

È necessario creare architetture applicative basate su un ambiente all'interno del quale i programmi possano interagire utilizzando protocolli affidabili e standardizzati. Il fine è un sistema informatico, composto da moduli intercambiabili, in grado di interagire in modo dinamico e automatico per adeguarsi al meglio alle richieste.

# Più dinamismo nell'enterprise

di Giancarlo Magnaghi

**A**lcuni decenni di evoluzione dell'informatica e il susseguirsi delle successive ondate tecnologiche hanno aumentato la complessità dei sistemi informatici presenti nelle aziende in modo incrementale.

Forzate ad aggiornare i sistemi IT da tutta una serie di fattori, che si sono trovati a coincidere in un trigono astrale di fine **millennio** (adeguamenti all'anno 2000, introduzione dell'euro e diffusione delle tecnologie Internet), molte aziende non sono state in grado di combattere su tre fronti con le sole forze interne e hanno adottato sistemi ERP **omnicomprensivi** o nuove soluzioni **best-of-breed** nella speranza di risolvere facilmente tutti i problemi relativi all'IT. Una volta passata l'euforia della **new economy**, e tramontata l'illusione che fosse sufficiente adottare un sistema ERP o qualche pacchetto di e-business per risolvere in modo automatico e indolore tutti i problemi IT, le aziende si sono svegiate **dalla** colossale sbronza di **fine** secolo con un grosso mal di testa e con la prova che anche nel mondo dell'IT non si possono fare miracoli, e che



per ottenere i risultati non bastano gli investimenti nelle tecnologie, ma ci vuole anche tanto duro lavoro.

Contemporaneamente, la recessione economica ha provocato il taglio dei budget IT e un feroce **controllo** dei costi, che però non è sufficiente per rendere competitive le aziende, ma deve essere anche accompagnato dall'introduzione di modelli di **business** flessibili, che richiedono architetture IT altrettanto flessibili per lo sviluppo software, l'integrazione e la gestione delle applicazioni. Quindi, è necessario introdurre pesanti innovazioni con risorse ridotte d'osso: "fare di più con meno". Per riuscire a risolvere questa difficile equazione, è chiaro che bisogna abbandonare i progetti faraonici di rinnovamento totale e bisogna fare tesoro delle vecchie

applicazioni legacy che, anche se hanno **mille** difetti, fanno funzionare i processi di base che mantengono in vita l'azienda. Spesso però i sistemi esistenti rappresentano una **pesantissima palla** al piede, poiché è difficilissimo sostituirli e costosissimo mantenerli, tanto che l'80% dei budget viene speso per mantenere i sistemi esistenti, che richiedono un continuo lavoro di sincronizzazione e aggiornamento delle applicazioni che interagiscono tra di loro in batch o in tempo reale. Il problema della gestione dell'esistente è aggravato ulteriormente da vari fattori: aumenta il **livello** sistemico di complessità da affrontare, l'aumento del livello di connettività, anche wireless, dell'impresa al suo interno e verso l'esterno diminuisce la sicurezza, aumentano le

applicazioni mission critical e aumenta di conseguenza la necessità di integrare le infrastrutture per creare un ecosistema IT più efficiente e dinamico.

## Un po' di dinamismo

La crescente complicazione dei sistemi IT è il peggior nemico dell'impresa dinamica, perché impatta non solo sui costi e sulla reattività dell'azienda, ma anche sulla governabilità e la sicurezza: un sistema non completamente sotto controllo è una mina vagante, che può scoppiare nel momento meno adatto. Per risolvere il problema, è necessario spezzare il sistema in un insieme di sottosistemi, creare architetture applicative basate su un ambiente d'interno del quale i programmi possano interagire utilizzando protocolli affidabili e standardizzati. Il fine è un sistema informatico, composto da moduli intercambiabili, in grado di interagire in modo dinamico e automatico per adeguarsi al meglio alle richieste esterne (autonomic computing) e fornire un livello di servizio adeguato per supportare tecnologicamente i processi di business in evoluzione.

L'obiettivo finale è quello di realizzare aziende dinamiche in grado di rispondere velocemente alle richieste del mercato. Tutti i principali modelli di azienda dinamica, che IDC chiama Dynamic Enterprise, IBM definisce On Demand Business e HP ha battezzato Adaptive Enterprise si riferiscono a un'organizzazione i cui processi di business possono rispondere con velocità sufficiente a richieste dei clienti, opportunità di mercato, rischi o condizionamenti esterni. Alla base delle organizzazioni dinamiche è necessario un modello di business componibile, per individuare gli elementi di business che creano valore in azienda e concentrarsi sui componenti

**Un'architettura SOA è costituita concettualmente da tre strati:**

- **Al livello più basso ci sono i componenti che forniscono i servizi.**
- **Nel livello intermedio (che può essere composto da vari sotto-livelli), c'è il middleware detto Enterprise Service Bus (ESB), una "conduttura" che trasporta tutti i servizi, si preoccupa di mappare i servizi disponibili sui processi di business che li richiedono e controlla tutte le funzioni accessorie (impostazione, autenticazione di chi chiama il servizio, logging, sicurezza).**
- **Al livello più alto, ci sono le applicazioni, che sono "composizioni" di diversi servizi.**

**L'utilità di queste architetture è tanto maggiore quanto maggiore è la**

**quantità di applicazioni esistenti che si possono riutilizzare.**

**Il fatto che anche il CICS, il vecchio monitor IBM che gestisce ancora oggi l'80% delle transazioni economiche in tempo reale a livello mondiale, può ora funzionare come provider e come requester di Web Services, è indicativo dell'importanza che anche IBM attribuisce a questa tecnologia. Grazie a questa evoluzione del CICS, anche le applicazioni Cobol transazionali sviluppate negli anni 60 possono ora essere utilizzate come Web Services nell'ambito di una struttura "On Demand, utilizzando WebSphere, l'architettura SOA IBM, o altre architetture d'integrazione basate sugli standard.**

che contribuiscono maggiormente a creare questo valore strategico.

## EAI e SOA

La ricerca relativa alle soluzioni di integrazione delle applicazioni EAI (Enterprise Application Integration) è iniziata negli anni 70 con le prime applicazioni EDI (Electronic Data Interchange) nate per scambiare documenti in forma elettronica tra le aziende, poi sono nati negli anni '90 i primi standard relativi agli oggetti distribuiti, come CORBA, e finalmente si è arrivati a XML, alle architetture SOA (Service Oriented Architecture) e ai Web Services. I Web Services rispetto ad altre tecnologie precedenti hanno il vantaggio di essere basati su protocolli e interfacce già largamente accettate dai produttori di software, come XML e SOAP, e quindi richiedono minori investimenti in software proprietario e minori rischi.

SOA e Web Services, che si trovano al punto di intersezione tra business e tecnologia pura, permettono all'impresa di adattarsi velocemente e mutate condizioni di mercato, fornendo un "service pool" IT adattivo e flessibile e configu-

rato in modo dinamico in base ai business requirement complessivi.

Le SOA sono modelli concettuali che definiscono i meccanismi attraverso i quali i moduli che richiedono particolari servizi interagiscono con i moduli che li forniscono, mentre i Web Services sono la tecnologia abilitante più diffusa (ma non l'unica). È prevista una netta separazione tra chi produce il servizio e chi lo consuma. Chi offre il servizio (service provider) deve poterlo descrivere (esporre) in modo che sia accessibile ai fruitori, mentre chi lo consuma deve disporre di un meccanismo di ricerca (discovery agency).

La discovery può essere realizzata in vari modi, per esempio tramite un registro, spesso realizzato secondo lo standard UDDI (Universal Description Discovery and Integration), un sistema distribuito basato su Web, per catalogare, descrivere e localizzare le funzionalità disponibili come Web Services, oppure attraverso un'indicizzazione esterna compiuta da un motore di ricerca, o infine attraverso modalità Peer-to-Peer (P2P) in cui i service requester hanno anche il ruolo di discovery agency.

Per realizzare questa struttura, bisogna definire gli indicatori e le metriche fondamentali di ogni processo, poi definire un modello dei servizi e infine implementare il modello.

Per la definizione del modello si utilizzano metodologie standard come BPEL (Business Process Execution Language) e WSDL (Web Services Description Language), il linguaggio basato su XML per la descrizione dei Web Services, che serve per capire come utilizzare i Web Services esposti da un server. Le descrizioni vengono generate automaticamente da programmi di utilità, dotate di interfacce grafiche amichevoli.

### Web Services Semantici

Il problema più difficile da affrontare per il funzionamento automatico dei Web Services, che sorge quando è necessario reperire automaticamente un modulo in grado di fornire un determinato servizio (business object), è quello della semantica, cioè della comprensione reciproca del significato dei servizi richiesti e forniti, la cui descrizione è contenuta in un apposito descrittore WSD (Web Service Description), che però non è sufficientemente esaustivo per fornire una definizione univoca e non ambigua. Il problema della comprensione semantica non può essere risolto dai soli Web Services ma richiede l'aggiunta di meccanismi specifici per rendere la semantica elaborabile dalle macchine, basati su ontologie che abilitano la comprensione dei simbo-

## WEB SERVICES

Un Web Service è un componente applicativo accessibile tramite i protocolli standard di Internet, che sono indipendenti dalla piattaforma IT e dal linguaggio. I Web Services rappresentano uno strumento di distribuzione e integrazione flessibile e la loro popolarità deriva dai componenti su cui sono basati: lo standard per la rappresentazione dei dati XML (Extensible Markup Language) e il protocollo SOAP (Simple Object Access Protocol), lo standard W3C per la comunicazione tra applicazioni in formato XML, un protocollo "leggero" per l'interscambio di dati, indipendente dalla piattaforma e dal linguaggio.

Moltissimi programmi possono utilizzare i Web Services in modo nativo, per esempio qualunque programma Java o qualunque applicazione in tecnologia Microsoft .Net. Tecnicamente, un server per Web Services è un server Web che accede ai servizi disponibili (per es. applicazioni COBOL o database) e sulla base delle informazioni estratte genera dei "tracciati" XML basandosi sulla stessa infrastruttura tecnologica utilizzata per i server Web. Le interazioni tra client e server sono analoghe a quelle di un normale server Web con un browser, con la differenza che la risposta destinata non a un fruitore umano ma ad altri programmi.

li da parte delle persone e nello stesso tempo la possibilità di elaborazione automatica (Semantic Web) mediante la descrizione formale delle relazioni tra i simboli. Mentre un semplice meccanismo basato sulla corrispondenza delle chiavi di ricerca (keyword matching) non risolve il problema degli omonimi (per es. ATM = Automatic Teller Machine, ATM = Asynchronous Transfer Protocol, ATM = Azienda Trasporti Milanesi), dei sinonimi (per es. paghe, retribuzione, salari, payroll) e delle forme gergali (parole che in un determinato contesto assumono un significato diverso da quello usuale, per es. "tunnel" di una VPN), aggiungendo altri criteri di ricerca come modelli di classificazione, tassonomie e dizionari di sinonimi, e limitando i campi di ricerca ad ambiti circoscritti (medicina, turismo, logistica), si riesce a migliorare notevol-

mente l'interpretazione semantica.

Oltre a un linguaggio per descrivere le ontologie, per ottenere l'interoperabilità semantica è necessario anche un modello concettuale per i Web Services. Sono in fase di approvazione da parte del W3C (World Wide Web Consortium) due nuovi standard: OWL-S (Ontology Web Language - Services), frutto del lavoro del programma americano DAML (DARPA Agent Markup Language - [www.daml.org](http://www.daml.org)), e WSMO (Web Service Modeling Ontology), prodotto dal gruppo di ricerca europeo WSMO ([www.wsmo.org](http://www.wsmo.org)), coordinati nell'ambito dell'iniziativa SWSI (Semantic Web Services Initiative - [www.swsi.org](http://www.swsi.org)), che hanno entrambi l'obiettivo di definire degli standard per descrivere le proprietà e le funzionalità offerte dai Web Services e per formalizzare le specifiche per la richiesta dei Web Services.

Questi nuovi standard, congiuntamente ad altri lavori in corso, dovrebbero portare, entro la fine di questo decennio, alla definizione della realizzazione pratica dei Semantic Web Services, in grado di condividere la semantica della descrizione dei servizi tra gli utenti e le macchine.

Alla base delle organizzazioni dinamiche è necessario un modello di business componibile, per individuare gli elementi di business che creano valore in azienda e concentrarsi sui componenti che contribuiscono maggiormente a creare questo valore strategico