



UNIVERSITÀ DI PISA

**DIPARTIMENTO DI INGEGNERIA DELL'ENERGIA DEI SISTEMI,  
DEL TERRITORIO E DELLE COSTRUZIONI**

**RELAZIONE PER IL CONSEGUIMENTO DELLA  
LAUREA MAGISTRALE IN INGEGNERIA GESTIONALE**

***Scrum – Agile in Bosch Rexroth  
Come gestire la migrazione di un software gestionale  
parallelamente allo sviluppo di altri applicativi,  
con uno stesso Development Team***

## **SINTESI**

---

RELATORI

Prof. Ing. Riccardo Dulmin

*Dipartimento di Ingegneria dell'Energia,  
dei Sistemi, del Territorio e delle Costruzioni*

Dott.ssa Bozza Angela

*Bosch Rexroth S.p.A.*

IL CANDIDATO

Silvia Manetti

[silviamanetti5@gmail.com](mailto:silviamanetti5@gmail.com)

Sessione di Laurea del 21/07/2021  
Anno Accademico 2020/2021  
Consultazione consentita

# ***Scrum – Agile in Bosch Rexroth: Come gestire la migrazione di un software gestionale parallelamente allo sviluppo di altri applicativi, con uno stesso Development Team***

---

**Silvia Manetti**

## **Sommario**

Il lavoro di tesi è il risultato di un periodo di stage di sei mesi presso l'azienda Bosch Rexroth S.p.A. di Cernusco sul Naviglio (MI), nel quale sono stata inserita nel reparto IT, nello specifico nel gruppo ICO (Innovation, Communication, Organization) che si occupa di sviluppo di applicativi e business intelligence, per portare a termine un progetto di migrazione dei processi aziendali da un applicativo gestionale non più in linea con l'attuale business, ad un applicativo più performante.

Tale obiettivo è stato perseguito soprattutto grazie all'introduzione della metodologia di gestione dello sviluppo Scrum – Agile, e il focus di questo lavoro sarà considerare le peculiarità con cui è stata introdotta, ovvero porre l'evidenza di quanto, pur mantenendo le fondamentali linee guida, sia stato necessario ridefinirla per calarla a pieno nella specifica realtà aziendale.

Il mio ruolo è stato quello di Product Owner di SpeedC, che verrà quindi opportunamente dettagliato.

## **Abstract**

This thesis work is the result of a six-months-long stage in Bosch Rexroth S.p.A. in Cernusco sul Naviglio (MI). During this period, I have been part of the IT department, specifically the ICO team, which is involved in application development and business intelligence. The aim of the project was the management of the migration of business processes from an obsolete software to a more efficient one.

The team adopted different development methods to achieve this goal: the focus of this thesis is the Scrum – Agile methodology and how it has been implemented in ICO.

In fact, although the fundamental guidelines have been maintained, the standard implementation of this method has been changed to be aligned with the specific needs of the organization.

My role was Product Owner of SpeedC, so it will be appropriately detailed.

# 1. Inquadramento generale

## 1.1. Obiettivo di progetto

Nel 2017 Bosch Rexroth S.p.A., in Fig.1 la sede di Cernusco sul Naviglio, si era posta come obiettivo quello di migrare i processi aziendali da OpenWork<sup>1</sup> a SpeeDC<sup>2</sup>, in ottica di miglioramento e allineamento all'attuale business, entro giugno 2020. In OpenWork si trovavano processi essenziali per il monitoraggio e il corretto funzionamento del lavoro quotidiano di svariati reparti, che riguardavano diversi contesti: dai reclami, alle nuove richieste o



Figura 1 - Sede Bosch Rexroth - Cernusco sul Naviglio (MI)

modifiche delle anagrafiche relative ai clienti e ai codici materiali, dalle variazioni di listino prezzo, alla gestione delle offerte e ordini, alle richieste di intervento.

A fine 2018 mancavano ancora quasi venti, dei trentatré processi da migrare; la situazione di evidente difficoltà ha spinto i responsabili del team a stravolgere lo status quo e ricercare tra le metodologie di progettazione quella che permettesse flessibilità, e al tempo stesso mettesse ordine nel processo di sviluppo: da qua l'introduzione di Scrum – Agile. Quando ad inizio del secondo quarter del 2019, sono entrata a far parte del team ICO, Scrum – Agile era stato introdotto da circa un mese.

## 1.2. I punti chiave analizzati

Il raggiungimento dell'obiettivo aziendale ha in particolar modo richiesto il focus su alcuni punti chiave:

- L'adattamento della metodologia di sviluppo Scrum - Agile, considerando un unico Development Team al quale sono assegnati parallelamente più progetti, con le conseguenziali criticità non comprese nella teorizzazione classica del metodo, che

---

<sup>1</sup> Applicativo per la gestione dei processi aziendali

<sup>2</sup> Applicativo per la gestione dei processi aziendali

prevede per ogni progetto uno specifico team di sviluppo. Il team ICO è infatti costantemente coinvolto in altri progetti, che non verranno trattati nello specifico.

- L'adozione degli strumenti di supporto più idonei alla stesura e alla gestione delle specifiche di sviluppo dei processi aziendali da migrare: Jira.
- La definizione di determinati ruoli all'interno del reparto per gestire in modo efficace ed efficiente il progetto di migrazione. Nel gergo proprio della metodologia Scrum – Agile i ruoli propriamente definiti sono: Development team, Scrum Master e Product Owner.

Nell'elaborato verrà quindi esposto come il team ICO sia riuscito ad implementare con successo una metodologia che di base non prevede di essere adottata per un certo tipo di team, e verranno riportate le attività operative più rappresentative del lavorare in Scrum – Agile.

Per quanto riguarda i ruoli, verrà posta l'attenzione sul mio ruolo di Product Owner di SpeedC e sulle attività che ne conseguono.

## **2. Scrum – Agile in Bosch Rexroth**

### **2.1. Il passaggio a Scrum - Agile**

Quando sono entrata a far parte dell'azienda, il team aveva da pochissimo introdotto la nuova gestione degli sviluppi secondo la metodologia Scrum - Agile, dopo che per circa un paio d'anni, l'implementazione dei processi in SpeedC era stata gestita seguendo la metodologia Waterfall. Con la gestione Waterfall, il team non aveva definito dei precisi ruoli ed anche l'analisi dei processi con i vari utenti era svolta direttamente dagli sviluppatori; inoltre, lo sviluppo del processo non veniva avviato finché tutte le attività non fossero perfettamente definite in tutti i dettagli, e successivamente, quando il tutto era stato implementato iniziava la fase di test con gli utenti. Volendo rappresentare visivamente la logica di sviluppo delle due metodologie a confronto, come visibile in Fig.2, potremmo pensare alla metodologia Waterfall come ad una lunga linea retta che prosegue sempre in un'unica direzione, in cui al finire di una fase inizia la successiva, senza prevedere formalmente la possibilità di tornare indietro, e ad Agile come a tanti blocchi consecutivi, all'interno dei quali le specifiche fasi sono percorse in maniera circolare, ciò significa che possono essere eseguite più volte finché non si è convinti di passare al blocco di azioni successivo. L'implementazione di un processo in

Agile procede a piccoli steps, e ogni nuovo sviluppo viene presentato e testato con gli utenti, per avere puntuali feedback.

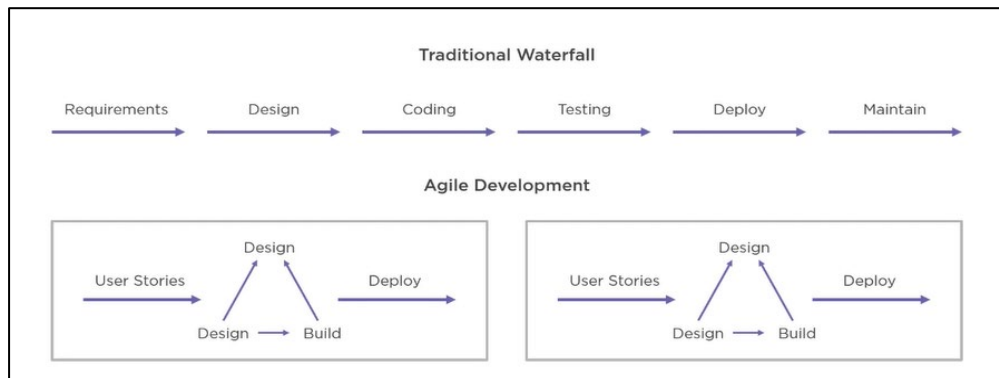


Figura 2 - Waterfall vs Agile

In un ambiente estremamente dinamico come quello della programmazione, Scrum- Agile permette di gestire il cambiamento come un elemento fisiologico alla buona riuscita del prodotto finale. L'implementazione della metodologia è molto strutturata, vengono infatti previste una serie di cerimonie ed artefatti da rispettare all'interno dello Sprint, che identifica un determinato slot di tempo di sviluppo, la cui lunghezza è decisa dal team in base a quello che ritiene più opportuno, nel nostro caso è stato assestato a due settimane. Un riassunto della struttura delle cerimonie presenti in ogni Sprint è riportato in Fig.3.

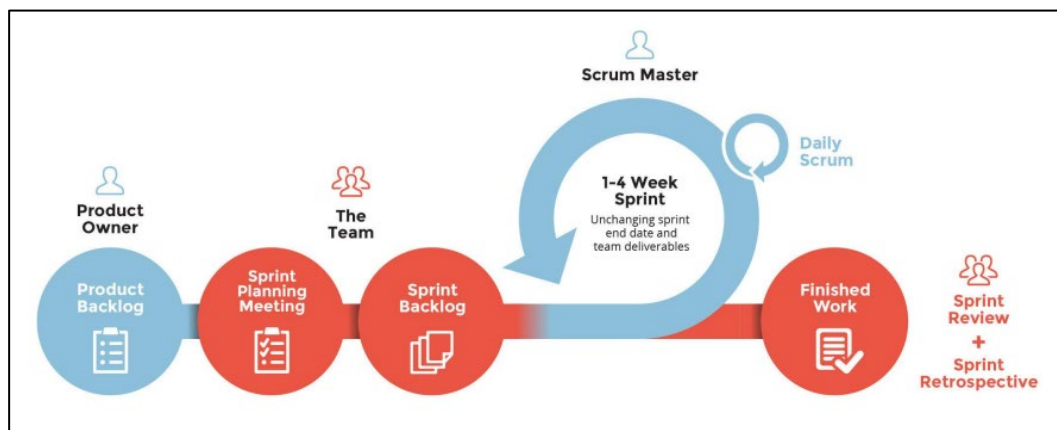


Figura 3 - Overview della metodologia Scrum

## 2.2. L'adattamento della metodologia al team ICO

La necessità di adattamento della metodologia è derivata da un assetto del team ed una condizione di necessità aziendale di sviluppi, diversa da quella teoricamente prevista.

- Implementazione classica di Agile: un team formato da un Product Owner e un Development Team con diverse aree di competenze informatiche, focalizzati su un unico progetto da portare a termine nel corso degli Sprint.

- Assetto peculiare di ICO: un unico Development Team con diverse competenze informatiche e due Product Owner coinvolti in media in cinque progetti per Sprint.

Lo specifico contesto aziendale che vedeva quindi più progetti da dover implementare in una condizione di scarsità di risorse, ha portato a dover affrontare due principali aspetti critici:

1. Decidere quanta capacità di sviluppo allocare su ogni singolo progetto
2. Calibrare il peso dei diversi sviluppi da implementare su ogni singolo membro del Development Team

Essere in due Product Owner, suddivisi nella gestione dei diversi progetti, ha significato innanzitutto avere una continua condivisione degli obiettivi per decidere su quali progetti dover allocare più o meno risorse.

Avere un unico Development Team su più progetti ha portato invece non solo a dover stabilire con precisione le diverse priorità, ma anche a capire se la totalità degli sviluppi per Sprint pesassero in maniera eccessiva su un unico sviluppatore.

La prima cerimonia di Agile ad essere adattata è stata il Backlog Refinement, cerimonia durante la quale noi Product Owner confrontiamo i nostri backlog di specifiche, per decidere quali portare nel prossimo Sprint perché prioritarie. Operativamente non solo si doveva equilibrare il quantitativo di sviluppi da portare in ciascun progetto, ma valutarne anche il complessivo mix per evitare che coinvolgessero quasi esclusivamente l'area di competenza di uno sviluppatore, rischiando di avere un membro troppo carico ed altri scarichi, in termini di specifiche assegnate. Al termine della cerimonia si arriva quindi a stabilire un quantitativo di punti massimo per ciascun progetto.

Le specifiche ritenute prioritarie vengono presentate al Development Team durante lo Sprint Planning: il Product Owner espone ogni richiesta che viene votata dal team, ovvero le viene attribuito un punteggio che ne rappresenta la generale complessità considerando diverse variabili di programmazione, e viene quindi portata dentro lo Sprint, fino a raggiungere il quantitativo di punti massimo stabilito nel Backlog Refinement. In ICO lo Sprint Planning deve prevedere la discussione di un progetto di seguito all'altro.

Per ogni Sprint vengono quindi portati avanti una serie di sviluppi riguardanti diversi progetti, che sono poi presentati ai vari utenti di riferimento durante le Reviews, in modo tale da avere

un feedback immediato e per far sì che ogni nuova implementazione futura si basi su sviluppi già avvalorati.

### **3. Lavoro svolto nel progetto**

#### **3.1. Il mio ruolo di Product Owner di SpeedC**

Il mio contributo al raggiungimento dell'obiettivo di progetto come Product Owner di SpeedC si è concentrato su:

1. Analisi e miglioramento dei processi in Openwork
2. Organizzazione delle specifiche da presentare al Development Team per l'implementazione dei nuovi sviluppi
3. Attività di Tester sugli sviluppi rilasciati in SpeedC
4. Definizione e coordinamento delle attività di preparazione al Go-Live di ogni nuovo processo con relative criticità
5. Supporto agli utenti post Go-Live

Verranno operativamente riportate le attività necessarie all'implementazione dei primi tre punti, ritenuti quelli più significativi.

1. Analisi e miglioramento dei processi in Openwork

Per ciascun nuovo processo, le due componenti a cui dover far riferimento, strettamente collegate tra di loro erano: modulo e diagramma di flusso. L'analisi inizialmente si concentrava sul rappresentare il flusso di processo AS-IS; per la stesura delle specifiche relative al flusso si è utilizzato un modello standard BPMN (Business Process Management and Notation) implementato attraverso il software Visio Studio che mi ha permesso di rappresentare graficamente le diverse attività, le connessioni tra di esse, gli eventuali eventi di interruzione o fine del flusso, e grazie all'utilizzo delle "swim lane" le specifiche assegnazioni per ogni attività.

Definito quindi il flusso con gli steps fondamentali per processare correttamente le diverse future richieste, andavo a definire le informazioni necessarie in ciascuna attività, ovvero i campi del modulo. Per tracciare i campi del modulo previsti all'avvio della richiesta in OpenWork e quelli nelle varie attività, inizialmente utilizzavo una tabella Excel che permetteva di raggruppare i campi in diversi blocchi, perché così si sarebbero poi presentati

nella pagina di SpeedDC, di indicare la tipologia di campo (testo, data, ecc.) e per i campi picklist tutti i possibili valori che gli utenti avrebbero potuto selezionare. Per ogni campo venivano inoltre indicate le eventuali condizioni in base alle quali un campo poteva essere obbligatorio o meno, nascosto o essere precompilato con diversi testi. Ci siamo però resi conto della difficoltà per gli utenti di capire in maniera chiara come il modulo di una futura richiesta si sarebbe presentato in SpeedDC a partire dal foglio Excel, e sono quindi passata all'utilizzo del programma Balsamiq Mockup (Fig.2), un software grafico che mi ha permesso di rappresentare tutte le informazioni mappate nell'Excel, ma in modo simile a come una richiesta sarebbe apparsa in SpeedDC, avendo quindi specifiche più efficaci.

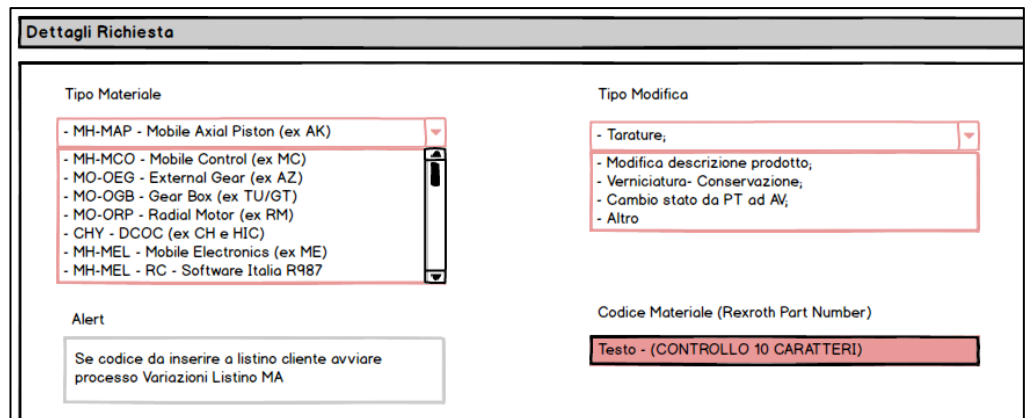


Figura 2 - Esempio rappresentazione campi in Balsamiq Mockup

Una volta presentata agli utenti la versione AS IS del flusso e del modulo, si discutevano e definivano le necessarie modifiche da implementare per ottimizzare al meglio le future richieste dello specifico processo.

## 2. Organizzazione delle specifiche da presentare al Development Team per l'implementazione dei nuovi sviluppi

L'introduzione della nuova metodologia ha portato anche a nuovi tool di supporto alla gestione, ed in particolare all'adozione di Jira: software che ha permesso di pianificare e mappare le diverse specifiche, oltre a monitorare i vari sviluppi ed avere quindi una overview sull'avanzamento dei vari progetti. Progetti che in Jira vengono diversificati in varie boards. Quando il flusso di processo e il modulo di SpeedDC con le varie informazioni erano stati definiti con gli utenti di riferimento, le diverse specifiche dovevano essere scritte nel backlog di Jira sottoforma di 'storie': la struttura che deve avere ogni storia è bene definita e visibile in Figura 3.



Local application development at DC Italy / DCETICOIT-10669

### As a CLP Manager I need to modify the flow of the process, so that the analysis of the claims will be faster

[Edit](#) [Comment](#) [Assign](#) [Log work](#) [Agile Board](#) [More](#) [Set Progress on hold](#) [Set Due Date](#) [Start Progress](#) [Close Issue](#)

**Details**  
 Type: **User Story** Status: **OPEN**  
 Priority: **Critical** Resolution: **Unresolved**  
 Affects Version/s: **None** Fix Version/s: **None**  
 Component/s: **Scrum SpeedC**  
 Labels: **1 M**

**Field Tab** **User Info**

User Department: **CLP**  
 Epic Link: **Logistic and commercial claims**  
 Sprint: **Doctor Strange**

**Description**

- In **Accreditati e resi per reclami gestionali** l'attività di **Verifica con cliente** e quella di **Inserimento numero SpeedC** saranno assegnate al richiedente
- All'avvio del modulo **Reclami gestionali** nella tabella **"Indicazioni per materiale da rendere"** la colonna **"Tecnologia"** deve essere uguale per tutte le eventuali righe: ovvero se nella prima riga viene ad esempio inserito **"Tecnologia" = AT - Assembly Technology**, nelle eventuali successive righe la tecnologia deve essere autocompilata con quella.
- In base a tale tabella modificare il flusso di processo come indicato nel Visio allegato

**Attachments**

Drop files to attach, or browse.

Modifiche flusso Reclami ge:  
 2021-01-18 11:00 284 kB

Figura 3 - Struttura di una storia

Il campo che riporta la dicitura 'Epic link', indica il singolo processo che fa parte della board generale di SpeedC. Nel caso riportato in figura si tratta ad esempio di una modifica per il processo di Reclami commerciali. Quando si iniziava un nuovo processo, in generale ogni storia rappresentava una singola attività da implementare.

Le storie venivano presentate al Development Team durante lo Sprint Planning.

### 3. Attività di Tester sugli sviluppi rilasciati in SpeedC:

Giornalmente in Agile è prevista la cerimonia del Daily Stand up, cioè un breve incontro che si svolge appunto in piedi ogni mattina, durante il quale ogni sviluppatore si assegna una o più tasks di una particolare storia, tasks che rappresentano le implementazioni operative necessarie al completamento della specifica richiesta dal Product Owner. Inoltre, lo sviluppatore indica le tasks su cui lavorerà quel giorno e che assumeranno lo stato di *In Progress* e quelle completate da portare in *Verify*. Quando una o più tasks venivano mosse in *Verify* iniziava la mia attività di Tester: nell'ambiente di sviluppo di SpeedC andavo a testare

le implementazioni effettuate per verificare che corrispondessero a quanto richiesto o fossero necessarie alcune modifiche. Una volta testato la correttezza degli sviluppi, passavo la task in *Done*: solo quando tutte le tasks di una storia sono chiuse, quella specifica è ritenuta completata e si può procedere a presentarla agli utenti a fine Sprint.

Il flusso di Fig. 4 schematizza graficamente i passaggi necessari per arrivare al completamento di un processo da OpenWork a SpeedDC e di come sia stato necessario lavorare con una metodologia che prevedesse di poter ripetere alcune fasi essenziali più e più volte, per giungere al risultato desiderato dagli utenti.

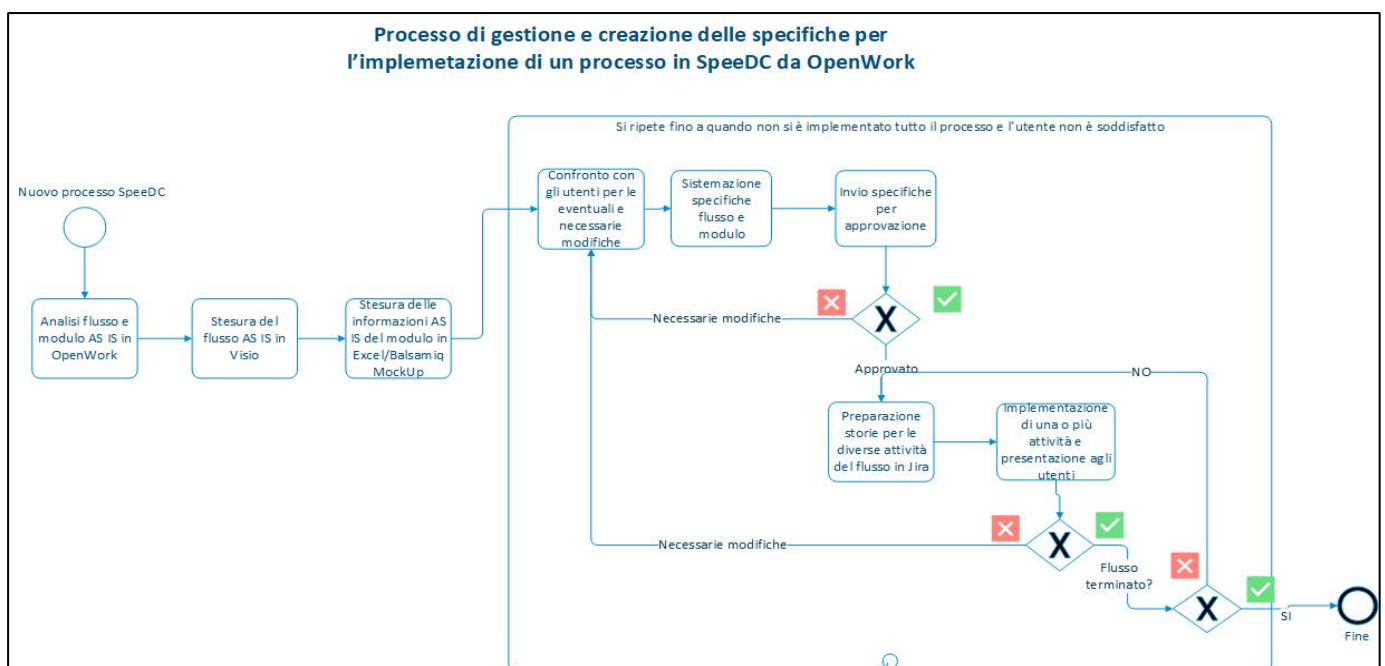


Figura 4 - Schema di gestione dell'implementazione di un processo in SpeedDC

Per ognuno dei processi migrati ciascuna fase ha avuto ovviamente una diversa complessità, in base a variabili quali la lunghezza del processo, le modifiche da apportare per allinearlo all'attuale business, gli automatismi da implementare e le modifiche richieste in corso d'opera.

## 4. Conclusioni

### 4.1. Il raggiungimento dell'obiettivo

L'obiettivo di migrare trentuno processi in circa due anni e mezzo, da OpenWork a SpeedDC è stato raggiunto. Il raggiungimento dell'obiettivo è dipeso fortemente dalla nuova metodologia di lavoro che è stata introdotta in ICO, infatti prima di lavorare in Scrum – Agile, ovvero fino al primo trimestre del 2019 erano stati migrati quindici processi, durante il 2019 e fino a maggio 2020 sono stati conclusi tutti i restanti. Durante tutto il mese di giugno 2020 sono stati inoltre implementati diversi reports per ogni modulo migrato per permettere agli utenti di consultare i dati delle vecchie richieste.

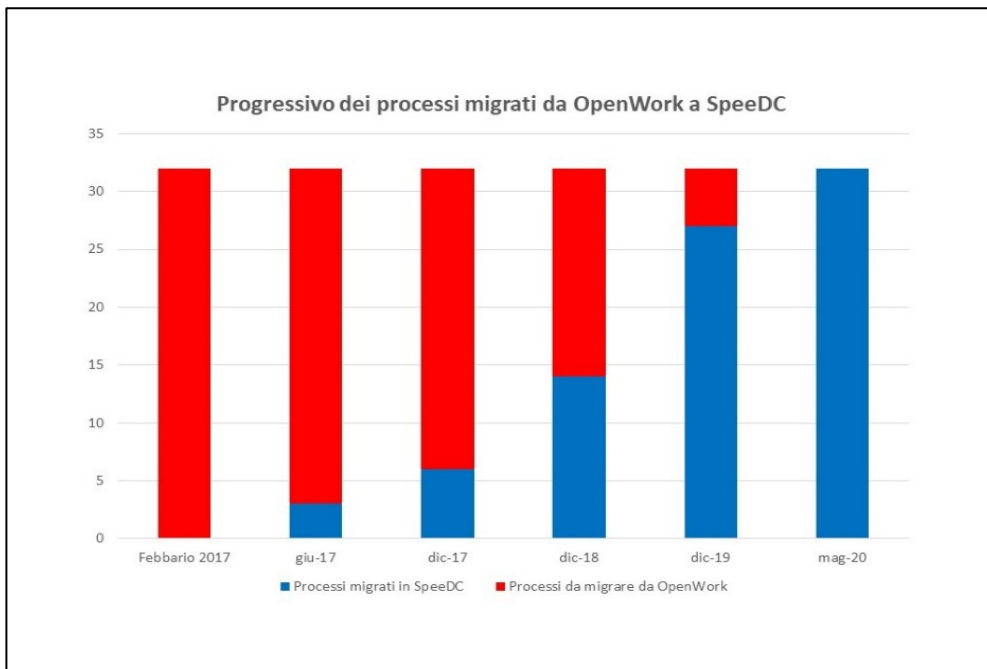


Figura 5 - Progressivo dei processi migrati

Il grafico di Fig.5 riassume il progressivo aumento, dall'introduzione di Scrum – Agile, dei processi sviluppati e migrati nel nuovo applicativo.

Dal punto di vista gestionale, aver avuto a che fare con svariate tipologie di processi, da quelli ritenuti standard per un'azienda commerciale ma pur sempre calati nella specifica realtà aziendale, a quelli peculiari del mercato e dei prodotti di Bosch Rexroth, mi ha permesso di avere a che fare con un'amplia gamma di utenti e capire meglio svariate dinamiche aziendali. Considerando la ristrettezza del team di sviluppo di ICO che porta a capacità di programmazione limitata, il quantitativo di processi da migrare e la stringente deadline per raggiungere l'obiettivo, è stato più che essenziale avere una metodologia di sviluppo che permettesse di identificare responsabilità e compiti in modo chiaro.

## **4.2. Considerazioni su Scrum-Agile in ICO**

Lavorare in Scrum - Agile ha permesso di avere una gestione del lavoro su vari progetti molto più organizzata, ed ha portato ad un radicale cambio di mentalità. Tale cambiamento non ha coinvolto solo il team ICO, ma anche tutti i reparti aziendali coinvolti nei diversi progetti di sviluppo. Soprattutto inizialmente, non è stato facile far comprendere agli utenti la nuova metodologia di gestione basata su priorità ben definite e la gestione molto rigida delle tempistiche, per la quale una specifica che non rientra in pianificazione dovrà essere rimandata obbligatoriamente di almeno uno Sprint, cioè due settimane. Questa organizzazione del lavoro è stata però essenziale per diffondere la consapevolezza all'interno di Bosch Rexroth, che a fronte del quantitativo di progetti contrapposto alla presenza di un unico Development Team, non fosse più plausibile una logica non basata su chiare e definite priorità, legittimate dai Product Owner per conto della Direzione.

Lavorare su svariati progetti contemporaneamente e con cadenza bisettimanale dover dare dei puntuali riscontri agli utenti, a livello di team è stimolante ma sicuramente anche psicologicamente stressante, per questo è importante avere un processo che sproni al miglioramento continuo anche di piccoli aspetti solo all'apparenza secondari, ma che permettono poi di poter gestire il tutto con maggiore serenità. Il momento di Retrospective che chiude ogni Sprint e nel quale il team si confronta sugli aspetti positivi e sui punti di miglioramento riscontrati nello slot di sviluppo appena concluso, diventa quindi fondamentale e in ICO è stato importante avere un senso di gruppo che abbia permesso di viverlo apertamente, facendo emergere importanti punti di miglioramento, poi implementati. Agile è quindi una metodologia che non può prescindere dal lavoro di squadra.

In conclusione, adottare la metodologia Scrum- Agile per gestire lo sviluppo informatico ha permesso senz'altro di avere una più precisa pianificazione dei progetti in corso e di affrontare in maniera rapida ed efficiente i cambiamenti, ed il successo della sua implementazione è dovuto sia al fatto di aver rispettato gli elementi fondamentali del metodo seppur adattandoli alle specifiche necessità, sia ad un team con la mentalità proattiva, che è riuscito a gestire lo stress di continue e parallele nuove realizzazioni grazie anche ad un forte spirito di gruppo.