi ibridati e, unendole, ne creano di nuove.

Oltre all'apertura garantita dalle API, uno degli elementi che favorisce la diffusione di mash-up è la **modularità dei servizi Web 2.0**. Molte applicazioni online, infatti, consentono l'allargamento del loro raggio d'azione operativo attraverso l'utilizzo di plugin, cioè di software aggiuntivi che forniscono nuove funzionalità ai servizi.

La cultura del plugin, già ampiamente diffusa nell'ambito del software tradizionale, in particolare nell'ambito della produzione grafica e multimediale, è parte fondante di molte applicazioni Web 2.0 che, grazie all'apertura nei confronti di plugin sviluppati da terzi, possono essere facilmente personalizzabili, adattabili alle esigenze dell'utente e rese facilmente interoperabili.

Per esempio un sistema di content management (oltre che una delle piattaforme di creazione/gestione di blog più diffuse al mondo) come Wordpress, deve gran parte del suo successo alla presenza di centinaia di plugin di vario genere, in grado di migliorarne le prestazioni, aggiungere nuove feature e interfacciare i contenuti con altri servizi Web 2.0.

Grazie ai plugin, per esempio, è possibile interfacciare Wordpress con Flickr, consentendo di pubblicare sul proprio blog le fotografie ospitate da quest'ultimo, o apporre i tag di Technorati in calce a ciascun post, consentendone una più facile indicizzazione e ricerca.

Per gli utenti la diffusione di una cultura aperta a terzi nello sviluppo evolutivo del software, contrapposta ad una cultura chiusa e "intimidita" dalle questioni di copyright tipica del software tradizionale offline, significa maggiore libertà di scelta, più varietà di software e servizi e, soprattutto, la possibilità di poter personalizzare i servizi.

2.3.8 Time to market: la versione beta

Lo scenario altamente competitivo che si è creato tra gli sviluppatori ha avuto un effetto considerevole sulle modalità di realizzazione di servizi Web 2.0, riducendone in maniera sostanziale i tempi. Il cosiddetto *time-to-market* – cioè il tempo che intercorre tra l'ideazione di un prodotto e la sua diffusione presso il pubblico – nel caso delle applicazioni Web 2.0 risulta estremamente ridotto.

Le cause di questa contrazione dei tempi sono sostanzialmente due:

- La riduzione del tempo permette agli sviluppatori di "arrivare prima" a un risultato, superando i prodotti concorrenti che sono ancora in fase di sviluppo. Trattandosi di applicazioni di massa in cui spesso il valore è portato dai contenuti degli utenti (motivo per cui a una maggior partecipazione corrisponde un maggior valore), spesso diffondere un prodotto

prima della concorrenza significa conquistare la maggior parte del pubblico e mantenerlo anche nel caso in cui i prodotti giunti successivamente siano migliori. Si tratta di un classico caso di rendita di posizione.

- La diffusione di applicazioni non del tutto complete o in fase di *beta testing* permette di ridurre enormemente i costi di sviluppo: saranno infatti gli utenti stessi a evidenziare pregi e difetti dei servizi in modo totalmente gratuito. Uno sguardo di panoramica sul mondo Web 2.0 rivela il proliferare di applicazioni in fase di beta o addirittura di pre-release. L'effetto non è solo imputabile al fatto che gran parte di questi servizi sono sviluppati da softwarehouse di piccole dimensioni, spesso impossibilitate a sostenere i costi di sviluppo e fine-tuning di applicazioni complesse, concepite per un uso di massa, ma sembra essere divenuto un trend operativo che ha coinvolto perfino giganti della Rete. Google, per esempio, ha preferito rilasciare sul mercato numerose applicazioni sociali con funzionalità minime, impegnandosi a fornire un costante sviluppo incrementale, in grado di integrare nuove features, risolvere bug critici e aumentare in genere le potenzialità del servizio offerto.

Nella valutazione di un'applicazione Web 2.0 è bene tenere conto del suo *status di release* e della *roadmap di sviluppo*, elementi critici per valutarne in prospettiva la qualità – anche in termini di stabilità – dei servizi offerti.

3. SOCIAL NETWORK: APERTURA ALL'UTENTE E ALLE SUE RELAZIONI



3. SOCIAL NETWORK: APERTURA ALL'UTENTE E ALLE SUE RELAZIONI



Dopo anni di pionierismo, seguiti da anni di applicazione di modelli appartenenti ad altri media e rivelatisi poco efficaci (il modello dell'advertising televisivo della pubblicità tradizionale online, il modello aziendalista puro della New Economy, ecc.), il Web comincia ad auto-produrre i propri modelli evolutivi.

Finalmente il Web non è più il derivato di altri mezzi di comunicazione di massa. Liberata da modelli evolutivi non propri, Internet è cresciuta in modo totalmente opposto alla televisione, che fino a qualche anno fa sembrava esserne la sorella maggiore.

Se nel mondo catodico regnano indisturbate le concentrazioni di potere e il tradizionale modello del broadcasting in cui pochi autori comunicano ad una massa enorme di pubblico sostanzialmente "muto", nel Web accade esattamente il contrario: la comunicazione diventa multi-a-molti.

Si tratta di uno scenario nuovo, in cui gli elementi "sociali" di un servizio online sono caratterizzanti tanto quanto quelli progettuali o tecnologici. Tenere conto della dimensione relazionale del nuovo Web e renderla proficua significa adottare strumenti innovativi per l'ideazione, progettazione e sviluppo di servizi su Internet. La nuova identità del Web, in coerenza con l'idea di Internet che si fecero Tim Berners-Lee e Robert Cailliau all'atto della sua invenzione, più di quanto non lo fosse la Rete al tempo dei "portali", è un territorio in gran parte da esplorare.

Siamo di fronte ad uno scenario nuovamente pionieristico e in via di maturazione, in cui un numero sempre crescente di prime-movers intuisce i naturali movimenti evolutivi del Web e li interpreta, traducendoli in servizi online e contemporaneamente sperimentando linguaggi, inventando strumenti e cercando di definire standard e best practices. Trattandosi di uno scenario alle prime fasi della sua evoluzione, al Web 2.0 manca una formalizzazione, una descrizione dello "stato dell'arte". In parte la comunità di sviluppatori, attraverso incontri, conferenze, seminari, sta cercando di stabilire pratiche comuni, standard e strumenti per l'interoperabilità dei servizi. Allo stesso tempo, gli utenti si auto-organizzano, sviluppano autonomamente gli aspetti sociali del nuovo Web e – più o meno consapevol-

mente a seconda dei casi – contribuiscono all'evoluzione della Rete, di cui finalmente sono parzialmente "autori".

Secondo la linguistica di Ferdinand De Saussure la "langue" è la componente sociale del linguaggio – termini, sintassi, morfologia –; per "parole" s'intende, invece, l'aspetto fonatorio individuale.

L'auspicio è che dalla lettura di questo documento, che di fatto cerca di evincere una *parole* applicativa da una *langue* estremamente intuitiva, dinamica, auto-organizzata

Tavola 5



SIX DEGREES OF SEPARATION

L'espressione "sei gradi di separazione" ("six degrees of separation" nella formulazione originale) si riferisce all'idea che ciascun individuo sia collegato a qualunque altra persona da un numero di passaggi mai superiore a sei, intendendo con "passaggio" una relazione di conoscenza diretta. Lo studio più conosciuto a questo riguardo, noto come "small world experiment", è stato condotto da Milgram nel 1967. In quell'occasione, lo psicologo americano inviò 60 lettere ad altrettanti volontari residenti in Nebraska, chiedendo loro di farle pervenire ad un agente di cambio nel Massachusetts effettuando tutte le consegne a mano, coinvolgendo nell'esperimento altre persone che avrebbero potuto raggiungere, direttamente o per mezzo di ulteriori conoscenti, il destinatario della lettera. L'esperimento di Milgram, pur non permettendo di individuare un numero costante, contribuì a consolidare la teoria secondo cui tutti gli individui sono collegati attraverso un numero molto limitato di passaggi intermedi. [...]

Nel 2001 Duncan Watts, un professore della Columbia University, cercò di ricreare l'esperimento di Milgram. Sfruttando Internet utilizzò un messaggio e-mail che raggiunse 48.000 persone in 157 paesi. Nel corso dell'esperimento Watts scoprì che il numero di intermediari era all'incirca pari a 6. Questa curiosa scoperta diede al mondo intero un'idea più precisa della natura e delle potenzialità della rete.

[...]

Nel gennaio del 2007 l'attore Kevin Bacon ha lanciato SixDegrees.org [<http://www.sixdegrees.org/>] un sito web che si basa sull'idea che "il mondo è piccolo" (*small world phenomenon*) per creare un social network di beneficenza e stimolare la partecipazione attiva a sostegno di varie cause e progetti.

(estratto da Wikipedia)

e non controllata, siano emersi elementi sufficienti a progettare e – nel caso di AJAX – direttamente sviluppare servizi innovativi online. L'opera di descrizione di uno scenario evolutivo e dinamico, tuttavia, evidenzia un limite congenito: la sua stretta natura temporale. Gli scenari presentati, infatti, sono in continuo mutamento e quotidianamente nascono servizi e prodotti innovativi che, non possono essere compresi e analizzati se non in epoche successive.

Tuttavia l'approccio adottato, cioè la scelta di analizzare i modelli applicativi del Web 2.0 in base alle loro tendenze evolutive e non alla descrizione di singoli casi fissati nel *hic et nunc*, fa sì che nelle linee guida siano evidenziabili alcuni scenari futuri che, combinati con l'intuizione e, a volte, la capacità di sperimentare degli sviluppatori, possono contribuire a creare servizi online.

3.1 UNA RETE DI IDENTITÀ

Per poter implementare funzioni di social networking in servizi online è necessario tenere conto di un requisito: consentire agli utenti di disporre di un'identità precisa online.

L'interazione tra soggetti, sia essa collaborativa su lungo periodo o puramente occasionale, acquisisce una maggiore efficacia quanto più i partecipanti sono definiti come soggetti dotati di un'identità personale, cioè di un profilo distinguibile. Il requisito, quindi, consiste nel far socializzare identità e non semplici "nomi" online. La socialità, infatti, nasce dalla constatazione di esperienze, argomenti e tratti comuni: se questi non sono specificabili, è difficile che si sviluppino conversazioni e condivisioni tra utenti.

Gran parte dei servizi Web 2.0 consente agli utenti di creare con notevole livello di dettaglio veri e propri profili personalizzati condivisi, in cui specificare ogni elemento caratterizzante la propria identità digitale.

Servizi come Flickr o MySpace⁴⁴ permettono di profilare i dati anagrafici, le preferenze e i gusti degli utenti. MySpace, per esempio, permette di cliccare sul titolo di un film per disporre di un elenco di utenti che lo hanno segnalato tra le loro pellicole preferite, ed è proprio sulla base di questi elenchi che è possibile creare relazioni più o meno complesse tra soggetti caratterizzati da profili affini.

L'esempio più lampante di funzionamento virtuoso della profilazione degli utenti è il social network per eccellenza, cioè LinkedIn. In questo servizio – che ha un'ottica marcatamente orientata al business e al reperimento di nuove opportunità di lavoro – gli unici contenuti consen-

titi sono i profili degli utenti. Ogni abbonato a LinkedIn ha la possibilità di specificare con notevole dettaglio il proprio curriculum studiorum e vitae, i propri interessi e le proprie aspirazioni. In questo modo il servizio permette di creare un'identità digitale (*business-oriented*) molto dettagliata e condivisibile con gli altri appartenenti alla propria rete sociale.

3.1.1 Linee guida

Per l'implementazione di reti sociali o per l'aumento del grado di interazione sociale all'interno dei propri servizi è consigliabile quindi:

- **Fornire un'identità sociale agli utenti**

Questo è un elemento critico; gli utenti possono partecipare al gioco relazionale (e sono spinti a farlo) solo in presenza di precise "leve" identitarie. L'utente anonimo, privo di identità, è solitamente condannato al ruolo di "lurker", cioè di un utente passivo che osserva senza partecipare.

Implementare nei servizi spazi di profilazione degli utenti, in cui questi possono definirsi in modo distinguibile e pertinente rispetto al tema del servizio stesso, è fondamentale per creare i requisiti per l'interazione sociale online.

Le pagine di profilazione vanno rese pubbliche e condivise, nei limiti del rispetto della privacy e della volontà degli utenti, che, in questo modo, possono confrontarsi su una base comune. Questa è la distinzione, nei tanti servizi di social-networking, tra identità digitali definite e, per esempio, chat tra anonimi.

- **Creare spazi definiti di relazione**

È importante che la relazione non si limiti al dialogo testuale, ma abbia specifici spazi d'azione, che rendano efficace e percepibile quella dimensione definita "the power of us", la forza delle comunità.

Le opzioni di sviluppo relazionale non mancano: la dimensione sociale può manifestarsi sia con servizi di comunicazione testuale sincrona o asincrona (chat, forum, post, commenti, wiki, ecc.), sia avere un focus sul giudizio (la valutazione di oggetti, pagine online, le user-review, ecc.) o sulla classificazione sociale. Si pensi ad esempio al social-bookmarking di servizi come del.icio.us, al social-tagging di Technorati e Flickr, e in generale a tutti i servizi che consentono la creazione di folksonomie, cioè tassonomie condivise tra comunità di utenti che, costruisci-

44_www.MySpace.com



scono sistemi intuitivi di ordinamento del caos informativo-relazionale.

3.2 TAGGING E FOLKSONOMY

Uno degli effetti più lampanti della diffusione del Web 2.0 è la crescita smisurata dei contenuti presenti online. Se nel Web 1.0 i collettori di contenuto erano relativamente pochi e molto istituzionalizzati (grandi portali, siti di aziende, siti pubblici ed educativi e poche pagine “private”), con il Web 2.0 si è moltiplicata la quantità di contenuti e si è disseminata in innumerevoli piccole entità fortemente frammentate (blog, forum, wiki, ecc.). Parafrasando Chris Anderson, caporedattore della rivista Wired, si è quindi creata una “coda lunga”⁴⁵ (<http://www.thelongtail.com/about.html>) di contenuti online che è impossibile mappare e definire con precisione, perché costantemente in evoluzione.

Questo fenomeno è dovuto alla mutata identità del cittadino digitale. Se nel Web 1.0 l’utente di Internet era un consumatore passivo di contenuti preconfezionati, col Web 2.0 i singoli utenti non si limitano a “fruire” i contenuti, ma ne producono altrettanti attraverso vari strumenti (blog, wiki, sistemi di social networking, ecc.). Si è parlato, a questo proposito, di “**read-write Web**”: una vera e propria rivoluzione per Internet, passata da un modello simile al broadcasting, in cui pochi comunicano a molti relativamente passivi, ad un modello nuovo in cui **molti comunicano a molti**.

La presenza di una mole costantemente crescente di contenuti pone, a livello globale, il grosso problema della loro archiviazione e del loro reperimento. Come fare ad orientarsi in una mole simile di contenuti? I tradizionali motori di ricerca full-text come Google stanno lentamente perdendo senso ed efficacia di fronte all’evoluzione quantitativa dei contenuti online.

La nuova frontiera, mutuata dalle tecniche biblioteconomiche e adattata alla versatilità della Rete, prevede la creazione di un’architettura esplicita dei contenuti archiviati online attraverso un’azione volontaria da parte dei loro stessi creatori e consumatori.

Ecco nato il fenomeno del tagging: gli utenti del Web catalogano i singoli oggetti di contenuto apponendo ad essi una o più “etichette” (tag, in inglese) virtuali che servono a identificarli e che compongono delle folksonomie. Una fotografia su Flickr di un bambino che mangia un gelato ad un concerto dei Rolling Stones a Torino, per esempio, potrebbe avere le seguenti tag: bambino, gelato, Rolling Stones, concerto, Torino.

Tavola 6



CODA LUNGA

L’espressione coda lunga, in inglese The Long Tail, è stata conosciuta da Chris Anderson

in un articolo dell’ottobre 2004 su «Wired Magazine» per descrivere alcuni modelli economici e commerciali come per esempio Amazon.com [<http://www.amazon.com>] o Netflix [<http://www.netflix.com>]. Il termine è anche utilizzato comunemente nelle scienze statistiche per definire modelli di distribuzione della ricchezza o di usi lessicali. In queste distribuzioni, una popolazione ad alta frequenza o ampiezza è seguita da una popolazione a bassa frequenza o ampiezza, che diminuisce gradatamente (*tail off*).

In molti casi, gli eventi poco frequenti o di bassa ampiezza – la *coda lunga*, rappresentata dalla porzione gialla della curva – possono cumulativamente superare in numero o in importanza la porzione iniziale della curva, di modo che presi tutti insieme rappresentano la maggioranza. Anderson sostiene che i prodotti a bassa richiesta o con ridotti volumi di vendita possono collettivamente occupare una quota di mercato equivalente o superiore a quella dei pochi bestseller o blockbuster, se il punto vendita o il canale di distribuzione sono abbastanza grandi.

[...] La coda lunga presenta implicazioni destinate a influenzare la cultura e la politica. Ove i costi di magazzino e distribuzione sono elevati, vengono venduti solo i prodotti più popolari. Ove al contrario la coda lunga funziona, i gusti delle minoranze vengono soddisfatti e gli individui hanno maggiore possibilità di scelta. Alcune delle aziende internet devono il loro successo allo sfruttamento del principio della coda lunga nel loro modello economico. Tra le grandi compagnie spiccano eBay (aste) [<http://www.ebay.com>] Yahoo! [<http://www.yahoo.com>], Google (motori di ricerca) [<http://www.google.com>], Amazon (vendita al dettaglio) e iTunes Store (musica e podcast) [<http://www.apple.com/itunes/>], seguite da imprese minori quali Audible (audiolibri) [<http://www.audible.com>] and Netflix (videonoleggio).

[...]

In occasione di un meeting nell’autunno 1994, cui parteciparono tra gli altri Marc Andreessen (fondatore di Netscape), e diversi membri dello staff di «Wired Magazine», Ken McCarthy, pioniere del commercio su internet e storico dei media, aveva affrontato il modello coda lunga dal punto di vista dei produttori. Dopo aver spiegato come l’industria medianica pre-internet basasse le proprie iniziative di distribuzione e promozione su una filosofia economica che privilegiava la sicurezza e non su criteri di qualità o sulla potenziale durata della domanda, McCarthy espose una dettagliata previsione dell’impatto che – a suo giudizio – internet avrebbe avuto sulla struttura dell’industria dei media, la quale si è in seguito rivelata estremamente precisa, prefigurando molte delle idee che avrebbero trovato espressione nel popolare saggio di Anderson.

(estratto da Wikipedia)

⁴⁵ Si faccia riferimento alla tavola 6.



Grazie al fenomeno del tagging, gli utenti che desiderano filtrare la grande mole di contenuti presenti online possono farlo semplicemente consultando i contenuti che corrispondono ai tag di loro interesse. Rispetto alla ricerca full-text i vantaggi sono innumerevoli. I motori di ricerca tradizionali, semplificando, si limitano a poco più che contare le occorrenze di una stringa di ricerca all'interno delle pagine che hanno indicizzato e forniscono dei risultati. Si tratta, però, di ricerche "alla cieca" in cui non vi è alcun legame semantico tra il testo ricercato e i contenuti trovati.

Con il tagging, invece, sono i creatori stessi dei contenuti ad apporre le parole chiave in base alle quali saranno reperibili. Questi si aggiungono al semplice conteggio delle occorrenze.

Il tagging diventa una pratica ancora più virtuosa nel momento in cui anche gli utenti possono apporre tag a ciò che consultano e possono condividere socialmente questi tag. In questo modo si sovrappongono due dimensioni di senso, una fornita dai creatori dei contenuti e una dai tanti suoi consumatori, a beneficio di tutti.

Riprendendo l'esempio precedente della fotografia del bambino al concerto, un utente potrebbe capitare sulla pagina di Flickr che la ospita e accorgersi che è scattata in controluce allo Stadio delle Alpi. In questo caso potrebbe aggiungere i tag: "controluce, Stadio delle Alpi", aumentando ulteriormente la reperibilità della fotografia e rendendola disponibile come soluzione a nuove esigenze.

3.2.1 Usare la folksonomy

La presenza di sistemi di categorizzazione condivisa all'interno di servizi online o di siti Web non è garanzia di un'immediata efficacia nel fronteggiare la complessità dei contenuti e la loro ingente quantità. Sono, infatti, altrettanto importanti i sistemi di utilizzo della folksonomy. Ovvero, dato un corpus di contenuti opportunamente dotato di tag socialmente condivisi, come renderlo efficacemente ricercabile e navigabile in modo soddisfacente per le esigenze dell'utente?

Si pone un problema di information retrieval. I tradizionali strumenti di filtraggio, che riportano come risultato di una ricerca dei tag tutti i contenuti che la soddisfano, sono normalmente efficaci, ma scontano un problema: come ordinare a loro volta i risultati del filtraggio.

Per esempio, se una ricerca di fotografie dotate del tag "superga" porta come risultato un elenco di 50 immagini, quale proporre per prima e quale per ultima? E, soprattutto, in base a quale criterio? Le soluzioni adottate online, in questi casi, sono molteplici e variano di

servizio in servizio.

Molti adottano criteri cronologici, per cui i risultati più recenti hanno la priorità su quelli più vecchi. Questa opzione può avere senso nel caso del reperimento di notizie, articoli, ecc. ma non è efficace nel caso in cui un utente esamini un elenco di immagini.

Alcuni servizi, come ad esempio Technorati, permettono di gerarchizzare i risultati delle ricerche attraverso la popolarità/rilevanza dei singoli risultati, cioè un parametro che misura "l'importanza online" di ciascuno. Solitamente questo parametro è basato sul numero di link che puntano ad un risultato: se un contenuto è linkato da molte pagine Web avrà la priorità in elenco rispetto a quelli meno linkati.

3.2.2 Linee guida

Per dotare siti e servizi online di sistemi di folksonomy è opportuno:

- **Implementare sistemi di tagging contestuali ai contenuti**

È prioritario indurre gli utenti, siano essi consumatori o produttori, ad apporre tag ai contenuti con cui si confrontano. Per ottenere questo risultato è importante rendere facile e immediata l'operazione di etichettatura, evidenziando lo spazio per l'aggiunta dei tag.

- **Sviluppare sistemi efficaci di utilizzo dei tag**

Grazie ai tag è possibile compiere numerose operazioni di tipo semantico sui contenuti, per esempio filtrarli escludendo tutti quelli che non rispondono ai criteri impostati, ma anche presentarli per la navigazione in modo non lineare e content-based. È il caso delle tag-cloud, "nuvole" di tag associate ai contenuti relativi a uno specifico ambito, la cui rilevanza viene espressa graficamente.

- **Integrare nell'interfaccia dei servizi le modalità di ricerca tag-based**

Questo significa integrare sia i box di ricerca tradizionali, sia strumenti che consentano di filtrare i contenuti con un solo click. Ogni contenuto, per esempio, potrebbe riportare in evidenza i tag che lo caratterizzano. Cliccando su ciascuno di essi potrebbe essere possibile raggiungere automaticamente contenuti simili (della stessa fonte o di fonti diverse), così come accade su Technorati, che tuttora è il più diffuso sistema di aggregazione e folksonomy rivolto ai blog di tutto il mondo.

3.3 SISTEMI DI GEOREFERENZIAZIONE E POSSIBILI APPLICAZIONI



Stabilita un'identità digitale dell'utente online, la nuova frontiera di interazione è capire dove si trova l'utente e fornirgli strumenti e contenuti personalizzati a seconda della sua posizione.

La diffusione di strumenti in grado di rilevare la posizione degli utenti, su tutti le antenne GPS che permettono – in esterna – di stabilire con esattezza le coordinate geografiche di ciascun utente, è fondamentale per implementare servizi che tengano conto del contesto spaziale/geografico in cui si trovano.

Poter stabilire con esattezza o relativa approssimazione la posizione di un utente può essere critico per la creazione di sistemi informativi personalizzati. Se un sistema è in grado di stabilire dove si trova ogni suo utilizzatore, diventa relativamente facile proporre prodotti, servizi e opportunità in qualsiasi zona si trovi l'utente, grazie agli strumenti di geolocalizzazione.

Le applicazioni generabili su questo modello sono innumerevoli. Basti pensare ai potenziali servizi di informazione turistica che si potrebbero creare: è sufficiente conoscere la posizione dei visitatori per inviare loro informazioni turistiche sui monumenti e sulle attrattive nella loro zona di pertinenza, ma anche proposte commerciali, segnalazioni di eventi, ecc.

Combinando la conoscenza del “dove” con l'identità digitale, quindi la conoscenza del “chi”, è possibile generare servizi che non si limitino a segnalazioni basate solo sulla posizione dell'utente, ma anche sui suoi gusti, sulle sue preferenze e, se possibile, sul profilo utente generato nel caso della sua esperienza online.

Per i cittadini il modello è simile: si possono generare servizi informativi in grado di segnalare, per esempio, le farmacie di turno e gli ambulatori più vicini all'utente. In presenza di un GPS e di un sistema di mapping, è possibile addirittura fornire indicazioni stradali, calcolare percorsi e, quindi, non limitarsi alla segnalazione ma svolgere anche il ruolo di accompagnatori. Con la diffusione dei telefoni cellulari e degli smartphone che integrano un navigatore GPS, dovrebbero nascere i primi servizi in grado di compiere queste operazioni in modo del tutto integrato.

Esiste, infine, una frontiera ulteriore, generata ribaltando la prospettiva dei servizi informativi. Gli utenti sul territorio, infatti, non sono meri consumatori di informazioni ma possono a loro volta diventare produttori di contenuti in mobilità. In questo caso, lo *user generated content* sarebbe facilmente geolocalizzabile – in particolare se prodotto attraverso terminali dotati di antenna GPS – e facilmente condivisibile online con gli altri utenti.

Questo può generare dinamiche positive per cui gli uten-

ti stessi commentano i servizi informativi, li arricchiscono di valore personale o, nel caso di servizi di utility, segnalano malfunzionamenti, consigli, segnalando ad esempio la situazione del traffico automobilistico in tempo reale.

3.3.1 Linee guida

Di seguito riportiamo alcune linee guida utili per la realizzazione di servizi di localizzazione dei contenuti e delle risorse.

- **Sviluppare sistemi di rilevamento automatico della posizione dell'utente**

Attualmente in Italia questo è possibile solo attraverso le antenne GPS, perché la triangolazione attraverso le celle GSM non è supportata dagli operatori telefonici e il Wi-Fi pubblico è poco diffuso, e non consente comunque un posizionamento preciso.

- **Implementare servizi informativi (guide, assistenti, mappe con punti d'interesse) in grado di adattarsi alla posizione dell'utente**

Il vantaggio del conoscere la posizione dell'utente sta proprio nel poter filtrare i contenuti dei servizi informativi e proporre solo quelli pertinenti, valorizzando l'elemento geografico.

- **Integrare la posizione dell'utente con la sua “identità digitale”**

Questo significa proporre agli utenti sistemi di profilazione o auto-profilazione tali per cui i sistemi informativi sono in grado di proporre informazioni basate non solo sulla posizione degli utenti, ma anche sulle loro preferenze.

- **Stabilire sistemi di immissione e condivisione in mobilità dello user-generated-content**

Questo significa aprire agli utenti i sistemi informativi e far sì che chi li utilizza possa contribuire socialmente, con diversi gradi di coinvolgimento. Si spazia dalla valutazione dei contenuti contrassegnati con le classifiche “stelline”, al social tagging/social bookmarking, fino alla produzione di commenti e contenuti ex novo.

- **Stimolare la produzione di user-generated-content geolocalizzato attraverso opportune politiche di rewarding**

In questo contesto di spazia dall'accesso ad aree riservate, alla possibilità di ottenere premi, sconti, vantaggi... Un rafforzamento motivazionale alla partecipazione, che, soprattutto in mobilità, può essere costosa per l'utente e deve essere sostenuta con incentivi e iniziative accessorie.

3.4 VALUTARE LA USER EXPERIENCE



La user experience, cioè l'esperienza avuta dai fruitori di un prodotto o servizio, è diventata negli ultimi anni uno strumento di marketing molto diffuso. L'idea alla base di questa considerazione è la cosiddetta "innocenza" del consumatore/cittadino, cioè un individuo guidato dalle proprie esigenze e teso alla loro soddisfazione. Ascoltare, valutare e condividere la user experience, quindi, può arricchire un servizio online, garantendogli contenuti più spontanei, non agiografici o parziali.

In particolare in ambito turistico la valorizzazione della user-experience può aggiungere credibilità ai servizi di promozione territoriale. Questo significa creare spazi in cui gli utenti di servizi turistici, i visitatori di musei, gallerie, opere d'arte, zone archeologiche, possano esprimersi, condividere socialmente sensazioni, opinioni, esperienze, immagini e filmati.

La creazione di aree dedicate alla conservazione dell'esperienza degli utenti non è una novità: nei musei anglosassoni è buona norma lasciare a disposizione dei visitatori un registro cartaceo in cui ciascuno può lasciare impressioni e, ovviamente, consultare quelle dei visitatori precedenti. Immaginare una versione moderna e aggiornata alle potenzialità della Rete del registro visitatori dei musei è una realtà praticabile. Attualmente, infatti, esistono numerosi servizi dedicati alla conservazione dell'esperienza.

Tutti i sistemi di creazione di gallerie fotografiche online o di archiviazione di video online non sono altro che grandi *arene di condivisione sociale dell'esperienza*. Molti turisti, cioè, di ritorno dai loro viaggi trasferiscono online e condividono le fotografie e i video catturati, raccontano i loro viaggi su blog e affini e si interfacciano con altri utenti della Rete, discutendo dei luoghi visitati. L'idea di centralizzare questa procedura, magari creando un servizio online che aggrega o che raccolga direttamente l'esperienza dei visitatori di un determinato luogo è realizzabile con relativa facilità, poiché si basa su tecnologie già esistenti.

La natura di questo servizio avrebbe un valore duplice: garantire ai visitatori la conservazione della loro esperienza e, soprattutto, fornire ai nuovi potenziali visitatori un'area ricca di contenuti, commenti (non necessariamente tutti positivi, ma proprio per questo più credibili) a tema: un sistema di promozione che sfrutta la credibilità/autorità dei singoli utenti e lascia a loro l'onere di produrre e classificare i contenuti.

3.4.1 Linee guida

Di seguito alcune linee guida utili per la realizzazione di servizi legati alla condivisione e conservazione della user experience.

- **Predisporre strumenti di archiviazione e condivisione della user experience relativa al luogo/evento/prodotto che si desidera promuovere**

Archiviazione e condivisione possono essere ottenute chiedendo agli utenti di utilizzare tag specifiche in modo tale da poter aggregare facilmente tutti i contenuti ad esse associati e riusarli liberamente. Un esempio interessante in questo senso è la manifestazione del BarCamp, un meeting tra il virtuale e il non virtuale fondato sulle logiche di scambio e interazione proprie del web 2.0, in cui i temi di discussione non sono prestabiliti ma vengono proposti liberamente dagli stesi partecipanti. Nel corso della manifestazione che si è tenuta a Torino⁴⁶, gli organizzatori hanno richiesto a tutti i blogger partecipanti di utilizzare i tag "barcampurin" sui loro blog qualora avessero scritto dei post relativi all'evento e, ad evento concluso, hanno aggregato su una singola pagina tutti i post e tutte le fotografie e video relativi al BarCamp, creando in modo automatico una rassegna Web multimediale.

- **Adottare precise politiche di gestione dello user-generated-content**

Questo si può ottenere evitando censure, correzioni agiografiche, ingerenze, ecc. Pensare, cioè, che la comunicazione dell'utente possa essere uno strumento di comunicazione più credibile della comunicazione istituzionale o di prodotto, soprattutto se gli utenti che se ne fanno carico dispongono di una buona reputazione nello specifico network sociale.

3.5 REPUTAZIONE, CREDIBILITÀ E CONSENSO

La moltiplicazione dei contenuti e il lavoro complesso che prevede il loro ordinamento attraverso la folksonomy sollevano un problema di autorità: chi è affidabile e chi no, in questo proliferare di contenuti e autori?

La risposta non è semplice, ma esclude a priori la possibilità di poter stabilire sulla base dell'esperienza ogni singolo caso di autorità o di mancanza di essa. Anche in questo caso, quindi, la soluzione può venire da un'analisi della dimensione sociale della comunicazione.

L'autorità, infatti, è uno status che si acquisisce social-

⁴⁶<http://barcamp.org/BarcampTurin>



mente: non ci si auto-proclama autorevoli, ma si ottiene lo status grazie agli altri e presso gli altri, conquistando la loro fiducia. Si tratta di uno status transitorio, limitato nel tempo e nelle competenze: un soggetto autorevole nel campo della fisica, ad esempio, può essere totalmente incompetente in altri ambiti e dunque non ritenuto credibile in contesti diversi dal proprio ambito di specializzazione.

Il problema dell'autorità e della fiducia, in una parola della credibilità, è tradizionalmente sentito online fin dai primi passi del web. Con il diffondersi del Web sociale, tuttavia, è possibile arrivare a soluzioni che cerchino di stabilire socialmente l'autorità di una fonte, di un utente, di un contenuto.

Sistemi come **eBay**, per esempio, basano l'autorità degli utenti, che nel caso specifico è sinonimo di affidabilità, sulla loro reputazione. I clienti di un venditore online sono invitati a segnalare al sito le loro impressioni, dargli un punteggio e, tramite un campo libero, segnalare eventuali note. Commenti e punteggi sono liberamente consultabili per ciascun utente e la valutazione globale, nella rete sociale di eBay, costituisce un parametro essenziale per la valutazione dell'affidabilità delle transazioni.

Sistemi complessi e da sempre aperti allo user-generated-content come **Amazon**, dalle recensioni dei prodotti ai commenti, dai forum di discussione ai wiki tematici sui prodotti, utilizzano un metro diverso: il grado di partecipazione sposato alla reputazione degli utenti autori, che è basata sui voti degli altri utenti, in merito all'utilità delle loro recensioni. In sostanza Amazon valuta e dà priorità agli utenti recensori che "lavorano" di più, cioè che producono più recensioni. Per loro viene addirittura prodotta una classifica che segnala e mette in evidenza i più attivi.

Altri servizi, come **Technorati** o **Digg.com** classificano l'autorità dei contenuti e dei loro utenti in base alla popolarità, cioè alla capacità di raccogliere link e segnalazioni online.

Visto in prospettiva, quello che emerge non è un sistema di valutazione esatto, ma una stima "qualitativa" di natura "sociale" realizzata su base *democratica o meritocratica*.

In tutti i casi, incluso Amazon, che ha appunto un sistema misto di meritocrazia e voto democratico, si identifica chiaramente un sistema di reputazione su base sociale: gli utenti danno un voto più alto a coloro che ritengono migliori siano essi più attivi, affidabili, credibili.

Di seguito riportiamo alcune linee guida utili per la realizzazione di servizi/ambienti legati alla social reputation.

- **Consentire agli utenti di valutare lo user-generated-content:** con questa operazione è possibile costituire un ranking dei contenuti, valutando i più popolari (cioè quelli con un maggior numero di voti) e i più graditi (quelli con la media più alta dei voti), cioè – con una metafora televisiva – l'indice *Auditel* (che misura quante persone assistono ad uno spettacolo) e l'indice di gradimento (che misura quanto sia gradito uno spettacolo).
L'ideale è consentire che i contenuti siano valutabili contestualmente, con sistemi intuitivi (le ormai abusate 5 stelline da colorare a seconda del gradimento espresso non sono che una delle tante soluzioni possibili, pur restando forse le più intuitive poiché usate in software di larga diffusione come Windows Media Player e iTunes per la valutazione dei file multimediali).
- **Consentire agli utenti di poter consultare la reputazione di ogni singolo altro utente:** in questo caso è sufficiente creare una pagina per ciascun utente attivo, che raccolga tutte le informazioni sulla sua attività e sui voti ottenuti (quantità e media).
- **Evidenziare la reputazione/esperienza di ciascun utente accanto a ogni suo intervento/contributo:** in questo modo è facile che si sviluppino immediatamente dinamiche in cui gli utenti riconoscono subito le fonti che reputano autorevoli.

4. ACCESSIBILITÀ E WEB 2.0



4. ACCESSIBILITÀ E WEB 2.0



Come anticipato nei paragrafi precedenti, una delle caratteristiche più interessanti del Web 2.0 è l'evoluzione del design delle interfacce verso un modello che si può definire "desktop-like", reso possibile dall'uso massiccio di scripting lato client e, in particolare, dalle potenzialità delle tecnologie del gruppo AJAX.

Da un uso accorto di AJAX, possono infatti risultare applicazioni più usabili, sia in quanto basate sulla ripresa di metafore e modelli di interazione efficaci e già noti all'utente, in grado quindi di migliorare le prestazioni, sia perché consentono un'esperienza d'uso decisamente più ricca rispetto alle applicazioni Web tradizionali. Si ricorderà infatti che la definizione classica di usabilità (norma ISO 9241-11, *Ergonomic requirements for office work with visual display terminals - Guidance on usability*) contempla, accanto all'efficacia e all'efficienza con cui l'artefatto oggetto di valutazione può esser fruito, anche la soddisfazione d'uso.

L'attenzione per l'usabilità che si è riscontrata nella progettazione della maggior parte delle applicazioni Web 2.0 trascende però l'utilizzo di AJAX, declinandosi anche in una serie di principi di buon design che, oltre ad aver contribuito a definire un aspetto "tipico" delle applicazioni Web 2.0, concorrono nel migliorarne l'usabilità in tutti i suoi aspetti. Il riferimento è alle ormai note linee di tendenza riscontrate nel design, secondo cui, ad esempio, vengono preferiti caratteri sovradimensionati, layout semplici, colori tenui e così via.

Indubbiamente, la cura posta nel valorizzare gli aspetti di usabilità delle applicazioni, accanto al desiderio di sperimentare liberamente le potenzialità creative di una nuova tecnologia, sottolinea la maturità raggiunta nell'evoluzione del Web Design.

Tuttavia, proprio il consolidarsi delle nuove tendenze di progettazione e la centralità acquisita da AJAX ha reso imprescindibile la questione dell'accessibilità, che risulta critica per molte applicazioni Web 2.0. Com'è noto, si definisce accessibilità la misura della capacità di una risorsa di essere fruibile da quante più persone possibile e in condizioni diverse, senza bisogno di modifiche. In particolare, il concetto di accessibilità viene riferito, da un lato, ai dispositivi utilizzati per accedere alla risorsa e, dall'altro, alle specifiche abilità fisiche e cognitive degli utenti.

Uno dei capisaldi stessi di AJAX, la delega di gran parte delle elaborazioni al client, rende evidente la difficoltà di progettare applicazioni che siano effettivamente accessibili, mantenendo le loro funzioni e caratteristiche su un gran numero di dispositivi diversi. D'altra parte, il supporto scarso e non uniforme che tecnologie assistive come gli screen readers o i lettori Braille offrono per linguaggi come Javascript rende difficoltoso l'accesso alle nuove applicazioni da parte di utenti non vedenti.

Malgrado usabilità e accessibilità condividano normalmente molti aspetti, soprattutto di ordine cognitivo, queste due dimensioni sembrano quasi definire un trade-off nell'ambito dell'attuale panorama di sviluppo del Web, per cui non è possibile accrescere il valore di una delle due senza sacrificare l'altra. Al progettista si pone quindi una scelta che è anche filosofica: creare applicazioni accessibili per tutti, ma che offrono un'esperienza d'uso ridotta rispetto alle potenzialità offerte dalle tecnologie AJAX (rinunciando anche a collocarsi a pieno titolo nel Web 2.0, con il valore aggiunto a livello di immagine che questo comporta), oppure estremamente usabili, ma escludendo una parte dei potenziali utenti dalla fruizione dei contenuti, in misura più o meno significativa. Si tratta di una scelta che idealmente non dovrebbe porsi, considerando non solo che sia l'usabilità sia l'accessibilità sono caratteristiche desiderabili per un'applicazione, ma anche che queste sono strettamente correlate e concorrono insieme nel determinare l'esperienza dell'utente. Si può infatti sostenere che l'accessibilità costituisca il requisito minimo affinché un contenuto possa essere considerato usabile; se un utente non può fruire di un certo contenuto, infatti, ulteriori valutazioni relative ad efficacia, efficienza e soddisfazione perdono ogni significato.

Una delle sfide principali a cui si trova di fronte chi progetta applicazioni e vuole aderire al paradigma del Web 2.0 consiste dunque nel trovare una riconciliazione tra usabilità e accessibilità. L'obiettivo è garantire un'esperienza d'uso ricca, senza rendere l'applicazione inaccessibile ad una parte degli utenti: si tratta, per usare una formula, di passare dalle RIA (Rich Internet Applications) alle ARIA (Accessibile Rich Internet Applications).



4.1 STATO DELL'ARTE SULLE ARIA (ACCESSIBLE RICH INTERNET APPLICATIONS)

4.1.1 Definizione RIA

L'acronimo RIA (*Rich Internet Applications*) è stato introdotto da Macromedia nel Marzo del 2002 per qualificare le applicazioni Web che presentano caratteristiche analoghe a quelle delle tradizionali applicazioni desktop. Le RIA, infatti, sono in grado di rispondere dinamicamente alle azioni dell'utente, sia modificando il proprio aspetto visivo, sia caricando in tempo reale i dati richiesti. Nel contempo, hanno il vantaggio di essere universalmente raggiungibili tramite Internet.

Tipicamente, le RIA demandano al client le elaborazioni necessarie alla gestione dell'interfaccia utente, mentre mantengono la maggior parte dei dati sul server. Inoltre, non richiedono l'installazione di alcun software, al contrario, sono solitamente accessibili tramite browser e vengono eseguite in una sandbox. Possono inoltre funzionare anche qualora non si disponga di una connessione continua.

Tra le tecnologie più comunemente utilizzate, oltre a quelle del gruppo noto come AJAX, di cui fa parte innanzitutto Javascript, vi sono Adobe Flash, Java (applets e application Web Start) e i controlli Active X.

L'architettura ibrida su cui si basano le RIA offre svariati vantaggi rispetto a quella delle applicazioni Web tradizionali. Queste ultime, infatti, sono strutturate secondo un modello client-server, in cui tutte le elaborazioni sono a carico del server, mentre il client si occupa esclusivamente della visualizzazione di contenuti statici. Il principale svantaggio di questo sistema è la sua relativa lentezza, dovuta proprio al fatto che tutte le interazioni devono necessariamente passare attraverso il server. Al contrario, eseguendo la maggior parte delle operazioni lato client, le RIA possono rispondere in modo più efficiente agli input dell'utente. In particolare, le RIA si caratterizzano per la presenza di uno strato di codice (solitamente chiamato *client engine*) che fa da ponte tra l'utente e il server. Il *client engine*, che viene solitamente scaricato all'avvio dell'applicazione e può essere completato da porzioni supplementari di codice integrate durante l'esecuzione, si comporta come una sorta di estensione del browser e si occupa di gestire sia l'interfaccia utente, sia le comunicazioni con il server, cosicché l'applicazione possa essere estremamente reattiva alle azioni dell'utente.

Vantaggi

Conseguenza dell'uso di un client engine è innanzitutto la possibilità di offrire un'interfaccia utente più ricca, che presenti una gamma di modelli di interazione più ampia rispetto a quella disponibile per le applicazioni Web tradizionali. Il *drag and drop*, l'uso di *slider* per modificare i dati o l'esecuzione di calcoli lato client sono esempi di tali possibilità di interazione aggiuntive. Avvantaggiandosi degli elementi descritti, le RIA più sofisticate possono arrivare a mostrare un *look and feel* molto simile a quello che caratterizza le applicazioni desktop.

Dall'uso di un *client engine* deriva anche un bilanciamento più efficiente del carico di lavoro tra il client e il server: nello specifico, viene liberata una parte delle risorse del server, cosicché quest'ultimo è in grado di gestire più sessioni utente allo stesso tempo.

Inoltre, nel modello RIA, client engine e server comunicano in modo asincrono, ovvero senza bisogno di attendere un input da parte dell'utente; quest'opzione è utile in particolare per minimizzare i tempi di attesa. Infatti, un'applicazione può prevedere la necessità di certi dati e scaricarli prima che l'utente li richieda in modo esplicito, accelerando così i tempi di risposta. Ad esempio, è possibile che un'applicazione che fa uso di mappe scarichi le porzioni adiacenti a quella visualizzata correntemente prima che l'utente le richieda, creando così un'esperienza di fruizione continua. Il fatto di scaricare in modo asincrono porzioni di codice può portare, tuttavia, ad un sovraccarico del traffico di rete: solo una parte dei dati recuperati dal client engine saranno infatti poi effettivamente richiesti dall'utente e quindi utilizzati. D'altro canto, è possibile che il client engine, che è specifico per una singola applicazione, risulti più "intelligente" di un browser generico nel determinare quali dati sia necessario scambiare con il server: in tal modo, è possibile che il traffico venga alleggerito, bilanciando così l'effetto contrario dovuto alle comunicazioni asincrone.

È inoltre importante sottolineare i vantaggi che le RIA presentano, in quanto fruibili attraverso il Web, rispetto alle applicazioni desktop. In primo luogo, le RIA non richiedono un'installazione e sono accessibili da qualunque macchina che disponga di una connessione Internet, qualunque sia (almeno nella maggior parte dei casi) il sistema operativo utilizzato. Similmente, gli aggiornamenti sono gestiti in modo centralizzato, senza che debba essere l'utente a preoccuparsene. Inoltre, le applicazioni Web risultano generalmente meno esposte al rischio di virus.

Si consideri anche che, mano a mano che la disponibilità di connessione Internet aumenta e, con essa, la consuetudine degli utenti a navigare sul Web, è possibile che questi ultimi risultino meno propensi a impegnarsi nella procedura di installazione di applicazioni sulla propria macchina, se si hanno a disposizione soluzioni alternative di immediato utilizzo.

Criticità

È importante sottolineare che, a fronte dei numerosi e significativi benefici descritti, le RIA presentano una serie di criticità.

In primo luogo, le RIA si basano generalmente su Javascript o altri linguaggi di scripting e, qualora un utente li abbia disabilitati, l'applicazione non funzionerà nel modo previsto. Allo stesso modo, poiché per motivi di sicurezza vengono eseguite in una sandbox, le RIA hanno possibilità di accesso limitate alle risorse del sistema e possono non funzionare appropriatamente qualora richiedano l'uso di risorse a cui non possono



accedere. Inoltre, benché non ci sia bisogno di procedere a un'installazione, è necessario scaricare il codice che compone il client engine almeno la prima volta che lo si utilizza, operazione che può richiedere tempi relativamente lunghi (per quanto gli sviluppatori di RIA cerchino di minimizzare i tempi adottando opportuni accorgimenti). Infine, è possibile che l'esecuzione dell'applicazione sia rallentata dalla necessità di interpretare in tempo reale linguaggi di scripting come Javascript (tale inconveniente non si presenta qualora si utilizzino linguaggi come Java, le cui prestazioni sono paragonabili a quelle dei tradizionali linguaggi compilati).

Per quanto riguarda i modelli di interazione con l'utente, si deve segnalare che le RIA non rispettano la convenzione consolidata secondo cui l'unità di navigazione e di visualizzazione devono corrispondere⁴⁷. Ciò significa che gli URL non sono sufficienti per identificare in modo univoco il contenuto visualizzato da una RIA in un certo momento, ma solo le pagine al loro stato iniziale (a prescindere, quindi, dalle modifiche apportate in seguito all'interazione con l'utente), in quanto non permettono una rappresentazione dei diversi possibili stati.

4.1.2 Accessibilità nelle applicazioni RIA

Benché, come si è detto, l'etichetta RIA sia stata introdotta ormai quattro anni fa, prima ancora che si cominciasse a parlare di Web 2.0, per descrivere concetti che erano noti già da tempo, queste applicazioni si caratterizzano ancora come una relativa novità. Il loro livello di diffusione e conoscenza tra gli utenti della Rete, infatti, non è ancora esteso, a causa di una serie di requisiti tecnologici che non tutti i possibili devices sono in grado di supportare.

In particolare, la maggior parte delle RIA funziona correttamente solo su browser moderni, capaci di interpretare Javascript e di supportare tecniche come XMLHttpRequest per la comunicazione con il server, DOM Scripting e CSS2 per la gestione dell'interfaccia utente. In aggiunta, bisogna considerare che le differenze nel comportamento dei diversi browser rendono difficile progettare applicazioni che funzionino in modo corretto e uniforme sulla maggior parte dei client. D'altro canto, anche quando un'applicazione funziona correttamente dal punto di vista tecnico, è possibile che l'utente incontri delle difficoltà nel servirsele. Benché, come si è cercato di mostrare, le RIA risultino generalmente più usabili, utenti ormai a proprio agio con il modello di interazione delle applicazioni Web tradizionali possono rimanere inizialmente disorientati qualora alcuni elementi mostrino un comportamento diverso da quello previsto. L'esempio diventato "classico" è quello del pulsante "indietro" del browser, che non sempre funziona nel modo atteso in interazione con Javascript. Tuttavia, se problemi di questo tipo si possono risolvere

in modo generalmente rapido, non appena l'utente familiarizza con il nuovo modello, di per sé piuttosto intuitivo, può risultare più ostico superare i limiti che si presentano sul fronte dell'accessibilità.

In particolare, nel progettare l'interazione con le RIA si devono affrontare due diversi aspetti critici nel dominio dell'accessibilità. Il primo è forse il più evidente: garantire che l'applicazione sia accessibile quando le tecnologie con cui è stata progettata (ad esempio, AJAX) non sono supportate dal client. Perché una RIA sia sempre accessibile è necessario considerare le tecnologie specifiche (ma non universalmente supportate) un valore aggiunto, che permette di ottimizzare particolari funzioni per le quali, comunque, siano previste anche modalità di fruizione alternative, sebbene non in grado di garantire un'esperienza d'uso analoga.

Il secondo problema diventa rilevante quando il client supporta le tecnologie utilizzate nell'applicazione: in tal caso è necessario assicurare che tutte le funzioni siano effettivamente accessibili e che l'applicazione funzioni in modo uniforme, secondo quelle che sono le intenzioni dei suoi progettisti. I problemi che possono emergere, in questo caso, sono legati principalmente alla rottura di convenzioni consolidate nell'interazione con il browser o al comportamento non uniforme manifestato dai client (ad esempio, degli *screenreader*) in risposta ad un certo evento.

4.1.3 Definizione di scenari nell'ambito accessibilità e AJAX

Come già sottolineato in precedenza, le criticità legate all'utilizzo di AJAX nello sviluppo di interfacce evolute e dinamiche è legato alla possibilità che script, applet o altri oggetti di programmazione siano disabilitati oppure non supportati da parte dell'utente. Senza i giusti accorgimenti, questo può provocare un mancato raggiungimento delle funzionalità fondamentali all'interno della pagina, compromettendone così l'accessibilità stessa. In questi casi tutti i vantaggi delineati in precedenza andrebbero completamente persi, danneggiando in ultima analisi l'esperienza dell'utente. L'effetto principale che caratterizza l'interazione con interfacce sviluppate in AJAX è legato alla modifica di parti che compongono la pagina (in modo appunto da velocizzare l'interazione stessa): il problema subentra nel momento in cui queste modifiche, avendo effetti diretti su specifiche funzionalità all'interno dell'applicativo, possono, se segnalate scorrettamente, non essere percepite dall'utente.

Prima ancora che il W3C⁴⁸ intervenisse direttamente sul tema dell'accessibilità di applicazioni interattive complesse con il proprio progetto WAI-ARIA,

⁴⁷ <http://www.usabilityviews.com/ajaxsucks.html>

⁴⁸ Il W3C, acronimo di *World Wide Web Consortium*, è un consorzio fondato nel 1994 dal "padre del Web" Tim Berners Lee, con lo scopo di migliorare i protocolli e i linguaggi per il Web. Tra i suoi membri si annoverano aziende informatiche e telefoniche (tra cui Microsoft, Siemens, Nokia, Google), associazioni come la Mozilla Foundation e The Open Group, università e istituti di ricerca.

in diversi contesti⁴⁹ è stato largamente discusso se e in quale modo l'impiego di AJAX potesse migliorare e arricchire l'interazione dell'utente, garantendo allo stesso tempo la piena accessibilità delle funzioni e dei contenuti. Gli esempi di script che non inficino l'accessibilità sono molto rari: la maggior parte degli esperti si trova infatti concorde nell'affermare che siano davvero poche le tecniche di scripting davvero accessibili⁵⁰. Le principali criticità sono quelle relative alle modalità attraverso le quali gli screen-reader gestiscono gli eventi Javascript: la maggior parte dei browser non visivi non è infatti in grado di supportare questa tipologia di elementi.

In particolare, quando si modifica un documento tramite uno script, è necessario che il contenuto aggiornato venga comunicato allo screen-reader⁵¹. Se non si prevede un meccanismo di notifica dei cambiamenti, gli utenti di screen-reader si trovano di fronte a due possibili scenari, altrettanto frustranti: nel primo caso l'utente non ha alcuna consapevolezza del fatto che siano stati effettuati degli aggiornamenti; nel secondo caso sa che una parte del contenuto è cambiata, ma è costretto a rileggere l'intera pagina per individuare gli aggiornamenti poiché lo screen-reader processa la pagina in una modalità di lettura lineare.

Negli ultimi mesi, molti gruppi di lavoro, hanno quindi cercato di verificare e delineare una metodologia in grado di assicurare il funzionamento delle tecnologie assistive – come gli screen-reader – quando queste hanno a che fare con AJAX. In linea di massima, non è pensabile contare su un funzionamento trasversale di AJAX su tutte le tecnologie assistive (così come su tutti i browser). È invece possibile definire una metodologia che permetta di informare gli utenti sia dei cambiamenti che avvengono all'interno di un documento, a fronte delle loro azioni, sia di come integrare con la nuova pagina proposta⁵².

Un caso piuttosto comune, che dimostra come sia cruciale gestire correttamente le modifiche che avvengono in una pagina a seguito delle azioni dell'utente, è rappresentato dall'inserimento di dati all'interno di form.

4.1.3.1 Un caso di studio: la gestione dei form

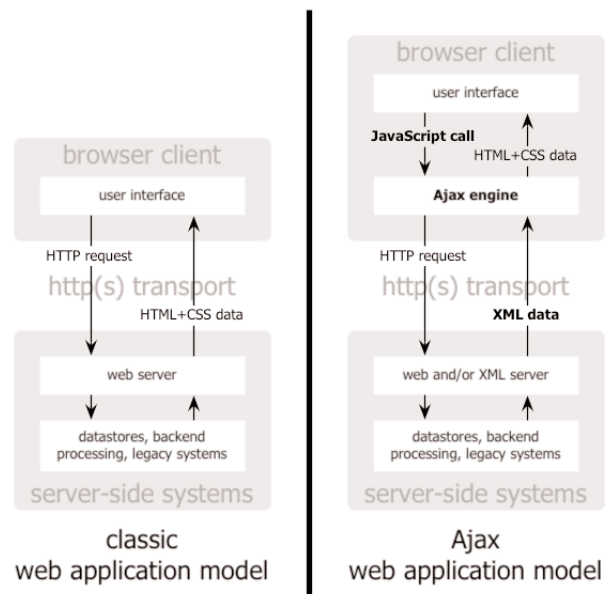
In questo caso l'impiego di AJAX può notevolmente aumentare l'esperienza finale dell'utente, sia per quanto concerne i tempi di attesa sia per le modalità di inserimento dei dati⁵³. Attraverso AJAX è infatti possibile verificare l'integrità dei dati inseriti dall'utente ogni volta che il focus passa da un elemento a quello successivo,

interrompendo così lo schema classico che prevedeva, una volta terminata l'operazione di inserimento, la verifica sui dati attraverso l'invio di questi al server. L'utente si aspetta che questa operazione avvenga in tempi lunghi, e tendenzialmente la associa ad un refresh della pagina. Con AJAX è invece possibile gestire queste azioni in maniera estremamente dinamica.

Il motore in AJAX, posizionato tra l'utente e il server, permette infatti di gestire l'interfaccia e comunicare contemporaneamente con il server. In ultima istanza l'interazione dell'utente con l'applicazione si svolge in maniera asincrona, riducendo in questo modo i tempi di attesa dell'utente al minimo.

Le azioni che normalmente prevedevano una chiamata HTTP si trasformano in una chiamata Javascript gestita appunto dal motore in AJAX. Le risposte alle azioni dell'utente che non richiedono una chiamata al server (come, la validazione dei dati) sono gestite direttamente dal motore. Qualora sia invece necessario accedere a dati – tipicamente in formato XML – disponibili solo tramite server, il motore AJAX si occuperà di gestire le comunicazioni con quest'ultimo in modalità asincrona⁵⁴ rispetto all'andamento dell'interazione con l'utente, così da ridurre i tempi morti.

Figura 1: Sulla sinistra il modello tradizionale delle applicazioni web, mentre sulla destra viene presentato il modello basato sull'impiego di AJAX. (<http://www.adaptivepath.com/images/publications/essays/ajax-fig1.png>)



49_ http://ajaxpatterns.org/Accessibility_Links

50_ Techniques and Failures for Web Content Accessibility Guidelines 2.0, <http://www.w3.org/TR/WCAG20-SCRIPT-TECHS/#N11799>

51_ Si veda in proposito: <http://juicystudio.com/article/making-ajax-work-with-screen-readers.php#informreader>.

52_ Per una descrizione dettagliata relativa al funzionamento di alcuni dei principali screen-reader: <http://juicystudio.com/article/making-ajax-work-with-screen-readers.php#screenmodes>

53_ Per una demo esaustiva sui possibili impieghi di AJAX nella compilazione delle form: <http://www.zapatec.com/website/main/zpform/demo/basic.html>

54_ Un esempio che chiarisce cosa si debba intendere per comunicazione asincrona è costituito dalla tecnica del prefetching, secondo cui un'applicazione può scaricare dal server alcuni dati senza che l'utente li richieda esplicitamente, se prevede che questi saranno necessari in seguito. Se le intenzioni dell'utente sono anticipate correttamente, la risposta dell'applicazione sarà immediata nel momento in cui questi effettuerà la propria richiesta.



Ne risulta che, grazie ad AJAX e all'XML, è possibile manipolare alcuni elementi della pagina senza effettuare un refresh della stessa: diventa così possibile progettare interfacce in grado di reagire istantaneamente all'inserimento di dati da parte dell'utente, che in questo modo non deve più attendere i tempi di refresh della pagina.

Dal punto di vista dell'accessibilità, l'interazione client-server classica, basata sul refresh dell'intera pagina ad ogni azione dell'utente, si adattava perfettamente alla modalità di lettura lineare tipica degli screen-reader⁵⁵. Il refresh, infatti, riportando il focus alla sezione iniziale della pagina, consentiva allo screen-reader di apprendere e notificare i risultati dell'interazione precedente con il sistema.

Al contrario, i cambiamenti che vengono apportati dinamicamente al DOM tramite AJAX possono riguardare elementi indifferentemente precedenti o successivi al punto a cui è arrivata la lettura: è evidente che, procedendo in modo lineare, l'utente può mancare completamente l'accesso agli aggiornamenti, oppure fruire dei nuovi contenuti, ma senza rendersi conto che questi sono diversi da quelli inizialmente disponibili.

Se talvolta perfino gli individui vedenti, che accedono ai contenuti senza essere vincolati ad una modalità di fruizione lineare, possono incontrare difficoltà nell'individuare una porzione di pagina modificata in modo dinamico, è evidente che chi accede ad un certo contenuto tramite screen-reader rischia di non avere alcuna consapevolezza degli aggiornamenti. Questo è particolarmente vero se lo screen-reader dell'utente ricorre ad una rappresentazione temporanea della pagina, conservata in un buffer virtuale⁵⁶: in questo caso, infatti, si perde anche la sincronizzazione rispetto allo stato effettivo della pagina.

Benché non si sia arrivati, come già detto, a definire una soluzione universalmente efficace, è possibile delineare una serie di linee guida per migliorare l'accessibilità dei form realizzati con AJAX⁵⁷.

In primo luogo, è utile segnalare all'utente che il form richiede il supporto di Javascript per funzionare correttamente. È possibile visualizzare comunque un avviso del genere, oppure rilevare automaticamente se tale tecnologia è supportata e mostrare un avviso solo nel caso in cui l'esito sia negativo. In tal modo, si evita all'utente un

lavoro inutile, se non dispone degli strumenti adatti per portare a termine il compito.

È importante informare l'utente del fatto che la pagina viene aggiornata dinamicamente: in tal modo, chi utilizza uno screen-reader può scegliere di richiedere esplicitamente il refresh della pagina, nei casi in cui risulta opportuno.

L'invio di un avviso ogni qual volta il contenuto di una pagina viene aggiornato facilita ulteriormente la fruizione tramite screen-reader: i semplici alertbox, ad esempio, vengono generalmente recepiti in modo corretto dagli screen-reader e possono essere accompagnati da un suono⁵⁸.

Infine, si tenga conto che progettare in modo accessibile non significa rivolgersi solo agli utenti diversamente abili: per rendere una pagina aggiornata dinamicamente più accessibile anche per chi non utilizza tecnologie assistive, è consigliabile evidenziare per breve tempo i contenuti modificati, ad esempio presentandoli per qualche secondo su uno sfondo giallo, che lentamente svanisce⁵⁹.

Se si considera, tuttavia, che una delle maggiori criticità riscontrate nel rendere AJAX accessibile è legata alla diversità nelle risposte dei vari device⁶⁰, risulta chiaro che non si può prescindere da un intervento diretto anche da parte degli stessi produttori di screen-reader. Solo attraverso un impegno congiunto dei diversi attori chiamati in causa è infatti possibile immaginare una risposta concreta ad un problema che, a fronte delle considerazioni tratte sino ad ora, appare complesso e di non facile e immediata soluzione.

4.1.3.2 AJAX e accessibilità: i possibili approcci alla progettazione

È incoraggiante rilevare che alcuni soggetti si sono già mobilitati in tal senso. Recentemente, ad esempio, la Mozilla Foundation⁶¹, in collaborazione con l'IBM e con la GW Micro (i produttori dello screen-reader Windows Eyes⁶²), ha iniziato a tracciare alcune linee guida relative all'"Accessible DHTML"⁶³ utilizzato nel browser Firefox. La finalità principale di questo documento è quella di introdurre funzioni e stati, definiti attraverso gli attributi degli elementi, in grado di specificare la natura e lo stato di un elemento. In teoria questo approccio potrebbe risolvere il problema, comunicando nella modalità

55_Per una descrizione del modo in cui uno dei più diffusi screen-reader, JAWS, permette di interagire con una form tradizionale, si veda: http://www.freedomscientific.com/fs_products/Surfs_Up/Forms.htm.

56_Una spiegazione esaustiva della modalità di funzionamento tramite virtual buffer si trova in: <http://juicystudio.com/article/making-ajax-work-with-screen-readers.php#jawsvirtual>.

57_Linee guida tratte da: <http://www.standards-schmandards.com/2005/ajax-and-accessibility>

58_Si noti tuttavia che non tutti gli screen-reader rispondono correttamente ad un alert: <http://www.sitepoint.com/article/ajax-screenreaders-work>

59_Si vedano in proposito: <http://www.37signals.com/svn/archives/000558.php> e <http://www.axentric.com/posts/default/7>.

60_Si veda ad esempio quanto riportato in: <http://www.sitepoint.com/article/ajax-screenreaders-work>.

61_<http://www.mozilla.org/foundation/>

62_<http://www.gwmicro.com/>

63_http://developer.mozilla.org/en/docs/Accessible_DHTML

appropriata anche agli screen-reader l'avanzamento dell'interazione dell'utente con il sistema, basato appunto su significato, comportamento e stato. La criticità risulta però legata alla compatibilità tra i diversi screen-reader e gli stessi browser.

Nell'aprile 2006, lo stesso W3C, rispondendo ad un'esigenza ormai molto forte in Rete, ha iniziato a definire i primi passi della roadmap⁶⁴ per lo sviluppo di linee guida relative alla progettazione di interfacce web dinamiche e accessibili. Le finalità e la struttura di questa iniziativa, che si distingue per il suo tentativo di standardizzazione, verranno descritte con maggiore attenzione nei paragrafi successivi.

L'analisi delle linee guida prodotte da gruppi di lavoro indipendenti ha delineato sostanzialmente quattro diversi approcci alla progettazione di RIA; questi si pongono idealmente su un continuum, che vede ad un estremo la realizzazione di un'applicazione alternativa non basata su AJAX e, all'altro estremo, una gestione strutturale dell'accessibilità, in modo che un'applicazione sviluppata in AJAX continui a funzionare correttamente anche quando Javascript e XMLHttpRequest non sono supportati.

4.1.3.2.1 Interfaccia alternativa senza scripting

La soluzione più semplice per permettere a qualunque utente di accedere ai contenuti di un'applicazione consiste nel fornire un'interfaccia alternativa, una versione solo testo o, comunque, priva di script, che possa essere visualizzata con qualunque device⁶⁵. L'accessibilità è rispettata se la pagina alternativa consente di eseguire almeno le funzioni principali; tuttavia, è bene tenere presente che, verosimilmente, le due versioni non saranno del tutto equivalenti, in quanto alcuni servizi non risultano gestibili senza ricorrere allo scripting.

Progettare un'interfaccia alternativa rappresenta comunque un approccio-limite, praticabile solo nel caso in cui non sia possibile offrire soluzioni che si integrino in modo fluido nell'applicazione principale.

Dalle Linee Guida per l'Accessibilità ai Contenuti Web 2.0 pubblicate dal W3C si evince infatti che:

“Le WCAG 1.0 e la Section 508 permettono varianti di solo testo unicamente nei casi in cui “l'originale” non possa esser reso accessibile in alcun modo e richiedono che la versione solo-testo sia aggiornata ogni volta che si modifica l'originale. Tale richiesta non è presente nelle WCAG 2.0, ma riteniamo necessario reinserirla.”⁶⁶

Le ragioni per cui il ricorso ad una versione alternativa non è considerato ottimale per chi voglia perseguire una reale accessibilità dei contenuti sono numerose.

In primo luogo, è possibile che la versione alternativa, se progettata senza la dovuta accuratezza, risulti essere essa stessa non accessibile; tra gli errori più frequenti, si riscontrano ad esempio la mancanza di testi descrittivi dei link. L'accessibilità di un'applicazione decade completamente se la sua versione alternativa è una mera descrizione che non è in grado di offrire informazioni e funzionalità “cruciali”. Il rischio maggiore, inoltre, è che la versione alternativa, per quanto progettata correttamente nella fase iniziale, non venga aggiornata con la medesima frequenza dell'applicazione principale.

4.1.3.2.2 Funzioni alternative senza scripting

L'uso di Javascript non rende necessariamente inaccessibile un'applicazione⁶⁷. Quando l'uso di uno script non è cruciale per portare a termine un compito, ma ha lo scopo di arricchire l'esperienza dell'utente (offrendo ad esempio suggerimenti contestuali, presentando informazioni aggiuntive, evidenziando i risultati oppure rendendo semplicemente più piacevole ed “interattiva” la grafica dell'interfaccia) non ci sono effetti negativi sull'accessibilità.

Secondo questo approccio, un'applicazione accessibile deve permettere di svolgere tutte le funzioni che la caratterizzano da una stessa interfaccia, senza bisogno di ricorrere allo scripting. Quest'ultimo può essere utilizzato, tuttavia, per consentire agli utenti dotati di un opportuno supporto tecnico di ottimizzare lo svolgimento dei diversi compiti. Ad esempio, uno script può essere utilizzato per compilare automaticamente alcuni campi di una form, a partire dai dati già inseriti: ai fini dell'accessibilità è importante che qualunque utente riesca a completare manualmente e ad inviare il modulo. La funzione di completamento automatico, che semplifica e velocizza il compito, ma non è determinante per il suo successo, verrà apprezzata solo dagli utenti che accedono all'applicazione con devices in grado di supportare lo scripting, senza danni per l'accessibilità.

4.1.3.2.3 Progressive enhancement

Come evidenziato nella sezione precedente, per risultare accessibile è necessario che una RIA funzioni anche quando lo scripting non è supportato. In questo senso, si è spiegato come sia possibile permettere a qualunque utente di portare a termine i compiti cruciali, offrendo però soluzioni ottimizzate a chi dispone di una device più evoluta.

64_ <http://www.w3.org/WAI/PF/roadmap/DHTMLRoadmap040506.html>

65_ Per quanto riguarda Gmail, Google ha ad esempio recentemente deciso di seguire questa seconda soluzione: <http://mail.google.com/support/bin/answer.py?answer=15046>

66_ <http://www.w3.org/TR/WCAG10/wai-pageauth.html>

67_ Si veda ad esempio: http://lau.csi.it/realizzare/accessibilita/linguaggi_programmazione/js_accessibile/js_accessibile_1.shtml#noimpact

L'approccio più strutturato alla progettazione di applicazioni di questo tipo prende il nome di "Progressive Enhancement" e prevede, appunto, che le diverse pagine siano progettate in primo luogo senza utilizzare AJAX (ma avendo come obiettivo una sua integrazione nell'applicazione⁶⁸), in modo da risultare universalmente accessibili. Solo in un secondo momento, le funzioni implementate vengono ottimizzate tramite AJAX, a beneficio degli utenti che utilizzano una device in grado di supportare lo scripting.

In particolare, i fautori di questo approccio⁶⁹ consigliano di iniziare progettando funzioni che si appoggino su una tradizionale struttura client-server, utilizzando linguaggi di programmazione server-side e ricorrendo al refresh dell'intera pagina per mostrare gli aggiornamenti. L'applicazione così progettata risulta accessibile a tutti gli utenti.

I link e i form usati per inviare le richieste al server possono in un secondo momento essere intercettati tramite Javascript, così da inviare i dati a un'istanza dell'oggetto XMLHttpRequest, in tal modo, si ottengono in risposta solo le informazioni necessarie, invece del caricamento dell'intera pagina. Il compito viene così ottimizzato tramite AJAX e l'esperienza dell'utente risulta arricchita.

4.1.3.2.4 Rendere gli script accessibili

La soluzione più radicale al problema dell'accessibilità delle RIA consisterebbe nel rendere gli script stessi accessibili. Per ottenere un risultato di questo genere è possibile ricorrere ad un approccio simile a quello utilizzato dalle normali GUI, che sfruttano delle API per l'accessibilità come ponte tra l'applicazione stessa e le tecnologie assistive attraverso cui l'utente fruisce di un certo servizio. Le API per l'accessibilità rappresentano un contratto tra l'applicazione e le tecnologie assistive, quali sono gli screen-reader, che permette a queste ultime di accedere ad un contenuto dinamico fornendone una descrizione semantica appropriata, tale da consentire la produzione di un'alternativa effettivamente usabile.

A tale scopo, è necessario definire correttamente la funzione e lo stato di ciascun elemento, così che questo possa essere presentato all'utente nel modo più appropriato rispetto alle caratteristiche del device utilizzato. Il progetto WAI-ARIA del W3C, a cui si è già fatto cenno, si può ricondurre proprio a quest'ultimo approccio.

4.1.4 Riflessioni conclusive: web 2.0 e accessibilità

A fronte di una relativa maturità del fenomeno Web 2.0, che ha permesso di individuare non solo una serie di elementi consolidati nella progettazione di applicazioni Web di nuova generazione, ma anche alcune linee evolutive che segneranno verosimilmente il prossimo futuro, si rileva come il tema dell'accessibilità delle nuove tec-

nologie, emerso solo recentemente, sia attualmente al centro di un acceso dibattito.

L'analisi condotta ci permette di affermare che, dopo una prima fase di sperimentazione in cui tecnologie come AJAX venivano adottate entusiasticamente, quasi costituissero di per sé un valore, la comunità dei progettisti ha iniziato un processo di riflessione volto a identificare, da una parte, le criticità insite nelle nuove tecnologie e, dall'altra, le possibilità di sfruttarne le potenzialità, garantendo al contempo un adeguato livello di accessibilità.

Come si evince dal quadro descritto nei paragrafi precedenti, tuttavia, le dimensioni e la complessità del problema non permettono ancora di presentare una soluzione generale. In molti casi, chi desidera progettare applicazioni accessibili si vede ancora costretto a sviluppare soluzioni ad hoc, che permettono di raggiungere il risultato desiderato solo in casi molto specifici.

La quantità e la varietà dei soggetti coinvolti, tra cui segnaliamo ancora una volta il W3C, con il suo sforzo di standardizzazione e di mediazione, costituisce tuttavia un ottimo presupposto per la definizione di una soluzione sufficientemente generale e condivisa. La capacità di coinvolgere nel processo decisionale anche i produttori di tecnologie assistive rappresenterà probabilmente uno dei fattori più importanti nel determinare un risultato positivo, soprattutto se consentirà di porre rimedio alle difformità riscontrate nel gestire gli script.

⁶⁸ Jeremy Keith, web developer inglese, sostiene ad esempio: "Plan for Ajax from the start... implement Ajax at the end". [<http://xtecho6.usefulinc.com/schedule/paper/29>]

⁶⁹ Il riferimento è, in particolare, a Jeremy Keith e a Ryan Campbell [<http://particletree.com/features/the-hows-and-whys-of-degradable-ajax/>].

5. CONCLUSIONI



5. CONCLUSIONI



Dopo questa panoramica, del mondo del Web che cambia ci preme sottolineare che la velocità della Rete impone anche la capacità, non sempre alla portata di tutti, di interpretare il cambiamento in termini di effettivo progresso dei sistemi e degli ambienti deputati alla comunicazione, alla collaborazione e alla condivisione.

Il ruolo del filtro è affidato anche in questo caso all'utente. Nel caso si volessero intraprendere attività di tipo economico usando qualche pezzo della filosofia del Web 2.0 consigliamo di pensare in "modalità business" ancor prima di stendere anche solo una riga di progetto.

Il mondo del Web non è mai stato permeabile a tutte le mode e la selezione è sicuramente dura, così come si rivela dura e impegnativa la ricerca di consenso e la creazione della reputazione.

Nel breve-medio periodo è presumibile che vedremo il nascere di molte applicazioni studiate sull'utente e in grado di migliorare l'interazione che questi ha con l'ambiente in cui vive e lavora o in cui passa il suo tempo libero, dedicandolo all'intrattenimento.

Chi opera nell'ambito della comunicazione pubblica avrà a che fare con nuove applicazioni nate per dare "voce" all'utente. Ci vorranno mesi, se non anni, per vedere forme di partecipazione mature, ma arriveranno e con loro arriverà anche la consapevolezza che in Rete la comunicazione top-down non può essere l'unica forma di dialogo tra enti e cittadini.

Le tecnologie sono sempre più trasparenti e usabili, il mercato è invaso di modelli e applicazioni innovative e molto creative, ma non tutte funzionano o funzioneranno. Come abbiamo visto esaminando le applicazioni Web 2.0 di maggior successo abbiamo di fronte un mix (potremmo dire mash-up) di applicazioni. Sopravvivranno e si imporranno pochi modelli, ben funzionanti e creati grazie alla collaborazione dell'utente e delle social network in cui verranno testati e diffusi.

Davanti a noi abbiamo un universo tutto da esplorare, creato dagli utenti per gli utenti e fortemente vincolato alla vita di tutti i giorni. Anche le operazioni più semplici saranno condotte con l'ausilio della Rete, soprattutto nel caso in cui l'utente in mobilità vorrà avere informazioni puntuali e precise del luogo

che sta "esplorando" e del mondo intorno ad esso, senza limiti di tempo e di distanza, consapevole che la Rete è fatta sì di siti web e di servizi, ma soprattutto di persone che condividono tempo, spazio, risorse e obiettivi.

ALLEGATO 1

APPLICAZIONI WEB 2.0 DI SUCCESSO

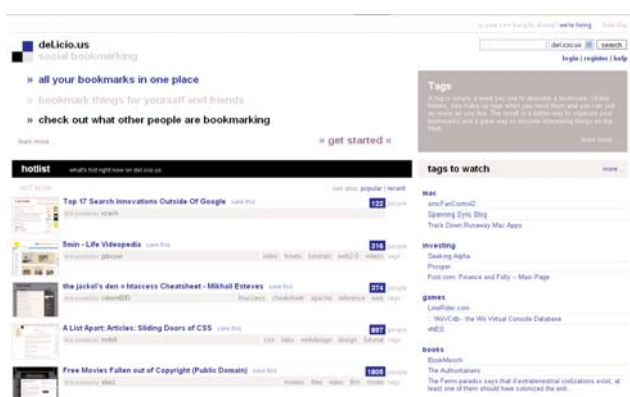
a1

APPLICAZIONI WEB 2.0 DI SUCCESSO ■ ■ ■

6.1 DEL.ICIO.US

Sistema di social bookmarking dedicato all'archiviazione, condivisione e ricerca di segnalibri.

Figura 1: del.icio.us - <http://del.icio.us/>



Del.icio.us è un sito di social bookmarking che permette di memorizzare gli URL delle proprie pagine preferite in uno spazio personale, al quale è possibile accedere da qualunque computer collegato alla rete. I propri preferiti sono inoltre visibili da qualunque altro utente e possono essere esplicitamente condivisi con contatti rilevanti.

Caratteristica dell'applicazione è la possibilità di taggare le pagine memorizzate per classificarle e identificarle, facilitandone il successivo reperimento. È inoltre possibile utilizzare i tag come “chiavi” per esplorare i segnalibri memorizzati da altri utenti. Del.icio.us presenta varie opzioni per organizzare i tag: aggregazione in gruppi, ordinamento secondo criteri alfabetici e di frequenza, visualizzazione in modalità “tag cloud⁷⁰” o “lista”, oppure basati sul numero di occorrenze.

Questo applicativo si caratterizza per essere uno strumento che offre vari servizi, ma che dipende totalmente, per quanto riguarda i contenuti, dal contributo degli utenti. È un servizio gratuito; l'utente può infatti esplorare liberamente i contenuti inseriti da altri, ma deve effettuare la registrazione per poter memorizzare i propri preferiti.

Per aggiungere un segnalibro, gli utenti possono seguire due procedure: selezionare il bottone “post” dalla propria pagina personale, oppure cliccare sul bottone “post to my del.icio.us” che compare tra i collegamenti nella finestra del browser, dopo essersi registrati al servizio.

I punti di forza di del.icio.us sono due:

- la completa accessibilità, a patto che si disponga di un computer connesso;
- il tagging.

Gli utenti dell'applicazione possono infatti accedere ai propri segnalibri da qualunque computer ed esplorare le liste di altre persone a partire dai tag che queste hanno utilizzato. In tal modo, è possibile accedere con facilità a risorse riguardanti argomenti di proprio interesse scoperte e memorizzate da altri utenti.

Rispetto ai tradizionali metodi di classificazione tramite cartelle, il tagging non obbliga gli utenti a ricordare esattamente il “luogo” in cui una risorsa era stata memorizzata, o a scorrere lunghe liste alla ricerca di un elemento di interesse. Al contrario, i preferiti vengono esplorati sulla base di un criterio semantico.

Gli utenti possono inoltre assegnare un numero indefinito di tag a ogni risorsa, permettendo di effettuare una successiva ricerca sulla base di molteplici criteri. I tag di ciascun utente sono visualizzati in uno spazio sulla destra dello schermo e possono essere organizzati in vario modo. Ad esempio, un utente può decidere di riunire i tag correlati in un unico “bundle”; concettualmente, si tratta di un'opzione simile alle tradizionali cartelle/directory, ma la particolarità è che, grazie ai tag, una certa risorsa può essere classificata in più gruppi.

Graficamente, i tag possono essere rappresentati tramite semplici liste o tramite “tag clouds”; queste ultime offrono una visualizzazione compatta, in cui i tag più utilizzati sono resi con un carattere più grande ed evidente.

È inoltre possibile utilizzare uno spazio detto “inbox” per “abbonarsi” ad un tag (presumibilmente collegato ad un argomento di interesse) ed essere aggiornati ogni volta che un utente lo associa a un segnalibro. Analogamente, è possibile monitorare le attività di altri utenti.

Del.icio.us mette inoltre a disposizione feed RSS per ogni pagina, cosicché gli utenti possono essere aggiornati in

⁷⁰ Modalità di visualizzazione dei tag. Per un approfondimento sul concetto e sull'applicativo di riferimento si veda il capitolo 6.9.

tempo reale sulle novità che riguardano singoli tag, combinazioni di tag, utenti o la propria inbox.

Pur possedendo varie caratteristiche che supportano l'usabilità, tipiche delle applicazioni Web 2.0, del.icio.us si presenta con un'interfaccia poco esplicativa, che costringe l'utente a esplorare le varie funzionalità, prima di arrivare a comprendere pienamente le potenzialità messe a disposizione dal sistema e il modo per sfruttare.

Sulla base delle osservazioni condotte, del.icio.us si qualifica come applicazione Web 2.0 in quanto possiede gran parte delle caratteristiche individuate come rappresentative di questo approccio. Si è già osservato come una delle sue caratteristiche principali sia la quasi totale dipendenza dai contenuti inseriti dall'utente: l'utilità di del.icio.us risiede, per il singolo utente, nella possibilità di raccogliere i propri segnalibri e di visualizzarli, indipendentemente dal computer che sta utilizzando. A livello collettivo, tuttavia, è soprattutto la condivisione di tali contenuti a rappresentare il valore aggiunto del servizio: in tal modo, ciascun utente può trovare facilmente risorse correlate a temi di interesse, senza bisogno di effettuare una ricerca con un motore. Inoltre, il fatto che i contenuti recuperati siano stati salvati da altri utenti aggiunge a essi una sorta di garanzia sociale circa la loro bontà che non può essere presente nell'interazione con un classico motore di ricerca. Tale garanzia può sia essere generica ("qualcuno" ha trovato interessante un certo documento, quindi è probabile che abbia "una qualche" utilità), sia essere legata all'autorità (o alla funzione di modello) che uno specifico utente riveste in un certo ambito.

La dimensione sociale di del.icio.us si manifesta anche nello stimolare esplicitamente le relazioni tra gli utenti: è possibile, infatti, memorizzare i profili dei contatti che si ritengono rilevanti in una pagina specifica ("*your network*") e tenere traccia delle loro attività sul sito. Inoltre, è possibile condividere i propri preferiti con altri utenti, o cliccando semplicemente sullo username dell'utente, se questi appartiene al proprio network, o indicando tra i tag lo username dell'utente, preceduto da "*for*".

Elemento pienamente in linea con la filosofia del Web 2.0 è l'uso del tagging come mezzo per classificare e recuperare le risorse salvate. I tag di del.icio.us permettono di trasformare un insieme complesso di informazioni potenzialmente interessanti in una base di conoscenze facilmente esplorabile secondo logiche diverse e a partire da criteri di ricerca che rivestono un senso ben definito per lo specifico utente.

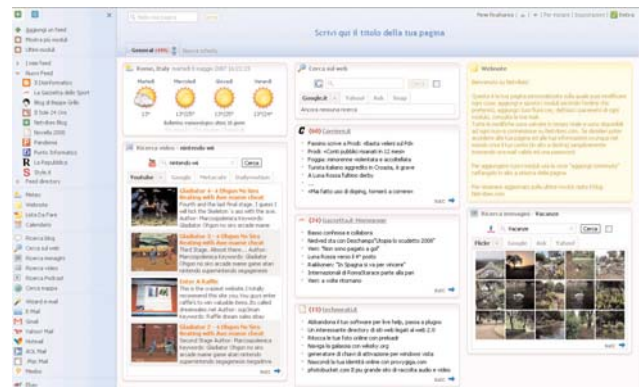
In un'ottica di riutilizzo delle informazioni e progettazione di nuove applicazioni, del.icio.us fornisce gratuitamente le proprie API che permettono di:

- controllare gli aggiornamenti (nuovi preferiti postati dagli utenti);
- ottenere una lista di tag e modificarla;
- ottenere una lista di post e modificarla;
- ottenere, creare e cancellare i raggruppamenti di tag creati da un certo utente.

6.2 NETVIBES

Applicazione web che, a partire dall'aggregazione di diversi feed, consente di creare la propria «start page» personalizzata

Figura 2: Netvibes - <http://www.netvibes.com/>



Netvibes è un'applicazione che permette di creare una home page personalizzata. Il servizio è gratuito e, una volta effettuata la registrazione, non richiede ulteriori operazioni di login, se l'utente accede alla propria pagina sempre dal medesimo computer (in questi casi, vengono infatti utilizzati i cookies per riconoscere l'utente). Il contenuto delle pagine personali è privato.

Caratteristica dell'applicazione è la possibilità di aggregare in un unico spazio contenuti di vario tipo: feed RSS, segnalibri, appunti, liste di cose da fare, motori di ricerca. L'utente ha inoltre la possibilità di gestire l'aspetto della pagina, spostando le varie finestre sul piano di lavoro e modificandone i colori.

Nel momento in cui un nuovo utente accede a Netvibes, gli viene presentata una pagina non personalizzata, ma già popolata con alcuni contenuti, permettendogli in tal modo di farsi rapidamente un'idea delle caratteristiche del servizio. Inoltre, è possibile provare immediatamente ad interagire con il servizio (ad esempio cancellando gli elementi che non interessano), cosicché la fase di apprendimento del funzionamento dell'interfaccia risulta molto rapida ed intuitiva.

Per aggiungere i propri contenuti, l'utente deve selezionare il bottone "*add content*" che compare in alto a sinistra e scegliere una specifica tipologia di contenuto dal menù che viene mostrato successivamente. Le opzioni offerte comprendono, tra l'altro, feed da fonti specificate dall'utente, feed da siti e blog promossi da Netvibes, vari motori di ricerca (Google, Yahoo, Ice Rocket, Wikipedia), le previsioni meteo, la propria posta (relativamente ad account Google o Yahoo), i bookmark di del.icio.us, spazi per scrivere i propri appunti, collegamenti a documenti di Writely, liste di cose da fare.

Per quanto riguarda in particolare l'aggiunta di feed, l'utente ha diverse possibilità: inserire direttamente l'indirizzo del file RSS che vuole utilizzare come fonte;



importare un file OPML⁷¹, aggiungere automaticamente, selezionando un'icona, un feed promosso da Netvibes, cercare un feed di interesse all'interno di una lista di directories.

Inoltre, l'utente può modificare l'aspetto della pagina utilizzando il pannello di controllo e specificando, per esempio, se vuole che vengano mostrati o meno il titolo della pagina e le favicons associate ad ogni contenuto.

È anche possibile creare, all'interno di una stessa pagina, più "schede" (tab) tra le quali suddividere i contenuti, semplicemente cliccando sul bottone "new tab" che compare accanto all'ultima scheda creata. L'utente può scegliere il nome, un'icona e il numero di colonne da visualizzare per ciascuna scheda e può, eventualmente, decidere di cancellarla.

Il servizio è disponibile, oltre che in inglese, in francese, tedesco, spagnolo, cinese, portoghese e hindi (sebbene non tutte le versioni sembrano effettivamente già funzionanti).

Netvibes rappresenta un buon esempio di progettazione conforme ai criteri che qualificano l'approccio Web 2.0, a partire dal fatto di proporsi nella sua versione beta.

Dal punto di vista dell'interfaccia, Netvibes sfrutta in modo intelligente il modello delle GUI per offrire agli utenti un ambiente familiare e usabile, in cui l'interazione si svolge principalmente grazie ad elementi quali icone, finestre, bottoni e menù. La fortunata metafora del desktop viene ripresa nell'impostazione generale dell'applicazione, laddove le varie finestre con i contenuti dell'utente sono disposte su una sorta di "scrivania". Le funzioni descritte sono abilitate dalle tecnologie AJAX.

In quanto "aggregatore" di contenuti, Netvibes si basa fortemente sull'idea di riuso dei dati, sfruttando il formato RSS e le API pubbliche per offrire informazioni e servizi (ad esempio, l'accesso alla propria casella di posta elettronica) provenienti da fonti diverse. In tal senso, non può prescindere da una concezione delle informazioni come bene pubblico e liberamente distribuito. Infatti, Netvibes stimola i propri utenti a condividere con gli altri le informazioni pubblicate su blog e siti personali, offrendo la possibilità di inserire in questi ultimi un bottone tramite il quale essi vengono aggiunti come "fonti" di Netvibes.

Inoltre, al servizio di pagine personali Netvibes affianca il cosiddetto "Netvibes ecosystem", tramite il quale gli utenti possono scambiarsi i moduli aggiuntivi che hanno progettato per arricchire le proprie pagine personali (ad esempio, erano disponibili moduli per la visualizzazione risultati delle partite dei Mondiali del Calcio in tempo reale). I moduli disponibili nel "Netvibes ecosystem" possono essere aggiunti velocemente alla propria pagina personale semplicemente cliccando su un bottone.

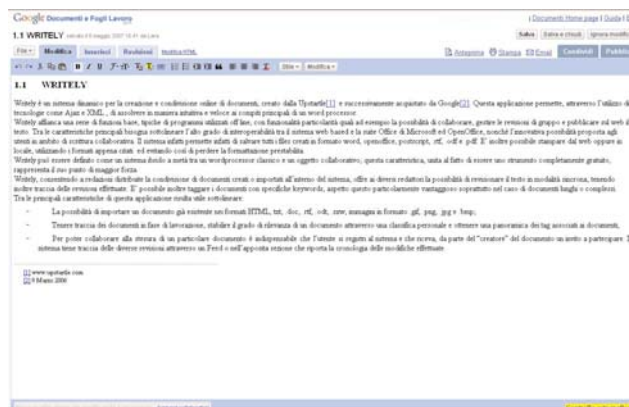
Inoltre, possono essere commentati da qualunque utente ed essere taggati dall'autore, permettendo di effettuare una ricerca sulla base di criteri semantici.

Tuttavia, proponendosi come pagina personale, Netvibes permette di controllare in modo molto restrittivo l'accesso al servizio, che dipende o dal possesso di una password, o dall'accesso al computer dell'utente, dove sono memorizzati i suoi cookies. Non è invece possibile configurare diversi gradi di accesso alla propria pagina personale.

6.3 WRITELY

Sistema dinamico per la creazione e condivisione online di documenti.

Figura 3: Writely - <http://docs.google.com/>



Questa applicazione, creata dalla Upstartle⁷² e successivamente acquistata da Google⁷³, permette di assolvere in maniera intuitiva e veloce ai compiti principali di un word processor attraverso l'utilizzo di tecnologie come Ajax e XML.

Writely affianca una serie di funzioni base, tipiche di programmi utilizzati off line, con funzionalità quali per esempio la possibilità di collaborare, gestire le revisioni di gruppo e pubblicare sul web il testo. Tra le caratteristiche principali bisogna sottolineare l'alto grado di interoperabilità tra il sistema web based e la suite Office di Microsoft ed OpenOffice, nonché l'innovativa possibilità proposta agli utenti in ambito di scrittura collaborativa. Il sistema infatti permette di salvare tutti i files creati in formato word, openoffice, postscript, .rtf, .odf e .pdf. È inoltre possibile stampare dal web oppure in locale utilizzando i formati appena citati ed evitando così di perdere la formattazione prestabilita.

Writely può essere definito come un sistema ibrido a metà tra un wordprocessor classico e un oggetto colla-

⁷¹ OPML, acronimo per *Outline Processor Markup Language*, è un formato basato su XML che viene generalmente utilizzato per esportare ed importare liste di feed tra diversi aggregatori.

⁷² www.upstartle.com

⁷³ 9 Marzo 2006. L'11 ottobre 2006 Writely è stato integrato a Google Spreadsheet, ed è stato così creato Google Docs & Spreadsheet.

borativo; questa caratteristica, unita al fatto di essere uno strumento completamente gratuito, rappresenta il suo punto di maggior forza.

Writely, consentendo a redazioni distribuite la condivisione di documenti creati o importati all'interno del sistema, offre ai diversi redattori la possibilità di revisionare il testo in modalità sincrona, tenendo inoltre traccia delle revisioni effettuate. È possibile inoltre taggare i documenti con specifiche keywords, aspetto questo particolarmente vantaggioso soprattutto nel caso di documenti lunghi o complessi.

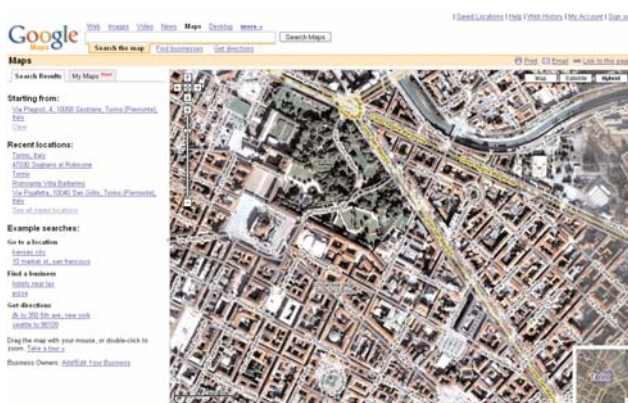
Tra le principali caratteristiche di questa applicazione risulta utile sottolineare la possibilità di:

- importare un documento già esistente nei formati HTML, txt, .doc, .rtf, .odt, .sxw, immagini in formato .gif, .png, .jpg e .bmp;
- tenere traccia dei documenti in fase di lavorazione, stabilire il grado di rilevanza di un documento attraverso una classifica personale e ottenere una panoramica dei tag associati ai documenti;
- collaborare alla stesura di un particolare documento, a patto che l'utente si registri al sistema e che riceva, da parte del "creatore" del documento un invito a partecipare. Il sistema tiene traccia delle diverse revisioni attraverso un feed o nell'apposita sezione che riporta la cronologia delle modifiche effettuate

6.4 GOOGLE MAPS

Sistema di ricerca e localizzazione di contenuti d'interesse su mappe geografiche.

Figura 4: Google Maps - <http://maps.google.com/>



Google Maps è l'innovativo servizio di Google che da qualche anno sta rivoluzionando il modo di presentare, attraverso la georeferenziazione, diverse tipologie di contenuto. Si tratta fondamentalmente della digitalizza-

zione di mappe digitali e vettoriali, liberamente navigabili via web o attraverso un software gratuito chiamato "Google Earth".

Le mappe presentate su Google Maps risultano avere un alto grado di interattività e riutilizzo. Ad oggi, sfruttando le API che Google ha reso libere⁷⁴, è stata infatti implementata una vastissima gamma di applicazioni.

Ad esempio integrando Google Maps con un sistema di archiviazione e condivisione di fotografie, come ad esempio Flickr, ha preso piede un fenomeno definito "memory maps". Vere e proprie mappe della memoria che consentono agli utenti di arricchire attraverso diversi contenuti la propria storia e identità personale, geolocalizzarla e condividerla con altri utenti.

Le modalità di utilizzo e interazione, come spesso accade per i prodotti marchiati Google, sono estremamente intuitive. Per quanto riguarda la referenziazione di una località all'interno della mappa la procedura risulta molto semplice: l'utente infatti deve inserire la località d'interesse all'interno di una form di ricerca per vedere immediatamente visualizzata la relativa porzione di mappa, all'interno della quale viene evidenziato il punto d'interesse. L'utente ha inoltre a disposizione strumenti di navigazione, quali per esempio lo zoom o il *drag and drop*, in grado di arricchire e potenziare l'esperienza finale di interazione con il sistema.

Google Maps mette a disposizione dell'utente una modalità alternativa di navigazione basata sui tasti di scelta rapida: utilizzando i tasti freccia è possibile spostarsi verso sinistra, destra, in alto e in basso; mentre con i tasti PagSu, PagGiù, Home e Fine è possibile allargare la panoramica e aumentare o ridurre lo zoom [attraverso o tasti (+) e meno (-)]. Come molte altre applicazioni Web 2.0, Google Maps ha implementato il servizio utilizzando la tecnologia AJAX, ottenendo così un'interfaccia estremamente usabile e interattiva⁷⁵.

I dati GIS⁷⁶ utilizzati per realizzare le mappe sono forniti da Tele Atlas⁷⁷ e da Navteq⁷⁸, mentre per quanto riguarda le immagini satellitari il fornitore a cui si è appoggiato Google è stato DigitalGlobe⁷⁹. L'immagine del globo, conosciuta con il nome *NaturalVue*, è stato derivato da Landsat 7, della MDA Federal.

Le possibili visualizzazioni di una porzione di territorio sono tre:

- **satellitare**: visualizzazione di fotografie satellitari ad alta risoluzione in alcuni casi in grado di rendere definiti anche dettagli di un metro;
- **mappa**: definizione politica (vettoriale) della cartina con il dettaglio di strade, autostrade e città;
- **ibrida**: entrambe le visualizzazioni sovrapposte

74_ <http://www.google.com/apis/maps/>

75_ AjaXSLT framework: <http://goog-ajaxslt.sourceforge.net/>

76_ GIS, o Geographic(al) Information System, è un sistema informativo geografico che permette l'acquisizione, la registrazione, l'analisi, la visualizzazione e la restituzione di informazioni derivanti da dati geografici (geo-referenziati).

77_ <http://www.teleatlas.com/>

78_ <http://www.navteq.com/>

79_ <http://www.digitalglobe.com/>



Il sistema oggi non consente di tracciare all'interno della mappa molteplici destinazioni ma supplisce a questa carenza attraverso l'integrazione con Google Earth⁸⁰, dove per esempio è possibile impostare diversi punti e calcolare le relative distanze. Il sistema consente inoltre di calcolare le coordinate di latitudine e longitudine di specifici punti selezionati e di stampare le mappe in diversi formati.

Grazie alla possibilità di visualizzare punti di interesse, viste in 3D in Quicktime VR, l'unione di Google Earth e Google Maps rende di fatto più semplice scegliere un percorso di navigazione permettendo di vederne un'anteprima a "volo d'uccello".

Una volta stabilito il percorso è possibile vederne le tappe in modalità stradale, fotografica o ibrida e stampare una versione adatta a viaggi "offline".

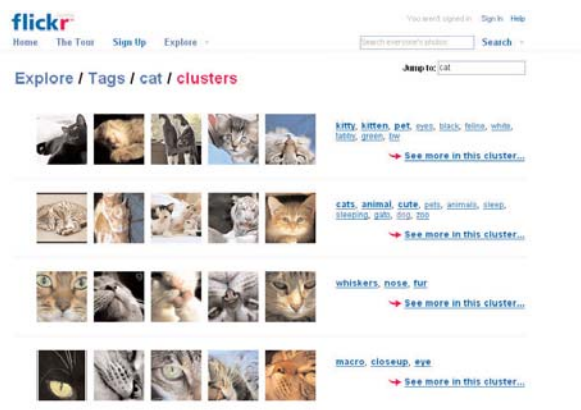
Da poco è anche possibile utilizzare una versione appositamente adattata per i dispositivi mobili, e fruire quindi di informazioni cartografiche in movimento. Gli sviluppi di questa tecnologia sono molto interessanti e porteranno in un futuro prossimo ad avere delle mappe di navigazione molto più realistiche e basate su dati in costante aggiornamento, offrendosi quindi al pubblico come un vero e proprio GIS che combina milioni di informazioni disponibili sul Web ad una base cartografica ad alta interattività.

Per quanto riguarda le modalità di ricerca Google Maps mette a disposizione dell'utente due opzioni. La prima, definita "Trova attività commerciale" consente di specificare l'"oggetto" della ricerca (ad esempio, biblioteca) e il relativo luogo. Così facendo per i diversi oggetti che soddisfano la ricerca effettuata il sistema posiziona all'interno della mappa un pin che, se selezionato, riporta una breve descrizione dell'oggetto in questione. La seconda opzione, definita "Ottieni informazioni", consente di visualizzare il percorso tra due punti e ottenere indicazioni dettagliate sul percorso. Selezionando una di queste il sistema fornisce un mappa ingrandita, un'immagine satellitare oppure ibrida del dettaglio oltre a stimare il tempo richiesto per coprire la distanza tra due punti specifici.

6.5 FLICKR

Applicazione per la raccolta e la condivisione di materiale fotografico.

Figura 5: Flickr - <http://www.flickr.com>



Flickr è un'applicazione che consente in maniera semplice e intuitiva di creare e condividere materiale fotografico. Si distingue dai sistemi tradizionali per il valore dell'interfaccia e per le modalità d'interazione con il sistema così come per alcune funzionalità (organizzazione dei contenuti, reperimento degli stessi, partecipazione, condivisione, ecc.) che la rendono una tra le applicazioni Web 2.0 di maggior successo.

Flickr è stato sviluppato dalla Ludicorp⁸¹, una compagnia di Vancouver fondata nel 2002. Nel marzo del 2005 la Ludicorp e Flickr sono stati acquisiti da Yahoo! I server si sono quindi trasferiti dal Canada agli Stati Uniti, e il sito adesso si adegua oggi alle leggi statunitensi.

Nato all'interno del progetto Game Neverending⁸² e conosciuto con il nome di FlickrLiva, Flickr era inizialmente focalizzato su di una chat room che consentiva agli utenti di scambiarsi fotografie in tempo reale. Solo successivamente sono state sviluppate le funzionalità del backend legate all'upload e all'archiviazione di materiale fotografico. Nel maggio del 2006, dopo un restyling a livello di funzionalità e di organizzazione dei contenuti, Flickr è passato dalla versione "Beta" a quella "Gamma". Flickr permette di inserire per ogni foto un titolo, i relativi tag e una breve descrizione, che risulta come il corpo di un post. Essendo un sistema di blogging fotografico consente inoltre ai diversi utenti di commentare le fotografie creando di fatto interazione e partecipazione. Flickr offre inoltre la possibilità di sottoscrivere un feed, ottenendo così gli aggiornamenti apportati dagli utenti, sia a livello di post sia di commenti.

Come detto in precedenza gli utenti registrati possono caricare le immagini all'interno del sistema e categoriz-

⁸⁰<http://earth.google.it/>

⁸¹<http://www.ludicorp.com/>

⁸²<http://www.gne.net/>

zarle attraverso l'utilizzo di keywords, o tag. È proprio per la modalità di gestione di questi ultimi che Flickr viene considerato il primo esempio di utilizzo effettivo della folksonomia. Gli utenti hanno inoltre diverse modalità per categorizzare le proprie fotografie, tra cui quella definita "set", ossia una galleria fotografica all'interno della quale far rientrare un insieme di foto definite dall'utente. Questo sistema di categorizzazione è più flessibile rispetto ai tradizionali sistemi *a cartelle o directory*: una fotografia può infatti appartenere a più set, a nessun set o a tutti. In questo senso i set di Flickr possono essere considerati una forma di categorizzazione basata sui metadati piuttosto che su di una rigida struttura gerarchica.

In Flickr sono proprio le fotografie, e non più gli album, le unità minime condivise con gli altri utenti. Le immagini possono poi essere raccolte in set, visualizzate attraverso uno slideshow, o caricate all'interno di gruppi (aperti anche al contributo di altri utenti).

Ed è proprio l'interconnettività che può nascere attorno ad una singola fotografia e ai suoi tag a rendere particolarmente interessante questo applicativo. Come già detto, infatti, gli utenti possono partecipare e contribuire all'operazione di organizzazione, e arricchimento, dei contenuti aggiungendo non solo commenti ma anche inserendo nuovi tag alle foto.

Una delle sezioni che maggiormente caratterizza Flickr come applicazione Web 2.0 è quella denominata "Organizr". Si tratta infatti di una web application utilizzata per organizzare le foto all'interno del proprio account. Organizr consente all'utente di modificare i tag, le descrizioni date alle singole foto o ai set interi attraverso un'interfaccia - sviluppata in AJAX - che riprende in quanto ad aspetto e funzionalità le caratteristiche dei sistemi desktop-based di gestione delle foto.

Un'altra delle principali caratteristiche di Flickr è quella relativa ai permessi di accesso attribuibili alle foto caricate all'interno del sistema. L'utente può infatti, per ogni foto, definire una particolare policy di accesso, considerando le foto pubbliche o private. Le foto etichettate come "private" sono visibili dall'utente stesso e dalle categorie di utenti, o "contacts", etichettati a loro volta come "family" o "friends".

Sempre in base ai permessi viene gestita un'altra delle funzioni "2.0", ossia partecipativa: le foto caricate all'interno del sistema sono infatti commentabili in diversi modi. In primo luogo ogni foto può essere commentata dagli altri utenti (sta all'utente stesso definire se questa operazione è fattibile da tutti gli utenti generici, purché registrati, o se invece è rivolta ai soli contatti). Gli utenti possono inoltre aggiungere personali "note" all'interno delle foto caricate nel sistema. Si tratta di uno strumen-

to semplice e intuitivo attraverso il quale segnalare, evidenziandole, specifiche porzioni all'interno della foto. Anche Flickr ha permesso agli utenti di creare applicazioni che sfruttano questo sistema e di includere in altre piattaforme le funzionalità di questa applicazione attraverso le API rese pubbliche. Questa operazione ha permesso la creazione di molti servizi anche per dispositivi mobili; Nokia, per esempio, ha sviluppato applicazioni per il suo sistema operativo Symbian, in grado di permettere un agevole inserimento sul proprio account Flickr delle foto scattate attraverso la fotocamera del cellulare.

Molti sistemi di blogging si interfacciano sia nativamente, sia attraverso l'utilizzo di specifici plugin con Flickr, permettendo una perfetta simbiosi tra piattaforme diverse. Queste sono solo alcune delle innumerevoli applicazioni (proprietarie e opensource) nate per interfacciarsi con questo applicativo che, proprio grazie al suo successo, sta permettendo di sviluppare piccoli modelli di business mai pensati prima attorno al mondo della fotografia amatoriale e non.

6.6 PANDORA

Web-radio che permette di creare stazioni radio a partire da brano o da un artista scelto dall'utente.

Figura 6: Pandora - <http://www.pandora.com>



Si tratta di una web-radio realizzata con la tecnologia Flash che, dato un punto di partenza (sia esso un artista o una singola canzone), genera un flusso musicale in streaming⁸³. Il sistema, partendo dal primo parametro inserito, propone le successive canzoni in modo che si avvicinino il più possibile alle qualità musicali caratterizzanti il brano inserito dall'utente.

L'idea alla base, sviluppata all'interno del Music Genome Project, è quella di catturare l'essenza della musica al suo livello più basso, quello dei suoni che la compongono. Nel corso degli ultimi sei anni il gruppo di ricerca del Genome Project ha analizzato le canzoni di oltre 10.000 artisti, da quelli più popolari a quelli cosiddetti underground, identificando centinaia di attributi in un vero e proprio genoma musicale. Attraverso questi geni, che descrivono tutti gli attributi

⁸³ A partire da maggio 2007 il servizio è accessibile solo per gli utenti statunitensi. Tim Westergren, il fondatore di Pandora, ha infatti dichiarato che le pressioni delle major così come i rischi di denunce per la violazione dei diritti d'autore stavano diventando troppo forti. La radio potrà continuare a trasmettere negli Stati Uniti grazie al Digital Millennium Copyright Act che, a differenza di quanto avviene in Europa e in Asia, permette alle radio online di trasmettere musica in streaming senza violare i diritti d'autore.



che caratterizzano una canzone (melodia, ritmo, strumentazione, arrangiamento) è possibile non solo descrivere un brano ma anche identificare le similarità tra due canzoni di artisti diversi, senza per questo ricorrere a classificazioni predeterminate.

Per fare in modo che lo streaming sia il più vicino possibile ai gusti dell'utente, Pandora consente inoltre di esplicitare un giudizio sui brani che vengono trasmessi. L'utente può infatti per ogni canzone dire se questa è di suo gradimento o meno, inserirla nella lista dei preferiti o saltarla. Tutte queste operazioni sono possibili grazie ad un'interfaccia sviluppata in Flash che rende ogni operazione estremamente semplice e intuitiva. Questa funzionalità consente al sistema di affinare sempre più la selezione di canzoni trasmesse e all'utente di disporre di un canale radio sempre più personalizzato. Ogni utente può creare più stazioni personalizzate (fino a 100) che hanno come punto di partenza un input particolare: questo permette di memorizzare le preferenze e le modifiche apportate alle singole stazioni musicali e di poter "risintonizzarsi" su una determinata stazione cambiando genere musicale.

Per quanto riguarda i requisiti tecnologici, è indispensabile una connessione Internet di almeno 500 Kbps dal momento che Pandora trasmette audio in streaming di alta qualità. Se la connessione è lenta o la rete è congestionata, infatti, il flusso musicale non risulta lineare.

Grazie alla funzione "Archiviazione Locale" sul player di macromedia Flash, Pandora tiene traccia della musica che viene trasmessa all'utente, ricordandone le preferenze e garantendo che tali informazioni non verranno utilizzate con altri fini.

I brani trasmessi su Pandora vengono fatti ascoltare in versione integrale e per ognuno è possibile sapere perché è stato inserito all'interno del canale radio (vengono infatti descritti i geni caratterizzanti uno specifico brano) oltre che inserirlo a sua volta all'interno di una lista di "preferiti". Ogni utente ha infatti una propria pagina personale all'interno della quale vengono raccolte tutte le canzoni segnalate. Questa lista oltre ad essere visibile agli altri utenti di Pandora, può essere inviata via mail oppure direttamente pubblicata su di un blog.

Il sistema offre la possibilità di accedere a una serie di informazioni aggiuntive sulla canzone, sull'album o sull'artista di riferimento e fornisce il link direttamente ad Amazon e iTunes per l'acquisto del cd o del singolo brano. Recentemente è stata aggiunta una sezione denominata "Backstage" all'interno della quale vengono raccolte le schede descrittive delle canzoni e degli artisti, di cui vengono proposti biografia e discografia. Selezionando uno specifico album vengono elencate le diverse tracce che lo compongono e di cui è possibile ascoltare un breve estratto e leggere la scheda relativa al suo genoma.

6.7 WORDPRESS E TECHNORATI

Piattaforma di blogging opensource e motore di ricerca online basato sui feed RSS creati dai blog

Figura 7: Technorati - <http://www.technorati.com/>



Figura 8: Wordpress - <http://wordpress.com/>



Wordpress è una tra le più diffuse piattaforme open-source per la creazione di blog. Di fatto è uno tra i migliori sistemi di content-management al mondo, per facilità d'uso, diffusione e consistenza dello sviluppo. Come tutte le piattaforme di blogging, Wordpress permette la creazione (singola o collettiva) di contenuti online, la loro categorizzazione e condivisione con altri utenti, che non sono soggetti passivi ma hanno la facoltà – ove consentito – di commentare quanto scritto dall'autore/dagli autori e la loro ricerca full-text.

Ciò che rende Wordpress leader nel suo ambito è sicuramente la base di sviluppo che da sempre la segue: come molti software open-source, infatti, Wordpress può contare su un lunghissimo elenco (in costante crescita) di plugin in grado di ampliarne le funzionalità e le capacità di relazione con altri servizi online. Attraverso i plugin, per esempio, è facile integrare – con una notevole raffinatezza – un blog con le immagini di Flickr, con le mappe interattive di Google e con Technorati.

Quest'ultimo, in particolare, è un motore di ricerca online che utilizza come base informativa i feed RSS generati dai blog (Wordpress li genera automaticamente, inclusi i sub-feed per ciascuna categoria di ogni blog) e i tag apposti a ciascun post. Grazie a questa struttura di base, Technorati è in grado di fornire informazioni pertinenti su temi "caldi", cioè le keyword più aggiornate dai blog a livello planetario, correlare e aggregare contenuti diversi e consentire agli utenti di ricercare l'intero corpus di contenuti attraverso una ricerca full-text e una ricerca per tag.

Wordpress, essendo una piattaforma per la creazione di blog, meriterebbe automaticamente di essere archiviata tra le applicazioni Web 2.0. Nei fatti, tuttavia, si distingue per alcune caratteristiche fondamentali:

- genera automaticamente feed RSS per tutti i post e per ciascuna categoria di post;

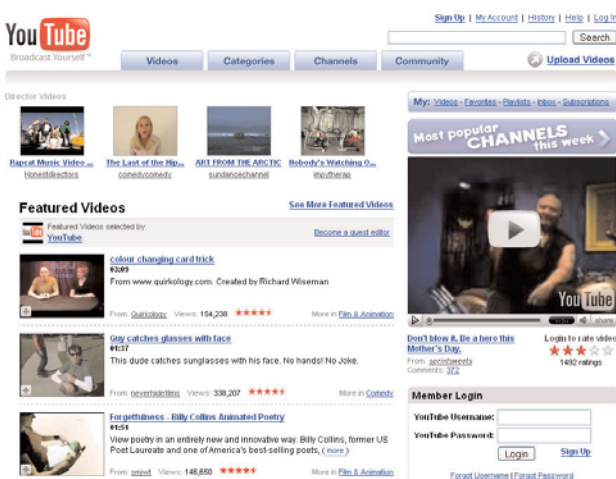
- permette, attraverso appositi plugin, di associare facilmente tag a ciascun post e si integra con altrettanta facilità con servizio come Technorati, che tengono traccia dei tag utilizzati;
- è un sistema aperto all'integrazione con API di terze parti (mappe interattive, gallerie fotografiche, ecc.) attraverso plugin gratuiti;
- permette agli utenti una notevole interattività, sia attraverso lo strumento dei commenti, sia attraverso una felice gestione degli accessi al backend dei blog, che consente di creare numerosi livelli di "iscritti" al sito (amministratori, autori, redattori, commentatori certificati, lettori, ecc.);
- dispone nativamente di strumenti che facilitano il riuso e la condivisione delle informazioni (ping e track-back);
- dispone di un'interfaccia – nel backend – di facilissimo utilizzo, agevolata – nella recente versione 2.0 – dall'integrazione di piccole parti in Ajax che consentono, per esempio, l'immissione di immagini nei post attraverso un'interfaccia "what you see is what you get".

Technorati si presenta come il motore di ricerca del mondo Web 2.0. Essendo basato su feed RSS e tag, in effetti, svolge una funzione estrema di aggregatore globale, in grado di associare decine di migliaia di fonti attraverso i tag.

6.8 YOUTUBE

Sistema per l'archiviazione e la condivisione di videoclip

Figura 9 : Youtube - <http://www.youtube.com>



Youtube è per i video ciò che Flickr è per le immagini: una galleria di videoclip condivisi online, associati a tag apposti dall'autore e a commenti da parte dei visitatori. La caratteristica fondamentale di YouTube, che recentemente si è confermato come servizio video più ricco e frequentato del Web, con oltre 100 milioni di video visualizzati ogni giorno (fonte: Hitwise), è la semplicità d'interfaccia, unita ad un'assoluta trasparenza tecnologica. Gli utenti, infatti, dopo una semplice registrazione pos-

sono immettere nel sistema i propri video attraverso un'interfaccia di uploading estremamente semplice. Il vero elemento critico della condivisione di video, cioè la conversione dei formati, avviene in maniera del tutto trasparente. YouTube, infatti, dispone di sistemi di conversione così raffinati ed efficaci da riconoscere numerose tipologie di file video e diversi tipi di encoding. In questo modo il servizio può accettare contenuti provenienti da fonti diverse, da hardware diversi e di qualità variabile. La conversione, inoltre, è un task che avviene in presenza dell'utente, che può valutare immediatamente la qualità e il funzionamento del video pubblicato online. Ad una forte vocazione alla semplicità tecnologica fa eco un'altrettanto forte predisposizione alla condivisione e alla relazione tra utenti. I video, infatti, non solo possono essere commentati da tutti gli utenti registrati, ma consentono anche di stabilire – tra utenti – una "risposta video", di fatto alimentando la nascita di thread di discussione interamente basati sul video.

Si tratta di un'applicazione Web 2.0 perché:

- permette di condividere contenuto attraverso una piattaforma unica dotata di strumenti di folksonomy (i tag)
- consente forme di interattività agli utenti attraverso i commenti
- rende fruibile il contenuto da tutti, attraverso l'uso di un player multimediale universale, compatibile con i principali browser e sistemi operativi
- favorisce la disseminazione di contenuti e il loro riuso su altre piattaforme, grazie alla versatilità del player multimediale, che può essere facilmente integrato nei blog, con un semplice copia-incolla di una stringa di testo (presentata con evidenza a fianco di ciascun video)
- genera feed RSS per le categorie di video (i più popolari, i più recenti, ecc.) e soprattutto per ciascun tag associato ad essi (es. tutti i video taggati con "calcio")
- dispone di un'interfaccia di pubblicazione estremamente facile, che solleva gli utenti da qualsiasi preoccupazione in merito al formato dei video condivisi: seguendo il wizard di pubblicazione, è sufficiente indicare al servizio il file che si desidera condividere e automaticamente Youtube provvederà alla conversione.

6.9 TAG CLOUD

Applicazione che analizza i feed RSS generando una rappresentazione grafica dei tag più usati.

Figura 10:

Tagcloud - <http://www.tagcloud.com/>



Le tagclouds non sono altro che rappresentazioni grafiche del sistema di tag presenti all'interno di un blog o di un sito. Di fatto sono "nuvole" di parole chiave, in cui la centralità, la dimensione e la posizione di ciascuna di esse ne definiscono la rilevanza all'interno dell'insieme di contenuti.



Il vantaggio delle tagcloud è rappresentare in modo grafico ed intuitivo il flusso di contenuti presente in blog, siti e servizi. Cliccando su ciascuna voce di una tagcloud, si accede ad un elenco cronologico di tutti i contenuti taggati con quella parola.

Rispetto a un elenco ordinato alfabeticamente, il vantaggio è la strutturazione automatica di una mappa concettuale in cui tutti i contenuti hanno pari dignità di presentazione (cioè non si sviluppano i classici casi di persistenza e primacy, tipici degli elenchi e inoltre – grazie alla dimensione del loro font – veicolano un'informazione in più: la rilevanza).

Il nome Tag Clouds, tuttavia, nasconde numerosi servizi con un'impostazione simile: si tratta di strumenti in grado di analizzare i contenuti di un sito Web e da esso estrarre le parole più rilevanti e più occorrenti, in modo tale da costruire uno schema grafico che le rappresenti. Normalmente questi sistemi, come TagCloud⁸⁴ per esempio, operano analizzando i feed RSS dei siti e permettono agli utenti di stilare una blacklist personalizzata di parole da escludere nel conteggio (solitamente si mettono all'indice preposizioni, congiunzioni, articoli, avverbi, ecc. e le "nuvole" sono basate solo sui nomi e sui verbi).

Allo stesso tempo, i servizi generano una tagcloud grafica e permettono di integrarla nel proprio sito o blog, con una procedura solitamente molto semplice. I servizi, infatti, generano automaticamente il codice che raffigura la nuvola; con una semplice procedura di copia e incolla nel codice del proprio blog, è facile integrare le tagcloud in una barra laterale o nel footer, per esempio.

Le tagclouds, tuttavia, non sono solo un servizio ma una modalità di fruizione dei contenuti. Servizi come Flickr e Technorati, infatti, dispongono di pagine in cui i tag più rilevanti sono graficamente mostrati sotto forma di tagcloud e garantiscono agli utenti una navigazione ragionata e pertinente all'interno del corpus di contenuti.

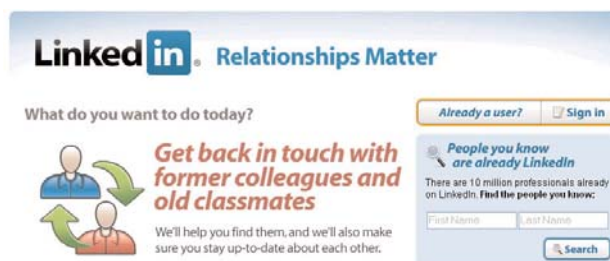
Le tagclouds appartengono al Web 2.0 perché:

- integrano il sistema di tag e di fatto ne costituiscono un'estensione;
- favoriscono l'accesso ragionato ai tag;
- mettono in relazione i contenuti, abbandonando l'artificiale ordine alfabetico o cronologico.

6.10 LINKEDIN

Sistema di social Networking finalizzato al business.

Figura 11: LinkedIn : - <http://www.linkedin.com/>



LinkedIn è un esempio applicativo di social networking utilizzato per finalità di business. Con più di 6 milioni di utenti e 130 industrie registrate LinkedIn viene utilizzato per ricerche di personale, proposte di lavoro e di progetti. L'idea dietro a questa business community è la teoria dei six degrees, secondo la quale si potrebbe raggiungere una qualsiasi persona passando per non più di cinque persone che si conoscono a vicenda⁸⁵. LinkedIn però non si limita a mettere in contatto le persone ma, con il supporto di strumenti estremamente semplici ed efficaci, tiene traccia dei percorsi attraverso i quali le persone possono raggiungersi e potenzia la rete di contatti che si viene a creare.

La registrazione, gratuita ma indispensabile per poter accedere al servizio, prevede una breve profilazione, basata principalmente sul percorso di studi e sulle esperienze professionali. Una volta registrato l'utente accede direttamente alla propria home page personalizzata: all'interno della sezione "account & setting" l'utente può inserire e modificare le informazioni personali e definire le modalità attraverso le quali essere ricercati dal resto della comunità.

Ciò che è particolarmente efficace in LinkedIn è la raffigurazione del network sociale di ogni utente. Questo, infatti, è navigabile attraverso un semplice elenco testuale, ma può anche essere esplorato attraverso un potente strumento che consente di limitare la ricerca su competenze o persone ad uno specifico grado di separazione. Un utente, quindi, può cercare collaboratori tra coloro che compongono il suo secondo grado di relazione, oppure può allargare lo spettro e fare ricerche più ampie. Durante tutto il percorso, LinkedIn tiene traccia in tempo reale del numero di contatti disponibile per ogni grado di separazione, di fatto rappresentando all'utente la "forza" della sua rete sociale.

Per quanto riguarda le operazioni di ricerca, oltre al motore interno al data base, l'utente può effettuarle sulla base delle persone (find people), delle opportunità di lavoro (find jobs) o sulla base di fornitori e consulenti (find services).

⁸⁴ www.tagcloud.com

⁸⁵ Si faccia riferimento alla tavola 5 di pag. 31



La rete di contatti è al centro del servizio offerto da LinkedIn e la sua costruzione è forse l'aspetto più interessante dell'intero applicativo.

L'utente può valutare i diversi contatti e le connessioni in base alle relazioni, alle referenze e alla reputazione. La relazione indica a che distanza si trova il contatto selezionato. LinkedIn prende in considerazione le relazioni che arrivano fino al terzo livello, partendo dal presupposto che oltre questo i legami sono troppo deboli. La relazione di primo livello rappresenta un contatto diretto mentre per quella di secondo si tratta del "contatto di un contatto". Le referenze sono impressioni personali dei contatti di primo e secondo livello su di un utente particolare mentre la reputazione deriva dal "sostegno" (endorsement) che datori di lavoro, colleghi, clienti, maestri, hanno attribuito a quella persona.

Ciascun utente, quindi, all'atto dell'iscrizione a LinkedIn può creare una cerchia di amici e conoscenti con cui condividere il proprio profilo personale, il proprio curriculum vitae e le proprie esperienze lavorative. Grazie alla teoria dei gradi di separazione, ciascun utente disporrà a sua volta di una cerchia di relazioni. La somma di queste relazioni, che cresce esponenzialmente mano a mano che crescono i gradi di separazione, permette ai singoli utenti di entrare in contatto con numerose persone qualificate e – grazie alle conoscenze comuni – attivare una comunicazione reciproca basata sulla fiducia e sull'autorevolezza dei "garanti" posti in mezzo tra i due capi della catena relazionale.

Il principio sviluppato da LinkedIn parte da una considerazione particolare fatta sul concetto di Rete, intesa come insieme di nodi interconnessi. Ogni nodo è caratterizzato da un proprio livello di competizione, o fitness, che influenza la capacità di "attrarre" altri nodi all'interno della rete. La competizione che si viene quindi a creare tra i nodi si basa sulla capacità di avere il numero massimo di link connessi. Secondo questa prospettiva, sono proprio i nodi più forti a tenere insieme la rete.

Come detto in precedenza una delle caratteristiche del Web 2.0 è proprio quella di basarsi sulla creazione di comunità, permettendo così la formazione di reti di contatti e il potenziamento delle connessioni. Nell'ambito della creazione di social network, LinkedIn rappresenta uno dei casi di maggior successo non solo per il numero di utenti che hanno aderito al servizio ma soprattutto per la modalità attraverso la quale LinkedIn ha saputo sfruttare la Rete e le sue potenzialità. Nei fatti, LinkedIn è una grande piattaforma di hiring, in cui domanda e offerta di lavoro (libero e dipendente) si incontrano su una base non casuale.

LinkedIn è un caso particolare di applicazione Web 2.0, perché:

- permette la condivisione di contenuti (nello specifico i curricula vitae), ma non la loro modifica da parte di terzi;
- pone come oggetto della condivisione la cerchia di relazioni che ogni individuo reca con sé: mettere in comune le relazioni significa sviluppare circuiti posi-

tivi in cui aumenta la fiducia tra soggetti e si riducono le distanze sociali tra attori;

- invoglia la collaborazione – sebbene non produttiva, ma relazionale – tra individui e facilita la creazione di relazioni basate sulla fiducia e sulla conoscenza;
- sviluppa sistemi di valutazione – sebbene non diretti – grazie allo strumento dell'endorsement (la raccomandazione che ciascun utente può associare a ciascun componente della propria cerchia di conoscenze).
- sviluppa sistemi di valutazione – sebbene non diretti – grazie allo strumento dell'endorsement: la raccomandazione che ciascun utente può associare a ciascun componente della propria cerchia di conoscenze).

ALLEGATO 2

ANALISI COMPARATIVA
DI ALCUNI FRAMEWORK AJAX MATURI

a2

ANALISI COMPARATIVA DI ALCUNI FRAMEWORK AJAX MATURI



7.1 INTRODUZIONE

AJAX è un approccio innovativo allo sviluppo di applicazioni per il Web. Non si tratta propriamente di una tecnologia a se ma piuttosto di un modo di combinare diverse tecnologie esistenti al fine di ottenere significativi vantaggi nella realizzazione di applicazioni interattive via web.

L'acronimo AJAX sta per **Asynchronous JavaScript and XML** ed è stato introdotto per la prima volta da Jesse Garrett in [Garrett] nel 2005.

AJAX combina le seguenti tecnologie:

- uso di XHTML (o HTML) e CSS per fornire una veste grafica all'interfaccia utente;
- accesso e manipolazione del DOM (Document Object Model) per leggere o manipolare in modo dinamico gli elementi che compongono l'interfaccia utente;
- uso di linguaggi eseguiti lato client all'interno del browser come JavaScript per la manipolazione del DOM;
- uso dell'oggetto run-time XMLHttpRequest (messo a disposizione dal browser) per poter scambiare informazioni con un server in modo asincrono;
- uso di un formato testuale facilmente parsificabile da una macchina (tipicamente XML) come strumento per lo scambio di strutture dati complesse.

7.1.1 Breve storia

La possibilità di interagire dinamicamente con il documento attraverso la manipolazione del DOM da parte degli script è una tecnica già affermata dal 1998 e andava sotto il nome di **DHTML (Dynamic HTML)**.

Esattamente come per AJAX, DHTML non rappresenta una tecnologia ma un insieme di tecnologie. L'obiettivo di DHTML è quello di rendere dinamica una pagina statica, sfruttando lo scripting lato client. Per pagina statica non si intende in senso stretto una pagina staticamente disponibile su un server, infatti può anche essere una pagina generata dinamicamente da uno script lato server. Con la manipolazione del DOM e l'utilizzo dei CSS, uno script lato client è in grado di alterare dinamicamente molti elementi della pagina senza richiedere alcuna interazione con il server. DHTML è stato inizialmente sfruttato per aggiungere effetti particolari alle pagine web, tra questi quello sicuramente più famoso è il *rollover*, che cambia dinamicamente un'immagine al passaggio del mouse sopra di essa.

La vera rivoluzione di AJAX è iniziata quando nel 1996 Microsoft ha introdotto il tag **IFRAME** all'interno di Internet Explorer 3 e con la successiva introduzione (1997) del tag **LAYER** da parte di Netscape in Navigator 4. Questi tag permettevano di includere pagine o codice Javascript esterni, per fornire gli effetti di dinamicità tipici delle moderne applicazioni AJAX. Successivamente si sono diffuse tecnologie proprietarie, come quella di Macromedia Flash, che già dalla versione 4 il suo player era in grado di scambiare dati in formato XML con un server.

Microsoft ha poi continuato a raffinare questa tecnica sino ad arrivare alla realizzazione di un oggetto **ActiveX** denominato XMLHTTP e che è stato aggiunto alla versione 4 di Internet Explorer. Microsoft ha sperimentato la prima volta questa tecnica, sviluppando **Outlook Web Access** fornito con Microsoft Exchange Server 2000. Successivamente altri browser, tra cui Safari e Mozilla, hanno messo a disposizione un oggetto Javascript con le medesime funzionalità e la stessa interfaccia (API), aspetto questo che ha sostanzialmente contribuito alla rapidissima diffusione delle applicazioni AJAX.

7.1.2 Funzionamento di AJAX

AJAX rivoluziona l'approccio allo sviluppo delle applicazioni web, introducendo un modello di interazione che prevede l'utilizzo di un'unica pagina per la realizzazione di tutta l'interfaccia grafica.

L'approccio classico usa massicciamente lo scripting lato server che si occupa di produrre gli elementi di interfaccia grafica in base al contenuto di un database o rispetto alle azioni precedentemente effettuate dall'utente e delle quali viene tenuto traccia attraverso un identificativo di sessione (mantenuto ad esempio all'interno dei cookie). In questa situazione al browser (client) viene lasciato pochissimo lavoro, generalmente il puro rendering grafico della pagina e in alcuni casi l'uso di tecniche DHTML per aggiungere effetti particolari alla pagina stessa.

In questo approccio, ogni componente della GUI deve essere supportata da una pagina html diversa, la quale viene generata dal server. L'esempio forse più classico è quello di un'applicazione Web che deve visualizzare un elenco di item presi da un database e successivamente (su richiesta dell'utente) visualizzare i dettagli dell'item selezionato. In genere questo tipo di interfaccia si articola su due pagine html: la prima genera l'elenco degli item



e la seconda viene invocata per fornire i dettagli su un determinato item.

Nell'approccio AJAX, si lavora su una sola pagina che si adatta dinamicamente in risposta all'interazione dell'utente o ad altri eventi non generati dall'utente (come, ad esempio, il trascorrere del tempo). Tornando all'esempio fatto in precedenza, con AJAX non risulta necessario sviluppare pagine diverse per fornire una vista sugli item e una vista sui dettagli di ciascuno di essi. Piuttosto si sviluppa una pagina dinamica che gestisce gli eventi dell'utente e in corrispondenza della richiesta di dettagli su un singolo item, richiede al server tali informazioni. Quando i dati sul singolo item sono stati ricevuti, lo script lato client, adatta l'interfaccia in modo da visualizzare i dati richiesti dall'utente, ad esempio visualizzando un pop-up. In pratica lo sviluppatore non deve gestire la generazione di un'interfaccia per ciascuna funzionalità. Nell'esempio fatto, lo script lato server che genera i dettagli dell'item, semplicemente consulta il DB e restituisce un testo parsificabile (tipicamente XML) con il risultato della query. Dal punto di vista dell'utente, invece, l'applicazione risulta più dinamica dato che non avviene il caricamento di una nuova pagina.

Per essere precisi, AJAX non elimina il problema della comunicazione con il server ma cambia la modalità con la quale questo avviene, al fine di raggiungere due importanti obiettivi:

- **riduzione dei dati scambiati con il server:** tali dati sono informazioni strutturate molto coincise, a differenza dalle pagine html che possono contenere diversi oggetti (come immagini e javascript);
- **disponibilità dell'interfaccia:** l'utente è in grado di interagire con l'interfaccia durante lo scambio dei dati con il server (modalità asincrona), a differenza della situazione classica (sincrona) in cui durante la ricezione della nuova pagina, l'utente non può fare altro che attendere che tutti gli elementi siano stati scaricati.

Risulta chiaro che il meccanismo di scambio asincrono delle informazioni da solo non permetterebbe la realizzazione dell'interfaccia grafica. Ad esso occorre affiancare lo **standard del DOM** e il **linguaggio Javascript** che permettono di manipolare l'interfaccia, interagire con l'utente e gestire lo scambio di dati con il server.

7.2 ELEMENTI DI AJAX

7.2.1 XMLHttpRequest

L'oggetto XMLHttpRequest permette a un codice Javascript di poter scambiare dati con un server. Per far questo, un oggetto XMLHttpRequest fornisce i seguenti metodi e attributi.

Metodi	
Sintassi	Descrizione
open (method,url[,mode,user,password])	Apri una connessione verso una determinata url. Tipicamente questi url rappresenta uno script lato server, ma potrebbe essere anche una risorsa statica. Il parametro method è il metodo HTTP da usare, tipicamente "GET" oppure "POST" . Infine, il parametro mode è un valore booleano che specifica (se true) che l'operazione deve avvenire in modo asincrono. La modalità sincrona serve per assicurarsi di ricevere i dati dal server prima di continuare una determinata operazione. I parametri user e password servono in caso url sia protetto con il sistema di autenticazione http.
abort()	Annula un'operazione in corso.
send (data)	Questo metodo avvia di fatto l'operazione. Se il metodo è di tipo "GET" allora il parametro data è posto a null . Altrimenti questo parametro contiene i dati che sono inviati al server (tipicamente un file testuale formattato in XML contenente i parametri di una certa operazione).
getAllResponseHeaders()	Restituisce gli header della risposta http ricevuta dal server.
getResponseHeader (header)	Restituisce il contenuto dell'header il cui nome è specificato nel parametro header .
setRequestHeader (name,value)	Permette di aggiungere un header alla richiesta, prima che sia inviata al server. Il parametro name contiene il nome dell'header mentre il value contiene il valore.
Attributi	
Sintassi	Descrizione
readyState	Questo è un codice numerico che va da 0 a 4 e che l'applicazione legge per sapere in che stato si trova la richiesta effettuata precedentemente: - 0 = non inizializzato; - 1 = aperto; - 2 = richiesta inviata; - 3 = risposta in ricezione; - 4 = risposta ricevuta.
status	Questo è il codice secondo lo standard HTTP relativo all'esito dell'operazione. Il codice 200 indica che l'operazione si è conclusa correttamente (l'operazione http si intende). Mentre 404 indica che la url inserita non esiste.
statusText	Contiene il messaggio di stato http, ad esempio "Not Found"
responseText	Stringa con il contenuto ricevuto dal server. Questo può essere semplice testo oppure avere un formato parsificabile.
responseXML	Quando il server restituisce un contenuto in formato XML, è possibile accedere al contenuto del documento attraverso le API DOM. Questo attributo contiene il riferimento al nodo DOM radice.
onreadystatechange	Questo attributo viene impostato dal client per specificare il nome di una funzione javascript (hook). Questa funzione viene richiamata ogni volta che readyState viene modificata. Tipicamente viene impostata una funzione che elabora i dati ricevuti dal server per poi effettuare delle modifiche all'interfaccia utente.



```
if (window.XMLHttpRequest) { // Mozilla, Safari, ...
    http_request = new XMLHttpRequest();
} else if (window.ActiveXObject) { // IE
    http_request = new ActiveXObject("Microsoft.XMLHTTP");
}
```

Il secondo passo consiste nel definire la funzione di hook, che viene chiamata ogni volta che lo stato dell'operazione cambia. Questa è la parte più complessa del codice, dato che deve gestire tutte le possibili situazioni di anomalia e dare all'utente il feedback più corretto. Nella sua realizzazione più semplice, questa funzione si limita a controllare che la richiesta sia terminata con successo. In tal caso accede ai dati restituiti dal server e

fa qualche tipo di aggiornamento dinamico della pagina per comunicare tali informazioni all'utente.

Nel seguente esempio la funzione di hook si limita a visualizzare il contenuto ricevuto dal server attraverso una messagebox (senza modificare il documento html). Questo esempio è fatto per evidenziare che questo sistema è indipendente dal fatto che si usi o meno la modifica dinamica della pagina.

```
http_request.onreadystatechange = function() { alertContents(http_request); };
...
function alertContents(http_request) {
    if (http_request.readyState == 4) {
        if (http_request.status == 200) {
            alert(http_request.responseText);
        } else {
            alert('There was a problem with the request.');
```

La richiesta invece viene fatta utilizzando la funzione **open()** e successivamente **send()** dopo aver correttamente impostato la funzione di hook. Il seguente esempio

mostra come poter effettuare una GET http ad un url di un testo. Il risultato verrà visualizzato in una finestra attraverso la funzione di hook precedentemente descritta.

```
http_request.open('GET', 'test.html', true);
http_request.send(null);
```

7.2.2 XML e altri standard di interscambio

L'esempio fatto in precedenza illustra molto bene l'opportunità di effettuare delle chiamate a un server per inviare e ricevere dei dati in modo sincrono. Tuttavia per rendere efficace questo meccanismo, occorre un sistema che permetta di rappresentare le informazioni da scambiare.

7.2.2.1 XML

Il linguaggio principe per inviare dati strutturati è naturalmente XML. Infatti XMLHttpRequest offre un oggetto DOM con cui il client è in grado di navigare gli elementi del file XML restituito dal server.

Con riferimento all'esempio descritto in precedenza, supponiamo di effettuare una GET sul file 'test.xml' e che questo contenga il seguente frammento di codice.

```
<?xml version="1.0" ?>
<root>
    I'm a test.
</root>
```

Nella funzione di hook invece di accedere direttamente al testo è possibile usare l'attributo responseXML che ci permette di usare le API DOM per ottenere le informazio-

ni di interesse. Il seguente esempio javascript permette di ricavare il contenuto del tag radice ('root') con una singola istruzione e di visualizzarlo con una messagebox.

```
var xmldoc = http_request.responseXML;
var root_node = xmldoc.getElementsByTagName('root').item(0);
alert(root_node.firstChild.data);
```



7.2.2.2 JSON

Quello di usare XML, non è l'unico metodo che permette di scambiare dati strutturati attraverso XMLHttpRequest. In linea di principio, si potrebbe fare uso di qualsiasi linguaggio utilizzato per costruire un parser. Con XML ci si risparmia la fatica di dover realizzare un parser, dal momento che l'oggetto ne mette a disposizione uno. Tuttavia Javascript dispone di una alternativa altrettanto comoda: **JSON**.

JSON che sta per **JavaScript Object Notation** è una notazione standard per lo scambio di dati basato su di un sottoinsieme delle regole sintattiche di Javascript. JSON è stato standardizzato da **ETSI (RFC 4627)** e il suo tipo **MIME** registrato è **application/json**.

JSON a differenza di XML ha una sintassi che serve a rappresentare tipi di base dei linguaggi di programmazione strutturati:

- Numeri (interi o reali)
- Valori booleani (true/false)
- Stringhe
- Array
- Oggetti (record)
- Valore nullo (null)

Il seguente frammento di codice rappresenta un array costituito da due elementi. Ciascun elemento è un oggetto che presenta diversi campi: i dati relativi ad una località.

```
[
  {
    "precision": "zip",
    "Latitude": 37.7668,
    "Longitude": -122.3959,
    "Address": "",
    "City": "SAN FRANCISCO",
    "State": "CA",
    "Zip": "94107",
    "Country": "US"
  },
  {
    "precision": "zip",
    "Latitude": 37.371991,
    "Longitude": -122.026020,
    "Address": "",
    "City": "SUNNYVALE",
    "State": "CA",
    "Zip": "94085",
    "Country": "US"
  }
]
```

Il vantaggio della sintassi JSON risiede nella possibilità di trasferire direttamente in variabili javascript il suo

contenuto tramite l'uso della funzione **eval()**, come illustrato nel seguente frammento di codice.

```
var the_object;
http_request.open("GET", 'test.php', true);
http_request.onreadystatechange = function () {
  if (http_request.readyState == 4) {
    if (http_request.status == 200) {
      the_object = eval("(" + http_request.responseText + ")");
      alert(the_object[0].zip);
    } else {
      alert("There was a problem with the URL.");
    }
  }
  http_request = null;
};
http_request.send(null);
```

Nell'esempio il codice dello script "test.php" quando invocato produce il file JSON con l'elenco delle località e le relative informazioni. La funzione di hook deve solo accedere al sorgente JSON utilizzando l'attributo `responseText` e fornirlo alla funzione **eval()**. Questa genera come risultato un oggetto che rappresenta il dato ricevuto, in questo caso un array di oggetti. Infatti nella successiva funzione

alert() si accede al codice "zip" della prima località utilizzando la consueta notazione javascript.

Il vantaggio di JSON rispetto a XML è che l'accesso alle informazioni è più rapido e intuitivo, dato che si usano i costrutti del linguaggio senza passare da API specifiche. Il grosso svantaggio è che risulta limitato nella tipologia di informazione che è in grado di rappresentare.



7.2.2.3 WDDX

WDDX (Web Distributed Data Exchange) è un linguaggio basato su XML che serve a rappresentare oggetti tipici di un linguaggio di programmazione. In particolare WDDX serve per rappresentare tipi primitivi (come interi o booleani),

strutture dati, vettori e record (normalmente il risultato di una query su DB). Questo formato, inizialmente ideato per ColdFusion, ora è divenuto una specifica aperta⁸⁶.

Il seguente frammento di codice mostra un esempio di un insieme di dati rappresentato nel formato WDDX.

```
<wddxPacket version='1.0'>
  <header comment='PHP' />
  <data>
    <struct>
      <var name='pi'>
        <number>3.1415926</number>
      </var>
      <var name='cities'>
        <array length='3'>
          <string>Austin</string>
          <string>Novato</string>
          <string>Seattle</string>
        </array>
      </var>
    </struct>
  </data>
</wddxPacket>
```

Di fatto WDDX e **XML-RPC** (nato nello stesso periodo) sono stati i precursori dello standard **SOAP (Simple Object Access Protocol)**, che rappresenta oggi il protocollo standard per l'accesso ai Web Services.

7.2.3 Presentazione e interazione

Nei precedenti paragrafi, è stato descritto come è possibile scambiare dati con un server e quali sono i formati di scambio utilizzati. Il passo finale consiste nella presentazione di tali informazioni all'utente.

Questo viene effettuato attraverso la possibilità di leggere e/o modificare gli elementi che compongono la pagina web. La maggior parte delle operazioni sono possibili sfruttando le funzionalità innovative introdotte nei fogli di stile CSS, come ad esempio il posizionamento assoluto degli elementi.

In pratica si tratta di utilizzare tecniche di html dinamico,

con i problemi di compatibilità cross-browser che questo ha sempre comportato.

Lo sviluppo di AJAX si deve in gran parte al lavoro del W3C che nel 1998 ha emanato le specifiche del DOM (Document Object Model) di livello 1. Questo descrive non solo il modello gerarchico dei dati che formano una pagina web, ma definisce anche le API standard che devono essere messe a disposizione per navigare e modificare questa struttura.

Allo stato attuale il W3C ha emesso le specifiche per il livello 2 del DOM. La componente più innovativa, e particolarmente importante per lo sviluppo di interfacce fortemente interattive, è il **modello degli eventi**.

Riprendendo l'esempio fatto nei precedenti paragrafi, è possibile utilizzare il DOM per andare e alterare il contenuto di una pagina come mostrato nel seguente esempio di codice.

```
...
<script>
...
http_request.onreadystatechange = function () {
  if (http_request.readyState == 4) {
    if (http_request.status == 200) {
      document.getElementById('message').innerHTML =
        http_request.responseText;
    } else {
      alert("There was a problem with the URL.");
    }
  }
  http_request = null;
};
...
</script>
...
<div id="message"></div>
```

⁸⁶ <http://www.openwddx.org/>



In questo caso nella pagina html viene inserito un elemento “div” di nome “message” il cui contenuto è vuoto. La funzione di hook utilizza la primitiva DOM **getElementById()** per ottenere un riferimento all’elemento div attraverso il suo identificativo. A questo punto viene usata la proprietà **innerHTML** dell’elemento (anche questa parte delle specifiche DOM) per poter scrivere all’interno del div in modo dinamico. L’effetto finale è quello di vedere comparire il messaggio ricevuto dal server all’interno della pagina, senza che sia stato effettuato un reload della stessa.

Modificando lo stile, oltre che il contenuto di uno o più elementi di una pagina, è possibile aggiungere effetti visivi molto accattivanti. Ad esempio è possibile far apparire una finestra in dissolvenza che notifica l’utente su un evento inviato dal server (ad esempio una news) mentre si sta navigando il sito. In generale la vera novità sta nella maggiore standardizzazione degli elementi di dinamicità con cui è possibile realizzare la presentazione dei dati e interagire con l’utente. Nonostante questo, restano ancora dei problemi di compatibilità cross-browser. Inoltre lo sviluppo diretto con queste tecnologie richiede competenze molto specifiche su DOM e CSS.

7.3 FRAMEWORK, TOOLKIT A AMBIENTI DI SVILUPPO

Quando ci si avvicina la prima volta ad AJAX si commette l’errore comune di pensare che sia la tecnologia che permette di realizzare applicazioni web spinte a livelli paragonabili alle *rich-client platform native*. Ad esempio è comune sui forum trovare utenti che chiedono come mai konqueror (il browser ufficiale di KDE) non sia compatibile con AJAX. In effetti konqueror, così come Opera e altri browser, nonostante supporti XMLHttpRequest non gestisce la maggior parte delle interfacce AJAX. Questo è in effetti un problema di compatibilità del DOM e del supporto CSS. Ad oggi Mozilla e Internet Explorer sono gli unici motori compatibili, benché in maniera differente, per la maggior parte dei siti AJAX. Naturalmente la speranza è che in prospettiva l’uscita e il consolidamento delle nuove specifiche DOM e CSS, unita alla maggiore maturità dei browser, permetta di eliminare la maggior parte dei problemi di compatibilità.

Nella scelta di un framework o toolkit per AJAX occorre sempre tenere presente che si tratta dell’unione di tre componenti:

- Motore di comunicazione asincrona con il server.
- Linguaggio di scambio di dati con il server.

- API di accesso agli elementi del documento ed ai relativi eventi.

Molti strumenti sono semplicemente delle librerie di oggetti grafici o effetti riusabili che permettono a chi lavora in Javascript di non avere una conoscenza specialistica del DOM o del CSS. Allo stesso tempo permettono di slegare l’applicazione rispetto ai cambiamenti che i browser fanno nel passaggio da una versione alla successiva.

Altri strumenti d’altro canto si concentrano sull’uniformare la comunicazione con il server e di conseguenza nascondono allo sviluppatore i dettagli dell’interfaccia XMLHttpRequest o del formato con cui i dati sono scambiati.

Strumenti più evoluti, cercano di supportare lo sviluppatore su tutti gli aspetti inerenti AJAX, nascondendone la maggior parte di dettagli. Questi solitamente sono strumenti di sviluppo che implementano il pattern **MVC (Model View Controller)** e quindi hanno l’ulteriore vantaggio di fornire un design ben organizzato che separa la logica applicativa (solitamente lato server) dalla logica di presentazione e interazione con l’utente.

Nei successivi paragrafi sono analizzati i principali strumenti AJAX disponibili. L’elenco di tali strumenti non è esaustivo, è tuttavia rappresentativo delle diverse tipologie di approcci allo sviluppo con AJAX.

7.3.1 Framework Javascript

La prima tipologia di framework che vengono analizzati sono quelli basati esclusivamente su Javascript. Questi operano esclusivamente lato client, mettendo a disposizione dello sviluppatore una libreria di oggetti.

Il principale vantaggio di questa tipologia di framework è che si possono integrare facilmente in qualsiasi strumento di sviluppo lato server.

Lo svantaggio è che richiedono un massiccio sviluppo in Javascript sul client, anche se molto semplificato grazie al livello di astrazione offerto dagli oggetti messi a disposizione.

7.3.1.1 Prototype

Prototype⁸⁷ è una libreria Javascript (contenuta in un singolo file) che fornisce un insieme di funzioni e oggetti attraverso cui è possibile leggere e manipolare gli elementi del DOM in un modo rapido e portabile rispetto a Internet Explorer, Mozilla e Safari. Fornisce, inoltre, delle funzioni comode che permettono di effettuare l’invio e la ricezione dei dati verso un server.

Ecco ad esempio il codice che permette di effettuare una richiesta asincrona verso un server:

⁸⁷<http://prototype.conio.net/>



```

var myAJAX = new AJAX.Request(
    url,
    {method: 'post', parameters: data, onComplete: AJAX_response}
);
function AJAX_response(originalRequest) {
    if(!bHasRedirect) {
        //process originalRequest.responseText;
    }
    else {
        bHasRedirect = false;
        AJAX_request(originalRequest.responseText, "");
    }
}
}

```

Come si vede dal frammento di codice, lo sviluppatore non deve scrivere codice specifico per creare un oggetto XMLHttpRequest, ma usa **AJAX.Request()**. Inoltre, nella funzione di hook (**AJAX_response**) non si deve preoccupare dello stato dato che il framework la chiama solo se la richiesta è stata completata (**onComplete**). Un'altra importante funzione che mette a disposizione questo framework è la variabile booleana **bHasRedirect**. Questa è impostata a true se il server ha risposto con un redirect. Usando XMLHttpRequest, deve essere lo sviluppatore a leggere la risposta del server e capire se contiene un redirect a una url differente.

Prototype è distribuito con licenza open-source MIT.

7.3.1.2 Script.aculo.us

Script.aculo.us⁸⁸ è una libreria Javascript costruita a partire da prototype che aggiunge funzioni per la realizzazione di molti effetti dinamici, tra cui:

- Effetti grafici (dissolvenze, animazioni, ecc...).
- Controlli dinamici (barre di scorrimento, campi con auto-completamento, ecc,...).
- Trascinamento.

Il seguente frammento di codice html visualizza un contenuto che scompare eseguendo uno zoom e contemporaneamente un effetto di dissolvenza.

```

<div class="example" id="demo-effect-puff" onclick="new Effect.Puff(this)">
    
    <span>Click for Effect.Puff demo</span>
</div>

```

Questa libreria, oltre a presentare un'architettura abbastanza unica e interessante, permette appunto la combinazione di diversi effetti per crearne dei nuovi. È molto utilizzata e di fatto resta integrata in diversi linguaggi e framework per lo sviluppo di applicazioni Web.

Anche questo software è open-source con licenza **MIT**.

7.3.1.3 DOJO Toolkit

DOJO⁸⁹ è un toolkit basato su Javascript che semplifica lo sviluppo di applicazioni DHTML. Caratteristica unica di questo toolkit, è quella di aver raggruppato le librerie in diversi pacchetti, in modo da consentire all'applicazione di caricare esclusivamente le componenti necessarie.

Questo toolkit offre non solo funzionalità legate alla realizzazione di interfacce grafiche, ma contiene anche tutto il supporto per la gestione delle comunicazioni asincrone con il server. Supporta sia JSON sia XML come sistemi di rappresentazione delle informazioni.

Un altro aspetto innovativo di questo sistema, è legato al modo con cui è possibile utilizzare i controlli raffinati della libreria all'interno della pagina html.

Lo sviluppatore può in prima istanza utilizzare attributi nuovi per convertire i controlli predefiniti dell'html nei controlli dinamici offerti dal toolkit. Il seguente esempio trasforma una classica casella di testo nella versione più ricca e interattiva, fornita da DOJO.

```

<!-- Versione classica -->
<input value="default">

<!-- Versione DOJO -->
<input dojoType="ComboBox" value="default" dataUrl="comboBoxData.js">

```

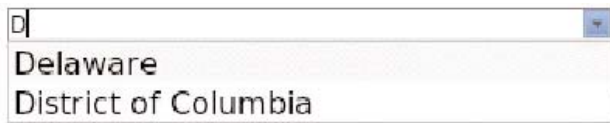
Se non si dispone di Javascript il controllo appare come una normale casella di testo. Con DOJO e Javascript attivo, viene visualizzata una finestra di selezione. Particolarmente interessante è l'aspetto AJAX fornito dal

parametro **dataUrl**. Questo fa riferimento ad un URL che fornisce i dati per la selezione. Questo tipo di controllo offre anche funzioni di auto completamento, come mostra la seguente figura.

⁸⁸ <http://script.aculo.us/>

⁸⁹ <http://dojotoolkit.org/>

L'utilizzo di attributi nuovi per creare l'interfaccia all'in-



```
<dojo:ComboBox value="default" dataUrl="comboBoxData.js">
```

DOJO offre le seguenti tipologie di funzionalità:

- controlli di base (combobox, menu, ecc., ...);
- controlli avanzati (SVG, calendario, task bar, ecc., ...);
- gestione del layout;
- gestione degli eventi;
- comunicazione asincrona XML/JSON;
- effetti speciali (animazioni, drag&drop, ecc., ...).

DOJO è distribuito sotto due licenze opensource: **AFL (Academic Free License)** e **BSD**.

Nonostante la comunità sia molto attiva, mancano molti contributi relativi alla documentazione dei controlli e in generale delle funzionalità offerte dal toolkit. D'altra parte è importante sottolineare come questo toolkit offra supporto dichiarato per un gran numero di browser:

- Safari 2.0.x+;
- Opera 8.5+;
- Internet Explorer (Windows) 5.5+;
- Firefox 1.0+/Mozilla;
- Konqueror 3.5+.

7.3.1.4 Qooxdoo

Qooxdoo⁹⁰ è uno dei framework opensource per AJAX più completi. Da un punto di vista dello sviluppo, questo toolkit è molto simile a quelli nativi come Swing, QT e GTK. Lo sviluppatore lavora sostanzialmente in Javascript, attingendo ad una serie di classi con cui è in grado di creare tool bar, menu, controlli, finestre modali, tabelle ecc. L'interazione viene gestita attraverso un modello ad eventi, che ricorda molto i toolkit nativi già precedentemente citati. Inoltre vengono fornite dal toolkit le classi per implementare il "remoteing", ovvero la comunicazione asincrona con il server. Nonostante ci sia una documentazione sempre aggiornata, attraverso un sistema di generazione simile a JavaDoc, resta minima la disponibilità di una documentazione più organica che permetta di capire come costruire le applicazioni.

Questo strumento è rilasciato sotto licenza LGPL.

7.3.1.5 Google Web Toolkit (GWT)

Google Web Toolkit⁹¹ è uno strumento sviluppato da Google per la realizzazione di applicazioni interattive attraverso AJAX.

Questo toolkit si differenzia dagli altri in quanto utilizza

terno delle pagine html porta ad avere una pagina non conforme allo standard W3C. Per evitare questo problema, il toolkit offre la possibilità (come mostrato nel seguente frammento di codice) di usare tag *custom* all'interno di uno spazio di nomi appositamente definito.

un approccio che serve a nascondere allo sviluppatore non solo html e CSS, ma anche Javascript. Questo obiettivo (come verrà spiegato in seguito) tendenzialmente viene ad oggi raggiunto utilizzando un approccio ibrido che prevede una parte sia client sia server. Google invece ha risolto il problema attraverso la realizzazione di un compilatore.

Il compilatore è stato introdotto per permettere all'utente di scrivere le proprie applicazioni utilizzando il linguaggio Java. Questo permette allo sviluppatore di utilizzare il suo ambiente di sviluppo preferito (IDE) e avvantaggiarsi di tutte le funzionalità che tali ambienti offrono (controllo dei tipi, debug passo passo, ecc...).

Il toolkit di Google offre sia un compilatore sia un browser integrato, utilizzato durante la fase debug. Questo browser è infatti in grado di integrarsi con la macchina virtuale. Per garantire il risultato migliore questo browser utilizza il motore di MSIE su piattaforme Windows e Geko/Mozilla su piattaforme Linux.

Per Eclipse è ad esempio disponibile un plugin che permette di sviluppare applicazioni dinamiche con il GWT. Tale plugin, denominato Googlipse⁹², è disponibile sotto licenza APL (**Apache Public License**).

L'API è fornita da Google con licenza APL, tuttavia il compilatore e il browser necessari per lo sviluppo sono rilasciati in forma binaria e sono soggetti a doppia licenza: commerciale e non commerciale.

7.3.2 Framework ibridi

Sotto la categoria dei framework ibridi rientrano tutti quelli basati sull'utilizzo non solo di una libreria Javascript ma anche di una libreria lato server, coadiuvata molto spesso da un run-time. L'obiettivo principale di questi framework è quello di migliorare l'architettura delle applicazioni AJAX, usando una serie di design-pattern consolidati o specificatamente studiati per questa metodologia di sviluppo.

In questi casi la valutazione è sempre un trade-off tra semplicità e flessibilità. Indubbiamente l'utilizzo di determinati design-pattern permette di rendere più robusta la struttura dell'applicazione, ma d'altro canto introduce delle rigidità che spesso è difficile aggirare. Inoltre la complessità dei design-pattern ne limita l'utilizzo a situazioni davvero complesse.

⁹⁰ <http://qooxdoo.org/>

⁹¹ <http://code.google.com/webtoolkit/>

⁹² <http://www.googlipse.com/>



In casi semplici potrebbe essere vincente la scelta di usare un toolkit Javascript da inserire nelle pagine generate server-side. In situazioni più complesse è invece opportuno usare un framework che integra le due componenti in modo sinergico.

7.3.2.1 Echo2

Echo2⁹³ è un framework AJAX basato su Java che permette allo sviluppatore di creare le applicazioni interattive utilizzando il modello ad eventi noto per le applicazioni native basate su Swing e AWT.

Il programmatore di fatto dispone di una libreria di classi che gli permette di creare finestre, di aggiungere controlli (pulsanti, liste, immagini, ecc.) di controllare il layout e ovviamente di associare metodi a determinati eventi.

Le applicazioni scritte con questo framework devono poi essere installate all'interno di un web container (ad esempio Tomcat).

Quando il client (browser) si connette all'URL su cui l'applicazione è stata installata, il framework genera (operazione di rendering) l'interfaccia AJAX. Quindi è il framework che si preoccupa di creare la pagina html e il codice Javascript in base al tipo di browser che si collega.

Quando l'utente effettua un'operazione per cui è richiesto l'aggiornamento del modello dei dati, il framework invia un evento al server utilizzando il sistema di comunicazione asincrona e XML. Quando il server riceve l'evento, notifica gli oggetti interessati a tale evento, richiamando gli opportuni metodi, cui passa un oggetto che descrive sia l'evento sia l'oggetto che lo ha generato.

Echo2 supporta anche una comunicazione di tipo push. In tal caso è il server che invia un evento al client.

Unico svantaggio di tale prodotto è il fatto che occorre apprendere la libreria per la realizzazione dell'interfaccia. Questa non contiene lo stesso numero di classi di Swing (manca ad esempio la parte per gestire la grafica 2D) ma è tuttavia molto complessa.

In aiuto per lo sviluppatore, esiste un plug-in per Eclipse che permette di disegnare l'interfaccia grafica e quindi di programmare la logica di business associata a ciascun evento. Ad oggi questo strumento è in versione beta.

Il framework Echo2 è completo ed è rilasciato sotto licenza MPL (Mozilla Public License) mentre il plug-in per Eclipse è distribuito con una licenza commerciale.

Echo2 (così come altri framework analizzati) cerca di nascondere i dettagli della programmazione Javascript e HTML al programmatore.

Questo approccio è sicuramente molto pulito, ma richiede che lo sviluppatore debba apprendere una nuova libreria GUI che spesso è di difficile utilizzo. In effetti questi framework si appoggiano quasi sempre sul supporto fornito da un IDE che semplifica notevolmente la fase di design dell'interfaccia.

7.3.2.2 ZK

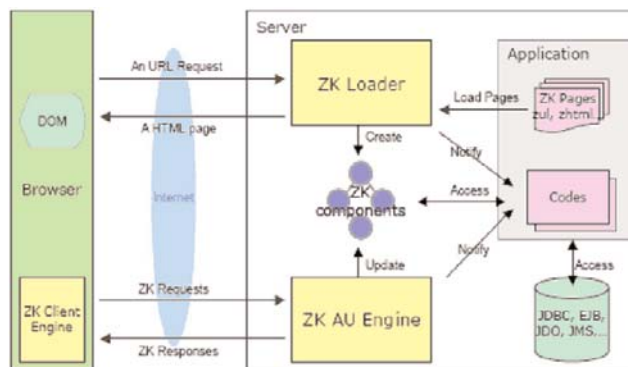
ZK⁹⁴ è un framework open source che cerca di compiere un passo evolutivo rispetto ai modelli analizzati in precedenza e che, per altro, è già ampiamente utilizzato da alcuni strumenti commerciali e open source. Questo passo, consiste nel fornire allo sviluppatore un linguaggio dichiarativo (basato su XML) per descrivere gli elementi delle GUI.

Alcuni esempi di soluzioni simili sono:

- libGlade, una libreria in grado di costruire un'interfaccia grafica GTK, a partire da una descrizione XML generata dal toolkit Glade;
- UIC (User Interface Compiler), un compilatore fornito nel toolkit QT, che genera codice C++ per la realizzazione di una interfaccia grafica descritta da un file XML;
- XAML (eXtensible Application Markup Language), è il linguaggio definito all'interno del framework di Windows Vista WPF (Windows Presentation Framework), per realizzare interfacce grafiche in .NET;
- XUL (XML User-interface Language), è un linguaggio introdotto da Mozilla per la realizzazione di interfacce grafiche attraverso la sua piattaforma rich-client.

In particolare ZK usa una versione estesa di XUL come linguaggio di definizione dell'interfaccia.

L'architettura è descritta nella figura sotto riprodotta.



ZK è costituita da una servlet che viene eseguita all'interno di un web container Java (ad esempio Tomcat).

L'utente (attraverso il browser) accede ad un file con estensione zul che contiene l'interfaccia grafica descritta in XML.

La servlet (ZK Loader) è configurata per essere attivata su questo tipo di URL ed esegue un parsing del file XML per poter realizzare la pagina da inviare al browser. Questa pagina di fatto contiene sia componenti HTML sia il motore Javascript (ZK Client Engine).

Quando un evento (dovuto ad esempio all'interazione dell'utente) viene generato sul client, lo ZK Client Engine invia questo evento allo ZK AU Engine. Questo aggiorna

93_ <http://www.nextapp.com/platform/echo2/echo/>

94_ <http://www.zkoss.org/>

gli oggetti che rappresentano l'interfaccia grafica (ZK Components) e che sono stati precedentemente creati da ZK Loader subito prima di restituire la pagina al browser. Questi oggetti possono poi scatenare eventi o chiamate verso altri oggetti che fanno parte dell'applicazione. La stessa applicazione può interagire con questi oggetti ad esempio per comunicare un cambiamento dell'informazione (il quale necessita una modifica dell'interfaccia grafica). Questa architettura non è molto diversa da quella di Echo2 e in genere, anche se con minori varianti, è l'architettura usata dai framework ibridi. Demandando allo ZK Loader i dettagli su come devono essere istanziati gli elementi dell'interfaccia utente, questo framework permette di ridurre notevolmente la parte

di codice che realizza la GUI, aumentando i tempi di sviluppo e diminuendo allo stesso tempo la quantità di codice non riutilizzabile (quella legata all'integrazione con il framework).

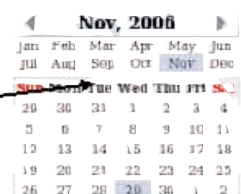
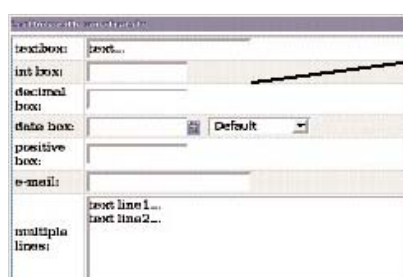
ZK permette anche di usare un linguaggio di scripting ZS (ZK Script) attraverso cui è possibile realizzare la logica applicativa in Java puro. Questo metodo è molto comodo dato che non richiede la compilazione dell'applicazione a ogni modifica, tuttavia è sconsigliato usarlo in produzione. L'ideale è usare ZK Script per semplici operazioni o per le fasi di prototipazione rapida.

Il seguente frammento di codice, illustra un esempio che mette in evidenza la semplicità con cui si può scrivere una applicazione con ZK:

```
<window title="textbox with constraints" border="normal">
  <grid>
    <rows>
      <row>
        textbox: <textbox value="text..." />
      </row>
      <row>int box:<intbox /></row>
      <row>decimal box:<decimalbox format="#.##0.##" /></row>
      <row>date box:
        <hbox>
          <datebox id="db" />
          <listbox onSelect="db.setFormat(self.selectedItem.value)"
mold="select" rows="1">
            <listitem label="Default" value="" />
            <listitem label="yyyy/MM/dd" value="yyyy/MM/dd" />
            <listitem label="MM-dd-yy" value="MM-dd-yy" />
          </listbox>
        </hbox>
      </row>
      <row>positive box:<intbox constraint="no negative,no zero" /></row>
      <row>e-mail:<textbox constraint="/.+@.+.[a-z]+/: Please enter an e-mail
address" /></row>
      <row>
        multiple lines:
          <textbox rows="5" cols="40">
            <attribute name="value">
text line1...
text line2...
            </attribute>
          </textbox>
        </row>
      </rows>
    </grid>
  </window>
```

Il risultato del codice precedente è rappresentato nella figura che segue. Occorre notare come sia semplice definire delle caselle caratterizzate da particolari vincoli con

l'attributo *constraint* o addirittura richiedere l'inserimento di una data attraverso l'utilizzo di un calendario con il tag *datebox*.



ALLEGATO 3

DESCRIZIONE DEL PROGETTO WAI-ARIA (W3C)

a3

DESCRIZIONE DEL PROGETTO WAI-ARIA (W3C)



Il 26 settembre 2006 il Web Accessibility Initiative (WAI) del W3C ha presentato i documenti di supporto allo sviluppo di applicazioni internet complesse ed accessibili⁹⁵. Il First Public Working Draft di WAI-ARIA (Accessible Rich Internet Application) si basa su tre documenti correlati:

- WAI-ARIA Roadmap;
- WAI-ARIA Roles;
- WAI-ARIA States and Properties.

Il primo documento intende approfondire l'approccio di sviluppo da adottare per assicurare l'interoperabilità tra le applicazioni Internet "evolute" e le tecnologie assistive utilizzate dagli utenti disabili. Gli altri due documenti sono invece finalizzati all'identificazione delle tecnologie necessarie per rendere accessibili tali applicazioni, nell'ottica delle criticità ad oggi delineate.

Come dichiarato dal WAI⁹⁶, "mentre gli utenti richiedono sempre più al Web – più informazioni, più applicazioni interattive ed esperienze coinvolgenti – si assiste ad una esplosione nello sviluppo di tecnologie, che esclude però dall'accesso al Web un numero troppo elevato di persone – sostiene Rich Schwerdtfeger, Distinguished Engineer dell'IBM e autore della WAI-ARIA Roadmap –. La pubblicazione di questa nuova suite di documenti è importante perché aiuterà gli sviluppatori ad avere accesso ai tool necessari al supporto di utenti disabili sul Web. ARIA è il nostro primo passo per fornire l'accesso ad una più ricca esperienza del Web dinamico a tutti gli utenti del Web, attraverso miglioramenti nelle tecnologie, ad esempio implementazioni migliori e più accessibili." Qui di seguito verranno descritti alcuni dei temi analizzati nell'ambito del progetto WAI-ARIA.

8.1 RUOLO (ROLE)

Poiché tutte le API per l'accessibilità attualmente disponibili si aspettano di ricevere informazioni circa il ruolo svolto da ciascun elemento di una GUI, uno dei passi necessari per rendere le RIA accessibili consiste nell'aggiungere un'indicazione di ruolo.

In particolare, si propone di rendere il ruolo un **qname** (*qualified namespace*; nome la cui interpretazione dipende dall'abbinamento ad un namespace) e di per-

mettere la definizione dei valori possibili in RDF. L'obiettivo, infatti, è lasciare che il *range* di ruoli possibili sia liberamente estensibile e che le specifiche che definiscono ciascun ruolo siano pubbliche. L'abbinamento tra il nome di un ruolo e la sua definizione è reso possibile dalla corretta dichiarazione del *namespace* di riferimento, in cui compare l'URL del documento RDF contenente le specifiche.

La forza di tale modello è che uno *user agent* può sempre recuperare il documento RDF che contiene la descrizione di un certo elemento e determinare quali proprietà esso supporta, permettendo così un corretto abbinamento con le API per l'accessibilità. Inoltre, uno *user agent* sarebbe sempre in grado di stabilire come gestire l'interazione anche con gli elementi creati *ad hoc* da un certo progettista.

La disponibilità delle specifiche RDF potrebbe anche facilitare la creazione di strumenti di Web design capaci di supportare gli autori nelle fasi di creazione e di test di applicazioni accessibili.

Oltre ai sopraccitati benefici in termini di accessibilità, l'indicazione del ruolo di ciascun elemento permetterebbe anche di procedere nella definizione del Semantic Web, abilitando tutte le funzioni di ricerca personalizzata e i servizi intelligenti ad esso associati.

Esempio:

Se si vuole indicare che un certo elemento riveste il ruolo di menù, si specifica l'attributo "role". Il valore di "role" sarà "prefisso:menu", in cui il prefisso sarà stato dichiarato in precedenza tra i *namespace*, indicando l'URL del documento RDF in cui il ruolo di menù viene definito.

8.2 STATO (STATE)

Lo stato di un oggetto sintetizza le proprietà significative che questo possiede in un certo momento, definendo, tra l'altro, il tipo di azioni che l'utente può compiere. Nel caso di contenuti dinamici, come quelli che caratterizzano le RIA, lo stato di un oggetto cambia costantemente: tali variazioni devono essere comunicate al *device* con cui l'utente fruisce dei contenuti stessi perché questi

⁹⁵ Dal momento che il processo di standardizzazione non si è ancora concluso, si faccia riferimento a questo indirizzo per l'aggiornamento sullo stato dell'arte della documentazione: <http://www.w3.org/WAI/PF>

⁹⁶ <http://www.w3c.it/pr/2006/aria-pressrelease-it.html>



risultino effettivamente utilizzabili. Anche in questo caso, le informazioni relative allo stato vengono indicate tramite un attributo qualificato da un namespace.

Esempio:

Se si vuole indicare che ad un certo elemento, ad esempio un menù, può essere abbinato un popup, si specifica l'attributo "prefisso:haspopup" e gli si attribuisce valore "true". Il prefisso utilizzato sarà stato dichiarato in precedenza tra i *namespace*, indicando l'URL del documento RDF in cui viene definito lo stato a cui si fa riferimento.

8.3 FOCUS

Riconoscere l'elemento su cui l'utente si focalizza in un certo momento permette di generare degli eventi in armonia con lo spostarsi dell'attenzione di quest'ultimo. Attualmente, i linguaggi HTML e XHTML permettono di gestire il focus solo per i form e gli elementi ancora. Tuttavia, la definizione del DOM richiede che qualunque elemento possa essere accessibile da tastiera: allo stato attuale, le limitazioni sulla gestione del focus rendono di fatto inaccessibile da tastiera la maggior parte degli elementi. In particolare, l'utente che utilizza la tastiera si vede costretto a scorrere tutti gli elementi della pagina mediante il comando "tab" per raggiungere quello desiderato. Normalmente, la navigazione degli elementi di un documento tramite tab prevede che vengano proposti per primi quelli per cui l'attributo "tabindex" ha un valore positivo, ordinati a seconda del valore stesso. Subito dopo, compaiono gli elementi che supportano il focus (form e link) e quelli per cui il valore di "tabindex" è zero, nell'ordine in cui appaiono nel documento.

La soluzione proposta consiste nel definire anche valori negativi per l'attributo "tabindex": in tal modo, sarebbe possibile focalizzare un certo elemento senza bisogno di scorrere tutto il documento, ma semplicemente attivando la funzione `element.focus()` tramite pressione di un certo tasto.

8.4 ACCESSO ALLE SEZIONI STRUTTURALI DI UN DOCUMENTO

I documenti ipertestuali si caratterizzano per la presenza di una serie di parti o strutture comuni, il cui ruolo viene definito contestualmente alle specifiche di XHTML (ad esempio, si hanno "navigation", "banner", ...). L'integrazione di tali ruoli nella tassonomia degli *user-agents* può permettere di fornire una modalità di accesso "prioritario" alle sezioni normalmente presenti in un documento, sfruttando le caratteristiche del *device*. Ad esempio, si potrebbero rendere accessibili le diverse sezioni sfruttando una combinazione di tasti per chi opera con uno strumento di tipo *desktop* (e, quindi, dotato di una comoda tastiera). In particolare, le sezioni potrebbero essere definite nell'header del documento, in un elemento "link" in cui l'attributo "rel" indichi il ruolo della sezione, l'elemento "href" la sua posizione all'interno del documento stesso e l'attributo "title", opzionale, ne dia una descrizione.

8.5 GESTIONE EVENTI

Perché un utente possa interagire efficacemente con un'applicazione, è necessario che questi sia consapevole delle azioni che può compiere su un certo elemento. Al momento attuale, tuttavia, le informazioni disponibili risultano, da una parte, non informative e, dall'altra, eccessivamente dipendenti dallo specifico *device*. La soluzione proposta dal W3C consiste nell'utilizzare XML per integrare la descrizione del comportamento (*listeners* e *handlers* degli eventi) con le specifiche di *markup* di un certo documento, sfruttando l'approccio dichiarativo offerto da tale linguaggio. In tal modo, le future versioni di XML permetterebbero di abbinare ai gestori degli eventi una descrizione che ne specifichi la funzione. L'uso di descrizioni ad alto livello consentirebbe di associare a ogni gestore, al momento della fruizione, una soluzione specifica, che sfrutti appieno le potenzialità offerte dal particolare *device* utilizzato dall'utente.

8.6 XFORMS

Gli "XForms" rappresentano l'ultima generazione di Web Form, come descritta da un'apposita specifica del W3C. Si caratterizzano per la separazione tra le informazioni semantico-strutturali e di presentazione: in particolare, gli XForms delegano a sezioni diverse la descrizione del funzionamento del form e del suo aspetto, permettendo molta flessibilità nelle scelte di presentazione. Ad esempio, è possibile includere dei form tradizionali nella definizione di un XForm.

Dal punto di vista dell'accessibilità, gli XForms esplicitano una lista di azioni che possono essere compiute su un form e che, in tal modo, dovrebbero risultare accessibili indipendentemente dal browser o dalla piattaforma con cui l'utente fruisce del servizio. Esempi delle funzioni disponibili sono: "cambio del focus", "invio", "reset". Inoltre, gli XForms offrono una semantica adatta ad informare lo *user agent* e, di conseguenza, l'utente, degli eventi che hanno effettivamente luogo sul form: ad esempio, "cambio di valore" "selezionato/deselezionato", "form inviato".

In questo caso, il ricorso ad un modello già standardizzato riduce il lavoro che il progettista deve compiere per aggiungere agli elementi della propria applicazione una serie di proprietà che la rendano effettivamente accessibile.

Le caratteristiche descritte nelle sezioni precedenti saranno integrate nella definizione di XHTML 2; nel tentativo di rendere tale linguaggio accessibile fin dalla base. Al contrario, XHTML 1.1 viene esteso e reso modulare attraverso l'uso di *namespaces*.

WEBGRAFIA E BIBLIOGRAFIA

WEBGRAFIA



[43things] www.43things.com	[Odeo] www.odeo.com
[Adaptivepath] http://www.adaptivepath.com/publications/essays/archives/000385.php	[Openwddx] www.openwddx.org/
[Amazon] www.amazon.com	[Orkut] www.orkut.com
[Apple] www.apple.com	[Pandora] www.pandora.com
[AtomEnabled] www.atomenabled.org/developers/syndication/atom-format-spec.php	[Particletree] http://particletree.com/features/the-hows-and-whys-of-degradable-ajax/
[Audible] www.audible.com	[Prototype] http://prototype.conio.net/
[Basecamp] www.basecamp.com	[Qooxdoo] http://qooxdoo.org/
[Campfire] www.campfirenow.com	[SAP Insider] www.gne.net/
[Digg] www.digg.com	[Script Aculo] http://script.aculo.us/
[Digitalglobe] www.digitalglobe.com/	[Semacode] http://semacode.org
[Discover] www.discover.com/issues/oct-05/departments/emerging-technology	[Semapedia] www.semapedia.org
[Dojotoolkit] http://dojotoolkit.org/	[Shopify] http://shopify.com
[eBay] www.ebay.com	[Shotcode] www.shotcode.com
[Flickr Bits] http://www.flickrbits.com	[Sixdegrees] www.sixdegrees.com
[Flickr] www.flickr.com	[Splinder] www.splinder.com
[Fluxiom] www.fluxiom.com	[Strongspace] www.strongspace.com
[Friendster] www.friendster.com	[Supereva] http://it.supereva.com
[Google] www.google.com	[Tadalist] www.tadalist.com
[Google AJAXSLT] http://goog-ajaxslt.sourceforge.net/	[Tagcloud] www.tagcloud.com
[Google Earth] http://earth.google.it/	[Technorati] www.technorati.com
[Google Code] http://code.google.com/webtoolkit/	[Teleatlas] www.teleatlas.com/
[Googlipse] http://www.googlipse.com/	[Thelongtail] www.thelongtail.com/about.html
[Harvard Law] http://blogs.law.harvard.edu/tech/rss	[Tribe] www.tribe.net
[Kibop] www.kibop.com	[Typo3] www.typo3.com
[Kosmar] http://kosmar.de/wp-content/webzomap.png	[Usabilityviews] www.usabilityviews.com/ajaxsucks.html
[Likedin] www.likedin.com	[w3c.org] www.w3.org/WAI/PF www.w3c.it/pr/2006/aria-release-pressrelease-it.html
[Ludicorp] www.ludicorp.com/	[Webopedia] www.webopedia.com/TERM/G/Graphical_User_Interface_GUI.html
[Ludicorp] www.netflix.com	[Xtech Conference] http://xtecho6.usefulinc.com/schedule/paper/29
[Myspace] www.myspace.com	[Yahoo] www.yahoo.com
[Navteq] www.navteq.com/	[Youtube] www.youtube.com
[Netvibes] www.netvibes.com	[Zkoss] http://www.zkoss.org/
[Nextapp] http://www.nextapp.com/platform/echo2/echo/	

BIBLIOGRAFIA

- _Hovland C.I. - Janis, I.L. - Kelley, H.H., *Communication and Persuasion. Psychological studies on opinion change*, New Haven, CO, Yale University Press, 1953.
- _Lee, J.D. - See, K.A., *Trust in automation: designing for appropriate reliance*, in "Human Factors", vol. 46, no. 1, Iowa, University of Iowa, 2004, pp. 50-80.
- _Meyrowitz J., *Oltre il senso del luogo. L'impatto dei media elettronici sul comportamento sociale*, Bologna, Baskerville, 1995.
- _Nielsen, J., *Usability Engineering*, Boston, AP Professional, 2003
- _Nielsen, J., *Web Usability*, Milano, Apogeo, 2000
- _Norman D. A. *La caffettiera del masochista*, Firenze, Giunti, 2005
- _Reeves B. - Nass C, *The Media Equation*, Stanford (CA), CSLI, 1996
- _Rizzo, A. - Marti, P. - Bagnara, S., *Interazione Uomo-Macchina*, in Burattini, E. - Cordeschi, R., "Intelligenza Artificiale. Manuale per le Discipline della Comunicazione", Roma, Carocci, 2001



Sede Legale e Operativa:
via Livorno, 60 - 10144 TORINO

Seconda sede operativa:
Viale Settimio Severo, 63 - 10133 TORINO

Tel. +39 011 4815111 - Fax +39 011 4815001 - E-mail: innovazione@csp.it