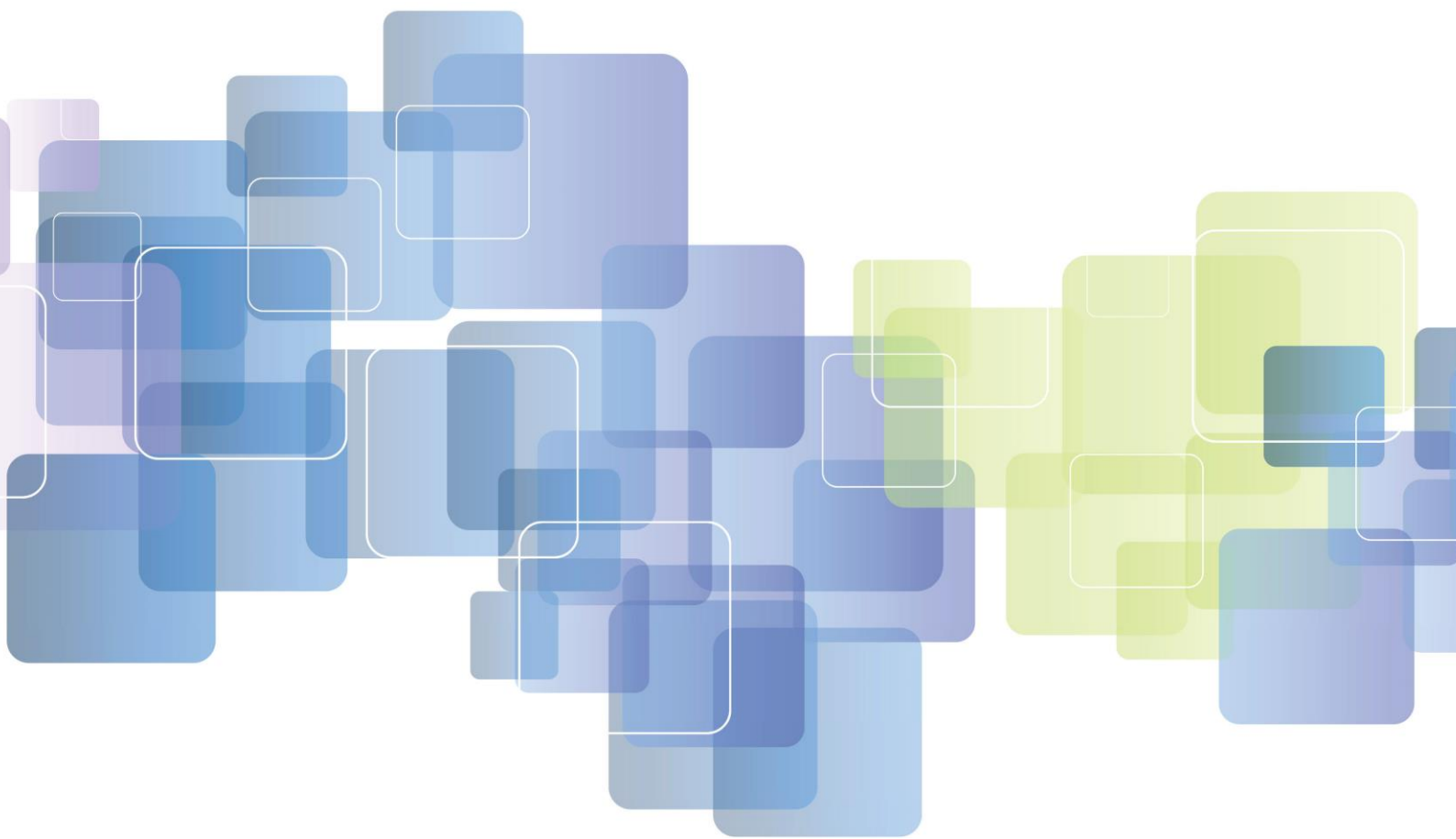


Sviluppo di applicazioni internet: Rad versus approccio formale

Mario Quirico



Come si fa a introdurre l'uso di tecniche formali nello sviluppo delle applicazioni internet senza incappare nelle spire di un ciclo di sviluppo convenzionale?

Sviluppo di applicazioni frenetico

Già la "rivoluzione" CLIENT / SERVER aveva fornito validi argomenti in favore di un ciclo di vita del software basato su uno sviluppo di applicazioni rapido (RAD - Rapid Application Development) in luogo del tradizionale ciclo a cascata sequenziale. Oggi, con l'avvento di applicazioni Internet/Intranet il concetto è stato portato alle estreme conseguenze al punto che, per dirla con Yourdon, si potrebbe parlare di FAD (Frantic Application Development) dove frantico dà l'idea di qualcosa di frenetico, convulso, quasi delirante. Ciò è dovuto anche al fatto che la tecnologia in questo campo ha fatto passi da gigante ed è in continua evoluzione; d'altro canto molti utenti hanno l'impressione che sofisticati siti WEB quali quelli visitati nelle loro "scorribande" su Internet possano essere creati istantaneamente.

In ogni caso, al di là degli estremismi, dato il livello di complessità raggiunto dai sistemi software attuali, diventa veramente improponibile oggi un processo di sviluppo sequenziale, che preveda di definire dapprima l'intero problema, disegnare una soluzione completa, costruire il software e quindi testare il prodotto. A questo proposito è da tenere in conto che, soprattutto nel caso di Internet:

- la maggior parte degli utenti non ha un'idea chiara di come potrebbe lavorare la nuova applicazione, di come sfruttare al meglio le nuove tecnologie per trarne il massimo profitto
- i tempi richiesti per il rilascio delle applicazioni diventano sempre più pressanti.

Necessità di un approccio formale all'analisi e al disegno

Oggi le parole d'ordine sono "RAD" e "prototipo", ma attenzione!, dopo un periodo che possiamo definire di "sperimentazione", nel momento in cui le applicazioni Internet/Intranet crescono in modo proporzionale con il tempo e non solo come numero, ma anche per ampiezza, volumi trattati, importanza strategica per le aziende e rischi associati, impatti sull'organizzazione del lavoro, appare sempre più evidente la necessità di un'analisi e di un disegno condotti in modo formale, per evitare di realizzare prodotti incontrollabili.

Ma allora come conciliare queste esigenze senza incappare nelle spire della "burocrazia" di un ciclo di sviluppo convenzionale?

Un approccio iterativo e incrementale al ciclo di vita del software

Proprio per superare questi problemi oggi si va sempre più affermando il concetto di un ciclo di vita iterativo e incrementale che consenta di:

- aumentare la comprensione del problema tramite successivi "raffinamenti"
- far crescere una soluzione efficace attraverso più iterazioni.

Un approccio di questo tipo permette inoltre una migliore flessibilità per accogliere nuovi requisiti, variazioni tattiche negli obiettivi del business. Fornisce la possibilità di frequenti feedback da parte di committente e utenza consentendo al progetto di identificare e risolvere eventuali rischi per tempo. Gli stessi autori di UML hanno in mente un processo di sviluppo del software di questo tipo, un processo:

- basato sui Casi d'Uso
- incentrato sull'architettura
- iterativo e incrementale.

Grady Booch nel suo libro ("Object Solutions - Managing the Object-Oriented Project", Addison-Wesley, 1996) chiarifica ulteriormente il concetto: "Iterativo è il processo che vede il progressivo affinamento dell'architettura del sistema e attraverso il quale applichiamo l'esperienza e i risultati di ciascun suo importante rilascio all'iterazione dell'analisi e del disegno che seguirà.

Ogni iterazione dell'analisi/disegno/implementazione viene ripetuta più volte sull'architettura OO. Il processo è incrementale nel senso che ogni passo attraverso il ciclo di analisi / disegno / implementazione ci porta ad affinare per gradi le nostre decisioni strategiche e tattiche, estende i nostri ambiti a partire da uno schema architeturale iniziale e sfocia nel prodotto software finale consegnato". Il lavoro delle iterazioni precedenti non viene scartato o rifatto, ma piuttosto rivisto e aumentato. Il processo è incrementale anche a un livello più fine di granularità, e cioè nel modo in cui ogni iterazione è organizzata internamente in una successione di passi realizzativi". Un processo di questo tipo presuppone comunque l'utilizzo di tecniche di analisi e disegno formali.

Formazione sui metodi formali di analisi e disegno

Le nuove generazioni di sviluppatori, che oggi spesso producono ad hoc su richiesta del committente o dell'utente finale, dovranno apprendere al più presto le basi teoriche e i metodi necessari per un'analisi e un disegno "rigorosi" delle applicazioni. Il compito della formazione sulle tecniche formali di analisi e disegno spetterà probabilmente ai progettisti "senior", quelli che hanno fatto esperienza di due o tre generazioni di variazioni tecnologiche.

Modellare i requisiti dell'utente

In conclusione, possiamo affermare che, sebbene in molti casi possa essere appropriato applicare qualche forma di RAD / prototyping alle applicazioni Internet/Intranet, ciò non

deve precludere la possibilità di impegnare all'inizio del progetto il tempo dovuto per modellare i requisiti dell'utente.

Internet/Intranet: un'occasione per il BPR

Non bisogna poi dimenticare che un'applicazione Internet/Intranet può essere una buona occasione per cambiare in meglio un processo aziendale: spesso le applicazioni Intranet hanno proprio l'obiettivo di migliorare processi aziendali "interni". D'altra parte, se si inizia a sviluppare un'applicazione Intranet solo perché "è qualcosa di nuovo da fare", senza valutare attentamente le opportunità che essa può offrire, il rischio di fallimento è molto alto: l'applicazione non sarà utilizzata perché probabilmente non apporterà nessun valore aggiunto per il business. Il buon risultato di un progetto Intranet dipende spesso da un'attenta analisi (BPR) dei processi esistenti che permette di comprendere dove possano essere raggiunti i migliori benefici dalla nuova applicazione.

Anche se non tutte le applicazioni Intranet sono buone candidate per il BPR, il successo di un'applicazione Internet/Intranet sarà tanto maggiore quanto più efficacemente modificherà il modo di operare degli utilizzatori, le modalità di esecuzione dei processi considerati importanti per il business.

Processi formali anche per il test, il configuration management ...

Insieme con i metodi e le tecniche appropriati per lo sviluppo di applicazioni, diventa cruciale l'adozione di processi formali per il test, la manutenzione e il configuration management e il versioning.

Queste ultime attività potevano essere ignorate o comunque eseguite in modo poco ortodosso nelle prime piccole applicazioni, ma nel momento in cui le applicazioni crescono esponenzialmente, queste attività diventano importanti in modo critico. Anzi sono probabilmente più importanti per progetti Internet / Intranet che per i progetti tradizionali in quanto nel primo caso la frequenza delle variazioni richieste all'applicazione aumenta in modo considerevole: le applicazioni Internet / Intranet possono richiedere aggiornamenti anche giornalieri.



Tecnet Dati s.r.l.
C.so Svizzera 185 -
10149 - Torino (TO), Italia
Tel.: +39 011 7718090 Fax.: +39 011 7718092
P.I. 05793500017 C.F. 09205650154
www.tecnetdati.com

