

Introduzione

Una Application Programming Interface (API) può essere paragonata a una porta: il suo compito principale è consentire l'accesso a qualcosa. Le porte possono essere di forme, dimensioni, colori e materiali diversi e possono offrire livelli di sicurezza diversi, per proteggere tutto quello che si trova dietro di esse (Figura I.1).

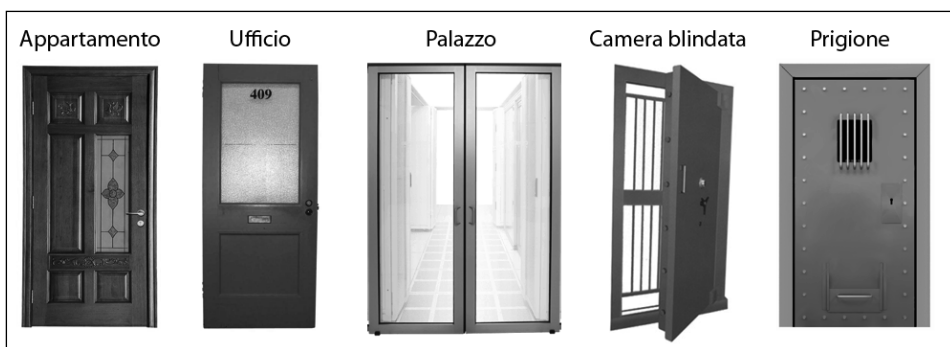


Figura I.1 Differenti tipi di porte.

Nel caso delle API, quello che “si trova dietro” sono risorse digitali come dati grezzi e ripuliti, immagini, video, documenti e addirittura funzionalità che eseguono calcoli complessi o elaborazioni dei dati sulla base degli input.

A volte, capita che le porte vengano progettate male o costruite nel modo sbagliato (Figura I.2).

Lo stesso vale anche per le API. La gestione delle API è una disciplina che si è sviluppata per fornire i processi e gli strumenti necessari per scoprire, progettare, implementare, usare o mantenere in esercizio API di livello aziendale. Cosa particolarmente importante, questa disciplina ha la responsabilità di gestire le comunità intorno alle API, comunità che possono essere costituite dagli sviluppatori che le costruiscono o le usano nelle loro app, ma anche da executive d'azienda e dell'IT che vogliono accelerare l'innovazione, a costi più bassi.



Figura I.2 Errori nella progettazione delle porte. Fonte: <http://www.constructionhunter.com.au/blog/industry-news/20-photos-that-will-make-you-question-your-faith-in-humanity>.

Possiamo concluderne che il vero obiettivo della gestione delle API è fornire valore. Per le aziende, questo può significare la riduzione dell'impegno dello sviluppo mediante l'uso di API esistenti (sviluppate internamente o da terzi). Il valore può venire anche dalla monetizzazione di API che offrono prodotti intangibili (risorse digitali), per i quali gli sviluppatori o gli executive sarebbero disposti a pagare.

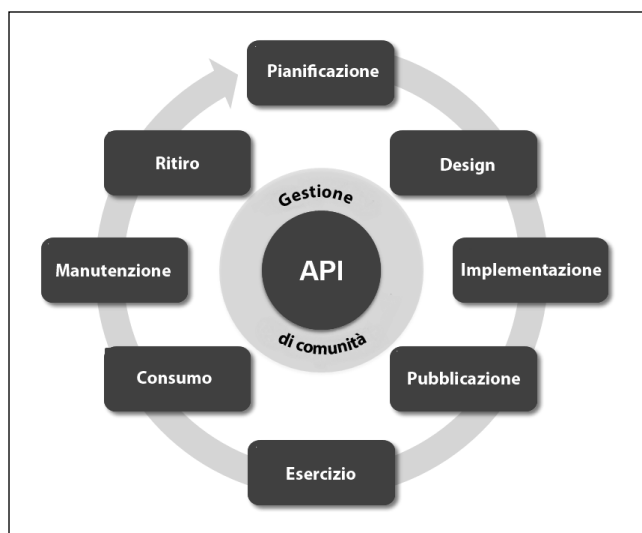


Figura I.3 Il ciclo di gestione delle API.

Si può fornire veramente valore solo se tutto il ciclo di realizzazione di qualcosa, in questo caso le API, è perfettamente compreso, ottimizzato e controllato. La creazione di una strategia API con finalità e obiettivi chiari è seguita dalla fase di avvio di una API attraverso

workshop di innovazione. Poi vengono la pianificazione, la progettazione, l'implementazione, la pubblicazione, l'esercizio e il monitoraggio, fino al "pensionamento" dell'API. La gestione delle API non è più solamente implementazione. Nei marketplace come programmableweb.com e [RapidAPI.com](https://rapidapi.com) si trovano migliaia di API pubbliche (e ogni giorno altre se ne aggiungono), ciascuna delle quali rappresenta una porta digitale alle offerte di prodotti digitali di un'azienda. Pensare che tutte le API debbano essere sviluppate internamente è del tutto sbagliato.

Per riassumere, la gestione delle API ha a che fare tanto con il fornire i mezzi per scoprire e usare API pubbliche quanto con l'implementazione di nuove API. Al centro di qualsiasi iniziativa di gestione delle API deve esserci la creazione di valore per le aziende, ma anche per gli utenti.

API come centro degli ecosistemi digitali

Quando le organizzazioni abbracciano l'adozione di API pubbliche o creano nuovi prodotti API, si verifica un effetto interessante: cominciano a crearsi nuovi ecosistemi, tutti abilitati attraverso le API, non appena si genera valore dall'adozione e dalla combinazione delle risorse digitali di altri nella creazione di nuovi prodotti.

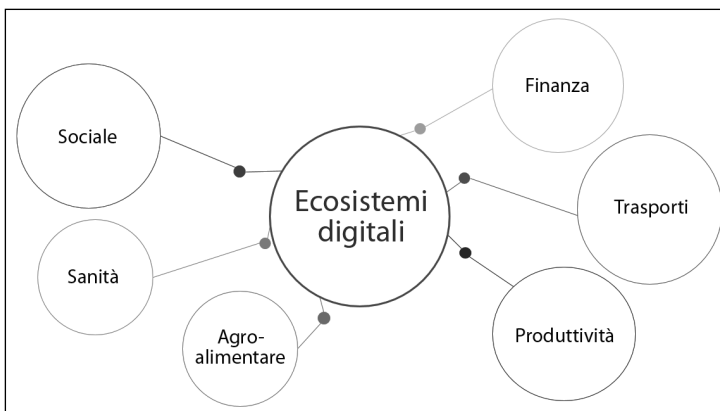


Figura I.4 Creazione di nuovi ecosistemi.

In effetti, uno studio condotto dalla McKinsey prevede che entro il 2025 gli ecosistemi digitali rappresenteranno circa il 30% delle attività economiche globali, corrispondente secondo questa società di consulenza a circa 60 milioni di miliardi di dollari.

NOTA

Lo studio citato si può trovare all'indirizzo <https://www.mckinsey.com/industries/financial-services/our-insights/insurance-beyond-digital-the-rise-of-ecosystems-and-platforms>.

Non si tratta solo di cifre enormi: far parte di questo ecosistema digitale sarà anche questione di sopravvivenza, per alcune organizzazioni.

API come paradigma in evoluzione

Le API non sono una novità, tutt'altro. L'uso del termine può essere fatto risalire al 1968, a una pubblicazione intitolata *Data Structures and Techniques for Remote Computer Graphics* di I.W. Cotton e F.S. Greatorex, Jr. Da allora il termine è ricomparso in protocolli proprietari come Remote Procedure Call (RPC) della Sun Microsystems, Common Object Request Broker Architecture (CORBA) e Distributed Component Object Model (DCOM). Lo abbiamo visto anche in standard pubblici come XML-RPC, che poi evolvendo è diventato il Simple Object Access Protocol (SOAP), diventato, con il Web Services Description Language (WSDL), la base di Web Services e dell'architettura orientata ai servizi (Service-Oriented Architecture, SOA).

C'è stato poi un cambiamento di paradigma, verso API centrate sulle risorse e più leggere, basate sullo stile di architettura REST. Ora siamo tornati a RPC con tecnologie emergenti come GraphQL e gRPC, la cui diffusione sta rapidamente aumentando.

NOTA

L'evoluzione delle API è descritta più in dettaglio nel Capitolo 6.

Tuttavia, quello che vediamo oggi non è solo un cambiamento tecnologico delle tecnologie API: l'emergere delle API come strumenti per abilitare ecosistemi digitali ha creato un'economia propria, un'economia API, che ha un impatto più fondamentale su come le aziende organizzano i propri team.

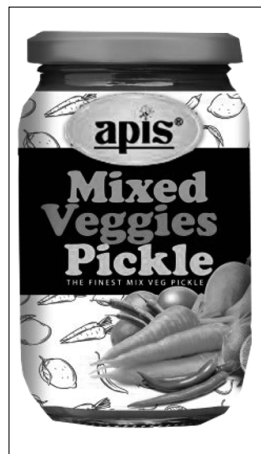


Figura I.5 API come prodotti di business. Fonte: <http://www.apisindia.com/>.

Se le aziende si rendono conto che le API possono essere prodotti di business a pieno titolo, i team che le realizzano non saranno più considerati semplicemente team di IT o, per dirlo brutalmente, centri di costo. Perché un'azienda abbia successo nell'economia API, deve abbandonare le modalità tradizionali di realizzazione dell'IT e passare a un approccio centrato sul prodotto, dove l'obiettivo principale è realizzare un prodotto API di successo e remunerativo.

Anche le capacità tecniche, in particolare ora che il cloud computing è diventato la normalità, devono evolvere per dare ai team tutti gli strumenti di cui hanno bisogno per creare prodotti innovativi e competitivi sul mercato.

A chi è rivolto questo libro

Questo libro sarà utile a sviluppatori, architetti e anche a executive dell'IT e del digitale che cercano fonti di ispirazione per definire e implementare:

- strategie API orientate dal business;
- architetture e pattern guidati dalle API;
- stili architetturali API da usare (per esempio, REST, GraphQL o gRPC);
- il ciclo di vita completo delle API, compresi i cicli correlati, come quelli del servizio e del consumatore di API;
- modelli operativi adeguati per i prodotti API.

Inoltre, poiché è indipendente dalla tecnologia e non offre concezioni o strumenti forti, il libro può essere usato anche come riferimento per confrontare prodotti diversi, non importa se commerciali o open source.

Argomenti trattati

- Il Capitolo 1, *Il valore di business delle API*, fornisce il contesto per le parti seguenti, chiarendo che cosa significhino le API per le aziende e perché dovrebbero essere adottate. Parla anche dei driver di business per le API e come determinarne il valore sulla base di una catena del valore delle API.
- Il Capitolo 2, *L'evoluzione delle piattaforme API*, fa un passo indietro ed esamina in dettaglio come le tecnologie e le piattaforme siano evolute dal middleware tradizionale e dalle architetture SOA centrate sul bus fino alle architetture pienamente federate, multi-cloud e basate su microservizi che consentono l'esistenza delle API e la loro gestione ovunque si trovino le informazioni.
- Il Capitolo 3, *Strategia API guidata dal business*, ha come obiettivo presentare un approccio generale alla definizione di strategie API che abbiano scopi e obiettivi chiari, concisi e centrati sul business.
- Nel Capitolo 4, *Architetture guidate dalle API*, esamineremo un'architettura di riferimento e tutte le capacità necessarie per implementare API moderne e disaccoppiare completamente i (micro) servizi. Questo capitolo definisce come deve essere fatto uno stack moderno.
- Il Capitolo 5, *Pattern delle architetture guidate dalle API*, prosegue il tema aperto nel precedente per esaminare come le diverse capacità descritte nell'architettura di riferimento possano essere combinate per fornire soluzioni corrette a problemi comuni.
- Il Capitolo 6, *Stili architetturali moderni per le API*, presenta una panoramica dettagliata degli stili più "trendy" (nel momento in cui è stato scritto il libro). Questo capitolo

può essere un'importante fonte di ispirazione per chiunque cerchi una prospettiva sui diversi stili API e i relativi pro e contro.

- Il Capitolo 7, *Il ciclo di vita delle API*, esamina tutto il ciclo di vita delle API e anche i cicli correlati, come quello dei servizi e quello del consumatore di API. È un capitolo di riferimento per quanti desiderino implementare processi e strumenti API end-to-end da zero e per quanti cerchino ispirazione su come migliorare quelli esistenti.
- Nel Capitolo 8, *Modello operativo target dei prodotti API*, esamineremo che cosa significhi realmente trattare le API come prodotti e quali conseguenze comporti per le organizzazioni. Dai concetti fondamentali ai diversi modelli operativi con i loro pro e contro, questo capitolo affronta un tema raramente discusso nel mondo delle API.

Immagini a colori

Potete scaricare un file PDF con la versione a colori di schermate e diagrammi utilizzati nel libro. Lo trovate all'indirizzo <http://bit.ly/apo-api>.