

POLITECNICO DI TORINO

Corso di Laurea Magistrale in Ingegneria Gestionale

Tesi di Laurea Magistrale

Lo sviluppo di un Digital Shadow all'interno di un'architettura di Digital Twin di magazzino



Relatore

Carlo Rafele

Correlatori

Giovanni Zenezini
Andrea Ferrari

Candidato

Raul Pintus

Dicembre 2021

Tabella dei contenuti

Introduzione.....	7
1. Il Digital Twin	9
1.1 Definizione	9
1.2 Step di sviluppo	9
1.3 Internet of Things	12
1.4 Intelligenza Artificiale.....	18
2. Struttura e architettura	22
2.1 Struttura	22
2.2 Architettura.....	24
3. Applicazioni	29
3.1 Manufacturing	29
3.2 Warehousing I	32
3.3 Warehousing II	35
4. Caso studio	37
4.1 Spazio fisico	37
4.2 Infrastruttura IT	41
4.3 Modello di simulazione	52
4.4 Modello UML della Digital Shadow	55
4.5 Modello UML Digital Twin	57
5. Esempio applicativo della Digital Shadow.....	59
5.1 Modello di simulazione di prova.....	59
5.2 Trasmissione delle informazioni	66
Conclusioni.....	75
Bibliografia.....	77
Sitografia	80

Tabella delle figure

Figura 1.1 – Simulazione CAD/CAE (Julien et al., 2012)	10
Figura 1.2 - Architettura Internet of Things (Adattato da: Al-Fuqaha et al., 2015).....	12
Figura 1.3 - Funzionamento Internet of Things (Revel G.M. e Naspi F., 2020).....	13
Figura 1.4 - Metodi di apprendimento del Machine Learning (Dey A., 2016)	19
Figura 1.5 - Rete neurale artificiale (Laptrinhx, 2019)	20
Figura 2.1 - Struttura del Digital Twin (Talkhestani A. et al., 2019).....	22
Figura 2.2 - Architettura Digital Twin (F. Tao and M. Zhang, 2017).....	24
Figura 2.3 - Shop-Floor Service System (F. Tao and M. Zhang, 2017a).....	26
Figura 2.4 - Calibrazione dei modelli (F. Tao and M. Zhang, 2017a)	26
Figura 2.5 - Processo di fusione dei dati (F. Tao and M. Zhang, 2017a).....	27
Figura 3.1 - Modello virtuale linea produzione (Zheng et al.,2019)	29
Figura 3.2 - Processo di acquisizione dei dati (Zheng et al.,2019)	30
Figura 3.3 - Processo di mappatura dei dati (Zheng et al.,2019)	31
Figura 3.4 - Modello virtuale del magazzino automatico (Leng J. et al., 2019)	32
Figura 3.5 - Architettura dell'applicazione warehousing (Leng J. et al., 2019).....	33
Figura 3.6 - Sincronizzazione Cyber-Fisica dell'applicazione warehousing (Leng J. et al., 2019).....	34
Figura 3.7 - Magazzino reale e modello di simulazione (Braglia et al.,2019).....	35
Figura 4.1 – Area automatizzata.....	37
Figura 4.2 – Trasloelevatore e scaffalature	38
Figura 4.3 – A sinistra stazione di kitting e a destra stazione di picking	39
Figura 4.4 – MiR	40
Figura 4.5 – Stazione di ricarica dei MiR.....	40
Figura 4.6 - Software gestionali	41
Figura 4.7 - Funzionalità WMS (Adattato da Mecalux, 2019b)	42
Figura 4.8 - Flusso informatico dalla ricezione alla fatturazione di un ordine.....	45
Figura 4.9 - Ruolo WCS (Adattato da Zdintegration).....	46
Figura 4.10 - Struttura logica INCAS	47
Figura 4.11 - Ambienti	49
Figura 4.12 – Processi del modello di simulazione del magazzino.....	54
Figura 4.13 – Layout del modello di simulazione in 3D.....	55
Figura 4.14 – Diagramma delle classi della Digital Shadow	56
Figura 4.15 – Diagramma delle classi del Digital Twin.....	58
Figura 5.1 - Verifica del caricamento dei dati durante la simulazione.....	59
Figura 5.2 - Agente Articolo	61
Figura 5.3 - Funzione toString dell'agente Articolo	62
Figura 5.4 - Costruttore dell'agente Articolo	62
Figura 5.5 - Agente Documento	63
Figura 5.6 - Agente Ordine.....	63
Figura 5.7 - Agente Cassetta	64
Figura 5.8 - Agente Main	64
Figura 5.9 - Blocco Source dell'agente Main.....	65
Figura 5.10 - Diagramma classi	67

Figura 5.11 - Primo passaggio creazione utente.....	70
Figura 5.12 - Secondo passaggio creazione utente.....	70
Figura 5.13 - Terzo passaggio creazione utente	71
Figura 5.14 - Primo passaggio accesso database da Anylogic	72
Figura 5.15 - Secondo passaggio accesso database da Anylogic	73
Figura 5.16 - Configurazione creazione degli agenti	74

Tabella delle tabelle

Tabella 4.1 - Hardware del server che ospita il database	49
Tabella 4.2 - Parametri di configurazione del database.....	50
Tabella 4.3 - Hardware del server che ospita gli applicativi	50
Tabella 4.4 - Prestazioni di rete cablata minime del sistema.....	51
Tabella 4.5 - Prestazioni di rete wifi minime del sistema.....	52
Tabella 5.1 - Elementi della Process Modeling Library utilizzati nel caso studio	60

Introduzione

Nel contesto tecnologico attuale, caratterizzato dalle innovazioni dell'Industry 4.0, le aziende puntano sempre più su nuove tecnologie per accrescere la propria competitività e per migliorare l'efficienza dei propri impianti produttivi. Uno dei principali modelli di business dell'Industry 4.0 è il Digital Twin. Quest'ultimo è una rappresentazione virtuale di un oggetto o di un sistema, collegato ad esso per tutto il ciclo di vita. Il Digital Twin, grazie a una costante comunicazione bidirezionale, è sincronizzato con il suo gemello fisico e cambia con esso. Questa tecnologia innovativa si basa sulla raccolta dei dati, l'analisi dei dati acquisiti e l'adozione di azioni specifiche con l'obiettivo di migliorare le prestazioni dell'impianto industriale in cui viene implementata. L'obiettivo di questa tesi è riuscire a creare un'architettura Digital Shadow di un magazzino automatico installato all'interno del Politecnico di Torino, ponendo le basi per una futura implementazione di un sistema Digital Twin. Per far ciò è stato creato un modello di simulazione in grado di prelevare da un database remoto tutti i dati necessari per il suo funzionamento. Il modello è stato sviluppato con il software di simulazione Anylogic e svolge delle semplici operazioni atte a dimostrare la corretta interazione col database. Quest'ultimo è stato creato da zero con la stessa struttura che presenta il database operativo ed è stato popolato con delle informazioni coerenti a quelle che vengono generate durante le operazioni di magazzino.

Lo studio è stato sviluppato in due fasi. Nella prima è stata svolta una ricerca in letteratura riguardo il concetto di Digital Twin. Per aiutare a comprendere meglio le potenzialità di questa nuova tecnologia sono stati approfonditi i principi di funzionamento dell'Internet of Things e dell'Intelligenza Artificiale. È stato approfondito anche l'aspetto legato all'architettura in cui va inserito il Digital Twin e alle modalità con cui avviene la sincronizzazione con il gemello fisico. In ultimo luogo sono state presentate tre applicazioni reali di questa tecnologia per dare evidenza di come sia uno strumento estremamente efficace e in grado di ottimizzare le performance del sistema in cui è inserito. Nella seconda fase è avvenuta la trattazione del caso studio. Inizialmente sono stati descritti tutti gli aspetti legati agli elementi che costituiscono il magazzino, quelli legati all'infrastruttura informatica che permette loro di comunicare ed è stata fornita una spiegazione del funzionamento del modello di simulazione attualmente funzionante. Successivamente è stato presentato un diagramma delle classi che descrive la struttura del sistema informatico necessario per implementare il Digital Twin. Questa fase si conclude con l'analisi dettagliata di tutti i passaggi che sono stati affrontati per implementare una Digital Shadow di prova.

La tesi è composta da cinque capitoli divisi nel modo seguente. Nel primo viene fornita una definizione di Digital Twin; vengono discussi gli step di sviluppo che portano alla sua creazione partendo dal Digital Model, passando per il Digital Shadow e infine giungendo al Digital Twin; sono state approfondite le tecnologie abilitanti Internet of Things e Intelligenza Artificiale, grazie alle quali il Digital Twin è in grado di ottenere la mole di dati di cui necessita e di elaborarle per riuscire a prendere le decisioni. Il secondo capitolo presenta una descrizione approfondita delle componenti che costituiscono il Digital Twin definendone la struttura; viene inoltre approfondita l'architettura in cui è inserito e che ne permette il

funzionamento. Nel terzo capitolo vengono analizzate tre applicazioni reali di questa tecnologia: una in ambito manufacturing e due in ambito warehousing. Nel quarto capitolo inizia la trattazione del caso studio. Viene descritta la composizione del magazzino fisico; viene presentata la composizione sia hardware che software dell'infrastruttura IT in cui è inserito; infine viene presentato il diagramma delle classi di un sistema Digital Twin. Nel quinto ed ultimo capitolo viene spiegato il funzionamento e l'implementazione del modello di simulazione di prova e viene descritto il procedimento di creazione del database e di configurazione del server che ospita il database.

Grazie a questa ricerca è stato possibile capire cos'è il Digital Twin, dove si colloca nell'attuale contesto tecnologico e quali sono i vantaggi che si avrebbero nella sua implementazione in ambito produttivo e logistico. Attraverso l'analisi di alcune applicazioni reali di questa tecnologia, questo lavoro ha permesso anche di comprendere quali sono e come interagiscono tra loro tutti gli elementi che costituiscono un'architettura Digital Twin. Questa base teorica è stata importante per riuscire a trattare il caso studio e implementare un sistema Digital Shadow. Quest'ultimo veste un ruolo fondamentale come supporto per le future attività di ricerca perché possiede tutte le caratteristiche proprie di una Digital Shadow. Inoltre, grazie alla descrizione dettagliata delle modalità di configurazione del database e del modello di simulazione, sarà possibile implementare la Digital Shadow anche in sistemi più complessi.

1. Il Digital Twin

1.1 Definizione

Nel contesto tecnologico attuale la capacità di archiviare e elaborare grosse quantità di dati per riuscire a estrapolare conoscenza o migliorare le prestazioni dei sistemi fisici ha una rilevanza sempre maggiore. Questa necessità dà lo slancio allo sviluppo di tecnologie in cui il dato costituisce una base essenziale: l'Internet of Things e l'Intelligenza Artificiale.

L'Internet of Things che, attraverso l'implementazione di sistemi in grado di rendere un qualunque oggetto capace di connettersi a Internet, consente uno scambio continuo di informazioni generando quindi una grande quantità di dati; l'Intelligenza Artificiale che, sfruttando algoritmi di Machine Learning, rende capace una macchina di svolgere tutte quelle operazioni caratteristiche dell'intelletto umano come la pianificazione, la comprensione del linguaggio, il riconoscimento di oggetti e suoni, l'apprendimento e la risoluzione dei problemi.

L'integrazione delle tecnologie menzionate motiva uno dei progressi più recenti e probabilmente uno dei più importanti nel campo della tecnologia, cioè il Digital Twin. Ad oggi, la definizione più comunemente usata di Digital Twin è stata proposta da Glaesegen e Stargel nel 2012: “digital twin means an integrated multiphysics, multiscale, probabilistic simulation of a complex product, which functions to mirror the life of its corresponding twin”.

Il Digital Twin è quindi più di una semplice simulazione, è un modello vivente, intelligente e in evoluzione che segue il ciclo di vita del suo gemello fisico per monitorare, controllare e ottimizzare i suoi processi e funzioni. Il Digital Twin è sincronizzato con il suo gemello fisico e cambia con esso, e il cambiamento riflette ed è governato dalle proprietà dell'oggetto fisico che viene simulato. Durante questa continua interazione i dati descrittivi vengono scambiati e aggiornati grazie alle funzionalità di caricamento e archiviazione dei dati. Inoltre è in grado di prevedere i comportamenti futuri di un sistema fisico e, nel caso in cui venissero predetti dei problemi, può trovare e applicare soluzioni preventive. Tuttavia, sebbene i gemelli digitali siano sistemi intelligenti, è importante capire che non sono sempre completamente autonomi. Infatti le applicazioni basate sull'intelligenza artificiale e i Digital twin richiedono ancora molto intervento umano, in particolare negli scenari in cui vengono utilizzati per testare nuove funzionalità e modifiche di risorse fisiche (B. R. Barricelli et al. 2019).

1.2 Step di sviluppo

In questo capitolo si prendono in esame tutte le fasi del ciclo di vita di un Digital Twin. Nel fare questo verranno analizzate le varie rappresentazioni virtuali e il livello di integrazione con il rispettivo gemello fisico.

1.2.1 Digital Model

Il primo livello è il Digital Model, un modello virtuale del sistema immaginato che può essere creato prima della sua costruzione fisica con lo scopo principale di mitigare i rischi tecnici e supportare il processo decisionale nella progettazione del concetto oppure può essere creato sulla base di un sistema già esistente con l'obiettivo di ottimizzarlo valutando virtualmente diversi scenari di design (Madni A.M. et al., 2019). Per la creazione del Digital Model ci si avvale dei tradizionali sistemi di simulazione che vengono eseguiti utilizzando applicazioni CAD/CAE, software che abbinano le funzionalità di un software CAD a quelle di uno CAE. Un software CAD si occupa di creare, modificare e ottimizzare un progetto con lo scopo di realizzare modelli 3D accurati. Ci fornisce un'anteprima digitale del prodotto finale e dei suoi componenti e migliora la qualità del design aggiungendo una maggiore precisione e quindi riducendo gli errori. Un software CAE invece è utile per svolgere analisi ingegneristiche sul prodotto CAD creato attraverso simulazioni virtuali atte a testare i vari scenari che il prodotto sperimenterà sul campo dando così l'opportunità di perfezionare e ottimizzare lo sviluppo del prodotto, fornire dati sulle prestazioni e consentire una risoluzione anticipata di eventuali problemi (Tracy Woo, 2016).

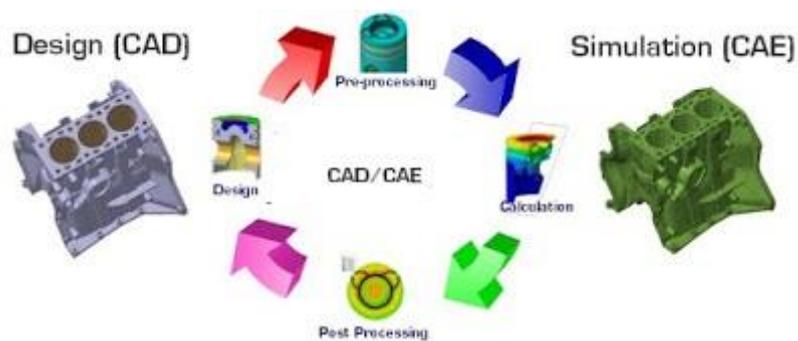


Figura 1.1 – Simulazione CAD/CAE (Julien et al., 2012)

In conclusione è importante sottolineare l'enorme distanza che intercorre tra questo stadio e quello del definitivo Digital Twin. In un Digital Model è completamente assente l'integrazione con la controparte fisica poiché di quest'ultima esiste solamente un prototipo virtuale.

A causa della mancanza del collegamento con il sistema fisico ovviamente non ci può neanche essere alcuno scambio automatico di dati e informazioni. Quindi è un sistema statico in cui è necessario l'inserimento manuale dei dati di input per effettuare le simulazioni dei vari scenari.

1.2.2 Digital Shadow

La Digital Shadow è una rappresentazione virtuale estremamente fedele di un elemento fisico capace di ricevere e incorporare dati. È inserita in un sistema informatico capace di generare tali informazioni nello spazio fisico e di inviarle automaticamente e in tempo reale allo spazio

virtuale rendendo la Digital Shadow perfettamente sincronizzata con il proprio gemello fisico. Questo strumento viene utilizzato per monitorare in tempo reale lo stato di salute o il livello delle prestazioni della controparte fisica permettendo una rapida rilevazione e risoluzione di eventuali problemi o di ottimizzarne il funzionamento. Grazie alla sincronizzazione costante eventuali modifiche manuali, una volta implementate, saranno immediatamente riflesse anche nella controparte digitale (Madni A.M. et al., 2019). Per ottenere la connessione tra mondo fisico e virtuale è necessaria l'implementazione di un Cyber-Physical System. Un sistema Cyber-Fisico è un sistema che, attraverso l'integrazione di sensori e attuatori, da in dotazione a un asset fisico (che può essere un oggetto, un sistema o un processo industriale) capacità computazionale, di comunicazione e di controllo al fine di generare e inviare dati alla propria controparte digitale. Questo canale di comunicazione, generato tramite Internet, è l'essenza del sistema Cyber-Fisico ed è questo a garantire una perfetta sincronizzazione in tempo reale (Fei Tao et al., 2019).

Il concetto appena illustrato di Digital Shadow si avvicina già molto a quello finale di Digital Twin. Sono presenti tutte le componenti principali quali il mondo fisico rappresentato dall'elemento simulato, il mondo virtuale di cui fa parte la Digital Shadow e il sistema di comunicazione tra le due realtà identificabile nel sistema Cyber-Fisico. A mancare però sono due aspetti fondamentali: la bi-direzionalità del flusso comunicativo tra gemello fisico e digitale e l'implementazione dell'Intelligenza Artificiale nel gemello virtuale.

1.2.3 Digital Twin

Il Digital Twin è una rappresentazione estremamente fedele di un elemento fisico capace di ricevere ed elaborare dati. È inserito in un contesto informatico che permette, in modo automatico, la generazione e lo scambio reciproco di informazioni in tempo reale e con il proprio gemello fisico. Si genera così un sistema in cui il Digital Twin e il corrispettivo fisico si influenzano vicendevolmente durante tutto il loro ciclo di vita. Inoltre è in grado di prendere decisioni e applicare soluzioni preventive grazie all'Intelligenza Artificiale di cui è dotato (Madni A.M. et al., 2019).

Il Digital Twin si può considerare un'evoluzione della Digital Shadow, infatti utilizza lo stesso sistema di funzionamento e mantiene tutte le caratteristiche di quest'ultima. Ciò che lo rende più evoluto sono, come anticipato alla fine del paragrafo precedente, la bi-direzionalità del flusso comunicativo, ossia la capacità di inviare dati, e l'Intelligenza Artificiale. Per bi-direzionalità del flusso comunicativo si intende la capacità del Digital Twin di inviare dati al corrispettivo gemello fisico. Una Digital Shadow non possiede infatti questa caratteristica (la sua funzione è solamente di monitoraggio e controllo) e delega ogni tipo di intervento agli operatori che lo effettuano manualmente. Per un Digital Twin invece è fondamentale lo scambio di informazioni reciproco tra lo spazio reale e virtuale poiché non è solo lo spazio fisico a definire quello digitale ma è anche il contrario.

In un Digital Twin l'interfaccia utente è adattiva, quindi sensibile alle preferenze e alle priorità dell'operatore. Questa caratteristica è conferita dall'Intelligenza Artificiale che apprende utilizzando i dati estratti dal gemello fisico su cui agisce l'operatore. L'obiettivo è quello di svolgere manutenzione predittiva e ottimizzare il sistema rendendolo più efficiente. L'Intelligenza Artificiale è una tecnologia innovativa capace di rendere intelligente il sistema nel quale viene implementata. Per raggiungere questo scopo fa uso di complessi algoritmi

chiamati di Machine Learning. Questi algoritmi usano metodi matematico-computazionali per apprendere informazioni direttamente dai dati. In questo modo migliorare le proprie prestazioni in modo adattivo man mano che aumentano le esperienze da cui apprendono fino a raggiungere il momento in cui saranno in grado di prendere decisioni autonomamente o prevedere scenari futuri.

1.3 Internet of Things

In questa sezione verrà svolto un approfondimento su una fondamentale tecnologia abilitante allo sviluppo del Digital Twin: l'Internet of Things.

L'Internet of Things è un concetto e un paradigma che considera la presenza pervasiva nell'ambiente di una varietà di oggetti intelligenti che sono in grado di interagire e cooperare tra loro tramite Internet per creare applicazioni e offrire nuovi servizi. In questa nuova prospettiva Internet quindi non è solo una rete di computer, ma si è evoluta in una rete di dispositivi di ogni tipo e dimensione: veicoli, smartphone, elettrodomestici, giocattoli, macchine fotografiche, strumenti medici o sistemi industriali. In questo modo cambia il modo di approcciarsi agli oggetti, soprattutto a quelli di uso quotidiano, perché si rendono identificabili, localizzabili e controllabili attraverso Internet (Patel et al, 2016). Tutti questi oggetti generano enormi quantità di dati, i quali, dopo essere stati raccolti e analizzati, rappresentano un importantissimo valore aggiunto per supportare le aziende nei loro processi decisionali.

1.3.1 Architettura

L'architettura Internet of Things è composta da una struttura a tre livelli di operatività (Figura 1.2). Il livello più basso è quello percettivo, poi abbiamo il livello network e infine quello applicativo.

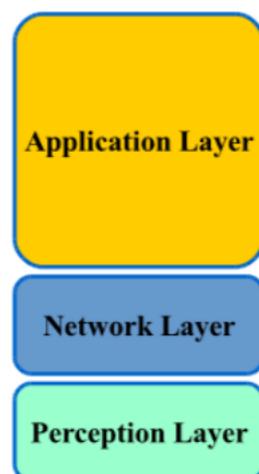


Figura 1.2 - Architettura Internet of Things (Adattato da: Al-Fuqaha et al., 2015)

Nel livello percettivo si trova la componente tangibile e integrata negli ambienti ed è costituita dai sensori e dagli attuatori, cioè i terminali del sistema. Essi si occupano di monitorare e acquisire continuamente vari parametri o di compiere azioni sulla base di istruzioni ricevute. Il livello percettivo si interfaccia con il network, cioè la struttura di connessione, che ha la funzione di processare e trasmettere i dati. Potendo contare su svariate tecnologie wireless e su numerosi protocolli di comunicazione, il network funge da ponte tra il livello percettivo e il livello applicativo (Jie Lin et al., 2017). In quest'ultimo risiedono il cloud e l'interfaccia utente. All'interno del cloud troviamo la componente analitica, il cuore della struttura IoT. Infatti, gli algoritmi hanno il ruolo fondamentale di svolgere i processi computazionali e decisionali al fine di raggiungere gli obiettivi preposti al sistema. Infine, le analisi si concretizzano in decisioni, inviate agli attuatori per essere eseguite, o in risultati mostrati nelle interfacce utente per permettere all'utilizzatore finale di prendere decisioni ponderate sullo stato dell'ambiente (Revel G.M. e Naspi F., 2020).

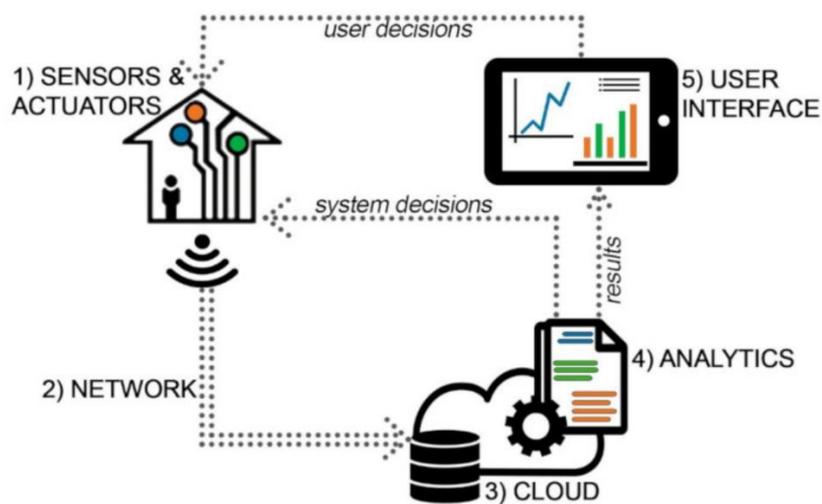


Figura 1.3 - Funzionamento Internet of Things (Revel G.M. e Naspi F., 2020)

1.3.2 Tecnologie abilitanti

Per la realizzazione dell'architettura di un sistema Internet of Things è necessario l'utilizzo di una serie di tecnologie. Alcune di queste sono: RFID, WSN, Bluetooth, NFC, Zigbee, Z-Wave.

L'RFID è una tecnologia di comunicazione basata sullo scambio di dati tramite segnali radio a breve distanza che viene utilizzata per identificare e tracciare oggetti. Questo sistema è costituito da un transponder, detto anche tag, da un'antenna e da un controller. I tag RFID sono delle memorie dotate di un apparato radio ricetrasmittente che, in seguito a un dialogo via radio con un apparato esterno, restituisce poi il proprio codice identificativo. Le antenne RFID, gestite dal controller, generano un campo magnetico che "risveglia" il tag, avviando così la comunicazione via radio. Il controller RFID è la componente di un sistema RFID dedicata alla lettura/identificazione e scrittura dei tag RFID e della comunicazione dei loro

codici ai sistemi di alto livello (Middleware o ERP) tramite le antenne. Rispetto ad altre tecnologie, l'RFID presenta i seguenti vantaggi: scansione rapida, durata, riutilizzabilità, grande memoria, lettura senza contatto, sicurezza, dimensioni ridotte, basso costo (Rfidglobal).

Una Wireless Sensor Network è costituita da dispositivi autonomi e indipendenti che sono distribuiti nello spazio e che sono in comunicazione tra loro. Questa rete fa uso di sensori per monitorare e analizzare le condizioni ambientali e fisiche come suono, vibrazioni, pressione, temperatura e movimento (K. Chopra et al., 2019). Si noti che sia RFID che WSN possono essere utilizzati per l'acquisizione di dati in IoT e la differenza è che RFID viene utilizzato principalmente per l'identificazione di oggetti, mentre WSN è utilizzato principalmente per la percezione dei parametri fisici del mondo reale associati all'ambiente circostante (Jie Lin et al., 2017).

Il Bluetooth è una tecnologia che funziona in modalità wireless per piccole distanze. È un'alternativa economica ad altre tecnologie wireless, funziona nell'intervallo da 10 a 100 metri e comunica a una velocità inferiore a 1 Mbps. Può essere utilizzato anche per creare Personal Area Network, infatti è in grado di collegare fino a 8 dispositivi contemporaneamente permettendo la condivisione dei dati (K. Chopra et al., 2019).

La Near Field Communication è uno standard per la comunicazione senza fili che aiuta a stabilire un collegamento di trasmissione radio a corto raggio (massimo 4 cm) tra due dispositivi. Questa tecnologia permette la comunicazione sia bidirezionale che unidirezionale: nel primo caso si parla di modalità attiva, nel secondo di modalità passiva. Il tag NFC passivo può essere un'alternativa al tag RFID per l'applicazione di comunicazione unidirezionale (N. Alam et al., 2017).

ZigBee è una tecnologia di rete wireless, progettata per comunicazioni a breve distanza e che funziona sul protocollo IEEE 802.15.4, 2,4 GHz. I vantaggi di queste reti includono basso consumo energetico, basso costo, bassa complessità, affidabilità e sicurezza. In più è anche altamente flessibile e supporta diverse topologie, incluse quelle a stella, ad albero e mesh.

Z-wave è una tecnologia di comunicazione wireless a breve distanza molto vantaggiosa grazie al basso costo, al basso consumo energetico e alla grande affidabilità. L'obiettivo principale di Z-wave è fornire una trasmissione affidabile tra un'unità di controllo e uno o più dispositivi finali. Sebbene sia ZigBee che Z-wave supportino la comunicazione wireless a corto raggio a basso costo e basso consumo energetico, ci sono alcune differenze tra loro. La principale differenza tra ZigBee e Z-wave è la banda di frequenza operata dal livello fisico. In ZigBee, la banda di frequenza del livello fisico è normalmente di 2,4 GHz, mentre la banda di frequenza in Z-wave è inferiore a 1 GHz (908,42 ~ 868,42 MHz). La rete ZigBee può supportare fino a 65000 dispositivi finali, mentre la rete Z-wave può supportare solo 232 dispositivi finali. Inoltre rispetto all'architettura ZigBee, Z-wave è più semplice nell'implementazione.

1.3.3 Sicurezza e Privacy

In questa sezione vengono presentate le proprietà in termini di sicurezza che un sistema internet of Things deve possedere e approfonditi alcuni degli attacchi che può subire in ogni

livello della sua architettura.

Un sistema Internet of Things deve possedere determinate proprietà di sicurezza per poter offrire dei servizi sicuri ed affidabili: riservatezza, integrità, disponibilità, identificazione e rispetto della privacy, tema quest'ultimo che verrà maggiormente approfondito (Jie Lin et al., 2017).

La riservatezza garantisce che i dati siano disponibili solo per gli utenti autorizzati durante tutto il processo e che non possano essere intercettati o interferiti da utenti non autorizzati.

L'integrità garantisce che i dati non possano essere manomessi da interferenze intenzionali o non intenzionali durante la consegna dei dati nelle reti di comunicazione, fornendo in definitiva i dati accurati per gli utenti autorizzati. La disponibilità garantisce che i dati e i dispositivi siano disponibili per utenti e servizi autorizzati ogni volta che vengono richiesti (X. Yang et al., 2015).

La disponibilità garantisce che i dati e i dispositivi siano disponibili per gli utenti e i servizi autorizzati ogni volta che vengono richiesti. Questa proprietà è importante perché i servizi sono comunemente richiesti in tempo reale e non possono essere forniti se i dati richiesti non sono disponibili (S. U. Maheswari et al., 2016).

L'identificazione e l'autenticazione garantiscono rispettivamente che i dispositivi o le applicazioni non autorizzati non possano essere collegati al sistema e che i dati forniti nelle reti e i dispositivi o le applicazioni che li richiedono siano legittimi (M.-C. Chuang and J.-F. Lee, 2014).

La privacy garantisce che i dati possano essere controllati solo dall'utente corrispondente e quindi che nessun altro utente possa accedere o elaborare i dati altrui. Andando più nel dettaglio sappiamo che tutti gli enormi dati raccolti e utilizzati nell'Internet of Things passano attraverso i seguenti tre passaggi:

- raccolta dei dati: rilevare e raccogliere i dati di stato degli oggetti;
- aggregazione dei dati: integrare una quantità di dati correlati in informazioni complete;
- analisi dati: estrarre il valore potenziale delle informazioni complete integrate.

Sulla base di questi passaggi i meccanismi di tutela della privacy possono essere suddivisi in tre categorie:

- tutela della privacy nella raccolta dei dati;
- tutela della privacy nell'aggregazione dei dati;
- tutela della privacy nell'analisi dei dati.

Poiché la privacy nella raccolta e nell'analisi dei dati può essere ampiamente preservata con varie tecniche (crittografia, gestione delle chiavi, ecc.), la maggior parte degli sforzi esistenti sulla conservazione della privacy si concentra sulla tutela della privacy durante l'aggregazione dei dati. In questo processo i dati rilevanti potrebbero essere elaborati in diversi luoghi diversi, e quindi è difficile ottenere la tutela della privacy attraverso i meccanismi di crittografia tradizionali. Pertanto, sono stati sviluppati diversi meccanismi di

tutela della privacy incentrati su questo processo e possono essere suddivisi nelle seguenti categorie:

- tutela della privacy basata sull'anonimato;
- tutela della privacy basata sulla crittografia;
- tutela della privacy basata sulla perturbazione.

Nella tutela della privacy basata sull'anonimato vengono utilizzate delle tecniche di anonimato per preservare la riservatezza delle informazioni di identificazione. Alcuni esempi sono: la K-anonymity, la L-diversity e la T-closeness (F. Qiu et al., 2015). Nella tutela della privacy basata sulla crittografia, vengono utilizzate delle tecniche di crittografia per garantire che i dati non vengano intercettati. Alcuni esempi sono: la homomorphic encryption, il commitment mechanism, la secret sharing e la zero-knowledge proof. Tuttavia, le tecniche di crittografia esistenti possono ottenere la riservatezza solo sulla trasmissione dei dati e potrebbero non funzionare bene sulla tutela della privacy (J. Girao et al., 2005). Nella tutela della privacy basata sulla perturbazione, vengono utilizzate delle tecniche per perturbare i dati grezzi. Alcuni esempi sono: la data customization, la data sharing e il random noise injection. A causa delle ottime prestazioni operando direttamente sui dati grezzi, gli schemi di tutela della privacy basati sulle perturbazioni sono tecniche molto popolari. Tuttavia, la tutela della privacy basata sulle perturbazioni raggiunge grandi prestazioni al costo di ridurre l'utilità dei dati (W. He et al., 2007). Con scarsa utilità, i dati potrebbero non supportare i servizi richiesti dalle applicazioni. Pertanto, la progettazione di schemi di tutela della privacy che mantengono un'elevata utilità dei dati rimane una grande sfida.

Terminata la trattazione sulle proprietà di sicurezza che deve possedere un sistema Internet of Things, proseguiamo approfondendo i principali attacchi a cui questo sistema è esposto dividendoli in base al livello dell'architettura verso cui sono mirati.

Nel livello percettivo gli attacchi alla sicurezza si concentrano sulla falsificazione dei dati raccolti e sulla manomissione dei dispositivi di percezione. Due esempi sono l'attacco di cattura del nodo e l'attacco di privazione del sonno. Lo scopo dell'attacco di cattura del nodo è catturare e controllare un nodo sostituendolo fisicamente o manometterne l'hardware. Se un nodo viene compromesso da quest'attacco, le informazioni importanti (chiave di comunicazione di gruppo, chiave radio, chiave di corrispondenza) vengono pericolosamente esposte. Un altro metodo di eseguire quest'attacco è copiare le informazioni importanti associate al nodo catturato su un nodo dannoso e quindi simulare il nodo dannoso come nodo autorizzato per connettersi alla rete o al sistema. In questo modo possono essere iniettati dati falsi alle applicazioni che a questo punto non riescono più fornire regolarmente i loro servizi. Questo influisce sulle prestazioni e l'efficacia delle applicazioni e delle reti IoT. Per difendersi dall'attacco di cattura del nodo, è necessario studiare schemi efficaci per monitorare e rilevare i nodi dannosi (M. V. Bharathi et al., 2012). Per quanto riguarda l'attacco di privazione del sonno lo scopo è quello di ridurre il ciclo di vita di un nodo. In un sistema IoT, la maggior parte dei dispositivi o dei nodi ha una capacità di alimentazione ridotta. Per estendere il ciclo di vita dei dispositivi e dei nodi, i nodi sono programmati per seguire una routine di sonno per ridurre il consumo di energia. Tuttavia, l'attacco di privazione del sonno può interrompere le

routine di sonno programmate e mantenere i nodi sempre svegli fino a quando non vengono spenti. Per contrastare questo tipo di attacchi e quindi estendere il ciclo di vita dei nodi, cambiare lo schema di raccolta di energia può essere una possibile soluzione, in modo che i nodi possano raccogliere energia dall'ambiente esterno (M. Sarkar and D. B. Roy, 2011).

Nel livello di rete gli attacchi alla sicurezza si concentrano sul rendere sovraccariche o inaccessibili le risorse di rete o sull'intercettazione delle informazioni. Due esempi sono: gli attacchi DoS (Denial of Service) e man in the middle. Gli attacchi DoS possono consumare tutte le risorse disponibili nell'infrastruttura IoT attaccando i protocolli di rete o bombardando la rete dell'IoT con un traffico massiccio, rendendo non disponibili i servizi. Questo li rende uno degli attacchi più comuni e rappresentano una minaccia molto pericolosa (S. U. Maheswari et al., 2016). In un attacco man in the middle, invece, un dispositivo dannoso si inserisce nel flusso di comunicazione tra due dispositivi che comunicano all'interno della rete IoT. In questo modo è in grado di intercettare e rubare le informazioni scambiate tra i due dispositivi. I due nodi non possono rilevare l'esistenza del dispositivo dannoso e credono che comunichino direttamente tra loro. Quest'attacco può violare la riservatezza, l'integrità e la privacy dei dati riservati. A differenza degli attacchi di cattura dei nodi, che devono manomettere fisicamente l'hardware dei dispositivi, l'attacco man in middle può essere lanciato facendo affidamento solo sui protocolli di comunicazione utilizzati nelle reti IoT. Tecniche di difesa efficienti per proteggersi da questo tipo di attacco è l'utilizzo di protocolli di comunicazione sicuri e schemi di gestione delle chiavi che possono garantire che l'identificazione e le informazioni chiave dei normali dispositivi non vengano intercettate (R. P. Padhy et al., 2011).

Nel livello applicativo i maggiori rischi riguardano gli attacchi software. Due esempi sono gli attacchi di auto-propagazione e gli script dannosi. Con i primi si infettano le applicazioni IoT con dei virus (worm, Trojan Horse) per ottenere o manomettere dati riservati. Per combattere questi attacchi devono essere implementati firewall affidabili e sistemi di rilevamento dei virus (W. Yu et al., 2010). Gli script dannosi sono degli script che vengono aggiunti, modificati o eliminati da un software con lo scopo di danneggiare le funzioni di sistema. Poiché tutte le applicazioni IoT sono connesse a Internet, l'avversario può facilmente ingannare i clienti eseguendo script dannosi quando richiedono servizi tramite Internet. Questi script possono comportare la fuga di dati riservati e persino l'arresto completo del sistema. Per difendersi da script dannosi, è necessario implementare nei sistemi IoT tecniche di rilevamento degli script efficaci, rilevamento di codice statico e rilevamento di azioni dinamiche (I. Andrea et al., 2015).

1.3.4 Cyber-Physical System e IoT

Questo paragrafo è stato scritto con lo scopo di chiarire le differenze e i punti di contatto tra CPS e IoT. Questa necessità nasce poiché, a causa delle evidenti somiglianze, non è immediato comprendere ciò che li contraddistingue e spesso a questi due concetti viene dato lo stesso significato.

Sia CPS che IoT mirano a realizzare l'interazione tra mondo cibernetico e mondo fisico. In particolare, CPS e IoT possono misurare le informazioni sullo stato dei componenti fisici

tramite dispositivi e sensori intelligenti senza l'input umano. Sia in CPS che in IoT le informazioni possono essere trasmesse e condivise tramite reti di comunicazione cablate o wireless. Dopo l'analisi delle informazioni sia CPS che IoT possono fornire servizi sicuri, efficienti e intelligenti alle applicazioni. La differenza fondamentale tra queste tecnologie è l'obiettivo che si pongono: l'essenza del CPS è il sistema e l'obiettivo principale di CPS è misurare le informazioni sullo stato dei dispositivi fisici e garantire il loro funzionamento in modo sicuro, efficiente e intelligente. Al contrario, l'IoT è un'infrastruttura di rete per connettere un numero enorme di dispositivi e per monitorare e controllare i dispositivi utilizzando le moderne tecnologie nel cyber-spazio. Pertanto, la chiave dell'IoT è "l'interconnessione". L'obiettivo principale dell'IoT è interconnettere varie reti in modo che la raccolta dei dati, la condivisione delle risorse, l'analisi e la gestione possano essere effettuate su reti eterogenee (Jie Lin et al., 2017).

1.4 Intelligenza Artificiale

L'Intelligenza Artificiale è una branca dell'informatica che si pone l'obiettivo di rendere capace una macchina di svolgere tutte quelle operazioni caratteristiche dell'intelletto umano come la pianificazione, la comprensione del linguaggio, il riconoscimento di oggetti e suoni, l'apprendimento e la risoluzione dei problemi. Una macchina intelligente può possedere tutte o solo alcune delle capacità elencate precedentemente e proprio per questo motivo l'Intelligenza Artificiale di cui è dotata può essere di due tipi: debole o forte.

L'intelligenza Artificiale debole è una forma di intelligenza artificiale progettata specificamente per essere focalizzata su un compito ristretto. Possiede solo quelle funzioni cognitive necessarie a svolgere le proprie mansioni. In ambito warehousing un possibile esempio di applicazione di questo tipo di intelligenza artificiale sono i Mobile Industrial Robot, nei quali viene implementato un sistema intelligente di calcolo del percorso più veloce per raggiungere le proprie destinazioni.

L'Intelligenza Artificiale forte invece è un sistema di con capacità cognitive umane generalizzate. Quando viene presentato un compito insolito, questo potente sistema è in grado di trovare una soluzione senza l'intervento umano. Le macchine dotate di questo tipo di intelligenza hanno la capacità di approcciarsi a qualsiasi problema, piuttosto che a un solo problema specifico (Spotle, 2019).

1.4.1 Machine Learning

Il Machine Learning è un sottoinsieme dell'intelligenza artificiale basato sull'utilizzo di algoritmi che, attraverso metodi matematico-computazionali, riescono a ottenere informazioni direttamente dai dati. In questo modo migliorano le proprie prestazioni in modo adattivo man mano che aumentano le esperienze da cui apprendono fino a raggiungere il momento in cui saranno in grado di risolvere problemi autonomamente.

Di seguito viene svolta una descrizione di tutti i metodi di apprendimento con un maggior approfondimento riguardante il metodo basato sulle reti neurali (Figura 1.4).

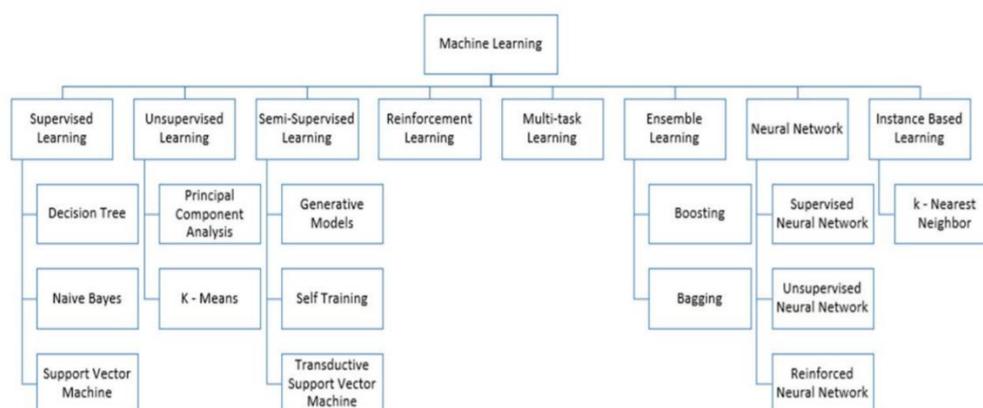


Figura 1.4 - Metodi di apprendimento del Machine Learning (Dey A., 2016)

Gli algoritmi di apprendimento automatico supervisionato sono quegli algoritmi che necessitano di assistenza esterna. Il set di dati di input è diviso in set di dati di addestramento e di test. Il set di dati del treno ha una variabile di output che deve essere prevista o classificata. Tutti gli algoritmi apprendono alcuni tipi di pattern dal set di dati di addestramento e li applicano al set di dati di test per la previsione o la classificazione (S.B. Kotsiantis, 2007).

Gli algoritmi di apprendimento senza supervisione apprendono poche funzionalità dai dati. Quando vengono introdotti nuovi dati, utilizza le funzionalità apprese in precedenza per riconoscere la classe dei dati. Viene utilizzato principalmente per il raggruppamento e la riduzione delle funzionalità (Dey A., 2016).

Gli algoritmi di apprendimento semi-supervisionato sono una tecnica che combina la potenza dell'apprendimento sia supervisionato che non supervisionato. Può essere fruttuoso in quelle aree dell'apprendimento automatico in cui i dati sono presenti solo senza etichetta e ottenere i dati etichettati sarebbe un processo noioso (X. Zhu et al., 2009).

L'apprendimento per rinforzo è un tipo di apprendimento in grado di apprendere e adattarsi alle mutazioni dell'ambiente. Questa tecnica si basa sul presupposto di potere ricevere degli stimoli dall'esterno a seconda delle azioni che vengono intraprese indirizzando le future azioni verso il risultato sperato con una politica di incentivi (azioni positive) e disincentivi (azioni negative) (R. S. Sutton, 1992).

L'apprendimento multitasking ha il semplice obiettivo di aiutare chi apprende a ottenere performance migliori. Quando gli algoritmi di apprendimento multi task vengono applicati a un'attività, ricordano la procedura di risoluzione del problema e utilizzano questi passaggi per trovare la soluzione di altri problemi o attività simili. Questo aiuto di un algoritmo all'altro può anche essere definito come meccanismo di trasferimento induttivo. Se gli algoritmi condividono la loro esperienza gli uni con gli altri, possono imparare contemporaneamente piuttosto che individualmente e possono essere molto più veloci (R. Caruana, 1997).

Quando diversi algoritmi individuali vengono combinati per formarne uno solo, quel particolare tipo di apprendimento viene chiamato Ensemble Learning. L'apprendimento d'insieme è un tema caldo sin dagli anni '90. È stato osservato che un insieme di algoritmi è quasi sempre più bravo a svolgere un determinato lavoro rispetto a uno singolo. Due famose tecniche di apprendimento in Ensemble sono boosting e bagging (D. Opitz et al., 1999).

Le reti neurali artificiali sono modelli matematici composti da neuroni artificiali ispirati alle reti neurali biologiche (quella umana o animale) che vengono utilizzati per risolvere problemi ingegneristici di Intelligenza Artificiale legati a diversi ambiti tecnologici. Le reti neurali artificiali possono essere descritte come modelli costituiti da almeno due strati, uno strato di ingresso e uno di uscita, e di solito anche da ulteriori strati intermedi (Figura 1.5). Più complesso è il problema da risolvere con la rete neurale artificiale, più strati sono necessari (James A. Anderson, 1997). Ogni strato della rete contiene un certo numero di neuroni artificiali specializzati. Questa rete può presentare moltissime interconnessioni tra i nodi dove lo schema tipico prevede la presenza di nodi input dove si trovano i dati da elaborare, i nodi intermedi che costituiscono il cuore della rete ed i nodi di output che hanno il ruolo di comunicare all'esterno i risultati del calcolo che la struttura ha effettuato. Ogni passaggio tra un nodo e l'altro della rete neurale è caratterizzato dall'elaborazione di più input che mediante un sistema di pesi contribuisce in maniera più o meno determinante alla formazione dell'output. Calibrando i pesi dei neuroni artificiale possiamo ottenere l'output che vogliamo per input specifici. Ma quando abbiamo una rete neurale artificiale di centinaia o migliaia di neuroni, è abbastanza complicato trovare a mano tutti i pesi necessari, quindi si fa uso di algoritmi che possono regolare i pesi in modo da ottenere l'output desiderato dalla rete. Questo processo di regolazione dei pesi è chiamato apprendimento o addestramento. L'allenamento inizia con pesi casuali e l'obiettivo è regolarli in modo che l'errore sia minimo (Laurene V. Fausett, 1993).

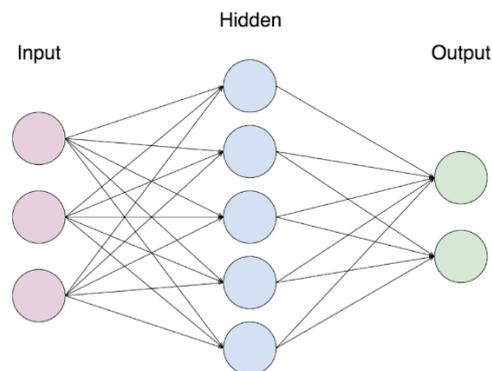


Figura 1.5 - Rete neurale artificiale (Laptrinhx, 2019)

In una rete neurale artificiale il processo di apprendimento può essere visto come il problema dell'aggiornamento dell'architettura di rete e dei pesi delle connessioni in modo che una rete possa eseguire in modo efficiente un compito specifico. La rete di solito deve apprendere i pesi della connessione dai modelli di formazione disponibili. Le prestazioni migliorano nel tempo aggiornando in modo iterativo i pesi nella rete. La capacità delle reti neurali di

apprendere automaticamente dagli esempi è uno dei principali vantaggi rispetto ai sistemi tradizionali (Gupta N., 2013).

Nell'apprendimento instance-based l'algoritmo impara un particolare tipo di pattern e cerca di applicare lo stesso modello ai dati che vengono successivamente inseriti. La complessità di questo algoritmo di apprendimento aumenta con la dimensione dei dati. Un noto esempio di apprendimento basato instance-based è il K-Nearest Neighbor (Dey A., 2016).

2. Struttura e architettura

2.1 Struttura

L'obiettivo di questo capitolo è fornire una descrizione dettagliata e approfondita di tutte le componenti che definiscono la struttura di un Digital Twin (Figura 2.1) (Talkhestani A. et al., 2019).

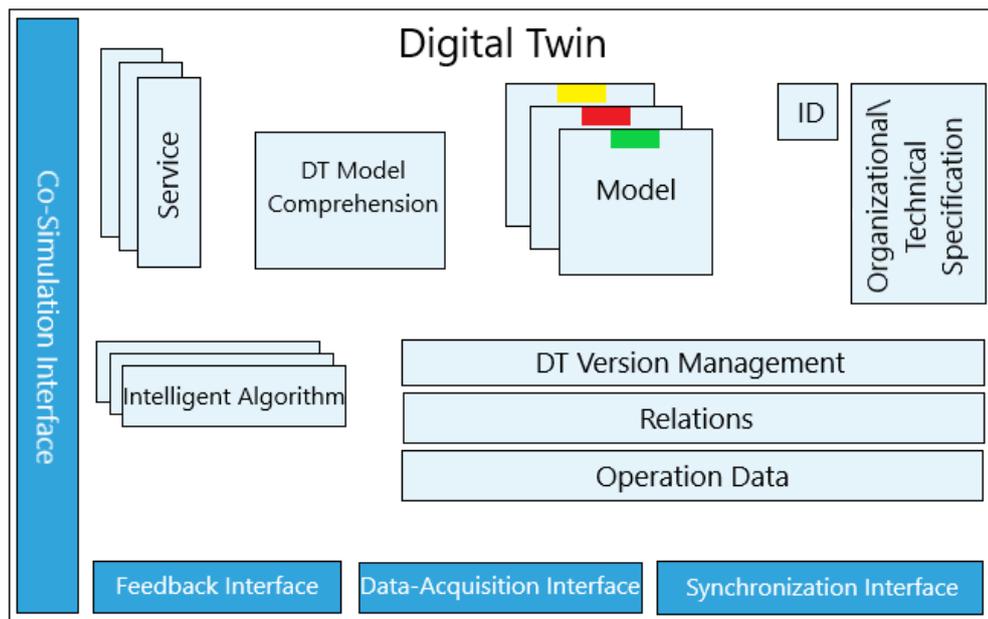


Figura 2.1 - Struttura del Digital Twin (Talkhestani A. et al., 2019)

- ID identificativo: questa proprietà identifica univocamente un determinato Digital Twin all'interno dello spazio virtuale in cui potrebbero esserne presenti più di uno. Grazie a questo identificativo, i dati e i modelli del Digital Twin vengono memorizzati come un modulo all'interno di un database e possono essere richiamati in qualsiasi momento;
- modelli e interfacce associate agli strumenti: un Digital Twin incapsula i modelli interdisciplinari di una risorsa. A questa categoria appartengono modelli CAD, modelli di schemi elettrici, modelli software, modelli funzionali e modelli di simulazione. Durante il processo di progettazione, ciascuno di questi modelli viene creato con uno strumento specifico e ciò rende molto difficile, se non impossibile, la comunicazione. Per ovviare a questo problema si utilizza un'interfaccia che permette l'interazione tra i vari strumenti di simulazione e, di conseguenza, la simulazione dell'intero sistema (il Digital Twin);

- gestione delle versioni dei modelli: un Digital Twin non contiene solo i dati e modelli attuali, ma anche tutti quelli generati durante il suo intero ciclo di vita. È possibile quindi accedere a tutte le versioni memorizzate dei modelli e alle loro relazioni e passare da una versione vecchia a quella corrente e viceversa in qualunque momento in base alle necessità. Inoltre la comparazione con i dati e le versioni precedenti è la base per l'implementazione dell'Intelligenza Artificiale;
- dati operativi dell'asset fisico: per riflettere accuratamente il comportamento e lo stato corrente dell'asset, il Digital Twin deve contenere i dati operativi. Può trattarsi di dati provenienti dai sensori che vengono continuamente trasmessi e registrati, oppure di dati di controllo che determinano lo stato del componente reale durante l'intero ciclo di vita. Questi dati vengono elaborati e memorizzati all'interno di un database e vengono utilizzati da algoritmi che operano sia online che offline per migliorare la comprensione dell'asset e perfezionare ulteriormente i modelli corrispondenti;
- dati organizzativi e tecnici dell'asset: questi dati contengono informazioni sull'asset fisico: come, quando e da chi è stato prodotto, progettato, sviluppato o da chi è stato commissionato. È presente anche tutta la documentazione prodotta durante il suo ciclo di vita, compresi quindi i documenti creati durante la fase di sviluppo (specifiche dei requisiti o i layout di progettazione) e durante il funzionamento (rapporti di manutenzione);
- informazioni sulle sue relazioni con gli altri Digital Twin: nel caso in cui all'interno di un sistema fossero presenti diversi Digital Twin, è necessario memorizzare all'interno di ciascuno le relazioni che si hanno con gli altri Digital Twin presenti nel sistema. Nello specifico vanno memorizzati in ognuno tutti i modelli interdisciplinari che sono correlati a modelli presenti in altri Digital Twin. Questa operazione è necessaria per garantire la coerenza e la coesione del sistema;
- interfaccia di co-simulazione: quest'interfaccia ha il compito effettuare la simulazione dell'intero sistema. Per fare ciò mette in comunicazione tutti i vari strumenti di simulazione indipendenti utilizzati per gestire ogni modello di ogni singolo Digital Twin presente nel sistema;
- interfaccia per la sincronizzazione di modelli e relazioni: quest'interfaccia ha il compito di sincronizzare il Digital Twin con il proprio gemello fisico. Grazie alla realizzazione di una descrizione semantica standardizzata di modelli, dati e servizi, ogni volta che avviene un cambiamento nel gemello fisico o nelle sue relazioni fisiche con altri asset, quest'interfaccia aggiorna i modelli interdisciplinari e, di conseguenza, le relazioni con gli altri Digital Twin;
- interfaccia per l'acquisizione dei dati: grazie a quest'interfaccia un Digital Twin è in grado di scambiare e memorizzare i dati. Questa capacità è fondamentale perché è necessario che la simulazione venga eseguita dinamicamente in parallelo al mondo reale. Infatti sono i dati raccolti dai sensori che permettono di costruire il comportamento effettivamente dinamico dell'asset in un ambiente simulativo;
- Intelligenza Artificiale: grazie all'Intelligenza Artificiale un Digital Twin è in grado di applicare algoritmi per condurre analisi sui dati operativi al fine di estrarre nuove conoscenze. Quest'ultime vengono incorporate nei modelli e permettono quindi al

Digital Twin di migliorare in modo incrementale il suo comportamento e le sue caratteristiche;

- servizi: un Digital Twin deve includere tutte le funzionalità che la risorsa può eseguire nel mondo reale. Queste funzionalità sono memorizzate come diversi servizi che, attraverso l'uso di standardizzazioni, possono essere compresi e utilizzati da altri Digital Twin per svolgere i propri compiti. Nel momento in cui variano le funzionalità della risorsa fisica, i servizi del corrispettivo Digital Twin si aggiornano di conseguenza;
- interfaccia di feedback: questa interfaccia consente il trasferimento di dati dal Digital Twin alle risorse fisiche. Pertanto, questa interfaccia può essere rappresentata come un canale di comunicazione o un canale di ritorno tra il DT e l'asset per la parametrizzazione dinamica e il controllo. Con l'ausilio di questo canale è possibile rendere autonomo il sistema Cyber-Fisico riducendo al minimo l'intervento umano.

2.2 Architettura

Per permettere l'implementazione del Digital Twin di un sistema produttivo è necessario che l'architettura del sistema comprenda quattro elementi: lo spazio fisico, lo Shop-Floor Service System, lo spazio virtuale e gli Shop-Floor Digital Twin Data (Figura 2.2).

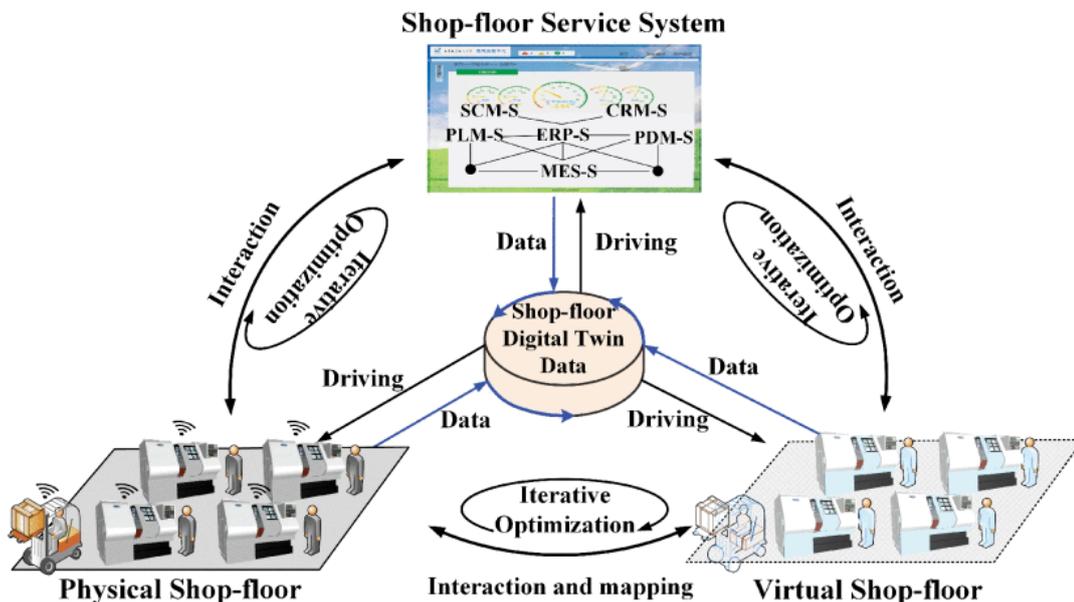


Figura 2.2 - Architettura Digital Twin (F. Tao and M. Zhang, 2017)

2.2.1 Lo spazio fisico

Lo spazio fisico è un ambiente complesso, diversificato e dinamico in cui sono presenti tutti gli asset fisici che costituiscono la linea di produzione. In quest'ultimi vanno integrati i

dispositivi in grado di realizzare l'interconnessione e l'interazione con lo spazio virtuale in modo da costituire il Cyber-Physical System. Questi dispositivi creano dati eterogenei e utilizzano protocolli e interfacce di comunicazione diversi tra loro rendendo difficile lo scambio delle informazioni. Per ovviare a questi problemi vengono implementati rispettivamente dei sistemi di integrazione dei dati che effettuano pulizie e conversioni del formato e moduli di accesso personalizzati che uniformano i protocolli e le interfacce di comunicazione. In questo spazio i gli asset fisici comunicano sia verticalmente con lo spazio virtuale inviando dati e ricevendo ordini di controllo che orizzontalmente tra loro per regolare reciprocamente i comportamenti (F. Tao and M. Zhang, 2017a).

2.2.2 Lo Shop-Floor Service System

Lo Shop-Floor Service System fornisce vari servizi per supportare la gestione e il controllo dello spazio fisico, nonché il funzionamento e l'evoluzione dello spazio virtuale.

Per svolgere il proprio compito questo sistema incapsula le risorse (dati, modelli, algoritmi, sistemi esistenti e metodi di visualizzazione) in sottoservizi e successivamente li seleziona per formare servizi compositi per le richieste dello spazio fisico o virtuale (Figura 2.3).

Le richieste dello spazio fisico includono principalmente la pianificazione e la programmazione della produzione, il test di qualità, il monitoraggio dei progressi e hanno come oggetto la risoluzione rapida di problemi esistenti e la prevenzione di possibili guasti durante la produzione.

Le richieste dello spazio virtuale invece richiedono principalmente calibrazione e test o data mining per supportare il funzionamento e l'evoluzione del modello. Queste richieste possono essere scomposte in sotto-richieste che di solito si concentrano su problemi come quali dati dovrebbero essere usati, quale modello è il più adatto e quale algoritmo fornisce la migliore soluzione. In funzione della determinata sotto-richiesta lo Shop-Floor Service System seleziona i sottoservizi appropriati, li combina e genera dei servizi compositi, i quali vengono monitorati in fase di esecuzione e, nel caso in cui non funzionino correttamente, vengono ricomposti. Nello spazio virtuale i servizi compositi vengono trasmessi direttamente ai modelli, mentre nello spazio fisico vengono trasmessi prima allo spazio virtuale per la verifica e poi tornano nello spazio fisico per l'esecuzione. Inoltre, i servizi possono essere programmati manualmente tramite il software applicativo sul computer o sul cellulare, se necessario (F. Tao and M. Zhang, 2017b).

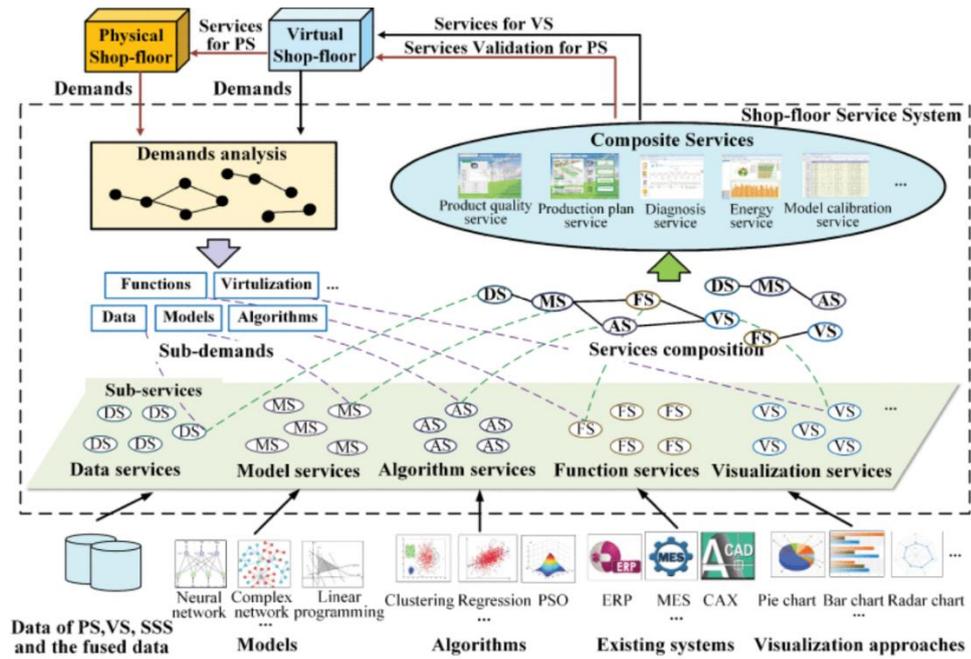


Figura 2.3 - Shop-Floor Service System (F. Tao and M. Zhang, 2017a)

2.2.3 Lo spazio virtuale

Lo spazio virtuale è il luogo in cui risiedono tutti i Digital Twin che costituiscono il sistema produttivo. Loro sono costantemente sincronizzati con il proprio gemello fisico e, come mostrato in figura 2.4, grazie all'uso di strategie di calibrazione è possibile rendere i modelli sempre più accurati.

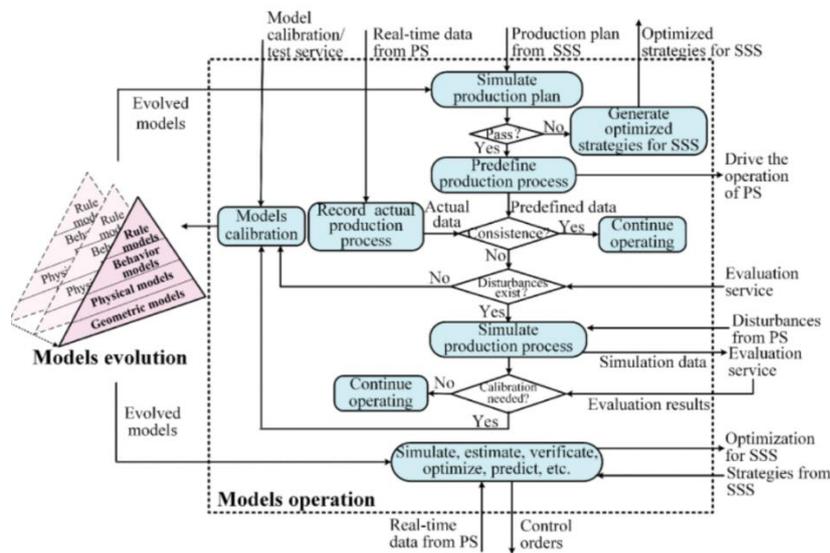


Figura 2.4 - Calibrazione dei modelli (F. Tao and M. Zhang, 2017a)

Nel dettaglio, il piano di produzione di Shop-Floor Service System viene trasmesso allo spazio virtuale per la verifica. Sulla base del piano verificato, il processo di produzione virtuale predefinito formerà e guiderà la produzione effettiva nello spazio fisico. Quando le entità fisiche iniziano a svolgere le loro operazioni, i dati in tempo reale da esse generati vengono registrati e confrontati con i dati del processo predefinito per valutarne la coerenza. Se i due tipi di dati sono allineati, viene confermata l'accuratezza dei modelli, altrimenti verrà programmato un servizio di valutazione per stimare il motivo dell'incongruenza. Se il servizio di valutazione non rileva disturbi nello spazio fisico, si può considerare che l'incongruenza è causata dai difetti dei modelli e sono necessarie calibrazioni. Se, invece, effettivamente esistono disturbi, saranno catturati e aggiunti ai modelli. La simulazione viene effettuata di nuovo con i modelli aggiornati. Se i risultati della simulazione non sono ancora coerenti con gli stati effettivi, sono necessarie calibrazioni, altrimenti i modelli si possono considerare accurati. Questo processo fa sì che ogni Digital Twin si evolve nel tempo col lo scopo di effettuare stime, verifiche, ottimizzazioni e previsioni per il processo operativo (F. Tao and M. Zhang, 2017a).

2.2.4 Shop-Floor Digital Twin Data

Gli Shop-Floor Digital Twin Data non sono altro che l'insieme dei dati generati dallo spazio fisico, da quello virtuale e dallo Shop-Floor Service System e dei metodi utilizzati per la loro modellazione ed elaborazione.

I dati prodotti nello spazio fisico riguardano principalmente i fattori di produzione, il processo di produzione e l'ambiente e sono generati direttamente dalla risorsa fisica senza ulteriore elaborazione. Questi tipi di dati vengono chiamati dati fisici.

I dati nello spazio virtuale si riferiscono a parametri del modello e dati di simulazione, valutazione, ottimizzazione e previsione, mentre i dati dello Shop-Floor Service System riguardano principalmente i dati dei vari servizi. Questi due tipi di dati vengono dedotti dai dati fisici e definiti come dati virtuali.

I dati fusi sono l'integrazione e la convergenza di dati fisici e virtuali attraverso il confronto, l'associazione, la combinazione e il raggruppamento. Il processo che porta alla creazione dei dati fusi è costituito da tre passaggi: la conversione, la pulizia e la fusione dei dati.

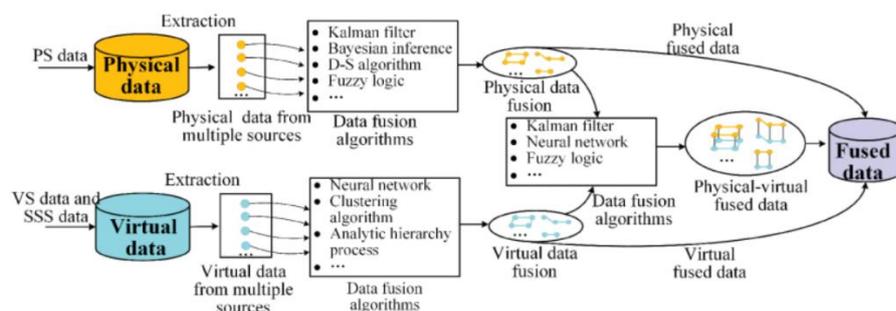


Figura 2.5 - Processo di fusione dei dati (F. Tao and M. Zhang, 2017a)

La conversione dei dati trasforma i dati provenienti dalle varie fonti con vari formati, tipi, strutture e incapsulamenti in modo tale da renderli omogenei; successivamente si effettua la pulizia dei dati che rileva e rimuove i dati sporchi (errori, dati duplicati, dati non validi) e ripristina i dati mancanti per migliorare la qualità dei dati; infine per ottenere interpretazioni coerenti e complete, viene applicata la fusione dei dati tramite l'esecuzione di specifici algoritmi. Grazie a questo sistema è possibile fornire una rappresentazione più coerente, accurata e completa di tutte le informazioni sia fisiche che virtuali.

Lo Shop-Floor Digital Twin Data è ottimizzato in modo iterativo. I dati storici vengono aggiornati e ampliati con l'unione dei dati in tempo reale, mentre i dati in tempo reale possono essere testati e corretti in base alle conoscenze accumulate nello storico. I dati fisici possono essere valutati e simulati dai dati virtuali, mentre il virtuale può essere confrontato con il fisico per confermarne l'accuratezza (F. Tao and M. Zhang, 2017a).

3. Applicazioni

Grazie alle sue caratteristiche, il Digital Twin trova applicazione in diversi ambiti industriali. In questo capitolo verranno analizzate in particolare tre applicazioni coerenti con il caso studio oggetto della tesi: una in ambito manufacturing e due in ambito warehousing.

3.1 Manufacturing

La prima applicazione che viene trattata riguarda l'implementazione di un sistema Digital Twin basato su una linea di produzione in cui avviene un'operazione di saldatura. È stato scelto di adoperare questa tecnologia al fine di migliorare la qualità dei prodotti finiti e l'efficienza produttiva.

Per prima cosa è stato costruito il modello virtuale della linea di produzione e sono stati raccolti in tempo reale i dati di monitoraggio della linea di produzione. In questo modo è stato possibile creare un collegamento che permette l'interazione in tempo reale tra la linea di produzione fisica e quella virtuale. Durante lo svolgimento delle operazioni vengono analizzati il tempo di funzionamento e il ciclo di produzione di ciascuna stazione ed è stato possibile realizzare un sistema di segnalazione in tempo reale dello stato del processo di produzione.

La figura 3.1 mostra il modello virtuale della linea di produzione dello stabilimento. Da sinistra a destra, sono presenti la saldatrice, il trasportatore per la consegna dei pezzi finiti e il magazzino multistrato per lo stoccaggio dei pezzi finiti (Zheng et al.,2019).

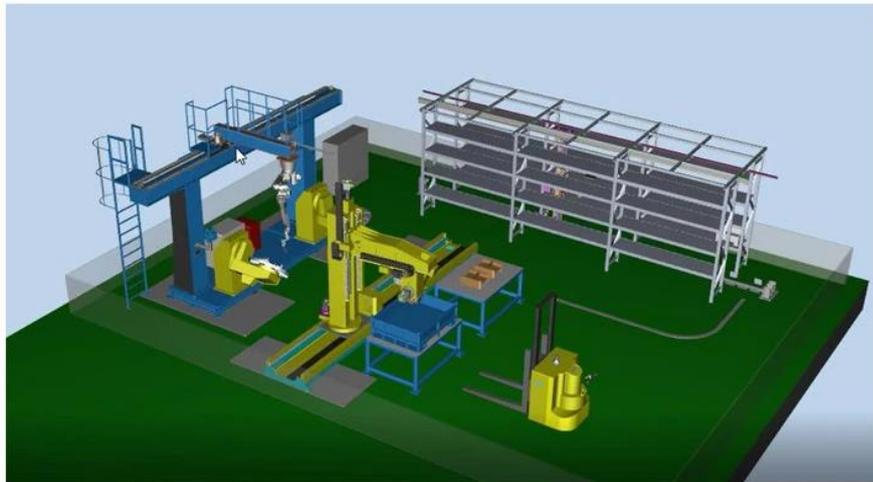


Figura 3.1 - Modello virtuale linea produzione (Zheng et al.,2019)

3.1.1 Modellazione della linea di produzione

Il modello Digital Twin della linea di produzione è composto da tre modelli: il modello geometrico 3D, il modello fisico e il modello cinematico. Il modello geometrico descrive la forma e le dimensioni del dispositivo; il modello fisico definisce lo stato meccanico e lo stato termodinamico del dispositivo nei dati parametrici; il modello cinematico definisce le coppie cinematiche dei dispositivi per descrivere la connessione tra i loro componenti. In base al modello, i dati di monitoraggio sono suddivisi in dati fisici e dati cinematici. I dati fisici corrispondono ai parametri specifici del modello fisico. Quando i dati fisici vengono ricevuti, il sistema li visualizza e li registra nel database per l'analisi e l'ottimizzazione dei dati storici. I dati cinematici corrispondono ai parametri specifici del modello cinematico. Quando i dati cinematici vengono ricevuti, il sistema li invia al dispositivo corrispondente e aggiorna le relative coppie cinematiche. Infine, il modello geometrico di ogni dispositivo viene aggiornato per realizzare il processo di monitoraggio 3D in tempo reale (Zheng et al.,2019).

3.1.2 Percezione delle informazioni e acquisizione dei dati

I dati vengono acquisiti principalmente dai robot industriali sulla linea di produzione. Per controllare la loro versione digitale nell'ambiente virtuale in tempo reale è necessario ottenere i dati di stato delle rispettive controparti fisiche e anche pre-definire, sempre nell'ambiente virtuale, le informazioni sulle coppie cinematiche.

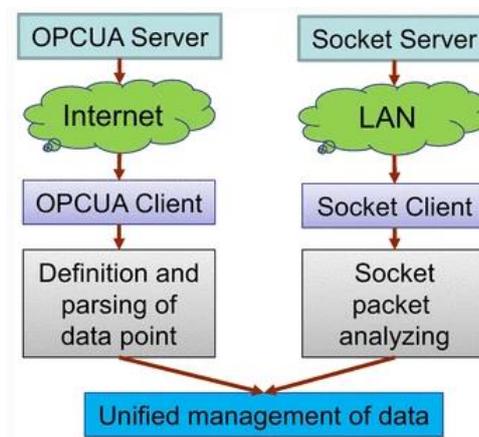


Figura 3.2 - Processo di acquisizione dei dati (Zheng et al.,2019)

I robot vengono collegati ad un server OPCUA che ha il compito di stabilire una connessione per permettere la comunicazione tra il sistema fisico e quello virtuale. In questo server i dati di stato provenienti dai robot fisici vengono raccolti, letti e infine inviati all'ambiente virtuale. A questo punto un client OPCUA riceve i dati, li analizza e li inserisce in un buffer in attesa che il sistema venga aggiornato. Quando l'ambiente virtuale viene aggiornato, i dati vengono estratti dal buffer dei dati e inviati al dispositivo virtuale corrispondente.

In questo processo per acquisire i dati vengono principalmente adottate le modalità di comunicazione OPCUA e Socket. Quando la comunicazione avviene in una Wide Area Network (Internet) si adotta la modalità di comunicazione OPCUA, che offre un'elevata affidabilità e sicurezza; mentre se la comunicazione avviene all'interno di una Local Area Network si adotta la modalità di comunicazione Socket per garantire una migliore efficienza di trasmissione (Zheng et al.,2019).

3.1.3 Processo di mappatura dei dati in tempo reale

Dopo aver stabilito le modalità di acquisizione dei dati, viene determinato il processo per la mappatura dei dati in tempo reale (Figura 3.3). Quest'ultimo comprende due passaggi: il primo consiste nella creazione dell'ambiente virtuale sulla base del sistema reale; l'altro è lo sviluppo del sistema applicativo Digital Twin basato su OPCUA e in grado di eseguire la mappatura virtuale-fisica.

Per prima cosa viene creato l'ambiente di simulazione virtuale della linea di produzione. Sulla base di questo, viene generato il file di layout da inviare al sistema applicativo Digital Twin. In quest'ultimo infine avvengono la configurazione del client OPCUA e l'implementazione del sistema di monitoraggio in tempo reale dello stato del processo di produzione (Zheng et al.,2019).

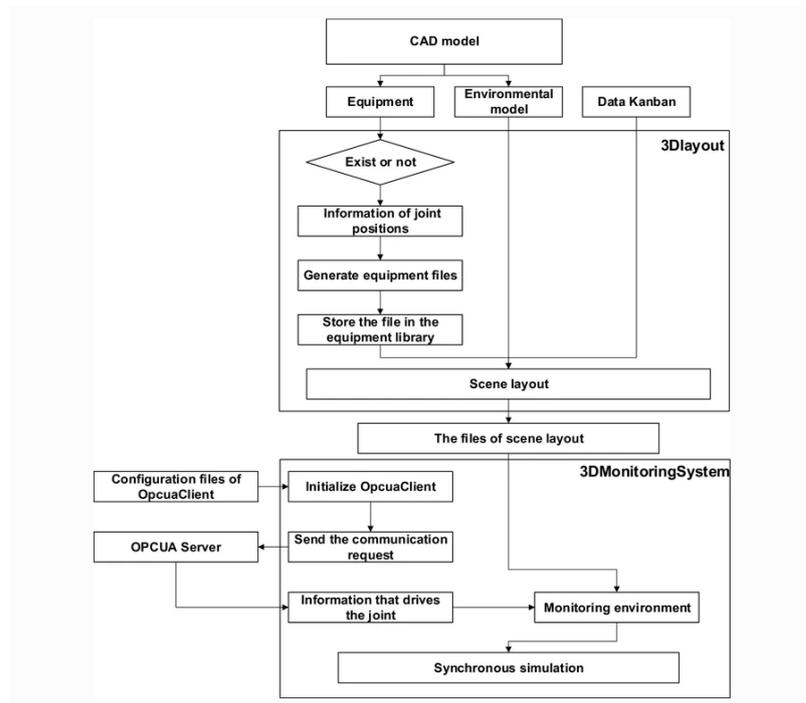


Figura 3.3 - Processo di mappatura dei dati (Zheng et al.,2019)

3.1.4 Effetto dell'applicazione

Il funzionamento sincrono del sistema di produzione virtuale è garantito dall'integrazione del modello fisico col modello virtuale. Il sistema di monitoraggio remoto e in tempo reale del processo di produzione fisico ci permette di verificare in qualunque momento l'affidabilità dei fattori produttivi e, in caso di problemi, offre una segnalazione puntuale e tempestiva.

Questo sistema può mostrare 19 tipi di dati della linea di produzione in tempo reale con un ritardo non superiore a un secondo. Questo sistema Digital Twin è in grado di garantire l'efficienza operativa dei fattori produttivi nella linea di produzione, oltre a fornire dati per la qualità della saldatura del prodotto (Zhang et al.,2019).

3.2 Warehousing I

In questa seconda applicazione viene implementata la tecnologia Digital Twin in ambito warehousing. Nello specifico si tratta di un magazzino automatizzato in larga scala adibito per lo stoccaggio e la distribuzione del tabacco.



Figura 3.4 - Modello virtuale del magazzino automatico (Leng J. et al., 2019)

Come mostrato in Figura 3.4, il corpo principale di questo magazzino è costituito da scaffali, transelevatori automatici, piani di lavoro del magazzino interni e esterni e da un sistema di imballaggio automatizzato. Gli scaffali hanno una struttura in acciaio di dimensione standardizzata. I transelevatori automatici si muovono tra gli scaffali per il carico e lo scarico delle merci che vengono identificate per mezzo di codici a barre, etichette magnetiche o della tecnologia RFID.

3.2.1 Sistema Digital Twin

Il sistema Digital Twin implementato in questo caso studio è costituito da due parti: il modello di ottimizzazione congiunta e il motore di simulazione semi-fisico (Figura 3.5). Il

primo, elaborando i dati generati dal magazzino fisico, è in grado di ottimizzare periodicamente tutte le operazioni che avvengono all'interno del magazzino. Successivamente questi risultati ottimizzati vengono inseriti nel motore di simulazione semi-fisica per verificare il loro comportamento nel sistema reale. Infine, se al termine di queste verifiche si ottiene il risultato atteso, avviene l'implementazione nel sistema reale (Leng J. et al., 2019).

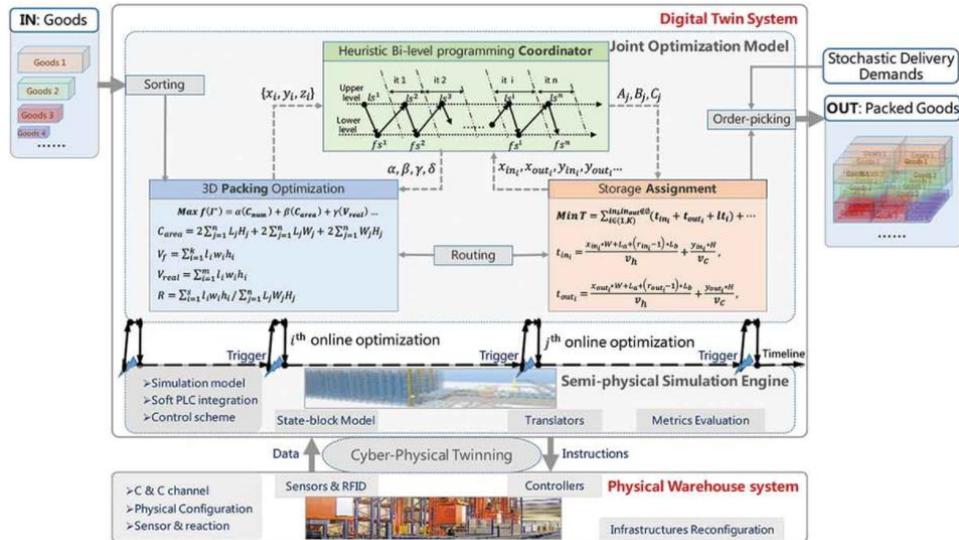


Figura 3.5 - Architettura dell'applicazione warehousing (Leng J. et al., 2019)

3.2.2 Motore di simulazione semi-fisica

La simulazione semi-fisica si riferisce alla creazione del canale di controllo e del canale di comunicazione tra il modello di simulazione, gli asset fisici e il motore di esecuzione del sistema. In questo modo quest'ultimo è in grado non solo di controllare il movimento del modello di simulazione, ma anche guidare l'azione dell'attrezzatura fisica.

La funzione del motore di simulazione semi-fisica include tre aspetti: mappare tutti i tipi di dati dei modelli virtuali del sistema fisico e quindi fornire l'input dei dati al modello di ottimizzazione congiunta; svolgere la verifica dei risultati ottimizzati dal modello di ottimizzazione congiunta; tradurre i risultati in istruzioni e distribuire a tutti i controllori nel sistema prodotto-servizio di magazzino.

Inoltre il motore di simulazione semi-fisico è in esecuzione durante l'intero processo di immagazzinamento per accumulare conoscenze e registrare le statistiche dell'effetto dell'implementazione del sistema Digital Twin. Può verificare lo schema di ottimizzazione e migliorare continuamente le prestazioni complessive del sistema (Leng J. et al., 2019).

3.2.3 Modello di ottimizzazione congiunta

Come visto precedentemente, oltre al motore di simulazione semi-fisico, il sistema Digital Twin include anche un modello di ottimizzazione congiunta. Questo modello è suddiviso in due moduli: modulo di ottimizzazione dell'imballaggio e modulo di assegnazione dello spazio di stoccaggio.

L'obiettivo del modulo di imballaggio è quello di impilare una serie di merci per massimizzare il tasso di utilizzo dello spazio occupato da ogni confezione. L'imballaggio delle merci è un processo chiave prima di immagazzinare le merci in un magazzino. L'ottimizzazione dell'imballaggio delle merci può ridurre l'uso di cartoni, garantire l'equilibrio del carico, ridurre i colli di bottiglia.

Il modulo di assegnazione dello spazio di stoccaggio è un insieme di regole che vengono utilizzate per assegnare la posizione di stoccaggio alle varie merci. Questo modulo elimina le risorse inutilizzate ed esegue un'assegnazione che tenta di bilanciare il carico tra le aree e le attrezzature disponibili. Una politica di assegnazione dello spazio di archiviazione efficace può ridurre i tempi di viaggio medi di recupero e prelievo degli ordini (Leng J. et al., 2019).

3.2.4 Sincronizzazione Cyber-Fisica

La sincronizzazione Cyber-Fisica è una componente fondamentale del sistema Digital Twin perché permette la comunicazione tra il sistema fisico e quello virtuale (Figura 3.6). Di seguito viene descritto il processo di implementazione.

In primo luogo, viene stabilito un canale di comunicazione bidirezionale tra le componenti fisiche e virtuali. In secondo luogo, in base ai flussi d'informazione o alle istruzioni inviate, viene creato un database per memorizzare i dati scambiati come l'istruzione di immagazzinamento, l'istruzione della macchina, i dati delle informazioni sul campo, i dati storici di una particolare macchina e i dati di background. Successivamente vengono definiti il protocollo di comunicazione, il formato delle istruzioni e il formato delle informazioni sul campo in modo da facilitare l'emissione delle istruzioni e il caricamento dei dati raccolti dai sensori sull'attrezzatura del magazzino. La sincronizzazione cyber-fisica è realizzata da un middleware che si occupa dell'integrazione dei protocolli pubblici e delle interfacce con il database. Infine, viene realizzata la connessione tra il modello di ottimizzazione congiunta e i controller all'interno delle apparecchiature e vengono visualizzati i dati statistici relativi alle prestazioni, in modo da effettuare il monitoraggio in tempo reale dell'intero magazzino (Leng J. et al., 2019).

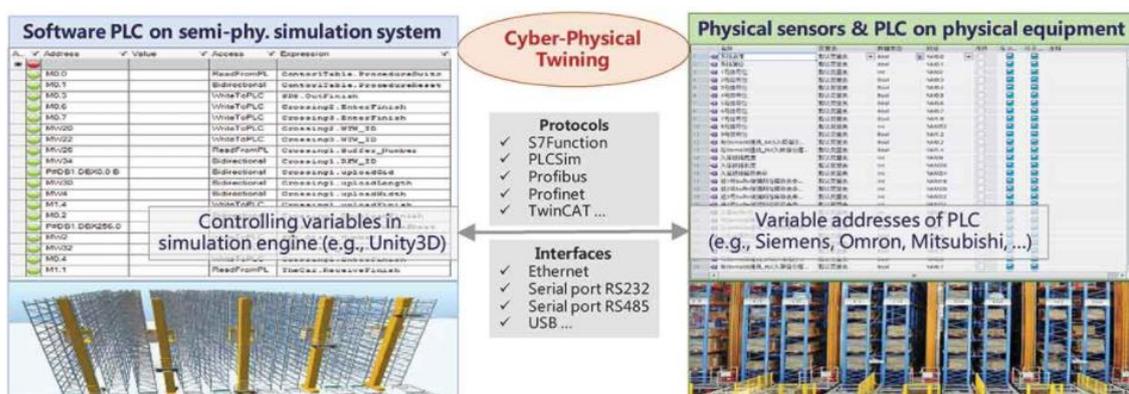


Figura 3.6 - Sincronizzazione Cyber-Fisica dell'applicazione warehousing (Leng J. et al., 2019)

3.3 Warehousing II

In questa seconda applicazione in ambito warehousing viene utilizzata la tecnologia Digital Twin con lo scopo di ottimizzare i flussi di carico-scarico dei pallet all'interno di un magazzino adibito allo stoccaggio di grandi quantità di confezioni di fazzoletti di carta. Questo magazzino è costituito da un numero variabile di baie parallele che formano lunghe corsie e può contenere fino a decine di migliaia di pallet contemporaneamente (Figura 3.7). Le diverse corsie sono servite tramite dei carrelli elevatori che si occupano di inserire i pallet in arrivo, prelevare i pallet in uscita richiesti e operare il servizio di pulizie notturne.



Figura 3.7 - Magazzino reale e modello di simulazione (Braglia et al.,2019)

L'implementazione di questo sistema Digital Twin parte dalla creazione di un modello di simulazione in grado di simulare il comportamento del magazzino. Successivamente, grazie all'utilizzo della tecnologia RFID, di algoritmi di data fusion e di una serie di sensori, vengono estrapolati, processati e immagazzinati i dati generati dai vari elementi che costituiscono il magazzino. Questi dati vengono inviati costantemente al modello, il quale esegue una serie di simulazioni che restituiscono le informazioni per ottimizzare i flussi di carico-scarico. Questi risultati, infine, vengono inviati automaticamente al magazzino sotto forma di specifiche da seguire per applicare l'ottimizzazione (Braglia et al.,2019).

3.3.1 Il Modello di simulazione

Per la modellazione del magazzino e dei suoi processi interni è stato adottato il pacchetto di simulazione Anylogic® Professional, che permette di creare applicazioni con un mix completamente funzionante di simulazione di eventi discreti e simulazione basata su agenti. Questi requisiti sono stati modellati utilizzando le tecniche Java® OOP e un approccio modulare gerarchico volto a semplificare l'intero processo. In particolare, gli elementi più importanti del magazzino del mondo reale sono stati accuratamente isolati e modellati come agenti indipendenti e intelligenti, in grado di interagire e comunicare tra loro. Questi agenti rappresentano, tra le varie entità, gli ordini in entrata e in uscita, i carrelli elevatori ed i

prodotti. Il modello è uno strumento prezioso per migliorare la produttività attuale del magazzino individuando problematiche e colli di bottiglia e per analizzare potenziali miglioramenti e modifiche future (Braglia et al.,2019).

3.3.2 Trasmissione dei dati

Per trasformare il modello di simulazione in un vero e proprio Digital Twin è necessario che il modello venga alimentato dai dati generati dal magazzino reale in maniera costante nel tempo. Per far ciò è stata costruita un'architettura di rete capace di metterlo in comunicazione in maniera bi-direzionale col magazzino reale. Tutti i carrelli elevatori sono stati dotati di hardware UHF-RFID, complessi sensori cinematici e un sistema di comunicazione wireless per interagire con una rete Wi-Fi. La posizione dei carrelli elevatori è determinata mediante algoritmi di data fusion, che combinano i dati di fase dei tag UHF di riferimento, posizionati a soffitto del magazzino, con quelli derivati dai sensori cinematici. La posizione del pallet viene quindi facilmente determinata dalle antenne posizionate sui montanti durante i processi di carico-scarico, riferendola alla posizione corrente del carrello elevatore. Ogni trenta minuti il server degli eventi centralizzato trasmette, tramite un'infrastruttura Wi-Fi, i dati al simulatore, tramite un protocollo predefinito. Questi dati contengono l'ubicazione di tutti i pallet, la posizione dei singoli carrelli elevatori, lo stato dei percorsi e gli ordini futuri in entrata e in uscita. La trasmissione avviene sotto forma di file Excel (o CSV), dove ogni riga rappresenta la coordinata geometrica di un pallet, la coordinata corrente di un carrello elevatore, lo stato dei percorsi disponibili (informazione necessaria per informare il simulatore se un percorso è attualmente accessibile) e, infine, il codice del pallet da caricare in o scaricare da una determinata posizione. La singola coordinata geometrica di un pallet è data considerando il codice di identificazione pallet, il codice di ubicazione, la corsia, la colonna e il livello da terra. Nel caso dei carrelli elevatori, la posizione è misurata in metri rispetto ad un'origine predefinita. Pertanto, la posizione dei pallet viene utilizzata per ottimizzare i flussi di carico-scarico, mentre la posizione dei carrelli elevatori viene utilizzata per scegliere il trasportatore più vicino per ogni percorso successivo.

Ogni volta che il simulatore ottiene i dati aggiornati esegue una serie di esperimenti personalizzati che simulano i successivi trenta minuti. Il risultato atteso è la messa a punto dell'assegnazione dei pallet e dei percorsi che riducono al minimo le distanze di movimentazione e di viaggio. Infine per applicare i risultati ottenuti, vengono ritrasmessi al computer del carrello elevatore i codici identificativi dei pallet da spostare sotto forma di lista di prelievo e la sequenza viene utilizzata dagli autisti del carrello elevatore per eseguire inserimenti e prelievi (Braglia et al.,2019).

4. Caso studio

Questo caso studio riguarda l'implementazione di un'architettura Digital Shadow di un magazzino automatico situato all'interno del Politecnico di Torino. L'obiettivo è quello di stabilire una connessione in tempo reale monodirezionale tra il magazzino fisico e il suo modello di simulazione. In questo capitolo verrà descritta inizialmente la struttura del magazzino automatico esaminando tutti gli elementi che lo compongono; successivamente verrà affrontata la descrizione dell'infrastruttura IT, sia hardware che software, che permette la comunicazione tra tutti gli elementi del magazzino; verrà spiegato il funzionamento del modello di simulazione; infine verrà presentato il modello UML che illustra tutte le relazioni tra le entità che costituiscono il magazzino.

4.1 Spazio fisico

Lo spazio fisico del sistema che verrà implementato corrisponde al magazzino automatico. Quest'ultimo viene utilizzato per lo stoccaggio di un piccolo numero di semilavorati che in seguito a un processo di assemblaggio si tramutano in prodotti finiti. Gli elementi che costituiscono il magazzino sono:

- l'area automatizzata;
- la stazione di ricarica dei MiR;
- l'area di assemblaggio.

L'area automatizzata, mostrata in Figura 4.1, è composta da una zona di stoccaggio a scaffalature verticali, da un trasloelevatore a basso carico, da una stazione di kitting, da una di picking e dai rulli trasportatori che collegano tutti questi elementi.



Figura 4.1 – Area automatizzata

Nella zona di stoccaggio vengono depositati e prelevati i contenitori al cui interno possiamo trovare i semilavorati, i prodotti finiti oppure possono essere vuoti in attesa di essere utilizzati. Dei prelievi e dei depositi se ne occupa il trasloelevatore, una macchina automatica composta da una colonna portante che si muove lungo un binario fisso al pavimento, da una piattaforma di accesso allo scaffale di destinazione che si muove verticalmente e da uno strumento che ha il compito di prendere il contenitore. In Figura 4.2 vengono mostrati il trasloelevatore e l'area di stoccaggio.

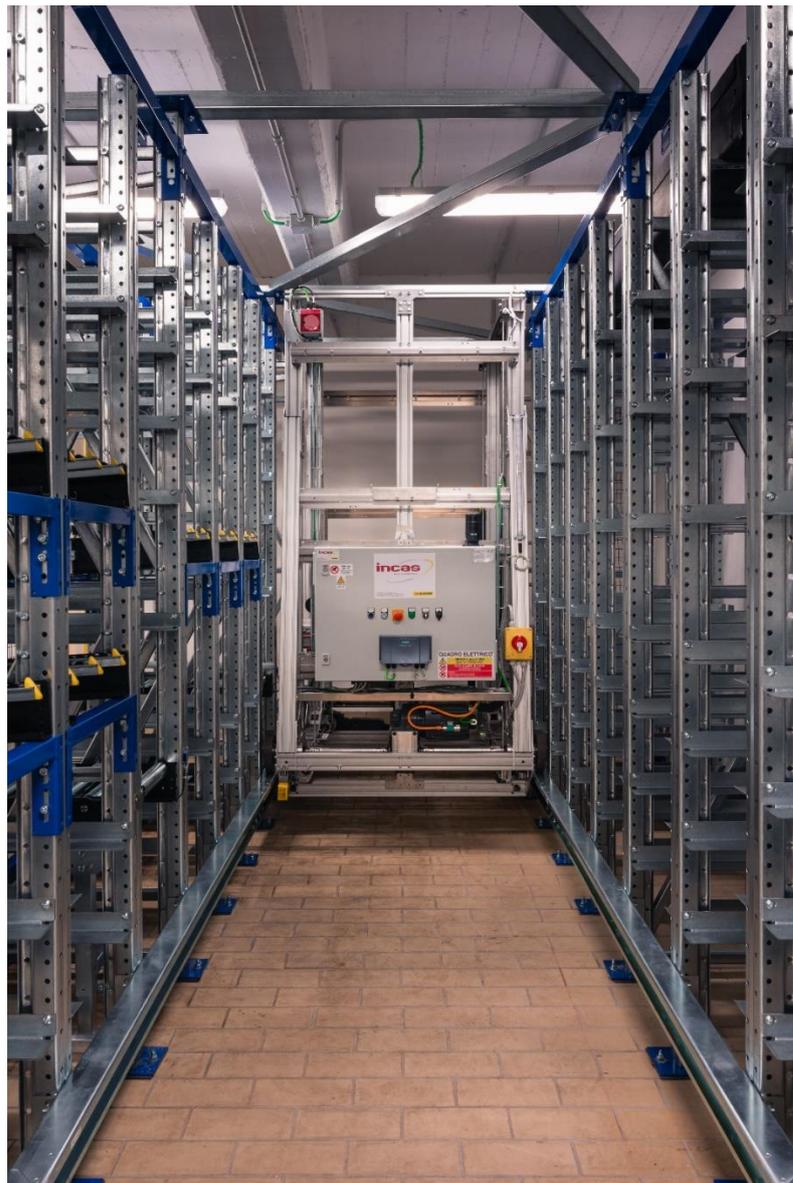


Figura 4.2 – Trasloelevatore e scaffalature

Nella stazione di kitting l'operatore si occupa, in seguito a un ordine di kitting, di prelevare i componenti dai loro rispettivi contenitori e di inserirli in un contenitore vuoto che verrà immagazzinato nell'area di stoccaggio. All'interno di quest'ultimo così troveremo tutti i

componenti che al momento dell'assemblaggio saranno necessari per costituire un determinato prodotto finito. Nella stazione di picking l'operatore riceve dalla zona di stoccaggio il contenitore con i componenti utili a creare il prodotto finito e inserisce in un contenitore vuoto quelli necessari a completare l'ordine di picking. In Figura 4.3 sono mostrate le stazioni di kitting e picking.



Figura 4.3 – A sinistra stazione di kitting e a destra stazione di picking

La movimentazione dei contenitori al di fuori dell'area automatizzata avviene attraverso l'utilizzo del MiR (Figura 4.4). Quest'ultimo è un robot mobile che automatizza rapidamente i trasporti e la logistica all'interno del magazzino. È in grado di trasportare autonomamente carichi fino a 100 kg e, grazie alle telecamere e ai sensori integrati, riconosce l'ambiente in cui si trova e sceglie il percorso più efficiente, evitando in tutta sicurezza ostacoli e persone. Permette di installare moduli superiori personalizzati come contenitori, griglie, elevatori e nastri. In questo caso studio il MiR è stato equipaggiato con dei rulli motorizzati dotati di un meccanismo di sollevamento incorporato che consente un prelievo più agibile dei contenitori.



Figura 4.4 – MiR

I MiR funzionano esclusivamente in modalità wireless e per questo motivo all'interno del magazzino si trova una stazione di ricarica (Figura 4.5) in cui i MiR si dirigono al termine di ogni missione e in cui si ricaricano in attesa di nuove istruzioni.



Figura 4.5 – Stazione di ricarica dei MiR

All'interno del magazzino troviamo infine un'area di assemblaggio in cui i semilavorati vengono prelevati da un contenitore, assemblati e successivamente impacchettati pronti per essere spediti.

4.2 Infrastruttura IT

L'infrastruttura IT è un elemento fondamentale per avere una corretta gestione di tutte le operazioni che avvengono all'interno di un magazzino automatico. È costituita da una parte software che comprende tutti i programmi gestionali e una parte hardware in cui sono presenti i server, i database e i dispositivi di rete.

4.2.1 Software gestionali

I software gestionali sono quegli strumenti informatici che permettono agli operatori di interfacciarsi con tutte le macchine e i dispositivi presenti all'interno del magazzino. Grazie a loro è quindi possibile controllare il funzionamento dei singoli macchinari, gestire gli ordini in entrata e in uscita dal magazzino e garantire il corretto coordinamento fra tutte le operazioni.

Come viene mostrato in Figura 4.6, l'architettura software del magazzino è basata su quattro livelli.



Figura 4.6 - Software gestionali

Nel livello più alto dell'architettura troviamo il sistema ERP (Enterprise Resources Planning). Si tratta di un sistema di gestione che integra tutti i processi di business rilevanti di un'azienda (vendite, acquisti, gestione magazzino, contabilità, etc.). La logica che sta alla base di un qualsiasi sistema ERP è la raccolta centralizzata dei dati che provengono da ciascun reparto dell'azienda, i quali vengono organizzati in un repository strutturato. Tutte le informazioni sono così raccolte in un database condiviso, aggiornate in tempo reale, immediatamente

disponibili ed elaborabili nei diversi processi aziendali. Gli ERP si basano, in generale, su un'architettura Client-Server, in cui si crea una rete dove tutti i clients dell'azienda si collegano e fanno confluire le loro informazioni su un unico server, che interfaccia il database management systems. Il sistema ERP è quindi un sistema modulare dove ogni modulo rappresenta una funzione specifica dell'azienda, e allo stesso tempo aperto, in quanto deve essere anche in grado di interfacciarsi con sistemi diversi ed eterogenei tra di loro. In questo caso studio il software ERP verrà fornito e configurato da MBM. (Datalog, 2019)

Al terzo livello dell'infrastruttura troviamo il software WMS (Warehouse Management System). Si tratta di un software di gestione magazzino usato per controllare, coordinare e ottimizzare i movimenti, i processi e le fasi operative che si svolgono all'interno di un impianto, dalle funzioni più elementari (come il controllo delle scorte e delle giacenze), fino alle attività più complesse (gestione del ricevimento delle materie prime o sincronizzazione con i sistemi ERP). Inoltre ci consente di migliorare l'efficienza logistica incrementando le performance del magazzino e migliorando le attività degli operatori. Porta a un impiego ottimale delle risorse grazie all'automazione delle procedure ripetitive, alla pronta segnalazione degli errori, alla drastica riduzione dei supporti cartacei e all'utilizzo di efficaci strumenti di controllo e rilevazione statistica. Andando più nel dettaglio le funzionalità di questo software possono essere suddivise in quattro macroaree: gestione delle merci in entrata, allocazione delle merci, controllo e gestione dello stock e gestione delle merci in uscita. In ognuna di queste macroaree si effettuano determinate operazioni. Un'illustrazione di quanto appena descritto è visibile in Figura 4.7 (Mecalux, 2019a).

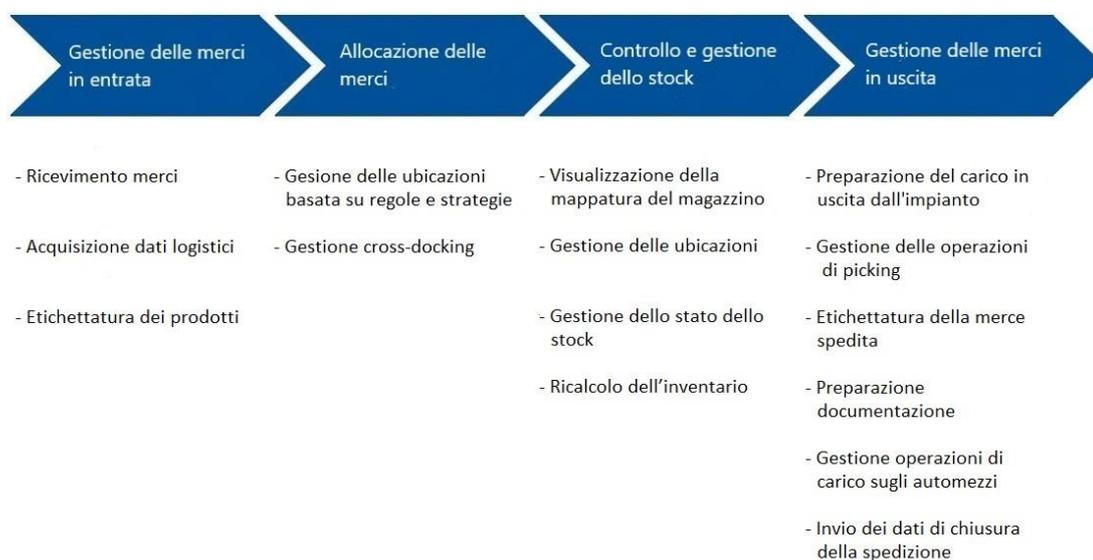


Figura 4.7 - Funzionalità WMS (Adattato da Mecalux, 2019b)

La prima macroarea è la gestione della merce in entrata. Durante il ricevimento merci si controlla scrupolosamente la conformità, la quantità, le caratteristiche e lo stato o la qualità della merce che entra nel magazzino e che, pertanto, diventa parte integrante dello stock. Le

informazioni raccolte vengono confrontate con l'ordine ricevuto e il WMS, dialogando con l'ERP, gestisce la fase operativa. Le operazioni che appartengono a questa macroarea sono:

- Ricevimento merci: gli articoli ricevuti possono essere pallettizzati o non pallettizzati. la differenza è che nel secondo caso prima di procedere allo stoccaggio è necessario effettuare il consolidamento e la registrazione degli attributi di logistica. In base alla tipologia di merce avviene lo stoccaggio all'interno del magazzino nelle apposite aree.
- Acquisizione dei dati logistici: la raccolta di informazioni, quali ad esempio il lotto di appartenenza del carico, la scadenza, il peso, la temperatura o il numero di serie consente di sapere con esattezza tutti i dati relativi alla consegna da parte di un fornitore. L'acquisizione di tali dati facilita la tracciabilità dello stock.
- Etichettatura dei prodotti: un WMS deve essere in grado di ricevere i dati inviati per mezzo della scansione dei codici a barre. In questo modo si riducono gli eventuali errori e i dubbi che potrebbero nascere durante la fase di movimentazione della merce e i processi diventano più performanti e aumenta la produttività. La documentazione relativa al ricevimento merci consente di generare report nei quali si indicano le differenze tra la merce prevista e quella effettivamente ricevuta, oltre ad ulteriori aspetti legati alla pianificazione temporale di queste operazioni.

La seconda macroarea riguarda l'allocazione delle merci. Qui si tengono in considerazione i processi di ubicazione, che gestiscono gli ordini di posizionamento della merce nel magazzino, stabilendo quale sia la migliore posizione per gli articoli in funzione di tipologia, caratteristiche, dimensioni. Inoltre il software WMS è fondamentale nel caso in cui la gestione del magazzino avvenga in modalità cross-docking. Esistono quindi due tipologie di gestione in base alla natura delle operazioni condotte all'interno del magazzino:

- Gestione delle ubicazioni basata su regole e strategie: la gestione della migliore ubicazione per ogni prodotto si basa su regole e strategie previamente pianificate che contribuiscono a ottimizzare sia le fasi operative sia quelle gestionali. A tale fine, il WMS gestisce gli ordini di posizionamento tenendo conto delle caratteristiche delle unità di carico e dei KPI di magazzino. Un esempio può essere: l'indice di rotazione, il tipo di contenitori industriali usati per il consolidamento della merce, le famiglie o tipologie di prodotti, la pericolosità o incompatibilità di alcuni prodotti, il packaging degli articoli o il loro volume.
- Gestione cross-docking: in un magazzino con questo tipo di gestione le merci vengono conservate per un periodo molto breve in attesa di transitare verso i clienti finali. Per far ciò viene applicata la tecnica del cross docking alla preparazione degli ordini, con cui si eliminano le tappe intermedie di stoccaggio delle merci. Se applicata correttamente, questa strategia permette di rendere più agile ed efficiente la fase distributiva, riducendo al contempo i costi operativi.

La terza macroarea riguarda il controllo e la gestione dello stock. Grazie a questa funzionalità è possibile ottenere in tempo reale un quadro relativo alla situazione delle scorte e delle giacenze. Conoscendo le quantità di articoli presenti in magazzino, non solo si preven

eventuali rotture di stock, ma si contribuisce a una migliore gestione amministrativa delle merci, in maniera più redditizia e scongiurando un pericolo di obsolescenza. Le operazioni che possono essere eseguite sono:

- Visualizzazione della mappatura del magazzino: permette di avere una visione dettagliata della posizione di tutti gli articoli all'interno del magazzino, indipendentemente che siano conservati su pallet o in contenitori.
- Gestione delle ubicazioni: consente di ottenere e modificare le informazioni relative al posizionamento, al tipo di ubicazione, di configurare i blocchi attivi, le dimensioni, le caratteristiche, le aree di stoccaggio a cui appartiene. In buona sostanza restituisce una mappatura intelligente dei beni in magazzino.
- Gestione dello stato dello stock: rende possibile un monitoraggio esaustivo dei dati relativi a quarantene, rotture, perdite, blocchi o riserve.
- Ricalcolo dell'inventario: fornisce feed specifici per ogni articolo, un'ubicazione o un'area specifica. Ciò consente di inviare comunicazioni al sistema ERP in caso di differenze o incongruenze di stock e tenere sempre sotto controllo l'inventario.

L'ultima macroarea è la gestione delle merci in uscita. I processi che la riguardano sono l'elaborazione degli ordini inviati ai clienti, i trasferimenti ad altri magazzini o i resi ai fornitori. Il WMS coordina tutti i processi legati a questi aspetti e in particolare si occupa della gestione delle seguenti operazioni:

- Preparazione del carico in uscita dall'impianto: il WMS ha il compito di raggruppare e assegnare la migliore posizione alle merci per gestire in maniera efficiente l'evasione degli ordini. Il software determina, ad esempio, in che punto delle baie di spedizione verrà lasciato il carico, quanti saranno gli operatori incaricati della preparazione, la maniera in cui verranno raggruppati gli ordini e la fascia oraria in cui avverrà tale operazione.
- Gestione delle operazioni di picking: il software stabilisce percorsi di picking, rendendo più performante le attività di selezione e prelievo, abbattendo tempi e numeri di movimentazioni, ma sempre seguendo i parametri specificati per l'ordine inviato dal sistema ERP.
- Etichettatura della merce che viene spedita: il programma svolge l'identificazione dei colli, utilizzando lo stesso processo visto nell'etichettatura della merce in entrata.
- Preparazione della documentazione relativa alle spedizioni: il software WMS crea la documentazione per i trasportatori e i report relativi alle discrepanze riscontrate.
- Gestione delle operazioni di carico delle spedizioni sugli automezzi: il software controlla la qualità della spedizione per ridurre gli errori.
- Invio dei dati al sistema ERP relativi alla chiusura della spedizione: si tratta di un passaggio fondamentale in quanto il responsabile della logistica riceve tutte le informazioni necessarie relative al numero di unità e di articoli da spedire, nonché al tipo di colli consegnati con ciascun ordine in uscita che viene evaso.

In questo caso studio come software WMS viene utilizzato EASYSTOR di INCAS, la quale si è occupata anche dell'installazione e della configurazione.

Facendo un confronto tra ERP e WMS emerge chiaramente che sono software completamente diversi sia per architettura che per funzionalità. Il primo è un sistema che comprende diversi moduli, mentre il secondo è un software *standalone* che non prevede l'integrazione di altri software per svolgere i propri compiti. Inoltre l'ERP è in grado di gestire trasversalmente tutti processi presenti in un'azienda, mentre il WMS si occupa di fornire una gestione approfondita solamente degli aspetti legati alla gestione del magazzino. Quindi l'ERP e il WMS non sono software sostituiti, ma complementari e sono entrambi fondamentali per una completa gestione di un magazzino. Per questo motivo spesso il WMS viene configurato come modulo extra volto ad arricchire le funzionalità di un ERP. Il colloquio tra ERP e WMS è di primaria importanza poiché tutte le transazioni di magazzino devono avere un riscontro fiscale ed un ritorno numerico sul sistema gestionale. Dall'ERP inoltre provengono le informazioni che permetteranno di pianificare e razionalizzare le attività logisticamente controllate ed automatizzate tramite il WMS. Lo scambio dati tra i due sistemi avviene attraverso la condivisione di tabelle di DB che vanno scritte e lette da entrambi i sistemi. La Figura 4.8 mostra il ruolo dei due software dalla ricezione di un ordine all'elaborazione dei pagamenti (Mecalux).

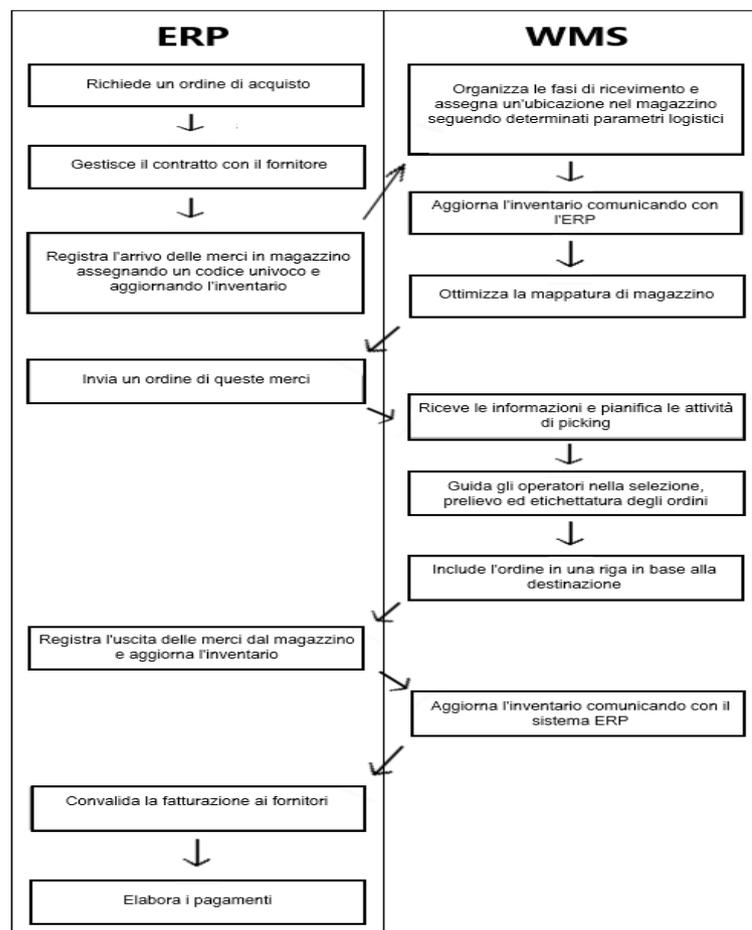


Figura 4.8 - Flusso informatico dalla ricezione alla fatturazione di un ordine

Al livello due c'è il WCS (Warehouse Control System). Questo software a differenza di quelli già trattati è in grado di controllare tutti gli elementi automatici presenti in un magazzino e quindi la sua presenza diventa fondamentale per applicare tutte le programmazioni e disposizioni che provengono dal WMS (Figura 4.9). Grazie al WCS è possibile:

- dirigere simultaneamente tutti i flussi di attività all'interno del magazzino calcolando quelli più efficienti per poi trasmettere i comandi ai controller delle apparecchiature per applicare i risultati desiderati;
- avere un'interfaccia integrata per la gestione dei sottosistemi di movimentazione dei materiali come i nastri trasportatori;
- raccogliere dati in tempo reale da tutti i sistemi collegati;
- monitorare le operazioni per rilevare rapidamente qualsiasi problema fornendo agli utenti un prompt immediato per risolvere il problema.

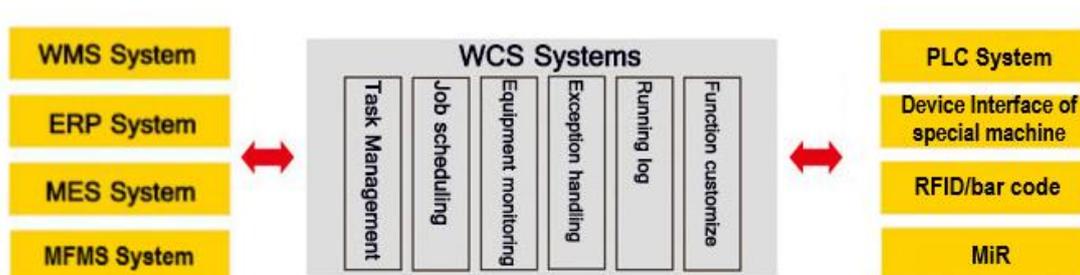


Figura 4.9 - Ruolo WCS (Adattato da Zdintegration)

In questo caso studio come software WCS viene utilizzato Easylogic di INCAS, la quale si è occupata anche dell'installazione e della configurazione. Easylogic riceve i comandi (missioni) da Easystor e si occupa di tradurli in un formato compatibile con la logica PLC, selezionando il percorso in base alla situazione dell'impianto e verificandone l'attivazione in base alle regole di movimentazione legate alla fisicità. Easylogic si occupa anche di informare Easystor riguardo lo stato di avanzamento delle missioni e la presenza di eventuali situazioni anomale. Il monitoraggio grafico del magazzino visualizza lo stato di funzionamento dei singoli macchinari, in modo da permettere agli operatori di riconoscere situazioni anomale evidenziate dalla grafica e prendere le adeguate misure correttive interagendo direttamente con il sistema tramite l'utilizzo del sinottico interattivo. Un'apposita interfaccia per ogni singolo dispositivo fisico permetterà all'utente di comandare manualmente l'impianto da questo livello, consentendogli di risolvere eventuali situazioni anomale senza la necessità di intervenire sul sistema Easystor.

Al secondo livello dell'architettura, oltre a Easylogic, si trova anche il Mir Fleet Management System (MFMS): un software fornito e configurato da TMP che si occupa della gestione delle missioni che devono essere svolte dai Mir e dell'interfaccia con la rulliera motorizzata. Per

quanto riguarda la gestione delle missioni è necessario che le tipologie di missioni siano preinstallate nel sistema delle navette, quindi sono state definite a priori tutte le tipologie di missioni che le varie navette possono svolgere. Dopo queste operazioni preliminare sono disponibili tutta una serie di funzionalità per la gestione delle missioni: la creazione di una nuova missione, l’inserimento di un nuovo punto su una mappa già esistente, la definizione delle modalità e del posizionamento del punto di ricarica e la possibilità di modificare la parametrizzazione della navetta entro un certo range di valori (velocità, accelerazione). Per quanto riguarda il corretto interfacciamento con la rulliera motorizzata il software esegue due tipologie di controlli: verifica che la postazione del Mir sia libera per accogliere un contenitore e che i rulli siano in marcia per poter completare il trasferimento. Al termine del trasferimento del contenitore il software invia un segnale di trasferimento completato alla rulliera.

Il livello uno è quello più basso dell’architettura software e qui si trovano i PLC (Programmable Logic Controller). Quest’ultimi sono dei computer specializzati che hanno il compito di controllare i macchinari automatici agendo in base ai comandi ricevuti dal WCS. Per far ciò i PLC eseguono dei programmi che elaborano i segnali digitali e analogici ricevuti dai sensori per poi trasmetterli agli attuatori installati sui macchinari.

4.2.2 Impianto di rete

Dopo aver descritto approfonditamente l’infrastruttura software impiegata nel magazzino oggetto di studio, di seguito viene presentata la descrizione dell’infrastruttura hardware, la cui struttura logica è mostrata in Figura 4.10.

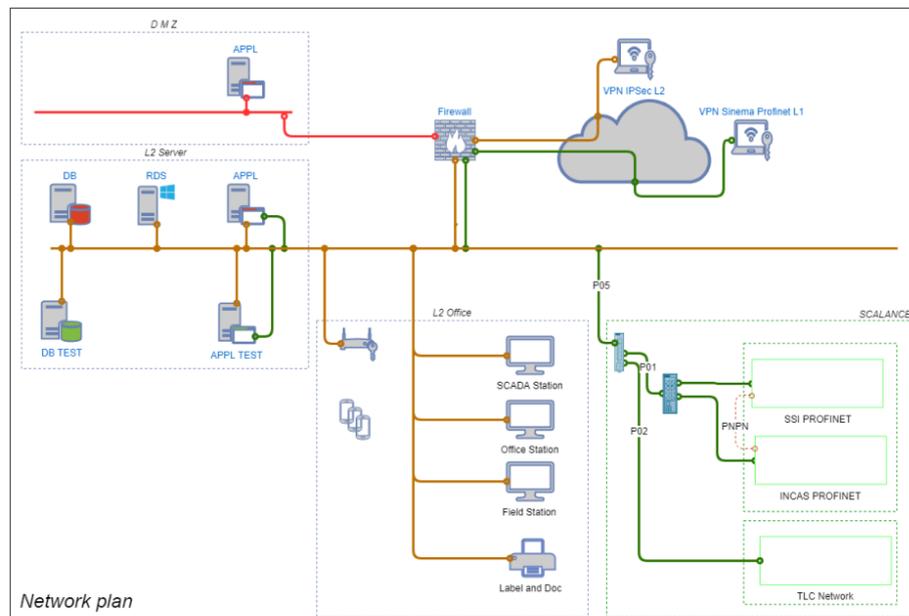


Figura 4.10 - Struttura logica INCAS

Come è possibile vedere dall'immagine l'infrastruttura di rete implementata nel magazzino è composta da sei elementi principali: una zona demilitarizzata (DMZ), il firewall, le VPN, la rete PROFINET, i server degli applicativi di livello due e i client degli applicativi di livello due. Prima di iniziare la trattazione di ogni singolo elemento viene precisato che verranno maggiormente approfonditi gli aspetti legati agli applicativi di livello due sia lato Server che lato Client.

- La zona demilitarizzata è una sottorete che si trova tra la rete interna e la rete esterna e il cui scopo è quello di proteggere la rete interna da eventuali attacchi alla sicurezza. Al suo interno sono consentite le connessioni dalla rete interna ed esterna, mentre le connessioni dalla DMZ sono consentite solo alla rete esterna. Ciò consente agli host della DMZ di fornire servizi alla rete esterna proteggendo la rete interna nel caso in cui intrusi compromettano un host nella DMZ. Per qualcuno sulla rete esterna che vuole connettersi illegalmente alla rete interna, la DMZ è un vicolo cieco (USA CISA);
- un firewall è un sistema, che può essere sia un elemento hardware che un'applicazione software, per la sicurezza della rete che permette di monitorare il traffico in entrata e in uscita utilizzando una serie predefinita di regole di sicurezza per consentire o bloccare gli eventi. La maggior parte dei firewall dispone di norme standard a cui l'utente finale può aggiungere altre personalizzate, in base alle proprie necessità. Il firewall si interpone tra la rete esterna, che comprende Internet, e la rete interna. La rete interna è considerata conosciuta, sicura, attendibile e protetta, mentre quella esterna è la presunta fonte di potenziali minacce, in quanto nel complesso è sconosciuta, insicura e non attendibile (Pandasecurity, 2019);
- una VPN (Virtual Private Network) è una rete privata virtuale che garantisce privacy, anonimato e sicurezza attraverso un canale di comunicazione logicamente riservato (tunnel VPN) e creato sopra un'infrastruttura di rete pubblica. Il termine virtuale sta a significare che tutti i dispositivi appartenenti alla rete non devono essere necessariamente collegati ad una stessa LAN locale ma possono essere dislocati in qualsiasi punto geografico del mondo. In questo modo viene realizzata una LAN "virtuale" e "privata" ma del tutto equivalente ad un'infrastruttura fisica di rete dedicata. Nell'infrastruttura di rete in questione vengono utilizzate due diverse reti VPN: una che si occupa di estendere la LAN interna del magazzino e che utilizza il protocollo di comunicazione IPsec L2; l'altra che invece viene utilizzata per l'assistenza remota dei macchinari all'interno della rete PROFINET attraverso l'utilizzo dell'applicativo Sinema (Cybersecurity360);
- la rete profinet è la rete locale che collega tutti i macchinari attraverso il protocollo Ethernet/Profinet.

Per quanto riguarda gli applicativi di livello due sono stati implementati due ambienti: quello di produzione e quello di test. La Figura 4.11 mostra l'architettura dei due ambienti.

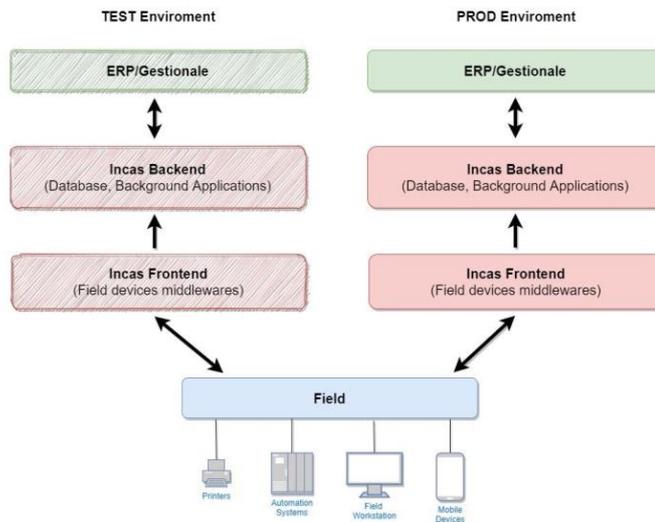


Figura 4.11 - Ambienti

Sia per l'ambiente di test che per quello di produzione sono presenti 4 livelli: l'ERP, il backend, il frontend e il livello campo.

Per quanto riguarda l'ERP abbiamo già affrontato la sua trattazione precedentemente.

All'interno del livello backend sono inclusi i database server e tutti gli applicativi automatici (servizi) che svolgono attività in background.

- I database su cui si appoggiano gli applicativi sono di tipo relazionale. In Tabella 4.1 è esposta la configurazione hardware del server in cui è installato il database e in Tabella 4.2 i suoi parametri di configurazione;
- gli applicativi automatici sono una serie di servizi Windows che svolgono attività automatiche in modalità *unattended*, cioè senza richiedere l'operatività costante di un operatore. Questi servizi a seconda della loro mansione potranno aprire connessioni dirette verso il database, i dispositivi di campo o entrambi. Dal momento che tutti gli applicativi di questo livello sono stati scritti con tecnologia Microsoft i sistemi operativi utilizzati in questo livello sono tutti Microsoft Server.

INFORMATION	DESCRIPTION
PROCESSOR	2 virtual CPU
MEMORY	8 GB
DISK	50 GB di spazio libero per dati e backup
OS	Microsoft Windows Server 2016 o superiore
RDBMS	Microsoft SQL Server 2016 Express o superiore
RUNTIME	Microsoft .NET framework 3.51

Tabella 4.1 - Hardware del server che ospita il database

DBMS	PARAMETERS
Microsoft SQL Server	Collation: SQL_Latin1_General_CP1_CI_AS Authentication: Mixed Mode Recovery model: FULL READ_COMMITTED_SNAPSHOT: ON
Oracle RDBMS	Istanza in ARCHIVELOG mode TABLESPACE/DATAFILE dedicati agli schemi Incas

Tabella 4.2 - Parametri di configurazione del database

Il livello frontend include tutti i middlewares che gestiscono la comunicazione da e verso i dispositivi di campo. Sono presenti diversi middlewares e l'utilizzo di uno rispetto a un altro dipende dal dispositivo che richiede la comunicazione.

- Per quanto riguarda gli applicativi sulle postazioni di campo, per esempio i PC, viene utilizzata la tecnologia Microsoft Remote Desktop Services che permette la comunicazione attraverso il protocollo Microsoft RDP. In Tabella 4.3 viene mostrata la configurazione hardware di questo server;
- quando l'applicativo è un terminale mobile con sistema operativo Android che richiede di comunicare tramite wifi, il middleware utilizzato è un server dedicato, chiamato "Android Radio Server", su cui gireranno i servizi dedicati a processare i dati che vengono inviati dall'App installata sul terminale. Il sistema operativo minimo dei terminali dovrà essere Android 5.0 ("Lollipop"), mentre l'ultima versione certificata è Android 10;
- infine nel caso in cui a richiedere la comunicazione sia sempre un dispositivo mobile con sistema operativo Android, ma che si collega tramite connessione LTE il middleware che si occupa di gestire tale circostanza viene connesso alla DMZ della rete.

INFORMATION	DESCRIPTION
PROCESSOR	2 virtual CPU
MEMORY	8 GB
DISK	50 GB
OS	Microsoft Windows Server 2016 o superiore – server in dominio per l'abilitazione dei ruoli RDS
RUNTIME	Microsoft .NET framework 3.51

Tabella 4.3 - Hardware del server che ospita gli applicativi

Il livello campo contiene tutti i dispositivi che vengono installati sul campo. Possono essere dati in dotazione ad un operatore oppure essere automatici. Su questo livello è presente la postazione multi-controllo, gli applicativi di livello uno e le stampanti.

- La postazione multi-controllo è una postazione operativa posizionata direttamente sul campo e che serve al gestore dell'impianto per avere un quadro completo sul suo funzionamento. Normalmente è dotata di tre monitor su cui vengono attivati i seguenti moduli: il cruscotto SCADA, il cruscotto sinottico WCS e le telecamere di visualizzazione dell'impianto;
- gli applicativi di livello uno sono quegli applicativi in esecuzione sui dispositivi PLC e direttamente connessi alla movimentazione di campo di cui è avvenuta la trattazione nel capitolo sull'infrastruttura software;
- le stampanti che vengono utilizzate sono di due tipi: document e label. Le stampanti di tipo document vengono utilizzate per la stampa di documenti e sono installate sulle macchine all'interno del livello frontend e del livello applicativo, in quest'ultimo sono necessarie per permettere la produzione di stampe automatiche. Le stampanti di tipo label invece vengono utilizzate per la stampa delle etichette. Quest'ultime vengono normalmente generate in modo automatico dagli applicativi a fronte delle varie necessità. Tutte le stampanti sono raggiungibili direttamente via rete e sono installate anche sul livello applicativo.

4.2.3 Networking

In questo paragrafo verranno descritte le tipologie di rete di comunicazione utilizzate con i rispettivi protocolli e le metodologie di autenticazione all'interno delle reti. La prima tipologia di rete che viene analizzata è quella cablata. La comunicazione tra i vari moduli e i vari dispositivi avviene tramite protocollo TCP/IP e per il corretto funzionamento del sistema le performance di rete devono rientrare entro i parametri definiti in Tabella 4.4.

CONNESSIONE	PARAMETRO
Latenza tra applicativi di livello 2 e di livello 1	<16 ms
Banda occupata da una sessione RDP	30 kbit/s

Tabella 4.4 - Prestazioni di rete cablata minime del sistema

Per una gestione più flessibile ed efficace l'indirizzamento avviene in maniera diversa in base al tipo di dispositivo. Per le postazioni di campo è stato scelto un indirizzamento dinamico, mentre per le stampanti, gli access point e i PLC si è preferito un indirizzamento statico. La seconda tipologia di rete è quella wifi. In questo caso l'aspetto fondamentale è quello di

garantire copertura costante su tutta l'area di utilizzo dei dispositivi e la continuità del servizio di connettività con segnale non al di sotto dei -70 dB e con una percentuale di perdita dei pacchetti mai maggiore dell'1 %. In Tabella 4.5 vengono mostrati i parametri che sono stati presi in considerazione per l'implementazione di questo tipo di rete.

CONNESSIONE	PARAMETRO
Latenza tra applicativi di livello 2 e terminali	<100 ms (consigliata <50 ms)
Banda occupata da una sessione RDP	30 kbit/s
Banda occupata da una sessione radio Android	30 kbit/s
Banda occupata da una sessione radio VT100	20 kbit/s

Tabella 4.5 - Prestazioni di rete wifi minime del sistema

La rete wifi opera su due diverse bande di frequenza: quella a 2,4 GHz e quella a 5 GHz. Sia per una che per l'altra sono stati utilizzati i canali 1, 6 e 11 con una larghezza di banda di 20 MHz. Anche in questo caso l'indirizzamento avviene in maniera diversa in base al tipo di dispositivo. Se si tratta di dispositivi mobili si utilizza un indirizzamento dinamico; nel caso di stampanti indirizzamento statico. L'ultima tipologia di rete utilizzata è la rete LTE, la quale non necessita di una configurazione ad hoc e l'unico parametro che deve soddisfare è quello di dover garantire una banda di 20kbit/s che corrisponde alla banda occupata da una sessione radio Android GPRS.

Per quanto riguarda le metodologie di autenticazione, tutti gli applicativi di livello due prevedono un database di security interno. I vari moduli si possono collegare al database in due modalità: con un account database standard oppure con l'utente di dominio che esegue l'applicativo. All'interno del database sono memorizzate le utenze applicative e il loro livello di autorizzazioni. Gli utenti dell'applicazione possono essere dichiarati di tipo "applicativo" oppure di "dominio". Ogni dispositivo PC o terminale radiofrequenza possiede un utente Windows dedicato.

4.3 Modello di simulazione

In questo paragrafo verrà svolto un approfondimento sul modello di simulazione del magazzino automatico. Inizialmente verrà fornita una spiegazione generica su cos'è un modello di simulazione e sui vari tipi di simulazione. In seguito si entrerà nello specifico del modello descrivendo le logiche di funzionamento e gli scopi per cui è utilizzato.

Un modello di simulazione è la rappresentazione virtuale degli elementi e dei processi che costituiscono un sistema reale. Al suo interno inoltre sono presenti tutti i meccanismi che descrivono i cambiamenti del sistema nel tempo e che permettono al modello di essere eseguito, generando così delle simulazioni. La creazione di un modello si rivela estremamente utile in fase di design prima dell'implementazione di un nuovo sistema con lo scopo di

prevenire, per esempio, un errato dimensionamento delle risorse impiegate o per valutare la presenza di eventuali colli di bottiglia. Un modello di simulazione risulta importante anche per risolvere dei problemi in un sistema già esistente, soprattutto quando l'esecuzione di esperimenti direttamente su di esso risulta essere troppo costosa o praticamente impossibile.

Per creare un modello di simulazione possono essere seguiti diverse tipologie di approcci, ognuno del quale rappresenta un paradigma. I principali sono: la dinamica dei sistemi, la simulazione a eventi discreti e la simulazione basata su agenti. La dinamica dei sistemi è un paradigma di alto livello utilizzato per sviluppare modelli che mirano a esemplificare le prestazioni dinamiche dei sistemi continuativamente nel tempo. Viene maggiormente impiegato nella risoluzione di problemi dinamici in ambito sociale ed economico. La simulazione ad eventi discreti è un paradigma di livello medio-basso basato sul concetto che nel sistema non avviene alcun cambio di stato tra due eventi consecutivi. I modelli basati su questo paradigma trovano applicazione principalmente negli impianti produttivi e nelle reti di distribuzione. La simulazione basata su agenti è un paradigma adatto a modellare ad ogni livello di astrazione in cui il sistema viene replicato attraverso l'interazione e le azioni di un certo numero di agenti. Questo paradigma trova applicazione in quei sistemi costituiti da un insieme di elementi individuali come una popolazione o il traffico stradale.

Il modello di simulazione del magazzino è stato creato attraverso il software di simulazione AnyLogic. Questo applicativo è stato sviluppato da The AnyLogic Company e permette di realizzare modelli basati sui principali paradigmi di simulazione. Offre una serie di librerie e di strumenti specifici che permettono di costruire modelli di sistemi appartenenti a numerosi settori, tra cui quello della produzione, dei trasporti, sanitario, ferroviario, del traffico e dei processi sociali. Per lo sviluppo del modello sono state usate la Process modelling library e la Material handling library. Il modello possiede una struttura gerarchica in cui al livello più alto si trova l'agente Main che incorpora al suo interno i seguenti dieci agenti: Component, Product, Tote, Box, AGV, Worker, Task, Arm, Crane e Order. Per quanto riguarda le logiche di funzionamento, in Figura 4.12 è mostrato il diagramma della attività, scritto con il linguaggio di alto livello UML (Unified Modeling Language), che descrive le modalità con cui il modello gestisce i processi che avvengono nel magazzino.

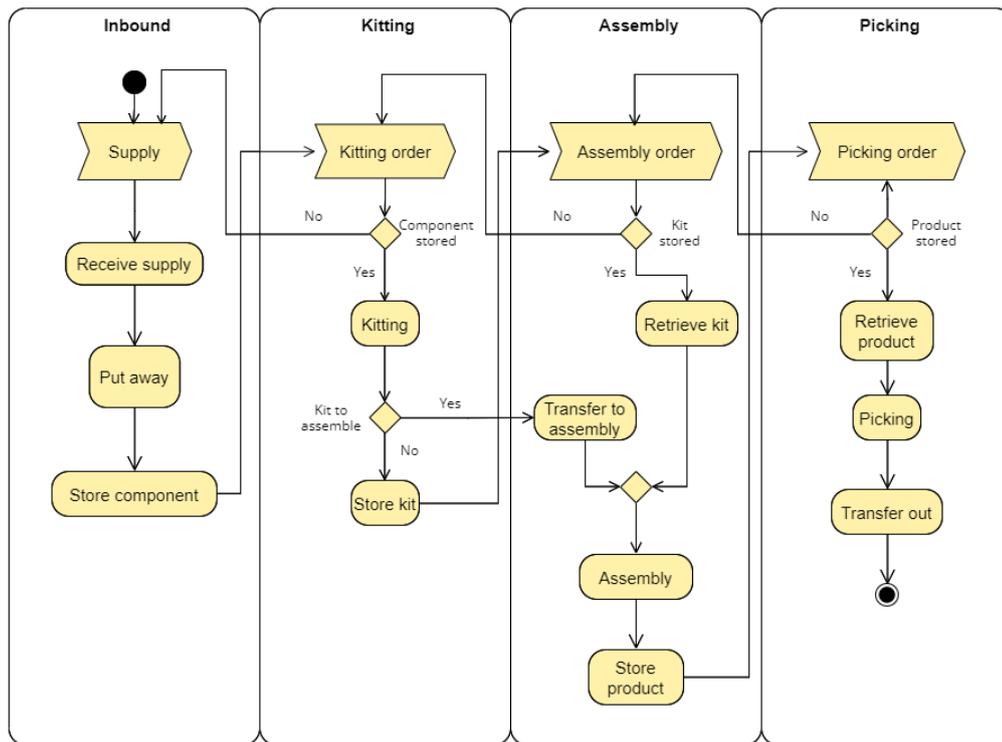


Figura 4.12 – Processi del modello di simulazione del magazzino

Il diagramma è costituito da quattro processi. In ordine di esecuzione sono: l'arrivo delle merci, il kitting, l'assemblaggio e il picking. Il primo processo inizia con l'emissione di un nuovo ordine di fornitura (nodo segnale "Supply") e si completa con i tre nodi azione "Receive supply", "Put-away", e "Store comp." che rappresentano rispettivamente la ricezione della merce, l'estrazione degli articoli dagli imballaggi e lo stoccaggio in magazzino. Il secondo processo, il kitting, inizia con la ricezione di un ordine di kitting (nodo segnale "Kitting order"), quindi, se i componenti necessari a completare l'ordine sono presenti in magazzino, viene eseguito il kitting (nodo azione "Kitting"), altrimenti si emette un ordine di fornitura dei componenti mancanti; al termine di questo passaggio il kit, se dovrà essere assemblato immediatamente, viene trasferito alla stazione di assemblaggio, altrimenti viene immagazzinato (nodo azione "Store kit"). Il terzo processo, l'assemblaggio, inizia con la ricezione di un ordine di assemblaggio (nodo segnale "Assembly order"), quindi, se il kit necessario a completare l'ordine è presente in magazzino, viene prelevato (nodo azione "Retrieve it"), altrimenti si emette un ordine di kitting; quando il kit è pronto viene eseguito l'assemblaggio (nodo azione "Assembly") e il processo si completa con lo stoccaggio del prodotto finito in magazzino (nodo azione "Store product"). Il quarto e ultimo processo, il picking, inizia con la ricezione di un ordine di picking (nodo segnale "Picking order"), quindi, se il prodotto finito necessario a completare l'ordine è presente in magazzino, viene prelevato (nodo azione "Retrieve product"), altrimenti si emette un ordine di assemblaggio; quando il prodotto finito è disponibile viene eseguito il picking (nodo azione "Picking") e il processo si completa con il trasferimento del prodotto finito (nodo azione "Transfer out").

Dato che il magazzino automatico non era ancora stato installato, si è deciso di sviluppare questo modello di simulazione per identificare il design ottimale con cui implementare il magazzino e per valutare le performance del sistema. A tali scopi sono stati svolti quattro esperimenti: il primo è stato eseguito per valutare le performance del sistema con il layout standard visibile in Figura 4.13; il secondo avvia delle simulazioni facendo variare solamente il tassi di domanda, in modo da comprendere qual è il rendimento del magazzino in diverse situazioni di domanda; il terzo prevede alcuni cambiamenti di layout degli elementi esterni all'area automatizzata, cioè la stazione di ricevimento delle forniture, la stazione dei MiR, la stazione di assemblaggio e i vari buffer; nell'ultimo si fanno variare le priorità dei processi per valutare la reazione del sistema a tali cambiamenti. Al termine di questi esperimenti è stato identificato come collo di bottiglia del sistema la stazione di assemblaggio. La stazione di kitting, la stazione di ricevimento delle forniture, il trasloelevatore e i MiR hanno presentato un basso livello di utilizzo, il che suggerisce che non sono elementi critici.

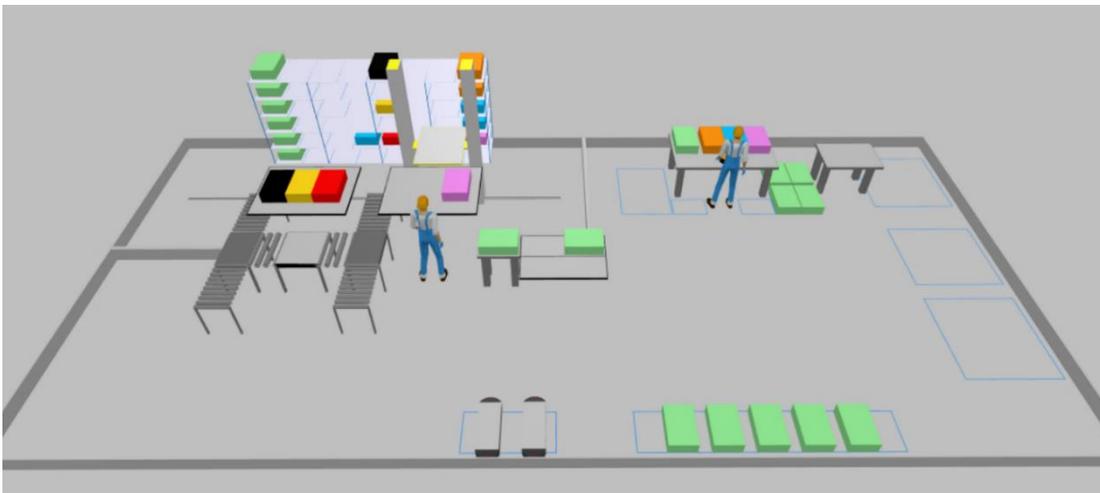


Figura 4.13 – Layout del modello di simulazione in 3D

4.4 Modello UML della Digital Shadow

In questo paragrafo verrà analizzato il modello concettuale che illustra il funzionamento della Digital Shadow del magazzino automatico. Questo modello è stato creato attraverso il linguaggio UML, più precisamente un diagramma delle classi (Figura 4.14). In questo tipo di diagramma ogni classe, rappresentata da un rettangolo, identifica un'entità e ogni collegamento identifica una relazione. Una classe è costituita da un nome, da una serie di attributi e dalle funzioni che svolge. In questo modello sono presenti due tipi di relazioni: le relazioni di composizione e le relazioni semplici. Le prime descrivono un'entità come parte o composizione di un'altra, le seconde invece definiscono il ruolo di un'entità rispetto a un'altra. Ogni relazione è inoltre corredata dalle cardinalità. Quest'ultima indica il numero di istanze di una classe che possono essere associate ad una singola istanza dell'altra classe.

Un dettaglio molto importante del modello è il colore delle classi. Quelle gialle appartengono allo spazio fisico, si riferiscono quindi a entità che hanno una collocazione fisica all'interno del laboratorio. Le classi verdi fanno parte dell'infrastruttura IT, la cui funzione è fare da ponte tra lo spazio fisico e quello virtuale. Le classi azzurre si riferiscono a tutte le entità che si collocano nello spazio virtuale.

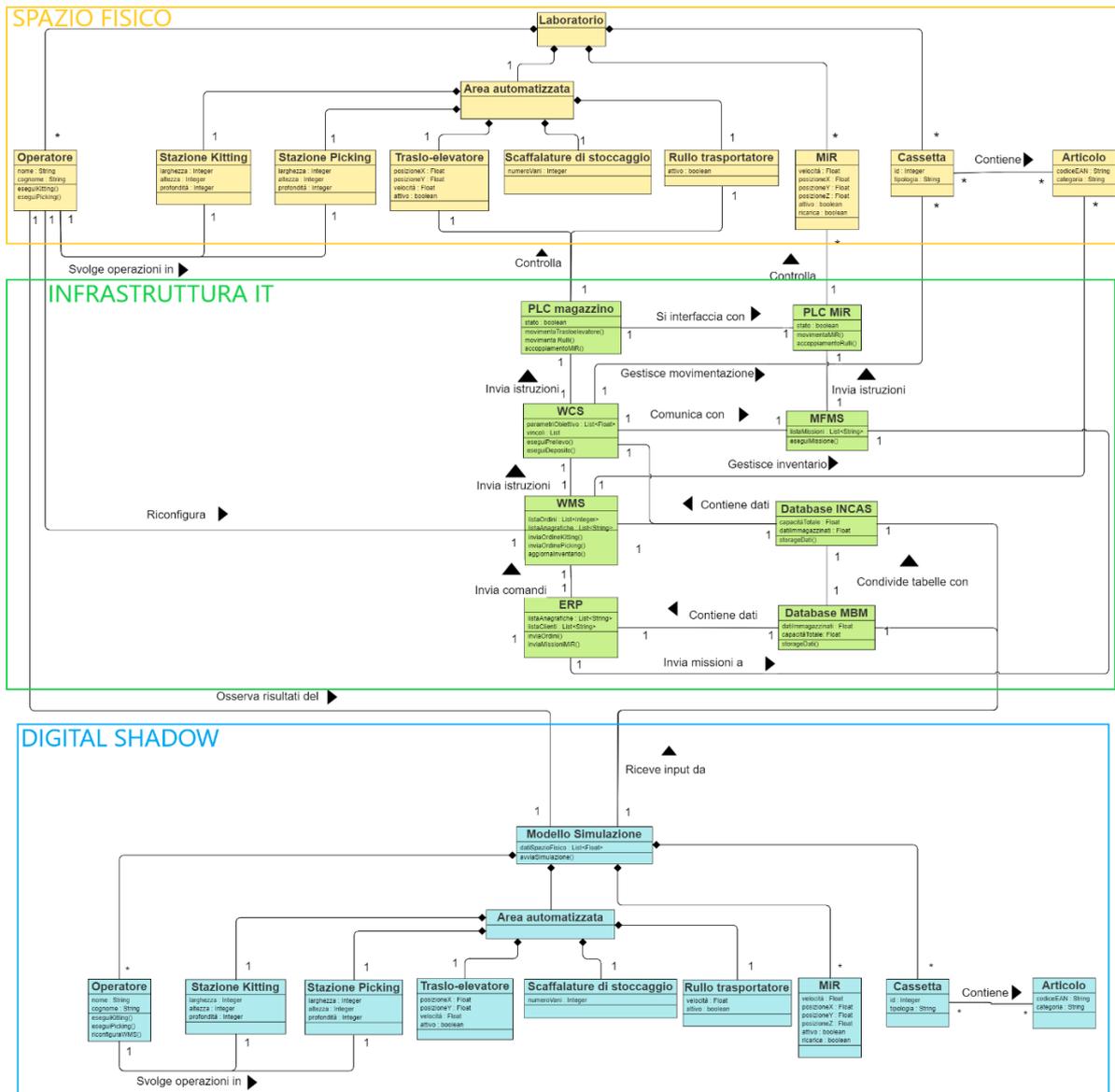


Figura 4.14 – Diagramma delle classi della Digital Shadow

La caratteristica fondamentale di una Digital Shadow è la connessione unidirezionale tra lo spazio virtuale e lo spazio fisico. Questa proprietà è rappresentata nel modello attraverso delle relazioni fondamentali. La relazione che collega i due database, quello di INCAS e quello MBM, al modello di simulazione rappresenta la capacità del modello di simulazione di funzionare utilizzando dei dati prelevati in maniera automatica dai database. Le relazioni che

collegano l'operatore al modello di simulazione e alla WMS invece rappresentano l'incapacità del modello di applicare in automatico i risultati ottenuti con le simulazioni e la necessità di far intervenire manualmente un operatore.

4.5 Modello UML Digital Twin

In questo paragrafo viene preso in esame il diagramma delle classi (Figura 4.15) che rappresenta l'architettura Digital Twin del magazzino. Facendo un paragone col diagramma delle classi della Digital Shadow, si può notare come i due diagrammi siano molto simili tra loro. Lo spazio fisico e l'infrastruttura IT sono identici nei due modelli, a cambiare sono lo spazio virtuale e le relazioni dell'entità Operatore. Lo spazio virtuale è caratterizzato dalla presenza, oltre del modello di simulazione, anche dell'Intelligenza Artificiale. Quest'ultima è connessa al modello di simulazione e offre la possibilità di svolgere delle ottimizzazioni sul sistema replicato attraverso la definizione di nuove policy. Per far ciò è necessaria un'enorme quantità di dati da utilizzare come input per gli algoritmi di machine learning. Dal momento che il sistema replicato è di piccole dimensioni, risulta eccessivamente oneroso in termini di tempo generare con esso una quantità di dati sufficiente. Per superare questo limite può essere utilizzato come fonte di dati il modello di simulazione che permette di eseguire un elevato numero di simulazioni in tempi ridotti. Prima di applicare nel sistema reale le nuove policy definite dall'Intelligenza Artificiale, il modello esegue una simulazione di validazione per confrontare il comportamento del sistema con le vecchie policy. Nel caso in cui è riscontrato il miglioramento atteso, si applicano automaticamente le nuove policy senza la necessità di un intervento da parte dell'operatore. In questa nuova architettura l'operatore non ha più il compito di riconfigurare i processi del sistema reale e si occupa solamente di eseguire il picking e il kitting nelle relative stazioni. Il confronto tra i due modelli evidenzia le differenze fondamentali tra un sistema Digital Shadow e uno Digital Twin, ovvero la presenza dell'intelligenza artificiale e la connessione bidirezionale tra lo spazio fisico e quello virtuale.

5. Esempio applicativo della Digital Shadow

In questo capitolo verrà presentato un esempio applicativo della Digital Shadow, spiegando dettagliatamente tutti i passaggi necessari per la sua implementazione. Lo scopo di questo esempio applicativo è dare evidenza della comunicazione monodirezionale, automatica e in tempo reale tra il magazzino fisico e il modello di simulazione, caratteristica fondamentale che contraddistingue una Digital Shadow da un semplice modello di simulazione.

Prima di addentrarsi nella trattazione del caso applicativo è importante precisare che per esigenze tecniche non sono stati utilizzati i sistemi già presenti, ma sono stati creati da zero un database e un modello di simulazione di prova.

5.1 Modello di simulazione di prova

In questo paragrafo verrà svolto un approfondimento sul modello virtuale di prova oggetto di questo caso studio. Inizialmente verranno presentate le operazioni svolte dal modello e successivamente verranno analizzati nel dettaglio tutti i passaggi compiuti per la sua implementazione.

5.1.1 Funzionamento del modello

Il modello di prova è stato creato unicamente con lo scopo di dare evidenza del corretto caricamento delle informazioni prelevate dal database e dell'utilizzo di quelle più aggiornate ogni volta che si esegue una nuova simulazione.

In questo nuovo modello vengono generati a intervalli regolari nuovi agenti che seguono un semplice percorso al termine del quale vengono poi eliminati. In questa operazione l'aspetto più importante e significativo è la generazione di nuovi agenti perché è in questo passaggio che il modello preleva le informazioni dal database e le utilizza per caratterizzare ogni nuovo agente. Per verificare che l'interazione col database sia avvenuta nella maniera corretta, sono state utilizzate delle variabili che durante la simulazione vengono valorizzate con le caratteristiche dell'ultimo agente generato (Figura 5.1).



Figura 5.1 - Verifica del caricamento dei dati durante la simulazione

Se le informazioni sull'agente mostrate dalle variabili corrispondono a quelle presenti nel database significa che sono state prelevate correttamente. Nel caso in cui durante una simulazione venissero aggiunte nuove istanze in una tabella del database o ne venisse modificata o eliminata una già esistente, il modello utilizzerrebbe le nuove informazioni aggiornate nella simulazione successiva e non in quella in corso.

Il modello appena descritto è inutile da un punto di vista operativo, però è la base da cui partire per creare un modello in grado di comunicare in tempo reale col proprio corrispettivo fisico e quindi in grado di costituire un sistema Digital Twin.

5.1.2 Creazione del modello

Il modello di prova è stato creato servendosi del software di simulazione Anylogic utilizzando come paradigma la simulazione basata su agenti. Per la costruzione del modello si è fatto uso della libreria Process Modeling Library. Grazie a questa libreria è possibile modellare i sistemi del mondo reale in termini di agenti, processi e risorse attraverso l'utilizzo di blocchi e di Space Markup Models. Un'altra caratteristica importante della Process Modeling Library è la capacità di creare animazioni molto sofisticate dei modelli di processo. In Tabella 5.1 è possibile consultare l'elenco di tutti gli elementi della Process Modeling Library utilizzati per la creazione del modello.

ICONA	NOME	DESCRIZIONE
	Percorso	Questo elemento serve a unire due o più nodi definendo il percorso che seguirà l'agente.
	Nodo Rettangolare	Un nodo è lo spazio in cui risiedono gli agenti.
	Source	Attraverso questo elemento possono essere generati nuovi agenti.
	Sink	Attraverso questo elemento possono essere eliminati gli agenti.
	Delay	Attraverso questo elemento è possibile far attendere un agente in un nodo per un determinato intervallo di tempo.
	Queue	Attraverso questo elemento è possibile mettere in coda gli agenti.
	Parametro	I parametri sono generalmente delle costanti che rappresentano le caratteristiche particolari dell'agente a cui si riferiscono. Diventano fondamentali nella descrizione di diverse istanze di uno stesso agente che differiscono proprio per il valore assunto dal parametro.
	Variabile	Le variabili possono cambiare durante la simulazione. Sono usate per modellare alcune caratteristiche dinamiche degli oggetti o per memorizzare i risultati della simulazione.
	Funzione	Le funzioni sono utili quando è necessario eseguire qualche operazione che è difficile da modellare direttamente con gli oggetti e gli elementi incorporati nel software. Sono in grado di restituire come risultato dei valori specifici eseguendo il codice definito al loro interno. Sono scritte in Java ed è quindi possibile sfruttare tutte le peculiarità di questo linguaggio.

Tabella 5.1 - Elementi della Process Modeling Library utilizzati nel caso studio

Il modello è stato costruito servendosi di cinque agenti:

1. Articolo;
2. Documento;
3. Ordine;
4. Cassetta;
5. Main.

L'agente Articolo, illustrato in Figura 5.2, contiene quattordici parametri e una funzione. Tutti i parametri sono stati configurati in modo da avere nome e tipo identici a quelli delle colonne presenti nella tabella Articolo del database. Tutte le altre proprietà dei parametri non sono state modificate. La funzione, denominata "toString", è scritta in linguaggio Java ed è utilizzata per definire una stringa composta dai valori che ogni parametro assume dinamicamente dal database separati da un ";". Di seguito un esempio:

"idCodice =1; codiceEan =FRE001; estensione =X; descrizione =Leva freno; categoriaMerceologica =mezzo trasporto; lunghezza =11; larghezza =2; altezza =7; peso =100; calcoloVolumetrico =X; pezziPerConfezione =6; dataScadenza =2022-05-01; dataInserimento =2021-05-01; tag =F"

La Figura 5.3 mostra la configurazione della funzione. Per completare la configurazione dell'agente è necessario definire un costruttore dentro la scheda "Advance Java" tra le proprietà dell'agente. Il costruttore è una funzione Java, mostrata in Figura 5.4, che ha il compito di creare l'oggetto Articolo partendo in questo caso dalle informazioni ricevute dal database. Questa funzione sarà richiamata all'interno dell'agente Main.

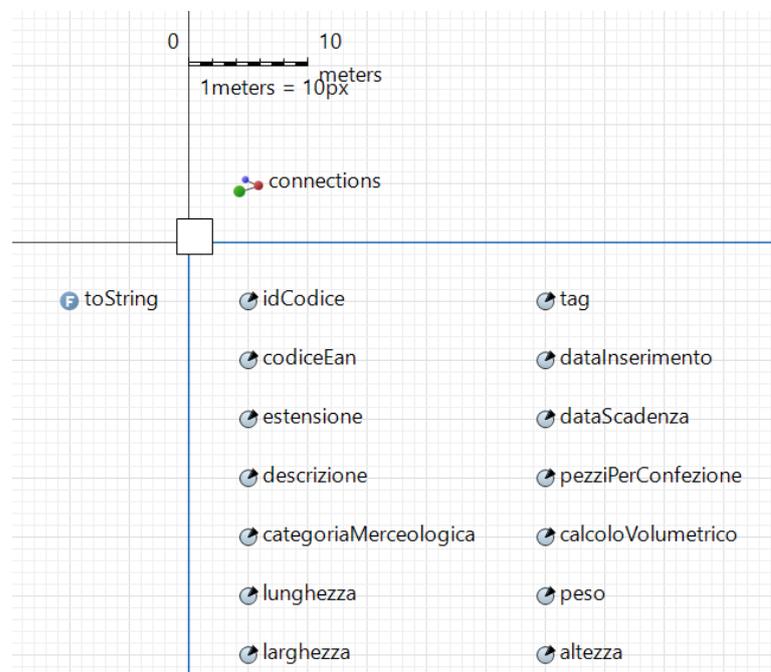


Figura 5.2 - Agente Articolo

```
Function body
return
    "idCodice =" + idCodice + ";" +
    "codiceEan =" + codiceEan + ";" +
    "estensione =" + estensione + ";" +
    "descrizione =" + descrizione + ";" +
    "categoriaMerceologica =" + categoriaMerceologica + ";" +
    "lunghezza =" + lunghezza + ";" +
    "larghezza =" + larghezza + ";" +
    "altezza =" + altezza + ";" +
    "peso =" + peso + ";" +
    "calcoloVolumetrico =" + calcoloVolumetrico + ";" +
    "pezziPerConfezione =" + pezziPerConfezione + ";" +
    "dataScadenza =" + dataScadenza + ";" +
    "dataInserimento =" + dataInserimento + ";" +
    "tag =" + tag;
Advanced
```

Figura 5.3 - Funzione toString dell'agente Articolo

```
Additional class code:
Articolo( Articolo a ) {
    this.idCodice = a.idCodice;
    this.codiceEan = a.codiceEan;
    this.estensione= a.estensione;
    this.descrizione= a.descrizione;
    this.categoriaMerceologica= a.categoriaMerceologica;
    this.lunghezza= a.lunghezza;
    this.larghezza= a.larghezza;
    this.altezza= a.altezza;
    this.peso = a.peso;
    this.calcoloVolumetrico= a.calcoloVolumetrico;
    this.pezziPerConfezione= a.pezziPerConfezione;
    this.dataScadenza= a.dataScadenza;
    this.dataInserimento= a.dataInserimento;
    this.tag= a.tag;
}
 Parameterized type
Advanced
```

Figura 5.4 - Costruttore dell'agente Articolo

L'agente Documento, illustrato in Figura 5.5, contiene nove parametri e una funzione. La configurazione di questo agente è stata eseguita allo stesso modo di quella dell'agente Articolo.

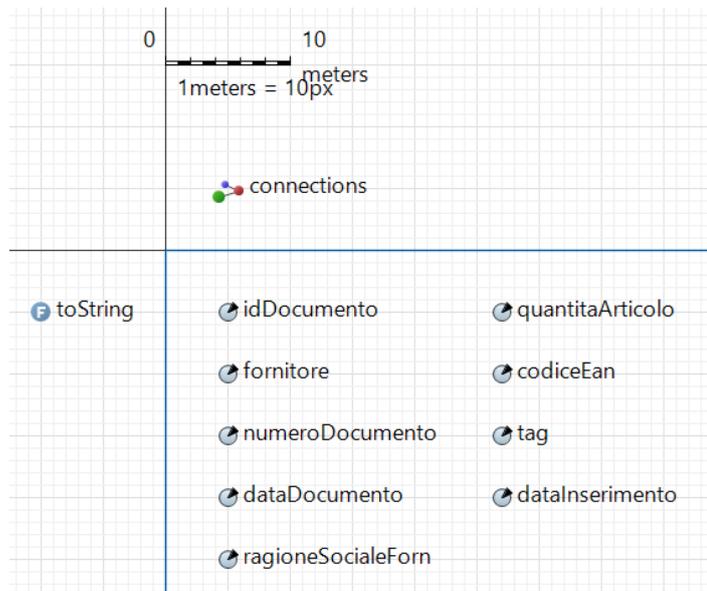


Figura 5.5 - Agente Documento

L'agente Ordine, illustrato in Figura 5.6, contiene quattordici parametri e una funzione. La configurazione di questo agente è stata eseguita allo stesso modo di quella dell'agente Articolo.

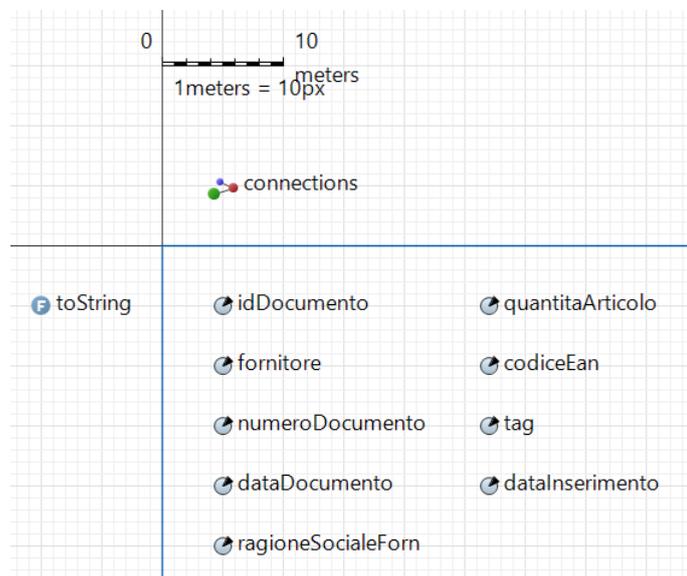


Figura 5.6 - Agente Ordine

L'agente Cassetta, illustrato in Figura 5.7, contiene sette parametri e una funzione. La configurazione di questo agente è stata eseguita allo stesso modo di quella dell'agente Articolo.

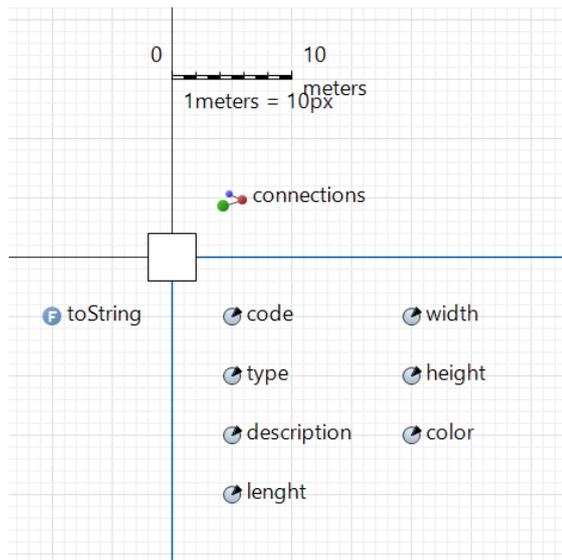


Figura 5.7 - Agente Cassetta

L'agente Main, illustrato in Figura 5.8, è l'agente radice e contiene quattro processi, quattro variabili, quattro nodi rettangolari collegati da due percorsi e una finestra 3D. Tutti e quattro i processi sono costituiti da quattro blocchi: una Source, una Queue, un Delay e un Sink. Questi blocchi sono stati configurati con la stessa logica in tutti i processi e differiscono solamente per l'agente a cui si riferiscono.

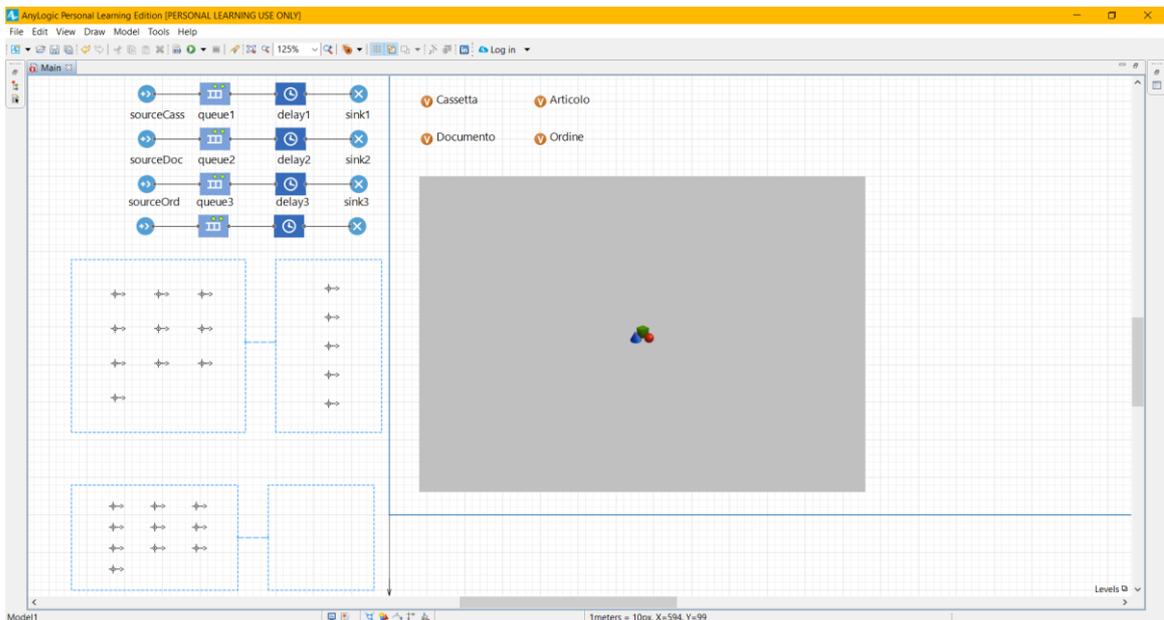


Figura 5.8 - Agente Main

Per la configurazione del blocco Source, in “Properties” è stato impostato un “Arrival Rate” di un agente al secondo; è stata spuntata l’opzione “Set agent parameters from DB” per importare i parametri dell’agente dal database; è stata scelta tabella in “Database tabel”, è stata spuntata l’opzione “Limited number of arrivals” per limitare il numero di agenti creati al numero di record presenti nella tabella selezionata; per gli agenti con una riproduzione fisica nel modello come Articoli e Cassette è stato definito il nodo di destinazione tramite l’opzione “Node”. Nella scheda “Agent”, nell’opzione “New Agent” è stato selezionato l’agente e sono stati mappate le proprietà specifiche dell’agente con la corrispettiva colonna nella tabella selezionata precedentemente. Nella scheda “Actions”, nell’opzione “On exit” è stato inserito il codice Java necessario a valorizzare il contenuto della variabile dell’agente creato servendosi del costruttore che abbiamo definito durante la sua configurazione. La Figura 5.9 mostra le impostazioni del blocco Source per la generazione di nuovi agenti Articolo. La configurazione del blocco Queue riguarda solamente il numero di agenti che possono rimanere in coda e la scelta del nodo in cui mettere in coda gli agenti. In “Properties”, nell’opzione “Capacity” è stato impostato il valore cento e nell’opzione “Agent location” è stato selezionato il nodo in cui vengono generati gli agenti corrispondenti. Per la configurazione del blocco Delay è stato impostato in “Properties” nell’opzione “Delay time” un tempo di attesa di dieci secondi e nell’opzione “Agent location” è stato scelto il nodo in cui l’agente, dopo essere stato generato, giunge tramite il percorso. Il blocco Sink invece presenta la configurazione di default. Grazie alla finestra 3D è possibile vedere, durante l’esecuzione della simulazione, i processi attraverso un’animazione 3D.

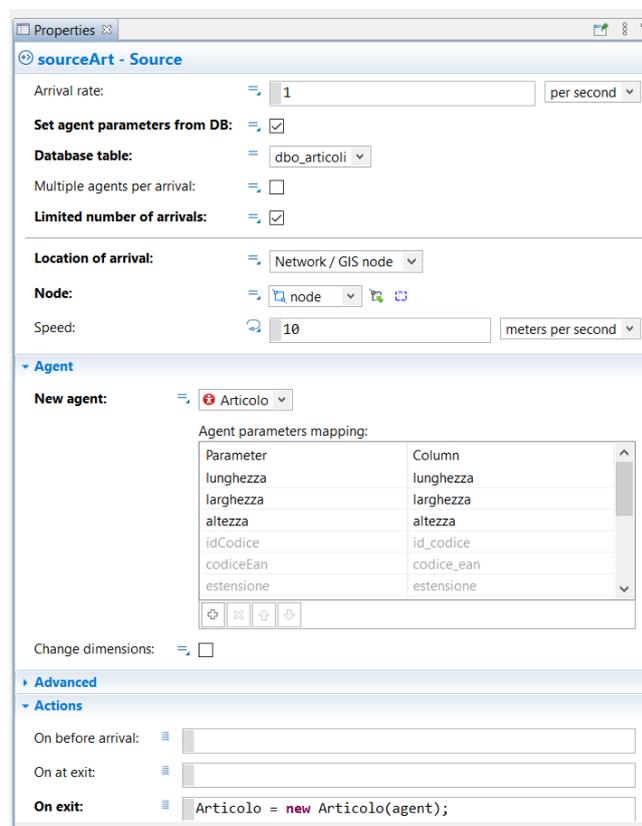


Figura 5.9 - Blocco Source dell’agente Main

5.2 *Trasmissione delle informazioni*

In questo paragrafo verranno descritte le modalità con cui è stato messo in comunicazione lo spazio fisico con quello virtuale. Per prima cosa verrà presentata un'analisi approfondita della struttura del database da cui vengono trasmessi i dati di input necessari ad eseguire le simulazioni. Successivamente verrà spiegato come è avvenuta la configurazione del modello di simulazione di prova per permettere l'accesso al database e utilizzare in automatico i dati più aggiornati.

5.2.1 Struttura Database

Come anticipato precedentemente, il database utilizzato in questo esempio applicativo non è quello attualmente operativo all'interno del magazzino, ma ne è stato creato uno da zero con una struttura coerente a quella del database operativo. Per dare una rappresentazione ad alto livello della struttura del database utilizzato è stato fatto uso di un diagramma delle classi scritto attraverso il linguaggio UML (Figura 5.10). Ogni classe in questo diagramma rappresenta una tabella del database e contiene al suo interno una serie di attributi che corrispondono alle colonne della relativa tabella.

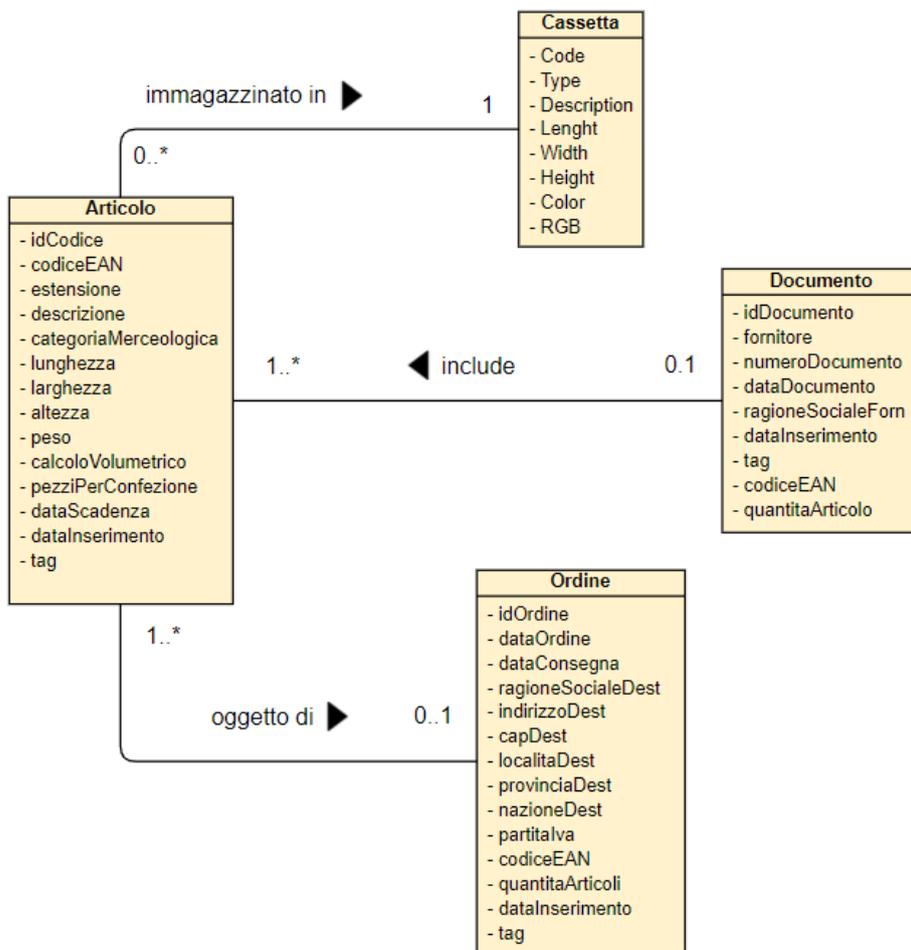


Figura 5.10 - Diagramma classi

Come è possibile vedere in Figura 5.10 sono state individuate quattro classi:

1. Articoli;
2. Casette;
3. Documenti;
4. Ordini.

La tabella Articoli contiene tutte le informazioni riguardanti i prodotti che passano attraverso il magazzino, quindi saranno presenti sia semi-lavorati che prodotti finiti. Questa tabella è costituita da 14 colonne:

1. idCodice: chiave primaria della tabella con valore numerico intero auto-incrementale valorizzato automaticamente al caricamento del prodotto nel database;
2. codiceEAN: stringa alfanumerica identificativa di un prodotto;
3. estensione:
4. descrizione: descrizione del prodotto;
5. categoriaMerceologica: descrizione della categoria merceologica del prodotto;
6. lunghezza: valore numerico intero espresso in cm riferito alla lunghezza del prodotto;
7. larghezza: valore numerico intero espresso in cm riferito alla larghezza del prodotto;

8. altezza: valore numerico intero espresso in cm riferito all'altezza del prodotto;
9. peso: valore numerico intero espresso in g riferito al peso del prodotto;
10. calcoloVolumetrico: carattere alfabetico che informa dell'applicabilità del calcolo volumetrico sul prodotto;
11. pezziPerConfezione: la quantità di prodotti presenti in una confezione;
12. dataScadenza:
13. dataInserimento: data in cui il prodotto è stato inserito nel database;
14. tag: carattere alfabetico di marcatura per importazione/esportazione dati.

Nella tabella Cassette si trovano le informazioni riguardanti le cassette in cui sono inseriti i prodotti. Le cassette possono essere di diverse dimensioni e quindi, in base alle dimensioni, possono contenere una diversa quantità di prodotti. In questa tabella sono presenti 8 colonne:

1. Code: chiave primaria della tabella con valore numerico intero incrementale;
2. Type: stringa alfanumerica che identifica il tipo di cassetta;
3. Description: descrizione estesa del tipo di cassetta;
4. Length: valore numerico intero espresso in cm riferito alla lunghezza della cassetta;
5. Width: valore numerico intero espresso in cm riferito alla larghezza della cassetta;
6. Height: valore numerico intero espresso in cm riferito all'altezza della cassetta;
7. Color: descrizione estesa del colore della cassetta;
8. RGB: valore numerico intero identificativo di un determinato colore.

Nella tabella Documenti sono presenti i documenti generati in seguito a un ordine in ingresso e contiene informazioni sulla quantità e la tipologia dei prodotti oggetto dell'ordine e sul fornitore che ha completato l'ordine. Questa tabella è costituita da 9 colonne:

1. idDocumento: chiave primaria della tabella con valore numerico intero auto-incrementale valorizzato automaticamente al caricamento del documento nel database;
2. fornitore: stringa alfanumerica identificativa del fornitore;
3. numeroDocumento: stringa alfanumerica identificativa del documento;
4. dataDocumento: data in cui è stato generato il documento;
5. ragioneSocialeForn: descrizione estesa della ragione sociale del fornitore;
6. dataInserimento: data in cui il documento è stato inserito nel database;
7. tag: carattere alfabetico di marcatura per importazione/esportazione dati;
8. codiceEAN: stringa alfanumerica identificativa di un prodotto;
9. quantitàArticolo: quantità di prodotti presenti nell'ordine a cui fa riferimento il documento.

Nella tabella Ordini sono presenti le informazioni riguardanti gli ordini in uscita. Questa tabella è costituita da 14 colonne:

1. idOrdine: chiave primaria della tabella con valore numerico intero auto-incrementale valorizzato automaticamente al caricamento degli ordini nel database;
2. dataOrdine: data in cui l'ordine è stato emesso;
3. dataConsegna: data in cui l'ordine è stato consegnato al destinatario;

4. ragioneSocialeDest: la descrizione estesa della ragione sociale del destinatario dell'ordine;
5. indirizzoDest: la descrizione estesa dell'indirizzo del destinatario dell'ordine;
6. capDest: il CAP del destinatario dell'ordine;
7. localitaDest: la descrizione estesa della località del destinatario dell'ordine;
8. provinciaDest: la descrizione estesa della provincia del destinatario dell'ordine;
9. nazioneDest: la descrizione estesa della nazione del destinatario dell'ordine;
10. partitaIvaDest: la partita iva del destinatario dell'ordine;
11. codiceEAN: stringa alfanumerica identificativa di un prodotto;
12. quantitàArticoli: quantità di prodotti presenti nell'ordine a cui fa riferimento il documento;
13. dataInserimento: data in cui il documento è stato inserito nel database;
14. tag: carattere alfabetico di marcatura per importazione/esportazione dati.

Le relazioni individuate sono state 3:

1. tra la classe Articolo e la classe Cassette;
2. tra la classe Articolo e la classe Documento;
3. tra la classe Articolo e la classe Ordine.

Tra queste relazioni le differenze sono definite solo dalle loro diverse cardinalità. Per esempio la cardinalità 1 della prima relazione significa che a un'istanza della classe Articolo corrisponde solo un'istanza della classe Cassette, ovvero che un determinato articolo può essere immagazzinato solamente in una cassetta. Sempre nella prima relazione, la cardinalità 0..* significa che a un'istanza della classe Cassette possono corrispondere 0 o più istanze della classe Articolo, ovvero che una determinata cassetta può immagazzinare 0 o più articoli.

5.2.2 Configurazione DBMS

Un passaggio fondamentale nella creazione della Digital Shadow è configurare il DBMS (Database Management System) e il modello di simulazione di prova in modo da permettere a quest'ultimo l'accesso ai dati più aggiornati in maniera automatica. In coerenza con quanto avviene nel magazzino reale, è stato utilizzato MSSMS (Microsoft SQL Server Management Studio) come DBMS. Di seguito un elenco dei passi che saranno affrontati:

1. Creazione di un utente su MSSMS con accesso al database;
2. Accesso al database da Anylogic con account appena creato;
3. Scelta delle tabelle da importare;
4. Utilizzo dati all'interno delle tabelle.

Per creare un nuovo utente su MSSMS è necessario accedere con l'utente amministratore e effettuare i passaggi seguenti corredati ognuno da una figura illustrativa. Per prima cosa andare in "Esplora oggetti" e selezionare il server, poi "sicurezza" e, cliccando col tasto destro su "Account di accesso", selezionare "Nuovo account di accesso..." (Figura 5.11).

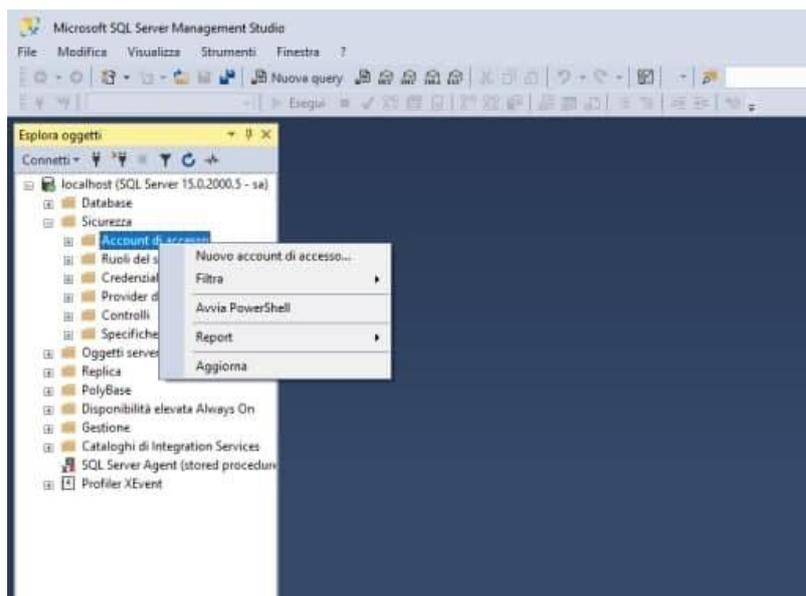


Figura 5.11 - Primo passaggio creazione utente

Si apre una finestra in cui va inserito il nome dell'account desiderato, successivamente selezionare "Autenticazione di SQL Server" e inserire la password scelta (Figura 5.12).

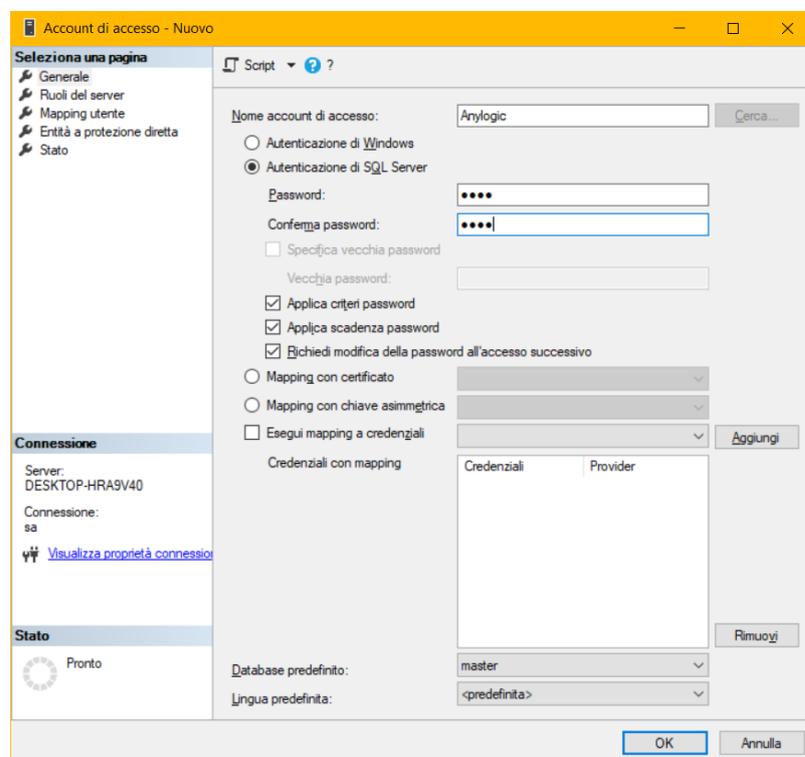


Figura 5.12 - Secondo passaggio creazione utente

Sempre nella stessa finestra andare in “Mapping Utente”, selezionare la spunta nel database che si intende utilizzare (in questo caso “tesi”), selezionare poi la spunta in “db_owner” e per concludere cliccare su “OK” (Figura 5.13).

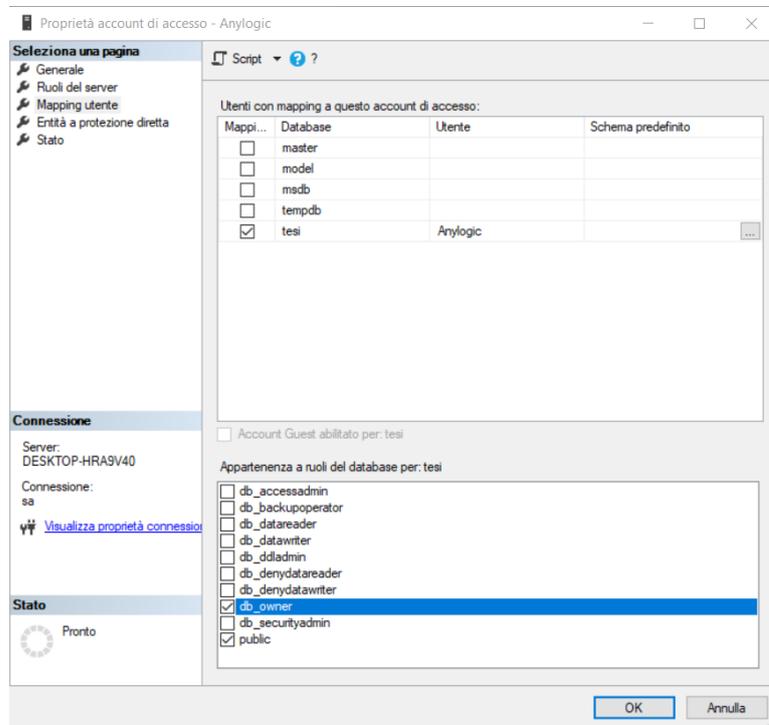


Figura 5.13 - Terzo passaggio creazione utente

Dopo aver creato un account di accesso al database, quest'ultimo va utilizzato direttamente da Anylogic. Si esegue il programma, si apre il progetto da utilizzare e nella scheda “Projects” selezionare “Database”. Si apre una scheda chiamata “Properties”, qui cliccare su “Create or import a table...” e nella finestra che si apre selezionare “Import database table(s)” (Figura 5.14).

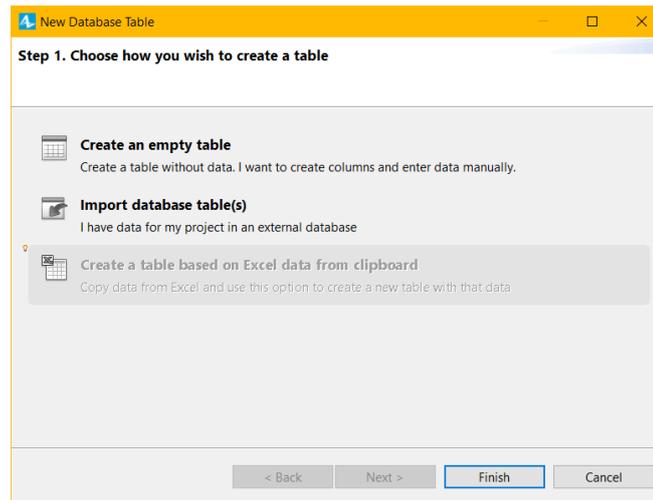


Figura 5.14 - Primo passaggio accesso database da Anylogic

Successivamente si apre una finestra in cui vanno settati tutti i parametri di accesso al database. In “Type” selezionare “Microsoft SQL Server”, in “JDBC driver” mantenere l’impostazione predefinita, in “Host” inserire l’indirizzo IP del database (“localhost” se si trova all’interno della stessa rete locale a cui è connessa la macchina in cui viene avviato Anylogic), in “Database Name” inserire il nome del database, in “Login” inserire il nome utente dell’account che è stato creato precedentemente e in “Password” inserire la password scelta in fase di creazione dell’account. Dopo aver compilato tutti i campi e aver premuto il pulsante “Show list of tables” sulla destra comparirà la lista di tutte le tabelle presenti nel database. Deselezionare la spunta in quelle di cui non si avrà bisogno e, se desiderato, rinominare quelle da importare. Selezionare la spunta nell’opzione in basso “Update data on the model startup” per utilizzare sempre i dati più aggiornati ogni volta che viene avviata una nuova simulazione. Infine cliccare su “Finish” (Figura 5.15).

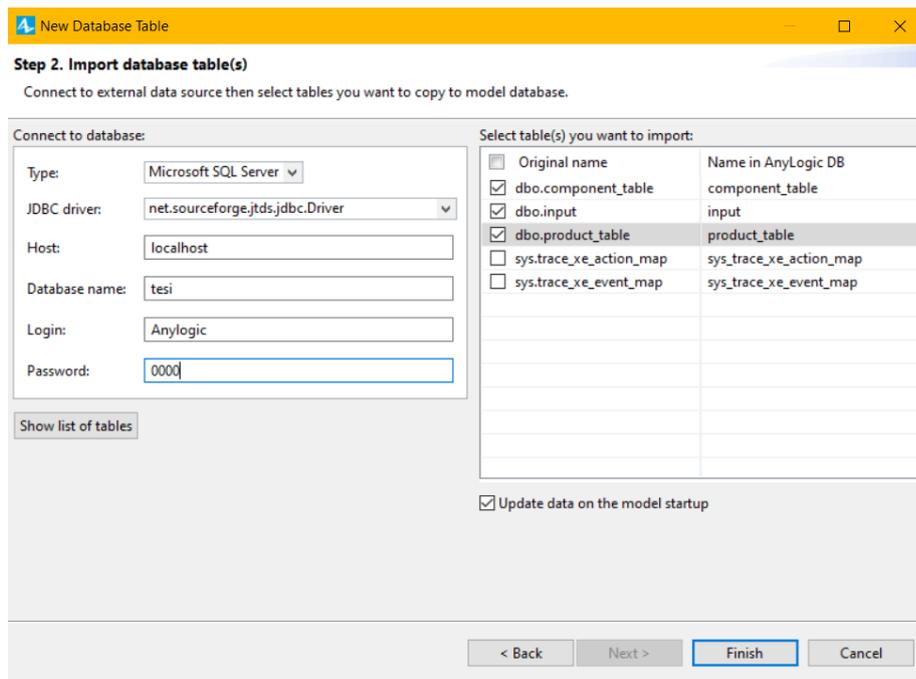


Figura 5.15 - Secondo passaggio accesso database da Anylogic

Per utilizzare i dati presenti nelle tabelle che sono state importate bisogna utilizzare il linguaggio SQL. Come esempio verrà mostrato il settaggio grazie al quale il software prende i valori dalla tabella `dbo_articoli` e li utilizza per definire i parametri di nuovi agenti di tipo Articolo. Per iniziare la configurazione bisogna andare nella scheda “Projects” e aprire l’agente “Main”. Poi bisogna selezionare la sorgente dalla quale si vuole creare gli agenti e andare nella scheda “Properties”. Qui è importante selezionare la spunta “Set agent parameters from DB”, selezionare la tabella dalla quale si ha intenzione di utilizzare i dati e selezionare la spunta “Limited number of arrivals” per limitare la creazione degli agenti al numero di istanze presenti nella tabella. Poi, nelle opzioni “Agent”, selezionare il tipo di agente che si intende creare e associare al parametro dell’agente la corrispondente colonna della tabella (Figura 5.16).

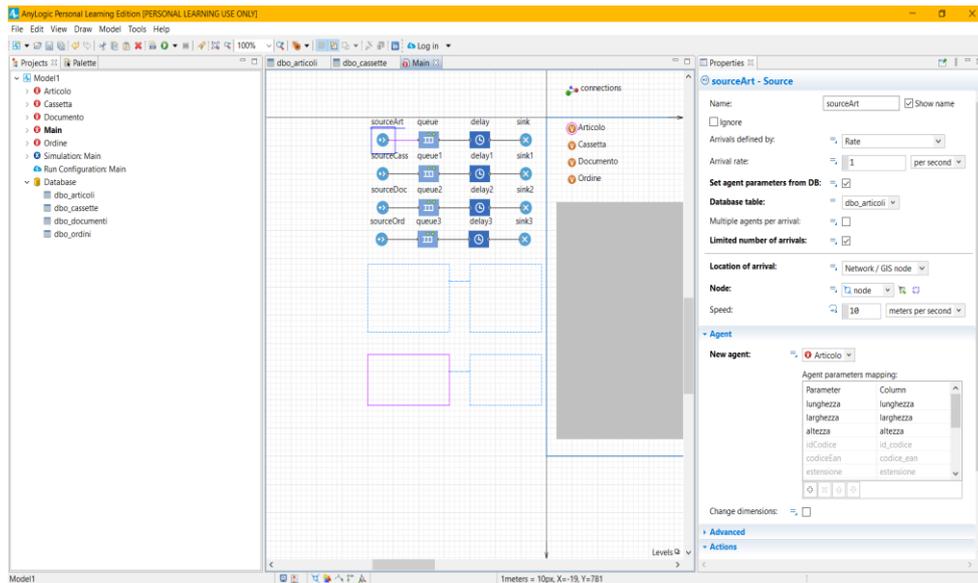


Figura 5.16 - Configurazione creazione degli agenti

Attraverso tutti questi passaggi Anylogic è configurato per comunicare con qualunque database Microsoft SQL Server ed è in grado di avviare le simulazioni utilizzando sempre i dati più aggiornati.

Conclusioni

L'obiettivo prefissato da questa tesi è riuscire a costruire un'architettura Digital Twin di un magazzino automatico installato all'interno dei laboratori del Politecnico di Torino. Per fare ciò sono state svolte delle ricerche in letteratura con lo scopo di riuscire a comprendere la struttura e le proprietà di un Digital Twin. Sono state anche analizzate alcune applicazioni reali di questa tecnologia per approfondire nel dettaglio gli aspetti legati alle caratteristiche che deve possedere l'infrastruttura informatica in cui va inserito. Durante lo svolgimento di questa ricerca è emerso che un passaggio fondamentale per lo sviluppo di un Digital Twin è l'implementazione di una Digital Shadow. Per questo motivo si è deciso di proseguire sviluppando non più il Digital Twin, ma la Digital Shadow del magazzino e modellando in maniera teorica l'architettura del Digital Twin. Terminate le ricerche in letteratura, ci si è addentrati nello specifico del caso studio. È avvenuto lo studio di tutte le caratteristiche degli elementi fisici che costituiscono il magazzino; sono stati approfonditi tutti gli aspetti legati all'infrastruttura informatica installata nel laboratorio, sia hardware che software; è stato analizzato il meccanismo di funzionamento del modello di simulazione del magazzino ed è stata eseguita una formazione basilare sull'utilizzo del software di simulazione utilizzato per creare questo modello.

Questa fase di studio ha permesso di avere a disposizione tutte le conoscenze per creare tramite il linguaggio UML un modello concettuale, precisamente un diagramma delle classi, che descrive l'architettura Digital Twin del magazzino automatico. Questo modello offre una visione chiara e d'insieme di tutte le logiche di funzionamento del sistema. Sono state rese esplicite le relazioni tra tutte le entità e definiti tutti gli attributi e le funzioni posseduti da ognuna. Inoltre, come esempio applicativo dell'architettura descritta dal modello UML, è stata sviluppata una Digital Shadow utilizzando un modello di simulazione creato da zero con il software di simulazioni AnyLogic e un database, anch'esso creato da zero, costruito con una struttura coerente a quello operativo. Grazie alla Digital Shadow è stato possibile dare evidenza della corretta connessione tra il database e il modello di simulazione e dell'utilizzo automatico dei dati più aggiornati all'avvio di ogni nuova simulazione.

Le maggiori difficoltà incontrate nello sviluppo della tesi hanno riguardato la creazione della Digital Shadow. L'idea di partenza era quella di integrare il modello di simulazione con delle modifiche che gli permettessero di comunicare con i database operativi. Per fare ciò però sarebbe stato necessario uno studio molto approfondito del modello e del software di simulazione. Per superare questo ostacolo si è deciso di creare un modello di simulazione da zero che avrebbe eseguito solamente delle semplici operazioni sufficienti a dimostrare la corretta connessione col database. In questo modo è stato possibile focalizzarsi specificatamente sulla realizzazione della Digital Shadow senza dedicare un tempo eccessivo allo studio del software e del modello di simulazione preesistente. Durante lo sviluppo della Digital Shadow si è presentata inoltre una limitazione da un punto di vista tecnico causata dal software di simulazione AnyLogic in versione Personal Learning Edition, il quale non permette la comunicazione tramite librerie con un ambiente di sviluppo esterno, per esempio

Eclipse. Utilizzare un ambiente di sviluppo esterno sarebbe stato necessario per avviare in automatico le simulazioni a intervalli di tempo regolari.

I risultati ottenuti al termine di questa tesi costituiscono una base importante per lo sviluppo di attività di ricerca future. Una naturale prosecuzione di questo lavoro è sicuramente rappresentata dall'integrazione del modello di simulazione preesistente con tutte le funzionalità appartenenti al modello di prova. In questo modo si riuscirà a creare una Digital Shadow capace di funzionare con tutte le informazioni del sistema reale. Un ulteriore sviluppo decisamente più ambizioso riguarda la creazione di un vero e proprio sistema Digital Twin. Infatti partendo da una Digital Shadow sarà più semplice compiere questo importante passo in avanti. Per raggiungere questo obiettivo sarà necessario implementare algoritmi di Intelligenza Artificiale e ridefinire parte dell'attuale infrastruttura informatica per garantire una comunicazione bidirezionale col modello di simulazione.

Bibliografia

B. R. Barricelli, E. Casiraghi and D. Fogli, "A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications," in *IEEE Access*, vol. 7, pp. 167653-167671, 2019, doi: 10.1109/ACCESS.2019.2953499.

Madni, A.M.; Madni, C.C.; Lucero, S.D. Leveraging Digital Twin Technology in Model-Based Systems Engineering. *Systems* 2019, 7, 7. <https://www.mdpi.com/404118> MDPI and ACS Style

Fei Tao, Qinglin Qi, Lihui Wang, A.Y.C. Nee, Digital Twins and Cyber-Physical Systems toward Smart Manufacturing and Industry 4.0: Correlation and Comparison, *Engineering*, Volume 5, Issue 4, 2019, Pages 653-661, ISSN 2095-8099, <https://doi.org/10.1016/j.eng.2019.01.014>.
(<http://www.sciencedirect.com/science/article/pii/S209580991830612X>)

Julien, Badin & Chamoret, Dominique & Roth, Sebastien & Jean-Remy, Imbert & Gomes, Samuel. (2012). Knowledge Based Simulation Driven Design for Crash Applications. *International journal of mechanics and applications*. 4. 10.5923/j.mechanics.20120205.01.

Patel, Keyur & Patel, Sunil & Scholar, P & Salazar, Carlos. (2016). Internet of Things-IOT: Definition, Characteristics, Architecture, Enabling Technologies, Application & Future Challenges.

Revel G.M. e Naspi F., 2020, Introduzione al mondo dell'Internet of Things

Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. (2015): Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications, in: *IEEE Communication Surveys & Tutorials*, Vol. 17, No. 4, 2015.

J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang and W. Zhao, "A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications," in *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1125-1142, Oct. 2017, doi: 10.1109/JIOT.2017.2683200.

X. Yang et al., "A novel en-route filtering scheme against false data injection attacks in cyber-physical networked systems", *IEEE Trans. Comput.*, vol. 64, no. 1, pp. 4-18, Jan. 2015.

M.-C. Chuang and J.-F. Lee, "TEAM: Trust-extended authentication mechanism for vehicular ad hoc networks", *IEEE Syst. J.*, vol. 8, no. 3, pp. 749-758, Sep. 2014.

S. U. Maheswari, N. S. Usha, E. A. M. Anita and K. R. Devi, "A novel robust routing protocol RAEED to avoid DoS attacks in WSN", *Proc. Int. Conf. Inf. Commun. Embedded Syst. (ICICES)*, pp. 1-5, Feb. 2016.

F. Qiu, F. Wu and G. Chen, "Privacy and quality preserving multimedia data aggregation for participatory sensing systems", *IEEE Trans. Mobile Comput.*, vol. 14, no. 6, pp. 1287-1300, Jun. 2015.

- J. Girao, D. Westhoff and M. Schneider, "CDA: Concealed data aggregation for reverse multicast traffic in wireless sensor networks", Proc. IEEE Int. Conf. Commun. (ICC), vol. 5, pp. 3044-3049, May 2005.
- W. He, X. Liu, H. Nguyen, K. Nahrstedt and T. Abdelzaher, "PDA: Privacy-preserving data aggregation in wireless sensor networks", Proc. 26th IEEE Int. Conf. Comput. Commun. (INFOCOM), pp. 2045-2053, May 2007.
- M. V. Bharathi, R. C. Tanguturi, C. Jayakumar and K. Selvamani, "Node capture attack in wireless sensor network: A survey", Proc. IEEE Int. Conf. Comput. Intell. Comput. Res. (ICCIC), pp. 1-3, Dec. 2012.
- M. Sarkar and D. B. Roy, "Prevention of sleep deprivation attacks using clustering", Proc. 3rd Int. Conf. Electron. Comput. Technol. (ICECT), vol. 5, pp. 391-394, Apr. 2011.
- R. P. Padhy, M. R. Patra and S. C. Satapathy, "Cloud computing: Security issues and research challenges", Int. J. Comput. Sci. Inf. Technol. Security, vol. 1, no. 2, pp. 136-146, Dec. 2011.
- W. Yu, N. Zhang, X. Fu and W. Zhao, "Self-disciplinary worms and countermeasures: Modeling and analysis", IEEE Trans. Parallel Distrib. Syst., vol. 21, no. 10, pp. 1501-1514, Oct. 2010.
- I. Andrea, C. Chrysostomou and G. Hadjichristofi, "Internet of Things: Security vulnerabilities and challenges", Proc. IEEE Symp. Comput. Commun. (ISCC), pp. 180-187, Jul. 2015.
- K. Chopra, K. Gupta and A. Lambora, "Future Internet: The Internet of Things-A Literature Review," 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 2019, pp. 135-139, doi: 10.1109/COMITCon.2019.8862269.
- N. Alam, P. Vats and N. Kashyap, "Internet of Things: A literature review," 2017 Recent Developments in Control, Automation & Power Engineering (RDCAPE), Noida, 2017, pp. 192-197, doi: 10.1109/RDCAPE.2017.8358265.
- Ayon Dey / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 7 (3), 2016, 1174-1179
- S.B. Kotsiantis, "Supervised Machine Learning: A Review of Classification Techniques", Informatica 31 (2007) 249-268
- X. Zhu, A. B. Goldberg, "Introduction to Semi – Supervised Learning", Synthesis Lectures on Artificial Intelligence and Machine Learning, 2009, Vol. 3, No. 1, Pages 1-130
- R. S. Sutton, "Introduction: The Challenge of Reinforcement Learning", Machine Learning, 8, Page 225-227, Kluwer Academic Publishers, Boston, 1992
- R. Caruana, "Multitask Learning", Machine Learning, 28, 41-75, Kluwer Academic Publishers, 1997
- D. Opitz, R. Maclin, "Popular Ensemble Methods: An Empirical Study", Journal of Artificial Intelligence Research, 11, Pages 169- 198, 1999

Neha Gupta, Artificial Neural Network, ISSN 2224-610X (Paper), ISSN 2225-0603 (Online), Vol.3, No.1, 2013-Selected from International Conference on Recent Trends in Applied Sciences with Engineering Applications

“Fundamentals Of Neural Network” by Laurene V. Fausett, (1993).

“An Introduction To Neural Network”, by James A. Anderson, (1997)

Dec 21, 2018, Titanic Prediction with Artificial Neural Network in R.
<https://laptrinhx.com/titanic-prediction-with-artificial-neural-network-in-r-3087367370/>
Laptrinhx

Ashtari Talkhestani, B., Jung, T., Lindemann, B., Sahlab, N., Jazdi, N., Schloegl, W., & Weyrich, M. (2019). An architecture of an Intelligent Digital Twin in a Cyber-Physical Production System, at - Automatisierungstechnik, 67(9), 762-782. doi: <https://doi.org/10.1515/auto-2019-0039>.

F. Tao and M. Zhang, "Digital Twin Shop-Floor: A New Shop-Floor Paradigm Towards Smart Manufacturing," in IEEE Access, vol. 5, pp. 20418-20427, 2017, doi: 10.1109/ACCESS.2017.2756069.

F. Tao, M. Zhang and J. Cheng, "Digital twin workshop: A new paradigm for future workshop", Comput. Integr. Manuf. Syst., vol. 23, no. 1, pp. 1-9, 2017

Zheng, Y., Yang, S. & Cheng, H. An application framework of digital twin and its case study. J Ambient Intell Human Comput 10, 1141–1153 (2019). <https://doi.org.ezproxy.biblio.polito.it/10.1007/s12652-018-0911-3>

J. Leng, D. Yan, Q. Liu, H. Zhang, G. Zhao, L. Wei, X. Chen, Digital twin-driven joint optimisation of packing and storage assignment in large-scale automated high-rise warehouse product-service system, Int. J. Comput. Integr. Manuf. (2019), pp. 1-18

M. Braglia, R. Gabbrielli, M. Frosolini, L. Marrazzini and L. Padellini, "Using RFID technology and Discrete-Events, Agent-Based simulation tools to build Digital-Twins of large warehouses," 2019 IEEE International Conference on RFID Technology and Applications (RFID-TA), Pisa, Italy, 2019, pp. 464-469, doi: 10.1109/RFID-TA.2019.8892254.

Sitografia

Tracy Woo, March 2016 <https://www.aberdeen.com/featured/computer-aided-development/>

Rfidglobal <https://www.rfidglobal.it/tecnologia-rfid/rfid-come-funziona/#:~:text=Le%20antenne%20RFID%2C%20gestite%20dal,dei%20tag%20ed%20all'ambiente>

Spotle, Munmun Ghosal, Jul 26, 2019 <https://spotle.ai/feeddetails/What-Is-AI-Weak-AI-Strong-AI-with-examples-Key-disciplines-and-applications-Of-AI-/3173>

Datalog, May 07, 2019 <https://www.datalog.it/cose-un-sistema-erp-e-come-ottimizza-la-gestione-dellazienda/>

Mecalux, 2019a <https://www.mecalux.it/blog/sistema-erp-significato#:~:text=Spesso%20ERP%20e%20WMS%20vengono,tipologia%22%20di%20architettura%20e%20funzioni>

Mecalux, 2019b <https://www.mecalux.it/blog/sistema-erp-significato#:~:text=Spesso%20ERP%20e%20WMS%20vengono,tipologia%22%20di%20architettura%20e%20funzioni>.

Zdintegration http://www.zdintegration.com/Management_software_system/

USA CISA https://us-cert.cisa.gov/ics/Control_System_Security_DMZ-Definition.html

Pandasecurity, 2019 <https://www.pandasecurity.com/it/mediacenter/sicurezza/che-cose-un-firewall/>

Cybersecurity360 <https://www.cybersecurity360.it/soluzioni-aziendali/vpn-cose-come-funziona-e-a-cosa-serve-una-virtual-private-network/>