

IL CICLO DI VITA del software logistico

La serie di articoli sul tema della software selection continua con un nuovo approfondimento dedicato al ciclo di vita del software per la logistica: quali sono le fasi principali e come gestirle per ottenere la massima efficienza operativa

Gli scenari in cui un'azienda può trovarsi in relazione ai software logistici sono diversi e il grafico 1 le illustra in modo efficace. Se un software logistico non è ancora presente, l'azienda deve decidere se svilupparlo internamente (make) o acquistarlo sul mercato (buy). Optando per l'acquisto, occorre scegliere

tra le tante soluzioni disponibili: questa fase è chiamata software selection, ed è l'oggetto principale di questa serie di articoli.

Anche se l'utilizzo di anglicismi può essere criticato, in questo caso il termine "software selection" è utile per esprimere un concetto preciso: si tratta di un processo strutturato e articolato in più fasi, come la gap analysis e il miglioramento

continuo (temi che approfondiremo nei prossimi articoli).

Un percorso a tappe

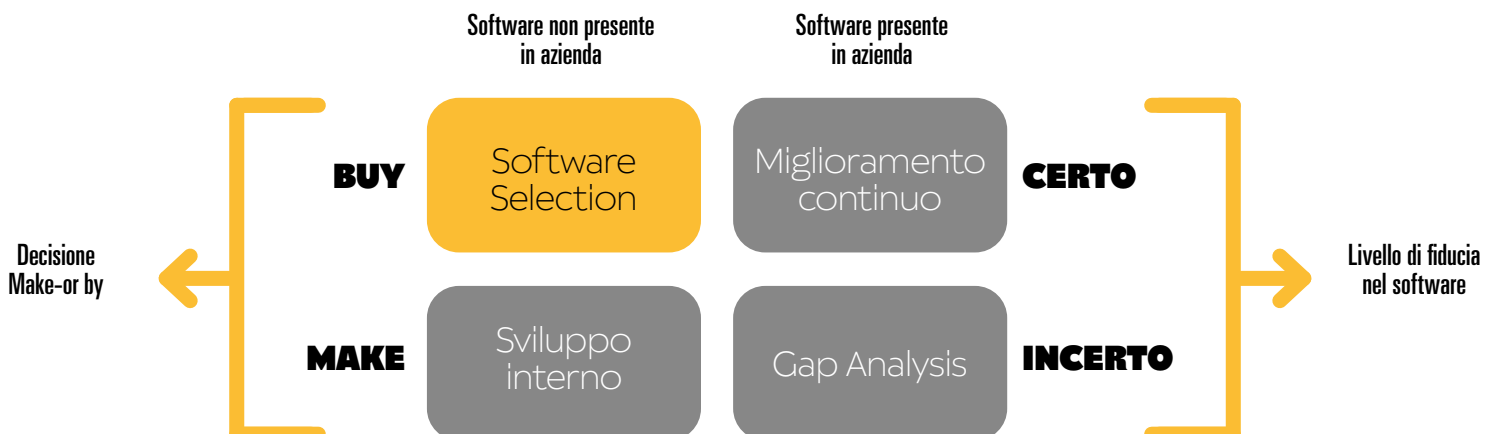
Questa è la seconda pubblicazione della serie di articolo dedicata alla software selection in ambito logistico.

- Il primo articolo ha analizzato le diverse tipologie dei software logistici
- Il prossimo tratterà il tema: software selection, cos'è e perché svolgerla in ambito logistico.

NOTE

1 Induvation è una società di consulenza manageriale che opera nell'ambito della Logistica e delle Operations, nata in Germania e attiva in Italia dal 2007. Come spin off del Fraunhofer IML di Dortmund, l'istituto di ricerca e sviluppo in ambito logistico più importante a livello europeo, l'azienda si occupa di intralogistica e gestione della produzione, con integrazione dei principi Lean. Ad oggi, numerosi sono i progetti seguiti dal team con aziende di dimensioni e settori merceologici differenti. Induvation collabora con i MotionMiners per l'efficientamento dei processi di magazzino e con il Team Warehouse Logistics di Fraunhofer Institute IML in ambito software per la logistica (WMS, TMS, ecc.).

Grafico 1 **Situazioni decisionali nella gestione del software logistico**



La decisione strategica tra sviluppo interno e acquisto

Sviluppare un software internamente (“make”) è una scelta molto strategica e al tempo stesso rischiosa e costosa (Grafico 2).

Numerosi esempi dimostrano come progetti di sviluppo interno siano falliti, causando ingenti perdite economiche.

Dunque, si dovrebbe scegliere di sviluppare internamente solo nei seguenti casi:

- Quando si punta a un reale vantaggio competitivo:

Se si ritiene che il software possa generare un vantaggio differenziale. Questo, tuttavia, è raro per i software operativi standard, data l’alto livello di personalizzazione che si può ottenere dai software logistici moderni.

- Quando si hanno le competenze e le risorse per mantenerlo:

Le aziende devono essere consapevoli che i cicli tecnologici richiedono riscritture o migrazioni complete ogni 5-8 anni. Inoltre, è difficile attrarre e mantenere programmatori aggiornati nel lungo termine.

- Quando il software richiesto non esiste sul mercato o è strettamente legato al core business.

Situazione rara, poiché esiste un’ampia offerta di soluzioni standardizzate altamente configurabili in qualsiasi settore merceologico.

Perché la software selection conviene (quasi) sempre

In generale, appoggiarsi ad un software esistente nel mercato è tendenzialmente molto meno rischioso di svilupparlo e mantenerlo internamente. Se si segue il processo strutturato della software selection, (spiegato in uno dei prossimi articoli di questa serie) e si sceglie un software molto parametrizzabile e molto vicino alle proprie esigenze funziona-

Grafico 2 Vantaggi e svantaggi della soluzione “Make”

VANTAGGI	SVANTAGGI
Personalizzazione totale	Costi elevati
Maggiore controllo	Tempi di sviluppo lunghi
Sicurezza e proprietà dei dati	Competenze tecniche necessarie
Possibile vantaggio competitivi	Manutenzione continua
Flessibilità nell’integrazione	Rischio di obsolescenza

li, il rischio dell’inserimento di un nuovo software logistico si riduce al minimo.

Dall’altra parte, lo sviluppo interno offre probabilmente più opportunità strategiche e, per alcune aziende con particolari peculiarità o modelli di business, potrebbe risultare una scelta preferibile, se non inevitabile.

Per la stragrande maggioranza delle aziende, dove la logistica rappresenta “solo” uno dei tanti fattori di successo per distinguersi dai concorrenti, risulta molto probabilmente più conveniente scegliere una soluzione già esistente, almeno per quanto riguarda la scelta dei due software logistici più diffusi, ossia WMS e TMS.

Va inoltre notato che il livello di servizio logistico dipende principalmente dalla corretta impostazione dei processi interni e da una loro eventuale automazione e non direttamente dal software stesso. Quest’ultimo, infatti, se ben realizzato, deve essere in grado di eseguire nel modo migliore i processi definiti.

Il ciclo di vita del software logistico e il fenomeno dell’invecchiamento

I termini “ciclo di vita del software” e “ciclo di vita del prodotto” sono concetti ben definiti.

Il primo descrive le diverse fasi della creazione di un software (ad esempio: analisi dei requisiti, progettazione, implementazione, verifica e test, distribuzione e rilascio – cfr. Pressman, Roger S. (2014): Software Engineering: A Practitioner’s

Approach, McGraw-Hill), mentre il secondo si riferisce alle fasi che un prodotto attraversa sul mercato (ad esempio: introduzione, crescita, maturità, declino – cfr. Kotler, Philip; Keller, Kevin Lane (2015): Marketing Management, Pearson Education). Entrambi i concetti, tuttavia, adottano principalmente il punto di vista di chi crea il prodotto. Qui e nei prossimi articoli, invece, il termine “ciclo di vita” verrà utilizzato considerando la prospettiva dell’azienda che sceglie e utilizza il software.

È importante comprendere che un software – pur essendo sottoposto a manutenzioni e aggiornamenti regolari – è comunque soggetto a un inevitabile processo di invecchiamento (software aging).

Questo fenomeno implica che, periodicamente, i software devono essere completamente riscritti o sostituiti con soluzioni più moderne.

Anche se oggi si adottano concetti come l’agile programming e le moderne architetture modulari, che consentono di aggiornare gradualmente i singoli moduli, il principio di fondo non cambia: alla fine, l’intero sistema deve essere ricreato.

Wikipedia descrive le cause di questo invecchiamento così: “nel corso del tempo, le condizioni dell’ambiente software cambiano: sono emerse nuove tecniche, requisiti e formati che non erano completamente conosciuti o prevedibili durante lo sviluppo. Una mancata adattabilità a queste condizioni può causare problemi, errori o semplicemente ridurre l’utilità

del programma, arrivando, nei casi estremi, a renderlo inutilizzabile. D'altra parte, proprio questi necessari adattamenti, correzioni di errori, espansioni delle funzionalità o altre modifiche portano spesso a un programma disorganizzato, codice sorgente difficile da comprendere e a nuovi errori.

Questi sono proprio i segni distintivi dell'invecchiamento del software".

Il fatto che anche i software logistici siano soggetti a invecchiamento è confermato dai dati statistici.

Il Grafico 3, tratto dal WMS Market Report 2024 del Fraunhofer IML, mostra che circa il 70% dei WMS viene utilizzato per un massimo di 14 anni.

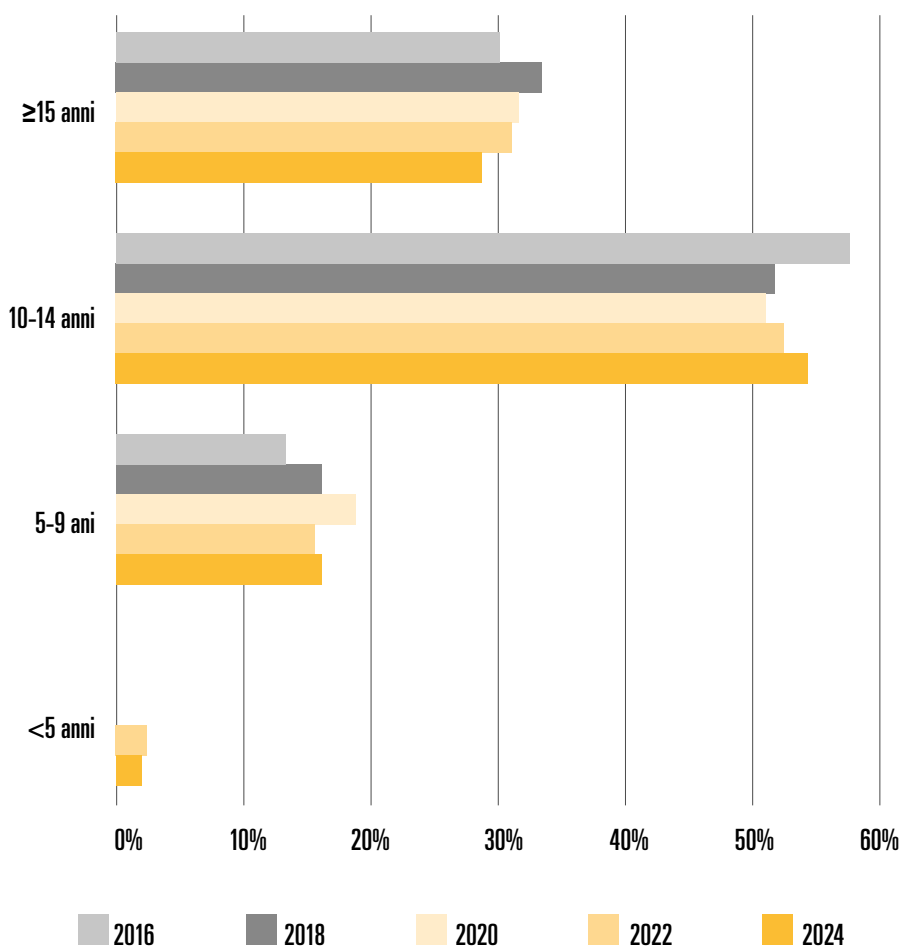
Questi dati provengono dai fornitori di software che partecipano annualmente alla piattaforma warehouse-logistics.com e che, ogni due anni, contribuiscono alla redazione del market report. Ne risulta che, in genere, i sistemi WMS vengono sostituiti dopo un certo numero di anni di utilizzo. Tuttavia, i dati indicano anche che oltre l'80% dei WMS (dati probabilmente rappresentativi anche di altri software logistici) rimane operativo in azienda per un periodo compreso tra 10 e 14 anni, e circa il 30% dei sistemi viene utilizzato addirittura per più di 15 anni. Questo sottolinea l'importanza di prestare particolare attenzione, in fase di selezione, alla stima del costo totale di ciclo di vita e al costo totale di proprietà del software.

Questo è un tema che approfondiremo in uno dei prossimi articoli.

Le fasi operative durante il ciclo di vita del software in azienda

Durante questo tempo, l'utilizzo del software in azienda attraversa diverse fasi, che corrispondono, in sostanza, alle situazioni decisionali nella gestione del software logistico (vedi grafico 1) – in poche parole, le fasi operative che un'azien-

Grafico 3 **Durata media di un WMS in azienda**



da deve gestire relativamente al software dipendono dal momento in cui si trova ogni software nel suo ciclo di vita:

1. Software selection (oppure sviluppo interno): questa fase non finisce con la sola selezione/decisione, ma contiene l'implementazione del software in azienda e l'affiancamento dei key users durante le prime settimane.
2. Miglioramento continuo: il miglioramento continuo rappresenta probabilmente la fase più sottovalutata nell'intero ciclo di vita del software logistico. Essa può essere ulteriormente articolata in diversi momenti:
 - Nei primi mesi dopo l'implementazione, è fondamentale concentrarsi sul perfezionamento della fase di avviamento.

Soprattutto nelle PMI – e in partico-

lare nei casi in cui si passa da una gestione manuale (su carta o con semplici fogli Excel) – si tende spesso a una mera digitalizzazione dei processi esistenti, senza sfruttare appieno le nuove potenzialità offerte dal software né senza intraprendere una vera ottimizzazione.

In questa fase, è importante che, man mano che i dipendenti si familiarizzano con il nuovo sistema, si inizino a valorizzare progressivamente tutte le funzionalità innovative del software adottato.

- Nel corso degli anni, invece, è necessario adeguare costantemente il software ai processi logistici aziendali, che sono in continua evoluzione.

Anche in questo caso, sia le PMI sia

le grandi imprese spesso faticano a mantenere il passo: operate dalle attività quotidiane, non sempre si accorgono che i processi operativi sono cambiati e che tali modifiche dovrebbero essere trasferite anche all'interno del software.

Questa mancata sincronizzazione porta frequentemente alla creazione di workaround manuali e dispendiosi, oltre a generare inefficienze operative. Inoltre, andrebbe effettuata regolarmente una valutazione delle nuove funzionalità sviluppate dalle software house per altri clienti e successivamente integrate nello standard del prodotto: un esercizio utile per capire se potrebbero portare vantaggi anche alla propria realtà e meritare quindi l'adozione.

- Infine, un terzo aspetto riguarda la gestione tecnica che riguarda:
 - il costante aggiornamento del software fornito dalle software house,
 - la formazione continua del nuovo personale,
 - la manutenzione e l'adattamento delle interfacce tra i vari sistemi aziendali (ad esempio, in seguito all'introduzione di nuove tecnologie).

3. Gap analysis: verso la fine della "vita utile" del software all'interno dell'azienda, iniziano a emergere crescenti dubbi sulla sua capacità di adattarsi ai nuovi bisogni. Se la selezione iniziale del software non è stata condotta con sufficiente attenzione – o senza una visione adeguata sul futuro – tali dubbi possono manifestarsi anche dopo pochi anni di utilizzo.

In ogni caso, diventa necessario simulare nuovamente un processo di software selection, costruendo un nuovo capitolato tecnico delle funzionalità richieste oggi e previste per il futuro.

Grafico 4 **Le fasi operative del ciclo di vita di un software logistico in ordine logico**



Questo permette di misurare il "gap" funzionale tra il software esistente e il fabbisogno aziendale attuale.

L'esito di questa analisi porta a una decisione chiave: sostituire il software oppure mantenerlo.

Si tratta di un ragionamento fortemente orientato ai costi, che implica la valutazione di:

- Quanto costa continuare a convivere con la mancanza di alcune funzionalità, sia in termini di costi interni (workaround manuali, inefficienze) sia di costi esterni (perdita di opportunità o clienti);
- Quanto costerebbe integrare o sviluppare le funzionalità mancanti attraverso modifiche da parte della software house;
- Quanto costerebbe invece adottare un nuovo software.

Una leva strategica di lungo periodo

I software logistici, come tutti i software, sono soggetti a un naturale processo di invecchiamento e, nella maggior parte dei casi, restano operativi in azienda per un periodo compreso tra i 10 e i 14 anni. Questo arco temporale può essere considerato come un vero e proprio "ciclo di

vita" dal punto di vista dell'azienda utilizzatrice, articolato nelle fasi di selezione/sviluppo interno, miglioramento continuo e gap analysis. All'inizio e alla fine di questo ciclo, è fondamentale valutare attentamente se sia più conveniente acquistare un software esistente oppure svilupparne uno internamente. Nel caso dei principali software logistici operativi – come WMS (Warehouse Management System) e TMS (Transportation Management System) – risulta nella maggior parte dei casi più vantaggioso optare per l'acquisto. Un progetto strutturato di selezione e implementazione viene definito software selection. Se condotto correttamente e se si sceglie un software moderno, caratterizzato da un'elevata adattabilità, il rischio legato alla scelta e all'implementazione si riduce al minimo inevitabile. Questa decisione è cruciale, perché il software rimarrà in azienda per molti anni e una buona scelta iniziale (best-fit) può aumentare significativamente la durata e l'efficacia della sua permanenza nel tempo.

Il prossimo articolo si intitolerà "Software Selection: cos'è e perché svolgerla in ambito logistico": discuteremo dei motivi di svolgere una selezione strutturata del software logistico. ✕