

**ARCHITETTURA ORIENTATA AI SERVIZI E STANDARD APERTI :
DALLA TEORIA DELLA VALUTAZIONE AL MECCANISMO PROATTIVO DI
REALIZZAZIONE**

ERCOLE VAGNOZZI

Polo Scientifico-Didattico di Rimini, Alma Mater Studiorum, Università di
Bologna

ercole.vagnozzi2@unibo.it

Abstract

La sempre più diffusa economia dei servizi stimola le aziende ad aprirsi verso nuove opportunità e nuove sfide manageriali e informative. La piattaforma dove perseguire queste sfide, ricavandone benefici misurabili, è quella della SOA - Service Oriented Architecture (Architettura Orientata ai Servizi), strutturata in modo tale da allineare i processi di business con i sistemi informativi.

Ogni giorno vengono dematerializzati molti oggetti fisici per cui conta di più l'elaborazione delle informazioni elettroniche che l'interazione con i prodotti fisici. Queste informazioni vengono spesso messe in rete a disposizione di tutti, realizzando una grossa potenzialità per l'aggregazione di ulteriori contenuti e creando al contempo una sorta di strutture e sovra strutture in cui le SOA svolgeranno la parte rilevante delle inter-connessioni. Questa attenzione particolare al progetto di interconnessione probabilmente susciterà nel futuro una riduzione dell'impatto economico del hardware e software, come accennato in diversi lavori sulla *comoditizzazione* del software e dell'hardware. L'articolo che segue induce alla riflessione a livello economico dell'intelligenza connettiva nell'ambito delle SOA e si avvale di esempi pratici sull'implementazione delle SOA da parte delle aziende al fine di concretizzare gli attuali obiettivi di business che possono essere sintetizzati in: Innovazione, Agilità e Valore di Mercato.

Introduzione

Dalla lettura di “state of the Art on Semantic IS Standardization, Interoperability & Quality”¹ è emerso che per capire ed implementare l'architettura orientata al servizio SOA è necessario comprendere cosa sono i sistemi chiusi e aperti. I sistemi chiusi sono caratterizzati da standard, protocolli, linguaggi e formati di dati strettamente protetti e di proprietà privata, che non sono disponibili all'esterno oppure sono disponibili solo sotto una licenza molto costosa. I sistemi

¹ Folmer, Erwin and Verhoosel, Jack (2011) State of the Art on Semantic IS Standardization, Interoperability & Quality. UT, CTIT, TNO en NOiV. ISBN 9789090260303

chiusi contengono tipicamente un software *proprietario* progettato con lo scopo di supportare un singolo sistema. Quando il sistema *proprietario* richiede aggiornamenti o manutenzione, il suo design unico rende gli aggiornamenti costosi e tecnicamente difficili, cosa che porta ad un aumento del costo del ciclo di vita del sistema.

Da quando i sistemi sono stati sviluppati per un singolo scopo, l'interoperabilità con altri sistemi ne ha sofferto molto. Molto spesso viene inserito del “*middleware*” per raggiungere l'interoperabilità tra i sistemi (*middleware* è il software che connette due diversi sistemi separati e chiusi attraverso l'uso di interfacce definite). Questo aggiunge un nuovo livello al sistema ed è potenzialmente più costoso da implementare e mantenere. Comunque, quando i sistemi utilizzano un'architettura aperta il *middleware* risolve il problema dell'interoperabilità.

A differenza dei formati aperti, i formati proprietari sono controllati e definiti da interessi privati (ed hanno, in genere, misure di restrizione o vincoli nell'utilizzo). I formati aperti sono un sottoinsieme degli standard aperti.

L'obiettivo dei sistemi è quello di avere una migliore performance e di essere più efficienti in termini di costi. I sistemi aperti possono raggiungere questo obiettivo. Nei sistemi chiusi, gli aggiornamenti che devono fornire maggiore capacità di elaborazione non possono essere completati senza revisionare il sistema corrente. Comunque, nel sistema aperto, hardware e software possono essere resi modulari, rendendo gli aggiornamenti più efficienti. Questo permette alla maggior parte delle attuali tecnologie di essere utilizzate, nonché permette la competizione tra industrie.

I sistemi chiusi legano il possessore del sistema ad un unico fornitore in possesso del codice sorgente limitando la concorrenza ma generando delle forti interazioni. L'obiettivo principale dei formati aperti è garantire l'accesso ai dati nel lungo periodo senza incertezza presente e futura riguardo ai diritti legali o le specifiche tecniche (interoperabilità). Un secondo obiettivo comune dei formati aperti è di incoraggiare la concorrenza invece di consentire a un solo produttore di mantenere il controllo su di un formato proprietario per inibire l'uso di prodotti concorrenti.

La tabella 1 fornisce una comparazione di alcuni aspetti dei sistemi aperti e chiusi.

Sistemi Chiusi	Sistemi Aperti
Integrazione difficile o più costosa	Alto grado di portabilità, connettività, interoperabilità e scalabilità
Componenti, interfacce, standard e implementazioni sono scelte sequenzialmente	Componenti, interfacce, standard e implementazioni sono scelte interattivamente
Concentrazione soprattutto sul costo di sviluppo e sull'incontro con la mission attuale	Concentrazione sul costo totale di possesso TCO, mantenimento e crescita
Costi globali maggiori	Costi globali inferiori
Importanza critica data a design e implementazione unica	Importanza critica data alla gestione dell'interfaccia e di convenzioni largamente utilizzate
Dipendenti dalla tecnologia del fornitore	Indipendenti dalla tecnologia del fornitore
Minimizzazione del numero di implementazioni	Minimizzazione del numero di tipologie di interfaccia
Poca enfasi sulla modularità	Pesante enfasi sulla modularità
Meno adattabile ai cambiamenti in minacce e tecnologie	Più adattabile a minacce e tecnologie in evoluzione
Fornitori tradizionali	Possono competere anche fornitori non tradizionali
Uso di fornitore unico	Uso di diversi fornitori
Relazione avversa verso i fornitori del sistema	Relazione di simbiosi con i fornitori del sistema
Influenza e controllo sul sistema rigidi e lenti	Sistema cibernetico in tempo reale di influenza e controllo
Sistemi con ciclo di vita minore	Sistemi con ciclo di vita superiore
Semplice test di conformità	Test di conformità più difficile
Trasferimento di tecnologia lento e costoso	Trasferimento di tecnologia veloce e poco costoso
Uso di interfacce, linguaggi, formati e protocolli strettamente protetti e privati (secondo standard legali oppure di proprietà del fornitore)	Uso di interfacce, linguaggi, formati e protocolli pubblici e largamente utilizzati
Uso di preferenze di una singola azienda per impostare e mantenere le specifiche	Uso di un processo con largo consenso per mantenere le specifiche del sistema
Utente come produttore del sistema	Utente come consumatore di componenti

Architettura orientata ai servizi

La SOA è un'infrastruttura tecnologica nella quale le applicazioni software sono scisse in componenti modulari, chiamate *servizi*, e posizionate in un deposito al quale sia gli utenti che gli altri servizi possono facilmente accedere. SOA è una raccolta di servizi auto-contenuti che comunicano tra di loro e svolgono i processi di business attraverso un linguaggio standard. Tecnicamente, la SOA promette sia

il riuso illimitato del software che l'interconnessione. La sfida iniziale della SOA è di sapere dove e come iniziare, dove entrambi sono necessari per costruire servizi che apportheranno servizi tangibili.

Con architettura orientata ai servizi (SOA) si indica, dunque, generalmente un'architettura software adatta a supportare l'uso di servizi Web per garantire l'interoperabilità tra diversi sistemi così da consentire l'utilizzo delle singole applicazioni come componenti del processo di business e soddisfare le richieste degli utenti in modo integrato e trasparente.

SOA è uno stile architetturale il cui scopo è ottenere un legame debole² tra attori-software che interagiscono. Un servizio è un'unità di lavoro svolto da un fornitore di servizi (service provider) per un utente (di servizi) con lo scopo di ottenere un determinato risultato finale, richiesto dall'utente. Sia il provider che l'utente sono ruoli giocati da software che agiscono su richiesta dei rispettivi proprietari.

Un SOA è progettato per il collegamento di richieste di risorse computazionali³, per ottenere un dato risultato per gli utenti, che possono essere utenti finali o altri servizi.

L'OASIS (Organizzazione per lo Sviluppo di Standard sull'Informazione Strutturata) definisce il SOA così:

“un paradigma per l'organizzazione e l'utilizzazione delle risorse distribuite che possono essere sotto il controllo di domini di proprietà differenti. Fornisce un mezzo uniforme per offrire, scoprire, interagire ed usare le capacità di produrre gli effetti voluti consistentemente con presupposti e aspettative misurabili.”

Solo il gruppo OASIS ha prodotto una definizione formale, applicabile profondamente sia alla tecnologia che ai domini aziendali. Tuttavia le definizioni fornite sono molteplici e si differenziano a seconda della caratteristica della SOA che evidenziano.

Le caratteristiche principali che rendono un'architettura software orientata ai servizi sono quindi:

² Legame debole, dall'inglese loose coupling un sistema di interconnessione dei componenti in un sistema o network sicché tali componenti, o elementi, nella minore misura possibile. I legami deboli semplificano le fasi di testing, manutenzione e risoluzione dei problemi in quanto i problemi sono facilmente isolabili.

³ Si intendono qui applicazioni e dati.

- supporto di servizi Web
- sistema integrato
- interoperabilità
- riuso dei servizi

Un modo molto intuitivo per descrivere le potenzialità ed il funzionamento dell'architettura orientata ai servizi è quello di ricorrere all'immagine dei mattoncini delle costruzioni. Associando ad ogni mattoncino che compone la struttura (business process) un servizio, è facile capire come questi mattoncini-servizi possono essere impiegati e agilmente rimescolati (ricombinati) per produrre processi di business-strutture differenti.

Mentre nelle architetture centralizzate esistono legami forti e vincolanti tra hardware software e network, e qualsiasi modifica implica una serie di procedure costose sia in termini di tempo sia di denaro, nelle strutture basate su architetture service-oriented l'approccio è dinamico.

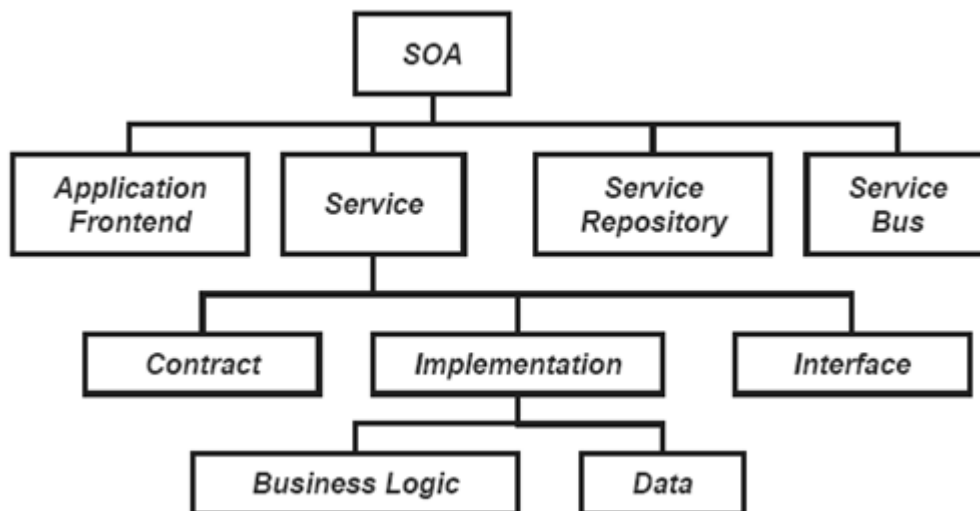


Figura 1: Elementi di una SOA, di Dirk Krafzig, Karl Banke, e Dirk Slama. Enterprise SOA. Prentice Hall, 2005

Come visibile nella figura 1, possiamo analizzare SOA scomponendola nelle parti che la compongono e spiegare ognuna di essa. Similmente ad un organismo vivente, ritroviamo parti quali “faccia”, “cervello”, “cuore” e “sistema nervoso”.

La SOA è un'architettura software basata sui concetti chiave di applicazione front end (interfacce che hanno come destinatario un utente), di (business) service, di service repository (raccolgitore di informazioni) e di service bus.

Un servizio è a sua volta composto da un contratto, una o più interfacce ed una Implementazione.

Application Frontend – Faccia:

Parte dell'applicativo a cui accede l'utente finale per usufruire dei servizi dell'applicazione, si occupa di instaurare e controllare tutta l'attività del sistema aziendale attivando un business process (che si scompone nei servizi atomici) e ricevendo i risultati.

Service – Cuore:

Componente software che incapsula un concetto di business ad alto livello, è il motore di tutto il sistema ed è sua volta scomponibile in:

- **Contract:** il design dell'interfaccia, l'aspetto che maggiormente viene curato;
- **Interface:** strato di comunicazione fra il servizio e la parte dell'applicativo che ne fa richiesta;
- **Implementation:** ossia come sono realizzate le operazioni, a sua volta si costituisce di: - Business logic (logica di esecuzione delle richieste);
- **Data:** dato concettuale richiesto nell'utilizzo del servizio.

Service repository – Cervello

Contiene le informazioni per accedere ai servizi e mette a disposizione i mezzi necessari per scoprire i servizi disponibili, non è un elemento essenziale in un primo momento ma lo diviene a lungo termine.

Service Bus – Sistema nervoso

Connette tra loro tutti i componenti di una SOA, spesso basati su linguaggi di programmazione e sistemi operativi eterogenei. Ha il compito di supportare diversi tipi di comunicazione e di fornire servizi "tecnici": es. logging, auditing, security.

La natura competitiva della globalizzazione costringe tutte le compagnie ad aumentare la collaborazione e a tagliare i costi incrementando la possibilità d'interazione tra i sistemi computerizzati. Un'architettura *service oriented* è un modo per riorganizzare le applicazioni software e le infrastrutture in un insieme di servizi interattivi.

Questi web services permettono alle aziende di mostrare le loro funzionalità di sistema sotto forma di componenti su Internet (o intra-net) attraverso linguaggi, basati su XML, e protocolli standard. I servizi web sono implementati tramite un'interfaccia auto-descrivente basata su standard aperti, come Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL) and Universal Discovery Description and Integration (UDDI).

Il concetto base di SOA è una relazione con tre tipi di partecipanti: il provider dei servizi, il servizio di registrazione e colui il quale ha richiesto il servizio. Le interazioni tra i partecipanti, che sono mostrate in figura 2, coinvolgono la pubblicazione di operazioni di ricerca e collegamento con protocolli aperti standard. Il fornitore dei servizi stabilisce la descrizione di un servizio web tramite WSDL e lo pubblica per un cliente o un registro di servizi. I servizi web vengono registrati presso un gestore di servizi con un deposito UDDI, in modo che i potenziali utenti possano trovarli facilmente. Il richiedente del servizio utilizza un'operazione di ricerca per trovare la descrizione del servizio nel registro dei servizi. Attraverso SOAP su protocollo http, il richiedente del servizio può usare la descrizione del servizio trovata nel registro dei servizi per collegarsi con il provider del servizio e interagire con l'implementazione del servizio.



Figura 1 Modello di architettura per ambiente SOA

Creazione di valore economico per l'impresa tramite SOA

Dall'analisi effettuata fin'ora di cosa sia una SOA e delle opportunità che essa offre e livello di new technology è emerso che il potenziale di questa nuova architettura software è senza dubbio di grande rilievo. Nei reparti ICT (Information and Communication Technology) si parla di SOA già da una decina di

anni, mentre è relativamente più nuovo l'interesse mostrato nei suoi confronti in ambito economico, dove i manager delle grandi imprese hanno iniziato a prendere in considerazione la SOA Aziendale e il valore economico che essa può produrre per loro. Volendo analizzare l'evoluzione di questo settore, SOA Aziendale, possiamo dire che si trova nella fase di introduzione/apparizione: le grandi software house promuovono il prodotto attraverso una nicchia di imprese, clienti di grandi dimensioni o fortemente orientate all'innovazione, se il settore di appartenenza lo consente.

Grazie all'effetto moda, o nella versione dispregiativa effetto gregge⁴, ed al lavoro del gruppo Gartner, che ha promosso la SOA, molte aziende si sono focalizzate sull'implementazione di questa nuova tecnologia senza verificarne l'effettivo guadagno.

Indubbiamente, il sopra citato effetto moda ha avuto il suo impatto anche nel settore IT ed è chiaro che, soprattutto nel caso di una nuova tecnologia o metodologia di sviluppo, l'attenzione facilmente si focalizzi su ciò che sta attraendo la massa, ma è anche vero che le SOA presentano vantaggi implementativi e prestazionali notevoli in una grande vastità di contesti.

L'osservazione del contesto ha portato Gartner a constatare come il 40% delle aziende, in realtà, non misurano effettivamente i risultati derivanti dall'utilizzo di SOA, con la conseguenza che, spesso, lo sforzo di aderire all'architettura non è quantificato né, tanto meno, comparato con quelle che sono le effettive motivazioni o il vantaggio derivante.

Per valutare il successo di un'architettura SOA sono stati proposti nove strumenti di valutazione:

1. l'effettivo miglioramento dell'efficienza rispetto alla normale (o attuale) esecuzione dei processi di business;
2. riduzione dei costi amministrativi dei processi;
3. aumento della visibilità dei processi di business, una sorta di conseguenza intrinseca nelle SOA, in quanto è parte della individuazione delle atomicità dei servizi, ma è anche un elemento che può dare un segno dell'aderenza del progetto all'architettura;
4. ridurre il numero di interventi manuali o basati su supporti cartacei;

⁴ Stephen Swoyer, Study Highlights SOA Herd Mentality, 2009

5. migliorare l'efficienza dei livelli di servizio;
6. velocizzare l'implementazione dei processi;
7. riduzione del time to market, cioè del tempo effettivo che intercorre tra l'ideazione di un prodotto e la sua effettiva commercializzazione: ovviamente tale criterio si concretizza anche nell'effettivo miglioramento che può dare una SOA in tutte le fasi del ciclo di vita del software;
8. velocità dei cicli di progetto;
9. globale riduzione dei costi di sviluppo e manutenzione del software.

In merito a quanto si evince dal lavoro del gruppo Gartner, David Linthicum⁵ propone una propria versione. Il punto di partenza dell'autore, probabilmente condivisibile, è che le misurazioni proposte da Gartner non sono specifiche delle SOA, ma sono applicabili a qualsiasi tipo di architettura software.

Le caratteristiche proposte da Linthicum, effettivamente, risultano più orientate alle peculiarità delle SOA.

1. La prima caratteristica da considerare è l'adattabilità dei processi alle modifiche degli schemi sottostanti: un'architettura SOA deve descrivere i processi come funzionalità atomiche ed eventuali cambiamenti nella struttura dei dati non dovrebbero impattare sui servizi offerti;
2. Il secondo termine di valutazione è il reciproco del precedente, ossia la possibilità di modificare i processi senza necessità di cambiare gli schemi sottostanti. Appare evidente, dai primi due principi, che l'attenzione è soprattutto rivolta alla necessità di un alto grado di astrazione nell'implementazione dei servizi;
3. Terzo elemento fondamentale è la possibilità di modificare il core business dei processi mediante operazioni di riconfigurazione, invece che intervenendo sul codice;
4. La quarta caratteristica è la possibilità di estendere i servizi al di fuori del contesto aziendale: il riferimento, in particolare, è al cloud computing⁶ e all'importanza di guardare alle opportunità che possono essere colte da tale metodologia;

⁵ CTO dei Blue Mountain Labs, ed esperto internazionalmente riconosciuto di cloud computing e SOA.

⁶ Si intende un insieme di tecnologie informatiche che permettono l'utilizzo di risorse hardware come storage e CPU o software distribuite in remoto.

5. Il quinto principio è l'inverso del precedente e, probabilmente, il più naturale: la possibilità di esporre processi e servizi all'interno del contesto aziendale.

Definiti quelli che sono gli strumenti e il sistema di valori per valutare il successo dell'implementazione SOA è possibile analizzare in che modo essa produce valore economico per le aziende.

La creazione di valore economico per le imprese tramite l'implementazione di un'architettura orientata ai servizi può essere ricondotta a 5 sostanziali situazioni di vantaggio come illustra IBM nei suoi "cinque progetti SOA che si ripagano in sei mesi".

Il valore economico può essere creato tramite:

1. il riutilizzo di prodotti e servizi;
2. riducendo il rischio;
3. riducendo gli sforzi.

Il riutilizzo di prodotti/servizi è realizzabile grazie alla capacità di questa architettura di ricombinare i servizi (ricordando la similitudine con i mattoncini delle costruzioni) per offrire all'utente finale il prodotto richiesto, precisando che per servizio e richiesta di servizio si intende un atto, il più delle volte, automatizzato. Ipotizziamo di essere un dipendente di una banca, esiste un servizio che noi chiamiamo "anagrafica comune" dove sono contenuti tutte le anagrafiche dei clienti, ed ipotizziamo tre interfacce (o moduli) chiamandoli rispettivamente "movimenti conto corrente", "richiesta mutuo" e "compleanno" (operazione automatica). Facendo uso di una qualsiasi di queste interfacce si accede sempre allo stesso servizio "anagrafica comune", estrapolando a seconda della richiesta i dati necessari.

La riduzione del rischio si esplica nella caratteristica intrinseca della SOA di interoperabilità, che permette di evitare problemi di incompatibilità a livello di hardware e software.

I cinque progetti IBM che si ripagano in sei mesi

Come menzionato precedentemente IBM ha portato a termine cinque progetti, ognuno dei quali dimostra un vantaggio apportato dalla SOA all'attività aziendale e dimostra anche, a fronte dell'investimento iniziale, che il costo sostenuto si

ripaga in sei mesi. Questo si spiega perché molte imprese non sono pienamente coscienti del numero di processi doppi, e quanto questi pesino in termini di costi, che avvengono utilizzando applicazioni separate e non comunicanti. L'utilizzo di un sistema basato su un'architettura service-oriented permette una gestione centralizzata dei processi, riducendo lo spreco.

Il vantaggio di implementare SOA consiste anche nella capacità di poter interagire fra sistemi eterogenei, sia all'interno della stessa azienda sia con aziende esterne con le quali si hanno soventi contatti. La SOA facilita e rende meno dispendiosa l'interazione fra applicazioni diverse.

Gli esempi riportati di seguito forniscono un'immagine chiara di tutto il processo decisionale che porta all'adozione di un'architettura orientata ai servizi. Superato l'impatto iniziale, la flessibilità e l'agilità acquisite dalle aziende grazie a SOA hanno gettato le basi per il raggiungimento e mantenimento di un vantaggio competitivo durevole negli anni.

Progetto 1: Servizio di notifica delle date di consegna: Fornire un'unica fonte di informazioni

Un' importante azienda del retail ha stimato l'impatto, in termini di ricavi, del progetto SOA in 20 milioni di dollari l'anno. Il progetto, che costa una piccola frazione di quei 20 milioni di dollari, consisteva nel creare un servizio centralizzato utilizzato come unica fonte di informazione per gli impegni riguardo alle date di consegna. Si è trattato quindi di implementare un servizio centralizzato per le date di consegna. Il nuovo sistema di evasione ordini dell'azienda, quando una data di consegna viene modificata, invia una notifica di modifica della data a questo servizio event-driven (servizi che si attivano nel momento in cui si verifica un determinato evento, in questo caso la modifica della consegna), attraverso l'ESB (Enterprise Service Bus) del retailer. Di conseguenza, il database del sistema degli ordini – e qualsiasi altro sistema che è sottoscritto a questo servizio – viene aggiornato immediatamente. Questa capacità consente agli addetti all'assistenza di rispondere e affrontare le problematiche dei clienti con più rapidità e precisione rispetto alla situazione precedente in cui la modifica della data di consegna avveniva su uno solo dei sistemi utilizzati dal retailer,

impedendo che l'informazione potesse essere aggiornata nel sistema elaborazione ordini.

Progetto 2: Servizio contestazione transazioni: Automatizzare i processi tra diverse aziende e utenti

Questo progetto di IBM ha consentito a un'organizzazione di servizi finanziari di ridurre significativamente i processi ad alta intensità di manodopera e i costi elevati associati alle transazioni contestate, che costituivano una parte importante dei costi di gestione. Automatizzare questo processo, grazie alla creazione di un servizio centralizzato, ha consentito all'organizzazione di realizzare un risparmio stimato di costi superiore a 200 milioni di dollari l'anno. Questo risparmio è stato conseguito sostituendo una serie di processi manuali e arcaici con un servizio automatizzato, semplice da implementare.

Per creare una soluzione, l'organizzazione di servizi finanziari ha rilasciato un'interfaccia esterna al proprio sistema transazionale centrale, che consente ai partner di trasmettere le richieste di contestazione all'organizzazione per conto dei clienti dei negozi dei partner. Per registrare una contestazione, i clienti ora si collegano semplicemente al sito Web dell'esercizio commerciale e visualizzano un elenco di transazioni imputate sul loro conto. Possono quindi scegliere le transazioni che desiderano contestare. Il sito Web invia questa richiesta al servizio contestazione transazioni dell'organizzazione di servizi finanziari. L'autenticazione fornita dai clienti durante il collegamento al sito Web consente all'organizzazione di evitare di richiedere la documentazione cartacea con la firma scritta a mano. Oggi il processo di contestazione delle transazioni richiede in media tre ore, anziché 20 giorni, e costa solo dai 40 ai 70 dollari per transazione, invece dei precedenti 400-700 dollari— vale a dire una riduzione dei costi del 90 per cento.

Progetto 3: Servizio di verifica dei documenti: Conseguire un risparmio grazie al riutilizzo dei servizi

In questo esempio, un'agenzia pubblica della regione Asia Pacifica utilizzava una combinazione di processi manuali e parzialmente automatizzati per verificare documenti quali passaporti, patenti di guida e certificati di nascita. Questi metodi

di verifica richiedevano molta manodopera ed erano lunghi da completare. Per eliminare questo flusso di lavoro manuale, l'agenzia ha deciso di implementare un servizio di verifica dei documenti totalmente automatizzato.

Dopo che questa prima agenzia ha implementato con successo il servizio, altre quattro agenzie della pubblica amministrazione hanno chiesto di utilizzare questa funzione automatizzata per le proprie procedure di verifica dei documenti. In base al successo ottenuto con le prime conversioni e rendendosi conto che questo progetto avrebbe potuto costituire il fondamento per servizi centralizzati simili, la pubblica amministrazione ha chiesto alla prima agenzia di ospitare il servizio in modo che potesse essere usato dalle altre quattro organizzazioni e realizzare così un servizio centralizzato. Per realizzare la centralizzazione del servizio di verifica dei documenti in tutte le cinque organizzazioni pubbliche, l'agenzia ha dovuto mantenere l'interfaccia dei servizi utilizzata dal proprio sistema esistente e sviluppare inoltre un'interfaccia che potesse essere usata tra tutti gli altri quattro sistemi.

Grazie all'uso della SOA, questa amministrazione della regione Asia Pacifica è riuscita a rilasciare in sole cinque settimane un'infrastruttura per ospitare i servizi in modo centralizzato. La funzionalità ESB⁷ fornisce all'amministrazione la flessibilità per integrare tecnologie e protocolli diversi, in modo che i componenti possano essere riutilizzati. Inoltre, l'infrastruttura costruita per supportare l'ESB è facilmente scalabile per ospitare altri servizi centralizzati in aggiunta alla verifica dei documenti. Adottando la SOA per riutilizzare i componenti dei servizi, questa amministrazione è riuscita a ridurre i propri costi IT e a fornire ai cittadini un servizio più rapido e più accurato.

Progetto 4: Servizio di connettività e-commerce: Vendere attraverso i siti web dei partner per aumentare il fatturato

Questo caso è incentrato su un'azienda che vende articoli da regalo esclusivi, possiede il proprio canale di vendita, ma per trovare un numero maggiore di potenziali clienti si propone di utilizzare anche i canali di vendita di terzi, cioè

⁷ Deriva da scalare, che indica la proprietà di un'architettura di poter aggiungere o togliere servizi senza dover modificare l'intera struttura software

partner a cui essa paga una commissione per utilizzarne lo spazio di vendita online.

L'intervento di IBM ha quindi lo scopo di collegare in modo integrato il catalogo dell'azienda, i sistemi di gestione magazzino e i sistemi di evasione ordini ai siti Web di front-end dei suoi partner.

La complessità della procedura dipende da diversi fattori quali (1) la diversità delle piattaforme usate dai diversi partner e a causa di tale diversità, (2) la lentezza nell'attività di integrazione B2B. La visione dell'azienda era di esporre le capacità in tempo reale del proprio sistema di evasione ordini, fornendo un'interfaccia in grado di consentire ai siti dei partner di riportare livelli di magazzino più accurati durante il processo di ordinazione e di instaurare un processo di ordinazione in real-time. Per realizzare questo obiettivo, l'azienda ha fornito ai propri partner una serie di interfacce SOAP. Questi servizi sono legati all'applicazione di elaborazione ordini tramite un collegamento WebSphere MQ (prodotto IBM che permette di collegare applicazioni e Web service supportando per oltre 80 configurazioni di piattaforme esistenti). Per integrare i servizi con la tecnologia WebSphere MQ, l'azienda trasforma le richieste da SOAP a WebSphere MQ e quindi effettua una trasformazione da WebSphere MQ a SOAP per la risposta. Fornendo i mezzi per effettuare le trasformazioni necessarie per l'interoperabilità di questo protocollo, l'azienda può riutilizzare i componenti di informazione esistenti, pur mantenendo i legami deboli dei componenti. Questo consente inoltre il riutilizzo futuro dei componenti, man mano che l'azienda migliora le sue capacità di integrazione sia interne che esterne.

Progetto 5: Servizio giudiziario penale: Costruire una SOA d'impresa utilizzando sistemi CICS

Per questo progetto, un'agenzia pubblica nel Nord America ha riconosciuto l'esigenza critica di aggiornare la tecnologia utilizzata per accedere al proprio sistema giudiziario penale. Questo sistema fornisce ai funzionari autorizzati importanti informazioni legali per rintracciare gli imputati, inclusi i dettagli vitali sugli arresti, le schedature e la libertà vigilata. L'agenzia si è resa conto che il sistema CICS- Customer Information Control System (un server transazionale che utilizza un mainframe IBM impiegato per gestire grandi volumi di transazioni) di base supportava le sue esigenze funzionali, ma che i terminali di interfaccia utente

utilizzati per accedere a queste informazioni erano troppo difficili da usare e non supportavano l'accessibilità richiesta ai pacchetti software comunemente in uso presso i professionisti legali. L'organizzazione ha deciso di mettere a riposo questi terminali e di riprogettare il modello di interazione, usando un'interfaccia Internet basata su portale. Questa soluzione offre un accesso semplificato a un pubblico molto più vasto, aiutando l'agenzia a ridurre significativamente i costi di formazione e consentendole di supportare un crescente numero di utenti. Inoltre, i professionisti possono usare questa nuova interfaccia insieme al comune software di settore, lavorando così con maggiore efficienza. L'agenzia può continuare a sfruttare la funzionalità esistente del sistema CICS e ha quindi risparmiato il tempo e i costi necessari per riscrivere questa logica su un'altra piattaforma.

Il progetto ha portato grandi vantaggi sia per l'agenzia che per gli utenti: l'agenzia ha amplificato il valore derivato dalle applicazioni esistenti, estendendone l'uso, e i professionisti legali hanno beneficiato di un migliore accesso alle informazioni giudiziarie penali sul sistema CICS nonché della possibilità di avere una vista unica e integrata delle informazioni, sia dal sistema CICS che da altre fonti.

Conclusioni

L'analisi effettuata finora, ha messo in evidenza il grande potenziale delle architetture orientate ai servizi nonché le problematiche legate al suo sviluppo ed al suo utilizzo in ambiti aziendali. Si è voluto seguire in questo lavoro un processo conoscitivo delle architetture orientate ai servizi in modo da renderne evidenti le caratteristiche che configurano SOA come un traguardo che molte aziende a livello globale stanno raggiungendo.

Sebbene lo scopo primario di questo lavoro sia analizzare l'impiego della SOA come supporto che renda possibile, efficace, sicuro e conveniente un sistema di architettura aperta, si è scelto di dedicare molto più spazio alla trattazione del concetto di architettura orientata ai servizi ed al suo funzionamento. Questa scelta deriva da un percorso cognitivo dove è emerso che per poter capire come sia possibile sfruttare ciò che SOA offre in questo ambito applicativo, è necessaria una piena comprensione di ciò che è, e di come IT e agenti economici possono collaborare al suo sviluppo, giacché si parla di una soluzione informatica al servizio di una necessità aziendale e istituzionale. Questa necessità comporta

anche uno sforzo per comprendere le dinamiche di programmazione e di funzionamento di questa architettura software e dei linguaggi di programmazione usati ed anche la difficoltà di tradurre in italiano terminologie specifiche che comunemente si utilizzano in lingua inglese.

Al termine di questo percorso si può affermare che SOA costituisce un cambiamento. Già l'avvento di internet aveva a suo tempo profondamente modificato il modo di concepire le interazioni, non solo tra privati, ma anche tra aziende e istituzioni; similmente, la diffusione della SOA comporta o comporterà in un prossimo futuro un profondo cambiamento, e si prospetta essere un passaggio obbligato nell'evoluzione del sistema aziendale sia a livello di gestione, ricordando la graduale trasformazione degli ERP da silos di informazioni chiusi a strutture aperte e interoperabili, SOA possiede qualità tali da poter creare un effettivo valore economico per le imprese, ma come ben specificato, essa deve essere implementata in ambienti in cui è chiara la modalità con cui si svolgono i processi di business senza incorrere nell'errore, come prospettato da Gartner, di rientrare in quel 40% di imprese che non riescono a misurare il contributo apportato da SOA né lo sforzo sostenuto per aderirvi.

Senza dubbio gli sforzi sia monetari che organizzativi per implementare un'architettura SOA e quindi reingegnerizzare in parte i processi di business sono consistenti, ma dagli esempi analizzati, sia nei 5 casi IBM sia negli esempi dei livelli di implementazione della SOA, si evince che i benefici sono evidenti e che i risultati sono subito riscontrabili.

BIBLIOGRAFIA

Liquid Life, 2005, Z Bauman

L'intelligenza collettiva: per un'antropologia del cyberspazio, 2002, Pierre Lévy, Feltrinelli

Oranization science, 2005, E Romanelli, OM Khessina

Dal *cloud computing* al SOA: l'importanza dell'interazione con il processo e l'ecosistema, 2009, Ercole Vagnozzi, AMS Acta.

Cinque progetti SOA che si ripagano in sei mesi, 05/2006, IBM.

Vagnozzi, E., Dal Cloud Computing al SOA: l'importanza dell'interazione con il processo e l'ecosistema, 2009, Ercole Vagnozzi, AMS Acta ISSN: 2038-7954 Università di Bologna.

Vagnozzi, E., Introduzione al valore economico dell'intelligenza connettiva nell'architettura orientata ai servizi, 2009, Ercole Vagnozzi, AMS Acta ISSN: 2038-7954 Università di Bologna,

Tapscott D., Williams A. D. (2006) Wikinomics 2.0:How Mass Collaboration changes Everything, Portfolio, USA, trad. It. La collaborazione di massa che sta cambiando il mondo, Rizzoli – ETAS, Italia

Shapiro C., Varian H. R. (1998) Information rules: a strategic guide to the network economy, Harvard Business School Press, USA, trad. It. Information rules. Le regole economia dell'informazione, Etas, Italia

Rifkijn J. (2001) The Age of Access., Tarcher/Putnam, USA, trad. It. L'era dell'accesso, Mondadori, Italia

Mahadevan L., Kettingery W. J., Paulz R., (2009) A Three Level Model of SOA Maturity:Toward Achieving Sense and Respond, AIS Electronic Library (AISeL), USA

Schelp J., Aier S. (2009) SOA and EA—Sustainable Contributions for Increasing Corporate Agility, Proceedings of the 42nd Hawaii International Conference on System Sciences – 2009, USA

Krafzig D., Banke K., Slama D. (2004) Enterprise SOA: Service-Oriented Architecture Best Practices, Prentice Hall, USA

David S. Linthicum (2009) Cloud Computing and SOA Convergence in Your Enterprise: A Step-by-Step Guide Web ISBN-10: 0-321-65939-2

Folmer, Erwin and Verhoosel, Jack (2011) State of the Art on Semantic IS Standardization, Interoperability & Quality. UT, CTIT, TNO en NOiV. ISBN 9789090260303