

ALMA MATER STUDIORUM – UNIVERSITÀ DI BOLOGNA
CAMPUS DI CESENA

DIPARTIMENTO DI INFORMATICA – SCIENZA E INGEGNERIA
Corso di Laurea in Ingegneria e Scienze Informatiche

DIGITAL TWIN NEL CAMPO DELLE
COSTRUZIONI
-
UN CASO DI STUDIO IN AMBITO
DOMOTICO

Elaborato in
SISTEMI EMBEDDED E INTERNET-OF-THINGS

Relatore
Prof. ALESSANDRO RICCI

Presentata da
GIOVANNI RADICCHI

Anno Accademico 2020 – 2021

*"My name is Ozymandias, king of kings:
Look on my works, ye Mighty, and despair!"
Percy Bysshe Shelley*

Indice

Introduzione	vii
1 Digital Twin: definizione, storia e proprietà	1
1.1 Definizione e storia dei digital twin	1
1.2 Proprietà dei Digital Twin	2
2 Applicazioni dei DT nel campo delle costruzioni	7
2.1 I DT applicati alle costruzioni e il rapporto con il BIM	7
2.2 La struttura del digital twin di un edificio	9
2.2.1 Architettura di un digital twin per un edificio	9
2.2.2 La Wireless Sensor Network e la raccolta dei dati	12
2.2.3 Maturity Elements e sviluppo incrementale dei digital twins	16
2.3 Benefici e vantaggi dei digital twin	18
3 Un caso di studio: Digital Twin di un sistema di riscaldamento	23
3.1 Scelta e analisi del caso di studio	23
3.2 Modellazione del caso di studio	24
3.3 Una proposta architettonica	25
4 Implementazione del prototipo con Azure Digital Twins	29
4.1 La piattaforma Microsoft Azure	29
4.2 Traduzione dei modelli in linguaggio DTDL	30
4.3 Creazione del grafo con Azure Digital Twins Explorer	33
4.4 Architettura della soluzione ed elementi principali	34
4.4.1 Simulatore di rete di sensori e device IoT	35
4.4.2 Gateway cloud	35
4.4.3 Funzione per la trasformazione dei dati	35
4.4.4 Elaborazione dei dati di telemetria	36
4.5 Test della soluzione e considerazioni sul suo funzionamento	37
Conclusioni	39

Ringraziamenti

41

Introduzione

Alla luce dell'aumento di interesse nei temi di sostenibilità e ecologia, è aumentata l'attenzione, anche da parte del governo italiano, nel rendere le abitazioni e gli edifici in generale più verdi, efficienti e sostenibili. Il superbonus 110% e il decreto legislativo 48/2020 [12] dimostrano come la domotica sia considerata uno degli attori principali nel miglioramento delle prestazioni energetiche degli edifici. Nonostante ciò gli investimenti effettuati ancora non rispecchiano l'effettivo potenziale di questi strumenti. Dei più di 8 miliardi di euro investiti in Smart building in Italia nel 2019, solo 2 mld. sono serviti a produrre soluzioni "effettivamente smart" [5], rivelando come nel mercato italiano il valore di queste soluzioni sia ancora sottostimato. Inoltre, anche a causa della pandemia globale di Covid19, nel corso del 2020 questi investimenti sono andati calando nonostante il superbonus [6]. Con l'arrivo dei nuovi finanziamenti previsti dal Piano Nazionale di Ripresa e Resilienza, gli investimenti nel campo dei Smart building potrebbero riprendere a crescere, ma è di fondamentale importanza che questi fondi siano diretti verso soluzioni veramente efficaci nella riduzione di costi, consumi ed emissioni degli edifici. La sola implementazione di sistemi domotici potrebbe non essere la soluzione migliore a lungo termine. Negli ultimi anni sta riscuotendo sempre più successo un nuovo approccio nella transizione digitale, non riguardante esclusivamente l'edilizia, ed è quello di associare a un prodotto, a partire dalla sua progettazione, il suo Digital Twin.

Con Digital Twin si intende un sistema di sensori, dispositivi IoT, software, database e applicazioni che creano una copia digitale e "viva", interattiva e in costante aggiornamento, dell'oggetto fisico. I DT permettono di raccogliere dati sullo stato operativo di un oggetto, analizzarli e ricavarne informazioni, eseguire simulazioni di utilizzo ecc. Le applicazioni per strumenti così flessibili sono tantissime, e non sono ancora state esplorate appieno, in quanto non è stato ancora mai implementato il DT completo di un oggetto.

L'obiettivo di questa tesi è quello di studiare la possibilità di utilizzare i DT per rendere più efficienti costruzioni già esistenti, e in particolare le abitazioni. Per fare questo, è fondamentale capire cosa effettivamente sia un DT, quali siano le sue proprietà fondamentali, le applicazioni e i vantaggi che porta, con

particolare attenzione verso la sua declinazione nel campo delle costruzioni. Questo studio è presentato nel primo e secondo capitolo. Per portare lo studio su un campo più pratico, è stato progettato e realizzato un semplice prototipo con lo scopo di mostrare in maniera pratica l'applicazione del concetto di DT alle abitazioni, le sue potenzialità e le relative problematiche. Le fasi di analisi e progettazione del prototipo sono presentate nel terzo capitolo, mentre la sua implementazione, realizzata tramite Azure Digital Twins, è descritta nel quarto capitolo.

Capitolo 1

Digital Twin: definizione, storia e proprietà

In questo primo capitolo viene esposto lo studio del concetto di DT effettuato sulla letteratura esistente. Verrà affrontato il concetto di DT, descrivendone la storia e l'evoluzione, e le sue proprietà generali, indipendenti dal contesto di applicazione.

1.1 Definizione e storia dei digital twin

Il termine digital twin (letteralmente “gemello digitale”) è stato introdotto per la prima volta dalla NASA nel Roadmap Report del 2010. In questo documento il DT viene definito come “an integrated multi-physics, multi-scale, probabilistic simulation of a vehicle or system that uses the best available physical models, sensor updates, etc., to mirror the life of its flying twin”; ossia “una simulazione integrata multi-fisica, multi-scala e probabilistica di un veicolo o sistema che usa i migliori modelli fisici, aggiornamenti sensoriali, ecc. disponibili per replicare lo stato del suo gemello in volo” [17].

Questa prima definizione di DT non rappresenta l'origine del concetto in sé. Già durante il programma Apollo la stessa NASA creava due copie del veicolo che sarebbe stato lanciato. Quella che rimaneva a terra aveva lo scopo di riprodurre le condizioni del veicolo in volo durante la missione. Il gemello veniva utilizzato sia per l'addestramento sia per simulare alternative in tempo reale durante le missioni, fornendo assistenza agli astronauti durante situazioni critiche. In questo senso qualsiasi prototipo costruito con lo scopo di simulare il comportamento reale di un oggetto può essere visto come un gemello [15].

La transizione del concetto di gemelli da una forma fisica a una forma digitale avvenne nel 2002 quando Michael Grieves della University of Michigan presentò a una conferenza della Society of Manufacturing Engineers l'idea

del product lifecycle management (PLM) [4]. In una pubblicazione successiva Grieves e John Vickers affermano che l'idea alla base del DT sia che “a digital informational construct about a physical system could be created as an entity on its own. This digital information would be a “twin” of the information that was embedded within the physical system itself and be linked with that physical system through the entire lifecycle of the system“. Tradotto: “un costrutto digitale informativo di un sistema fisico può essere creato come un'entità a sé stante. Le informazioni digitali diventano dei “gemelli” delle informazioni che sono integrate nel sistema fisico stesso e vi sono collegate durante tutto il ciclo di vita del sistema”. Sebbene non fosse chiamato digital twin il PLM corrispondeva già a questa descrizione, in quanto presentava, oltre al sistema fisico, il sistema virtuale e il collegamento per lo scambio di informazioni tra i due sistemi. Essi permettevano al sistema virtuale di replicare lo stato del suo gemello fisico [9].

1.2 Proprietà dei Digital Twin

Come abbiamo visto, il concetto di Digital Twin si è evoluto digitalizzando pratiche di prototipazione già note e applicate. Da quel momento, la ricerca sui DT ha proseguito l'esplorazione delle sue funzionalità e applicazioni proponendo molti modi sia per la realizzazione che per il funzionamento degli stessi. Mancando una standardizzazione a livello globale, il rischio è quello di creare sistemi incapaci di interagire fra di loro. Il survey di Minerva et al. del 2020 [14] si propone, quindi, di identificare un set di proprietà più o meno essenziali alla realizzazione di un DT esaminando la letteratura esistente. All'interno del survey, gli autori identificano il DT come il riferimento a un Physical Object (PO), ossia un oggetto materialmente presente nel mondo fisico, e la sua o le sue controparti software, chiamate anche Logical Object (LO), connesse al PO da forti relazioni. E' importante sottolineare come il rapporto fra PO e LO non deve necessariamente essere di 1 a 1, in quanto anche un rapporto di 1 a n può rappresentare un DT, dove ciascun LO si focalizza sul rappresentare una parte del PO, o viene utilizzato per servizi o scopi differenti. E' importante però che tutto il DT si riferisca a un singolo PO in maniera univoca, indipendentemente dal numero di LO al suo interno.

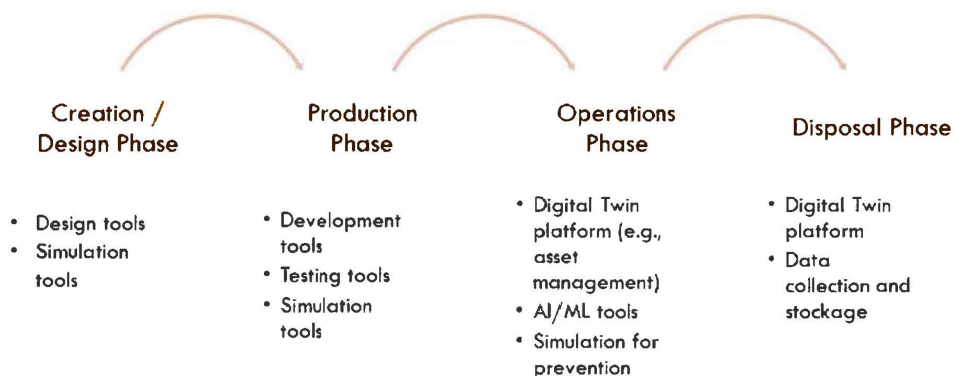


Figura 1.1: Lifecycle di un DT, con strumenti e funzioni necessarie [14].

Gli autori identificano anche quello che è il lifecycle di un DT, in questo caso di quelli sviluppati in ambito manifatturiero. Come si può vedere dall'immagine, i DT intervengono in ogni fase del ciclo di vita del PO che modellano. Se in fase di design il LO risulta predominante sul PO, che potrebbe non esistere nemmeno, dalla fase di produzione il PO inizia a diventare predominante nell'invio di dati e informazioni al LO. Questo non vuol dire che il LO smetta di inviare informazioni al PO, può anzi rivelarsi utile in molte situazioni che sia il primo a trasmettere informazioni e comandi al secondo. La connessione e lo scambio di informazioni fra le due parti del DT è quindi di fondamentale importanza per renderlo efficace.

Conclusa questa premessa, il survey procede con l'identificare le proprietà di un DT dividendole in due insiemi. Il primo insieme rappresenta le proprietà fondamentali di un DT, senza le quali non si può nemmeno parlare di implementazione di un DT:

- **Reflection:** il LO deve riflettere lo stato del PO nella maniera più accurata possibile. Al momento, considerando le possibilità dei sistemi IoT, questa proprietà si riferisce principalmente a tutto ciò che è misurabile nel PO. Le misurazioni effettuate rispecchiano lo stato del PO in una posizione e un momento ben definiti. Affinché il LO rifletta nel modo migliore il PO, deve includere un set di questi valori tale che il modello ottenuto rappresenti sufficientemente il PO;
- **Replicazione e virtualizzazione:** il PO può essere virtualizzato e replicato molte volte per soddisfare le richieste di servizi e applicazioni differenti. Anche un LO può essere replicato con lo stesso scopo. E' importante che ciò sia possibile all'interno di un DT per distribuire al meglio le risorse e i carichi di lavoro, in risposta alle richieste di applicazioni e servizi;

CAPITOLO 1. DIGITAL TWIN: DEFINIZIONE, STORIA E PROPRIETÀ

- **Entanglement:** il concetto di DT può essere espresso come il collegamento fra un PO e il suo LO. La natura di questo collegamento deve essere quanto più vicina possibile a un trasferimento di informazioni in tempo reale fra i due oggetti, così da garantire che lo stato corretto del PO sia sempre rappresentato e visualizzato da applicazioni e servizi;
- **Rappresentatività e contestualizzazione:** la creazione di un DT deve avvenire avendo bene in mente il suo scopo e il contesto nel quale deve operare. Pertanto il LO deve rappresentare tutte quelle caratteristiche, proprietà e comportamenti che ha il PO all'interno del contesto per cui si vuole utilizzare il DT. Questo vuol dire che il PO non deve essere replicato per intero, ma idealmente solo ciò che riguarda lo scopo per il quale verrà utilizzato il DT, mantenendo comunque la rappresentazione dello stato quanto più simile allo stato reale del PO.

Il secondo insieme rappresenta delle proprietà più accessorie, ma che contribuiscono ad espandere e incrementare il valore e le funzionalità di un DT:

- **Composability:** la possibilità di un DT di essere composto da più parti potenzialmente indipendenti, gestibili e monitorabili individualmente, ma al tempo stesso in grado di operare come insieme, creando un sistema unico più complesso. Rappresenta anche la facilità con cui un dato DT può comporsi con altri andando a formare un sistema più grande e complesso;
- **Accountability/Manageability:** la possibilità di gestire il DT completamente e accuratamente. Questa proprietà non si riferisce solo all'accesso a tutte le funzioni e informazioni del DT, e quindi del PO e del LO, ma anche alla capacità di accedere al LO anche in caso di malfunzionamenti del PO. Idealmente, il DT deve poter continuare a esistere e funzionare anche dopo la fine del ciclo vitale del PO;
- **Memorizzazione:** quanti dei dati ricevuti dal PO vengono memorizzati e conservati. Questi dati potrebbero non essere utili nel contesto in cui viene creato il DT, ad esempio se è di interesse solo lo stato attuale del PO, ma memorizzare gli stati passati del PO può essere utile per una serie di funzionalità aggiuntive, come la creazione di applicazioni che imparano dai dati storici;
- **Persistenza:** questa proprietà si collega alle due precedenti, in quanto la persistenza dei dati e del funzionamento del DT è fondamentale per per la *memorizzazione* e la *manageability* nel DT. Il LO è l'abilitatore principale in questa proprietà, e deve essere robusto e persistente nel

CAPITOLO 1. DIGITAL TWIN: DEFINIZIONE, STORIA E PROPRIETÀ

mantenere i dati e le funzionalità sempre attive, anche quando il PO non è disponibile o funzionante;

- **Augmentation:** la possibilità del DT di migliorare e incrementare le sue funzionalità. Anche se questo può non essere possibile per il PO, il LO, essendo un oggetto digitale, può cambiare ed evolvere nel tempo, aumentando le funzionalità e i servizi forniti;
- **Ownership:** questa proprietà è molto importante per un'azienda che investe nella creazione di DT. Difatti, creare un DT associato al PO fornisce un numero maggiore di potenziali fonti di guadagno, in quanto, oltre al PO, possono essere venduti anche i LO ad esso connessi, e i servizi associati a ciascun LO. L'identificazione della proprietà dei dati raccolti e dei diritti d'uso degli stessi risulta importante per gli utenti finali, anche da un punto di vista legale;
- **Servitization:** questa proprietà è strettamente collegata al valore di mercato del DT, in quanto si riferisce alla sua usabilità ed efficienza nei confronti dei suoi utenti, intermedi o finali. In maniera simile all'*augmentation*, si tratta di fornire quanti più servizi e funzionalità possibili, ma più incentrata verso gli utenti, che possono interagire con il DT tramite software, strumenti o interfacce appositamente create;
- **Predictability:** è la capacità del DT di effettuare test e simulazioni per predire situazioni specifiche e le sue interazioni con altri oggetti una volta inserito in un determinato ambiente.

Tutte queste proprietà consentono di identificare bene l'identità, l'uso e lo scopo dei DT, anche se molto deve essere ancora fatto per raggiungere un vero e proprio standard che permetta una visione unica del concetto di DT. Non aiuta il fatto che, anche a causa del vasto interesse che i DT stanno suscitando, il dominio applicativo dei DT sia popolato da prototipi e soluzioni proprietarie anche molto diverse tra loro, tra l'altro a volte non complete o operative, oppure create strettamente per essere usate in un solo settore, mancando quindi della generalità necessaria per essere usate in maniera diffusa. Fortunatamente diverse organizzazioni stanno procedendo a stilare standard per lo sviluppo di questi sistemi, come ad esempio ISO con lo standard ISO/AWI 23247.

Capitolo 2

Applicazioni dei DT nel campo delle costruzioni

In questo capitolo si prosegue lo studio sui DT, iniziato nel capitolo precedente, concentrandosi sulle applicazioni nel campo delle costruzioni. Verrà effettuato un paragone con il Building Information Modeling (BIM), una metodologia di progettazione delle costruzioni di poco meno recente dei DT che condivide con essi molte caratteristiche. Verranno discusse alcune architetture e proprietà specifiche per i DT di costruzioni, oltre ai vantaggi e benefici che questi possono portare alle figure coinvolte nella loro realizzazione e utilizzo.

2.1 I DT applicati alle costruzioni e il rapporto con il BIM

Dallo studio svolto finora sono emerse caratteristiche importanti del concetto di DT, ma la sua declinazione in domini applicativi specifici può richiedere degli adattamenti ulteriori. E' il caso dei DT delle costruzioni, dove l'implementazione dei DT può beneficiare del metodo di progettazione noto come Building Information Modeling (BIM). Nell'ambito della progettazione di costruzioni la metodologia BIM sta prendendo sempre più piede, in quanto beneficia della collaborazione di tutte le figure professionali coinvolte nella progettazione e costruzione dell'opera. BIM crea un modello al cui interno sono presenti tutti gli oggetti che compongono la costruzione, e ai quali sono associati tutti i dati relativi come materiali, dimensioni, ecc. Sebbene richieda un lavoro più lungo in fase di progettazione il modello così prodotto offre numerosi vantaggi. Può essere utilizzato per eseguire simulazioni prima che l'opera venga effettivamente costruita al fine di correggere errori e applicare modifiche al progetto. Facilita la comunicazione fra le diverse figure e fasi

della progettazione e costruzione, poiché riduce gli errori e le perdite di dati e aumenta l'efficienza e la produttività. Risulta utile anche durante la fase operativa, fornendo informazioni necessarie e di facile accesso per la manutenzione della costruzione e, al termine del suo ciclo di vita, permette di pianificarne e simularne l'abbattimento. Il BIM risulta quindi "un contenitore di informazioni sull'edificio", che "garantisce la validità dei dati inseriti nel manufatto in ogni momento del suo ciclo di vita" [13] [3].

Data la somiglianza fra DT e BIM negli scopi che ricoprono e nelle funzionalità, sono numerosi gli esempi nella letteratura in cui sistemi che sono, di fatto, in qualche modo paragonabili a un DT vengono comunque scambiati per BIM [7]. Occorre, quindi, sottolinearne le differenze per capire se i due sistemi possano coesistere, se non addirittura integrarsi per facilitare il raggiungimento dei rispettivi obiettivi. Occorre quindi avere una chiara idea dell'identità del concetto di DT applicato alle costruzioni e, a questo proposito, si esprime il documento noto come "Gemini Principles" [2], redatto nel 2018 dal Centre for Digital Built Britain (CDBB). Qui il DT di una costruzione viene definito come "a realistic digital representation of assets, processes or systems in the built or natural environment. What distinguishes a digital twin from any other digital model is its connection to the physical twin.", ossia "una rappresentazione digitale e realistica di beni, processi o sistemi in un ambiente costruito o naturale. Quello che distingue un gemello digitale da qualsiasi altro modello digitale è la sua connessione con il gemello fisico.". Questa distinzione esclude il BIM dalla definizione di DT, in quanto la differenza sostanziale fra questi due modelli è proprio il collegamento del DT con l'asset fisico. I dati all'interno del modello prodotto con BIM sono inseriti manualmente dai professionisti che lavorano al progetto e alla costruzione dell'opera, e non è previsto che vengano visualizzati dall'utente finale. I DT invece automatizzano l'inserimento dei dati in fase operativa, mettono a disposizione servizi e app non solo per accedere ai dati, ma anche per interagire con il DT, sia a livello virtuale che a livello fisico. Difatti, i DT possono, a livello teorico, modificare lo stato della costruzione, anche in maniera automatica, mentre il BIM non dispone in nessun modo di un tale collegamento con l'opera.

Inoltre, il BIM concentra il suo utilizzo nelle fasi di progettazione, messa in opera e demolizione della costruzione, essendo principalmente una metodologia di progettazione, mentre il DT interviene in maniera molto più marcata durante la fase operativa a tutto tondo, fornendo strumenti di analisi e intervento in tempo reale e concentrandosi sulle interazioni fra l'opera e l'ambiente circostante e gli utenti, soprattutto gli utenti finali [1]. Sebbene il BIM possa essere utilizzato per fornire informazioni utili ai manutentori, e risparmiare parte delle attività di sopralluogo, i DT sono più efficienti anche in questo ambito, potendo potenzialmente segnalare direttamente la fonte del problema,

e permettendo interventi più rapidi e mirati.

In conclusione, il DT può essere visto come un'espansione del metodo BIM, in quanto aggiunge il collegamento fra asset logici/virtuali e asset fisico, abilitando una comunicazione bidirezionale e lo sviluppo di app e servizi che espandono le funzionalità della costruzione, oltre a poter svolgere simulazioni ancora più efficaci. D'altro canto, in fase di progettazione un DT può beneficiare dell'applicazione della metodologia BIM, grazie alla quale il modello creato sarà più preciso e implementerà meglio le proprietà fondamentali di un DT.

2.2 La struttura del digital twin di un edificio

In questa sezione viene analizzata la possibile struttura di un DT, utilizzando come fonti articoli e studi recenti in questo ambito, prendendo in esame sia le componenti dei DT, sia possibili framework più generali che descrivono l'intero DT.

2.2.1 Architettura di un digital twin per un edificio

Quando si progetta un sistema complesso come un DT, è fondamentale avere in fase di progettazione un'architettura generica su cui fare affidamento, per stabilire lo scopo che avranno le componenti progettate. In letteratura si possono trovare diversi articoli e studi che si sono focalizzati sul proporre una possibile architettura più o meno generica per i DT. Particolarmente interessante è un articolo del 2019 [11] pubblicato da alcuni ricercatori dell'università di Cambridge, in cui viene proposta un'architettura per i DT di edifici piuttosto dettagliata, con tanto di applicazione a un caso di studio reale per verificarne la correttezza e possibilità di implementazione. L'architettura proposta dagli autori si compone di 5 livelli:

1. **Acquisizione dati:** è il livello più basso, e costituisce le fondamenta di ogni DT. Si occupa di raccogliere i dati necessari al funzionamento utilizzando varie tecnologie a seconda delle necessità, fra cui dispositivi IoT, sensori, tag ecc. Il design di questo livello deve dipendere non solo dall'architettura del DT, ma anche dall'edificio per il quale viene implementato, in quanto sarà direttamente collegato ad esso;
2. **Trasmissione:** lo scopo di questo livello è di trasformare i dati raccolti dal livello sottostante per inviarli ai livelli superiori. I dati devono essere ricevuti da fonti molto diverse fra loro e posizionate in luoghi anche distanti tra loro, pertanto i protocolli di comunicazione devono essere efficienti e affidabili. La tecnologia più usata e conosciuta è il Wi-Fi;

3. **Modellazione digitale e dati complementari:** rappresenta il modello digitale del DT, di solito il BIM, e fornisce supporto ai livelli superiori collegandoli ai livelli di acquisizione e trasmissione dei dati. La composizione del livello varia a seconda delle funzioni che devono essere svolte dal DT che si sta progettando. A seconda delle simulazioni che dovrà svolgere, ci potranno essere diversi modelli digitali specifici per il compito richiesto;
4. **Integrazione dati/modello:** rappresenta il kernel di questa architettura, include l'immagazzinamento e la gestione dei dati e dei modelli, per poi svolgere tutte le operazioni di trasformazione dei dati in informazioni utili, aggiornando lo stato del modello. Ospita tutta la parte di machine learning del DT;
5. **Applicazione:** la parte del DT che interagisce con l'utente finale, siano essi i gestori o gli utenti dell'edificio.

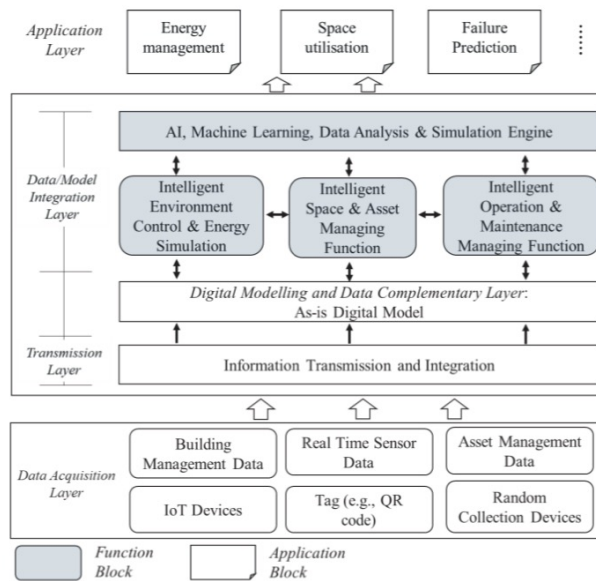


Figura 2.1: Schematizzazione dell'architettura di un DT appena esposta[11].

L'architettura è stata testata creando un DT nel campus dell'università di Cambridge, nello specifico nell'edificio dell'Institute for Manufacturing. I primi due livelli sono stati implementati tramite una Wireless Sensor Network (WSN), ossia una rete di sensori allacciati a dei gateway che comunicano con

un server centrale. È evidente come i sensori e i gateway rappresentino rispettivamente i livelli di acquisizione dati e trasmissione, pertanto occorre analizzare come la WSN sia stata implementata per adattarsi all'edificio protagonista del progetto. È stata scelta un'architettura a stella per la rete, con i sensori e i gateway che comunicano tramite sequenze radio (radio frequencies, RF). Secondo quanto riportato nello studio dagli autori, la combinazione usata permette ai sensori di comunicare fino a 90 metri di distanza non in linea d'aria (quindi con ostruzioni parziali), con una durata della batteria dei sensori approssimativamente di due anni trasmettendo i dati una volta al minuto (dati presi dai costruttori dei dispositivi usati, la Monnit). Gli autori hanno trovato particolarmente indicata, per la natura distribuita dell'architettura di un DT, la comunicazione via RF, considerato il suo basso costo. Infine il collegamento fra gateway e il server centrale è stato implementato usando il protocollo HTTP. Sul server centrale viene hostato il Sensor Manager, un software sviluppato in .NET che, tramite le Monnit Mine API, si occupa di gestire e inizializzare il collegamenti fra gateway e server.

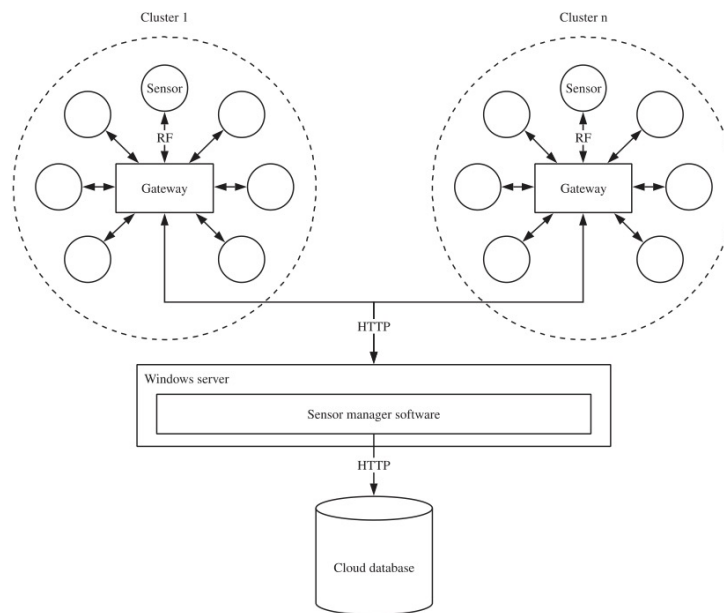


Figura 2.2: Schema della WSN costruita per raccogliere dati dall'edificio[11].

Il terzo livello è stato implementato attraverso un BIM altamente dettagliato, comprensivo di tutti i dati architettonici, strutturali e relativi ai sistemi MEP (mechanical, electrical and pumping), inseriti all'interno di un modello digitale rappresentante ogni dettaglio dell'edificio.

Gli ultimi due livelli sono stati implementati integrando, oltre ai dati real-time provenienti dalla rete di sensori, anche i dati provenienti dai Building Management System (BMS), asset management system (AMS) e space management system (SMS). Si tratta di sistemi di controllo computerizzati atti ad agire e controllare i sistemi meccanici ed elettrici del palazzo. In questo modo può essere implementato anche il sistema di feedback del DT sul gemello fisico. Infine il progetto ha proposto e sviluppato due applicazioni per interagire con il DT, una per monitorare le condizioni dell'edificio in tempo reale, e un'altra per svolgere simulazioni sulle sue performance future, in modo da predire futuri interventi.

Lo studio si conclude con delle considerazioni effettuate sui risultati conseguiti. Sebbene non siano ancora stati raccolti abbastanza dati per permettere un'analisi dei risultati approfondita, gli autori ritengono che l'architettura da loro proposta sia promettente, e che valga la pena investire ulteriormente nella ricerca e nello sviluppo della stessa.

2.2.2 La Wireless Sensor Network e la raccolta dei dati

Abbiamo già visto come la WSN risulti una parte fondamentale in un DT, in quanto implementazione fisica del collegamento da gemello fisico a gemello digitale. È bene quindi approfondire ulteriormente le sue caratteristiche, operando confronti fra le varie tecnologie disponibili. Uno studio del 2019 diretto da alcuni ricercatori dell'Università di Aalto, in Finlandia [10], si concentra particolarmente sull'implementazione della WSN per la realizzazione di un DT di una delle facciate del campus, analizzando varie tecnologie disponibili. Nella loro ricerca gli autori usano una struttura di DT composta, oltre dall'edificio fisico, da tre elementi fondamentali:

1. Il BIM dell'edificio, che costituisce il modello digitale e il database per i dati;
2. La WSN, che fornisce i dati real-time sull'attività dell'edificio;
3. Data Integration and Analytics, una serie di algoritmi di machine learning che analizzano i Big Data collezionati dalla WSN, imparando da essi e ottenendo informazioni utili alla gestione dell'edificio.

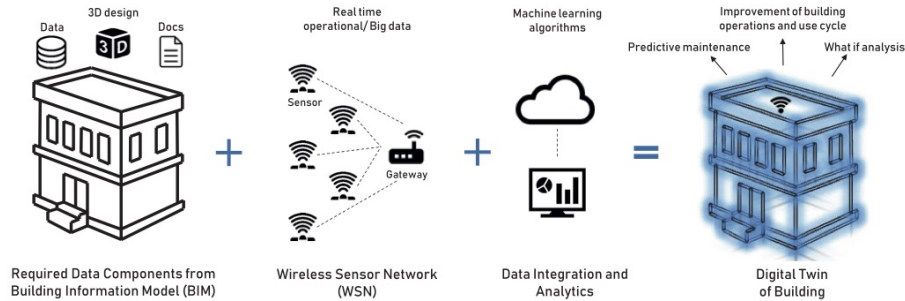


Figura 2.3: Rappresentazione degli elementi costituenti un DT[10].

La WSN implementata è composta da una rete a stella di sensori, i Texas Instruments (TI) Sensortag CC2650, che sono stati scelti sia per il basso consumo delle batterie, sia per l'ottimo supporto fornito dai produttori e dagli utenti, che rendevano disponibili numerosi software user-friendly per programmare i sensori. L'efficienza energetica è un aspetto fondamentale per una WSN, in quanto non essendo allacciata alla rete elettrica dell'edificio, un alto consumo imporrebbe un frequente cambio delle batterie, oltre alla possibile perdita di dati dovuta allo spegnimento dei sensori stessi. La possibilità di impostare cicli di attività e di risparmio energetico ha reso i TI CC2650 preferibili rispetto alle altre opzioni testate. Come gateway per la rete sono stati scelti i Raspberry Pi 3B+, che inviano i dati alla piattaforma cloud tramite Wi-Fi. È stato scelto Raspbian come sistema operativo e sono stati programmati in Python. La comunicazione tra sensori e gateway avviene con il protocollo Bluetooth Low Energy (BLE), che in questa particolare situazione è stato preferito al protocollo Zigbee, in quanto supportato da un numero maggiore di dispositivi, tra cui il gateway scelto dai ricercatori. Inoltre, BLE porta ad un consumo minore di energia e una velocità di trasferimento più alta, a scapito di una minore distanza percorribile dai dati, aspetto non penalizzante nello specifico ambito di questa ricerca, che non richiedeva di collocare i sensori a grandi distanze dai gateway. Nonostante ciò, gli autori considerano Zigbee il protocollo più adatto alla creazione di DT nel mondo reale, non solo per la comunicazione a raggio maggiore, ma anche e soprattutto per la possibilità di costruire reti più robuste usando architetture diverse da quella a stella. Difatti la tecnologia BLE supporta solo questo tipo di architettura, molto semplice ma che, in caso di errori nella comunicazione dei dati, può portare a interruzioni e conseguente perdita degli stessi. Zigbee invece supporta anche altre architetture, come mesh e albero, permettendo quindi l'installazione di meno gateway e fornendo robustezza, in quanto i sensori inviano i dati attraverso una rete di router, i

nodì Zigbee, che poi instradano i dati al gateway. L'alta affidabilità rende le reti Zigbee più indicate alla realizzazione di un DT.

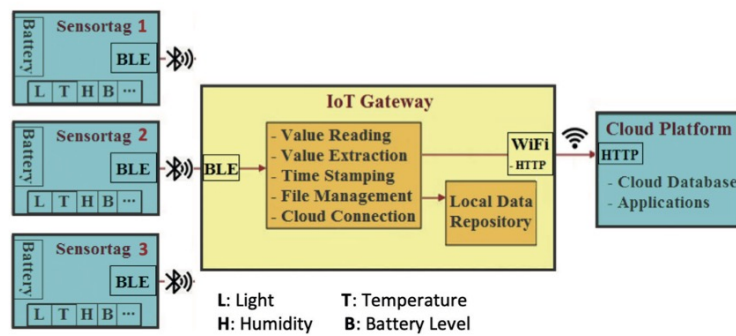


Figura 2.4: Schema della WSN implementata nell'articolo[10].

È importante sottolineare che Zigbee è una tecnologia più adatta alla realizzazione di un DT anche rispetto alle RF analizzate nella sottosezione precedente. La Tabella 1 mostra le specifiche di diverse tecnologie per la comunicazione in una WSN, e Zigbee risulta avere range decisamente più alti, uniti a un consumo di batteria minore e una flessibilità maggiore nella costruzione della rete, rendendolo probabilmente lo standard più adatto alla creazione di WSN complesse o particolarmente estese.

S.No	Feature	Wi-Fi (IEEE 802.11B)	RF (ISO/IEC 14443-2)	Bluetooth (IEEE 802.15.1)	Zigbee (IEEE 802.15.4)	GSM/GPRS CDMA/1X RTT
1	Radio	DSSS#	FHSS	FHSS*	DSSS	-
2	Data Rate	11 M bps	150 kbps	1 M bps	250 kbps	64-128+
3	Nodes Per Master	32	65	7	64000	1
4	Slave Enumeration Latency	Up to 3s	Up to 3s	Up to 10s	30 ms	-
5	Data Type	Video, audio, graphics pictures, file	Compatibility with print, digital and magnetic media	Audio, graphics, pics, files	Small data packet	Wide Area Voice And Data
6	Range(M)	100	35	10	70	10+
7	Extendibility	Roaming possible	Yes	No	Yes	Yes
	Batter Type	AAA battery power	Lithium-ion polymer	Li-ion battery pack	alkaline batteries	Nickel cadmium
8	Battery Life	Hours	6 to 12 months	1 Week	>1 year	1.7
9	Complexity	Complex	Simple	Very complex	Simple	Simple
10	Frequency	2.4GHz	3kHz-300 GHz	2.4GHz	850-930MHz	1,710-1,785 MHz
11	Security	Authentication service set ID(SSID)	-	64 bit,128 bit	128 bit AES and application layer user defined	-
12	Operating Environment	+32°F to +104°F	32° F to 122° F	32° F to 122° F	0 to 55°C (32 to 131°F)	- 30 °C ~ + 85 °C
13	Success metrics	Speed ,flexibility	Compatible	Compatible, easy to handle	Reliability , power ,cost	Reach, quality

Figura 2.5: Tabella riepilogativa delle principali tecnologie di comunicazione wireless[10].

Alla fine dell'articolo gli autori hanno mostrato il DT ottenuto, che consisteva in una rappresentazione delle luce a cui era esposta la facciata del campus. A seconda dell'ora del giorno, o anche in seguito al passaggio di una persona davanti al sensore, il modello si colorava di una diversa tonalità di giallo, tanto più quanto più era bassa la quantità di luce ricevuta dal sensore. Tuttavia, come affermato nelle conclusioni dagli autori stessi, lo studio non è andato oltre al creare una WSN relativamente limitata, e anche il DT risultante era piuttosto semplice, in quanto non attuava nessun tipo di feedback sul gemello fisico partendo dalle informazioni acquisite.

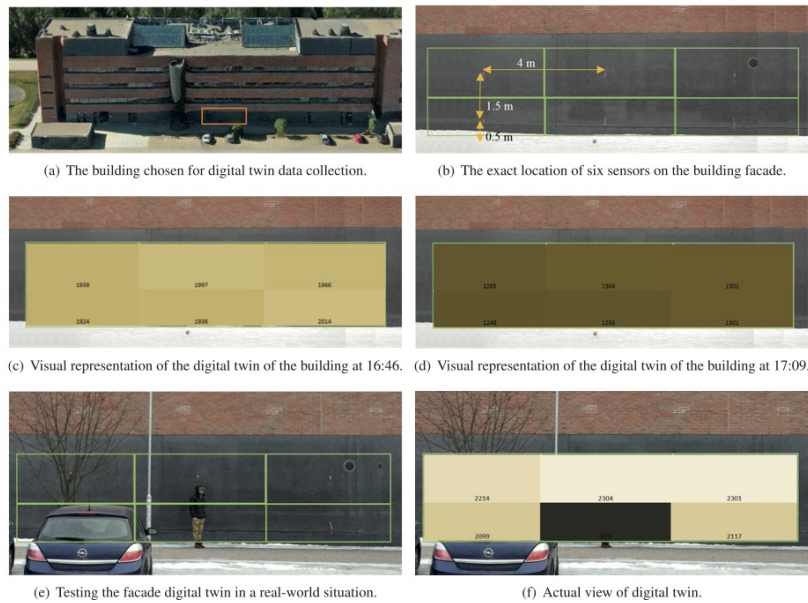


Figura 2.6: Collocazione e resa visiva del DT sviluppato. Viene visualizzata l'intensità luminosa percepita dai sensori. Possiamo notare come venga rilevato anche il passaggio di un pedone, che abbassa la luminosità percepita dal sensore quando ci passa davanti[10].

2.2.3 Maturity Elements e sviluppo incrementale dei digital twins

Un DT alla massima espressione del suo potenziale è un'ideale difficile da raggiungere ai giorni nostri, le tecnologie impiegate sono ancora costose ed è necessario parecchio tempo per ottenere risultati proporzionali all'ingente investimento iniziale, in quanto gli algoritmi di machine learning che sono al centro delle sue funzioni più avanzate necessitano di una gran mole di dati per raggiungere un'efficienza elevata, dati che possono essere prodotti solo nell'arco di diversi anni. In un articolo pubblicato dall'Institution of Engineering and Technology (IET) e l'azienda Atkins [8] vengono mostrate alcune statistiche interessanti riguardo a questo aspetto. Nonostante il mercato globale dei DT nel 2019 sia stato valutato per 3,8 miliardi di dollari (valore che si prevede aumenterà, arrivando a 35,8 miliardi nel 2025), soltanto il 5% delle imprese ha iniziato a implementare i DT in qualche forma, e solo l'1% degli asset è attualmente provvisto di un DT. Secondo gli autori dell'articolo, questa difficoltà nella diffusione dei DT è da attribuire a un modo errato nel porsi verso la creazione del DT di un asset. Considerato l'elevato costo e le difficoltà tecni-

che e tecnologiche della sua creazione nella forma più completa, molte aziende rinunciano a investire in questo campo, ritardando l'acquisizione di dati utili a rendere profittevole in meno tempo un DT. Considerando il fatto che i vantaggi dei DT renderanno in futuro universale la sua presenza in tantissime industrie, iniziare a implementarli a step potrebbe rivelarsi una strategia ottimale per iniziare a raccogliere dati e generare valore già in questi stadi iniziali della diffusione della tecnologia. Inoltre la crescita del mercato spingerebbe sullo sviluppo delle tecnologie e tecniche per la creazione dei DT. Per questo motivo gli autori dell'articolo hanno delineato 6 elementi chiave, detti "maturity elements" (elementi di maturità) che costituiscono un DT al massimo potenziale. Questi livelli non sono sequenziali o incrementali, possono essere implementati in qualsiasi ordine e ciascuno di essi contribuisce a generare valore nel momento in cui viene implementato. Alcuni sono già presenti in molti edifici esistenti, mentre altri sono di più complessa e costosa realizzazione:

1. **Elemento 0 – Cattura della realtà:** è l'elemento di partenza, rilevante solo per edifici esistenti. Vi sono numerose tecnologie disponibili, come il point cloud scanning, l'uso di droni ecc. Il risultato è un modello digitale accurato che contiene i dati sulla geometria e gli spazi dell'edificio, permettendo una maggiore consapevolezza nel suo utilizzo, specialmente in contesti di edifici fatiscenti, che possono esporre i lavoratori a rischi di infortunio;
2. **Elemento 1 – Mappatura 2D e modellazione 3D:** è il punto di partenza nell'implementazione di un DT per i nuovi edifici. Nella maggior parte delle nuove costruzioni è presente fin dalla fase di design, e viene riutilizzato durante l'operatività dell'edificio con lo scopo di ottimizzarne l'utilizzo. In questo stadio non si può ancora parlare di BIM, in quanto non sono presenti dati integrati nel modello. A differenza dell'elemento 0 non viene creato dopo che l'edificio è stato costruito, e per questo motivo potrebbe discostarsi dalla vera natura dell'oggetto fisico. Pertanto un edificio con modello digitale può ugualmente beneficiare della presenza dell'elemento 0, per aggiornarlo e renderlo più preciso;
3. **Elemento 2 – Connessione con dati e metadati statici, implementazione del BIM:** l'aggiunta al modello digitale di una serie di dati presi da diverse fonti, come specifiche dei materiali, dati di gestione ecc., permettono la costituzione del BIM. Il principale vantaggio dato da questo elemento è che tutti i dati che potrebbero servire per qualunque operazione da svolgere sull'edificio sono raccolti in un'unica fonte, riducendo errori, incertezze e costi. Inoltre permette di condurre simulazioni sia all'interno del modello sia con applicazioni esterne. Grazie a queste

simulazioni è possibile predire l'impatto che avranno alcune modifiche sull'intero edificio;

4. **Elemento 3 – Arricchimento con dati dinamici real-time:** con l'installazione di sensori e dispositivi IoT avviene la raccolta di dati dinamici e operativi, che permettono di ottenere lo stato quasi real-time dell'immobile. Il flusso di dati è unidirezionale dall'asset fisico a quello digitale, ma questi dati possono essere analizzati per predire il comportamento dell'edificio e operare scelte informate nella sua gestione, generando indirettamente un feedback sui sistemi esistenti. A partire da questo elemento si inizia a parlare di DT, in quanto viene implementato il collegamento fra gemello fisico e digitale. Fra gli elementi visti finora questo è il primo che richiede investimenti considerevoli;

5. **Elemento 4 – Integrazione ed interazione bidirezionale:** L'aggiunta di sensori e attuatori meccanici permette un'interazione bidirezionale fra i due gemelli. Con questo elemento si può cambiare lo stato del gemello fisico interagendo solamente con il gemello digitale, e il risultato di questa operazione verrà a sua volta reinserito nel gemello digitale per aggiornarlo. Questo elemento si può implementare anche per collegare fra loro diversi asset digitali. Nel momento in cui diverse applicazioni digitali sono collegate al DT, il cambiamento realizzato da una di queste deve riflettersi su tutte le altre, in modo da poter analizzare simulazioni, design ecc. da ogni possibile punto di vista;

6. **Elemento 5 – Automatizzazione operativa e di manutenzione:** l'aggiunta di algoritmi di machine learning e IA all'interno del DT gli permette di creare conoscenza dai dati, fino a diventare autonomo e applicare automaticamente risposte e soluzioni a problemi che si presentano nell'immobile, con requisiti di interazione umana via via sempre più bassi. Il potenziale di questo elemento, così come i benefici quantificabili, devono essere ancora compresi appieno, in quanto le tecnologie a nostra disposizione non ne permettono la realizzazione se non in aspetti relativamente limitati.

2.3 Benefici e vantaggi dei digital twin

Chiarita la natura e la struttura dei DT, possiamo analizzare quelli che sono i benefici che possono essere tratti dall'implementazione del DT di un immobile. I DT rappresentano sistemi estremamente versatili, in grado di raggiungere diversi gradi di complessità e svolgere numerose applicazioni diverse. Pertanto

è di fondamentale importanza identificare in maniera precisa lo scopo del DT che si vuole implementare per poter stabilire con precisioni i vantaggi che può portare. Nel documento dei “Gemini Principles” [2] vengono riportate una serie di applicazioni dei DT, suddivise in varie categorie:

1. Varietà di scopi:

- (a) Attività di previsione: supporto alla strategia e alla pianificazione, esecuzione di scenari “what if?”, regimi di manutenzione predittiva e/o preventiva;
- (b) Stato attuale: monitoraggio e controllo dello stato real-time, diagnostica di eventuali problemi, interventi di gestione;
- (c) Storico: mantenere i dati relativi al passato dell’edificio, per trarne conoscenza.

2. Varietà di scala spaziale:

- (a) A livello di bene o immobile;
- (b) A livello di rete o quartiere;
- (c) A livello di sistema, città o regionale;
- (d) A livello nazionale.

3. Varietà di scala temporale:

- (a) Durante l’operatività dell’immobile;
- (b) Nel momento in cui si fanno manutenzioni predittive;
- (c) Nel momento in cui si fanno manutenzioni pianificate;
- (d) Nel momento in cui si decidono investimenti di capitale.

4. Varietà di approcci alla modellazione:

- (a) Modellazione geometrica;
- (b) Modellazione matematica/computazionale;
- (c) Apprendimento tramite IA e algoritmi di machine learning.

La grande varietà di utilizzi possibili di un DT è dovuta anche all’elevato numero di persone che vengono coinvolte nel ciclo vitale di un edificio. Architetti, ingegneri e designer vengono coinvolti nella sua progettazione, operai e tecnici nella sua costruzione, mentre il proprietario dell’edificio, gli investitori, gli affittuari e gli utenti hanno necessità differenti tra loro durante l’operatività

dell'immobile. Tutto questo insieme di persone viene definito "stakeholders", e per ciascuno di essi i DT portano funzioni e benefici profondamente differenti.

Per quanto riguarda la fase di progettazione, i dati raccolti tramite DT già attivi possono ridurre sensibilmente il numero di errori commessi. Il BIM di cui abbiamo parlato in precedenza svolge già in parte questa funzione, creando "un'unica versione di verità" [8] per tutti i dati relativi all'immobile. Il DT migliora ulteriormente questo aspetto, in quanto la raccolta dei dati durante l'operatività dell'edificio fornisce uno strumento utile a misurare l'efficienza ed efficacia di alcune soluzioni piuttosto che altre. In questo modo si raccolgono informazioni e conoscenze utili per i progetti futuri, che diventeranno via via migliori. Questo aspetto impatta anche sul lavoro svolto da operai e tecnici durante la costruzione dell'immobile, lavoro che risulta anch'esso meno soggetto ad errori grazie ai BIM, con il risultato finale di minori spese extra-budget per la correzione di errori o minori guasti e problemi durante la fase operativa. Secondo quanto riportato in un white paper redatto dalla Siemens [18], la presenza di BIM e DT migliora l'affidabilità del budget per la costruzione dell'immobile, con riduzioni anche del 40% nelle spese non previste.

Oltre agli aspetti già citati, proprietari e investitori beneficiano dell'implementazione di un DT anche durante la fase operativa, rendendola più efficiente in molteplici situazioni. La capacità di un DT di analizzare i dati di illuminazione, areazione e utilizzo degli spazi permette di trovare schemi di efficientamento nell'uso delle stanze, oltre a fornire informazioni sul miglioramento del design e della progettazione delle stesse. Tutto questo porta ad utilizzare meglio gli edifici già costruiti, e la possibilità di costruire in futuro edifici perfettamente progettati attorno al compito che dovranno svolgere e al numero di occupanti previsti, ad esempio eliminando gli spazi che non verrebbero utilizzati. Inoltre l'immobile risulta notevolmente più efficiente anche dal punto di vista del risparmio energetico. Tutti i sistemi MEP possono beneficiare largamente dall'integrazione con un DT, con conseguenti benefici sia per il proprietario che per l'ambiente. Infine, un DT abilita la possibilità di effettuare manutenzioni sia in previsione di un guasto, attraverso simulazioni che predicano l'insorgenza di malfunzionamenti in sistemi che si stanno deteriorando, sia programmando manutenzioni di tipo preventivo. Queste possibilità permettono sia risparmi sul costo e la frequenza dei lavori, sia un'organizzazione migliore dei lavori stessi, che intralciano meno l'attività degli occupanti. Nel complesso, questi processi di efficientamento portano a una riduzione dei costi di operazione pari al 9% rispetto a edifici in cui il DT è assente e un'occupazione degli edifici del 3,5% maggiore rispetto a immobili sprovvisti di questi elementi [18].

Gli utenti dell'edificio trovano un'ampia gamma di vantaggi nella presenza di un DT. A partire dai locatari dell'immobile, la presenza di un DT può,

ad esempio, migliorare sensibilmente tutti gli aspetti riguardanti la sicurezza dello stabile. Un DT può eseguire simulazioni sulle evacuazioni, identificare in maniera più precisa i fattori di rischio e i guasti anche nei sistemi di sicurezza stessi, aiutando a prevenire situazioni di emergenza e migliorando la risposta a esse. Il DT può contribuire a creare un ambiente più soddisfacente sia per chi ci lavora, sia per gli utenti in generale, aumentando a tutto tondo la produttività. Tutto ciò che abbiamo elencato porta, nel complesso, anche a un miglioramento nell'immagine dell'azienda che usa DT, che risulta più attraente per nuovi impiegati e clienti. Infine, l'elevata predisposizione dei DT all'integrazione con dispositivi e tecnologie IoT porta a interessanti interazioni fra l'edificio e i suoi utenti, non solo migliorando il modo in cui si interfacciano all'immobile, ma con la possibilità di creare funzionalità non ancora esistenti.

Tutti i vantaggi indicati presentano un trend interessante, dove lo sviluppo di nuove tecnologie e la quantità di dati raccolti dal DT portano a benefici via via maggiori. Agli estremi di questo schema possiamo notare come i benefici che gli utenti trovano in un DT verranno attivati molto più dalla quantità di dati rispetto alle tecnologie, che già oggi permetterebbero l'implementazione di nuove funzionalità per gli utenti. Al lato opposto, i vantaggi possibili per architetti e proprietari saranno attivati in misura molto maggiore dalla creazione di nuove tecnologie e tecniche per i DT. Ancora una volta possiamo evidenziare come lo sviluppo dei DT beneficia molto di più da un'implementazione progressiva, in quanto il tempo svolge una funzione chiave nell'attivare le potenzialità degli stessi.

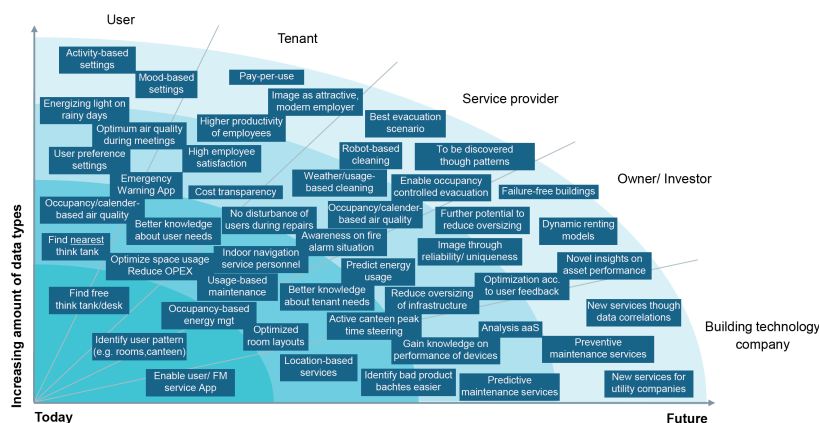


Figura 2.7: Grafico che raccoglie svariate possibili applicazioni dei DT. Gli assi del grafico sono rappresentati dalla quantità di dati disponibili (asse y) e dalle tecnologie disponibili, intese come disponibili oggi o in un futuro più o meno prossimo (asse x). I vantaggi sono anche stati divisi in base agli stakeholder interessati[18].

Per concludere questa capitolo sui DT, è importante fare una considerazione sui risultati raggiungibili in futuro attraverso questa tecnologia. Una delle sfide principali che la società umana dovrà affrontare nel futuro prossimo sarà quella di invertire i danni ambientali procurati nei recenti secoli di storia. In tutti i campi di ricerca e sviluppo l'attenzione all'ambiente riveste un ruolo chiave, e i DT non fanno eccezione. Abbiamo visto come un DT al massimo stadio di sviluppo renda di fatto l'edificio uno "smart building", provvisto di svariati dispositivi e sistemi interconnessi con un preciso scopo e risultati definiti e quantificabili [10]. Collegare in rete numerosi edifici provvisti di questo strumento spianerebbe la strada alla realizzazione di una smart city, aumentando in scala tutti i vantaggi che già abbiamo elencato per i DT, e aggiungendo possibilità e soluzioni altrimenti irrealizzabili. Con le opportune modifiche, una rete di DT potrebbe diventare essa stessa una sorta di enorme DT di città, regioni o intere nazioni. Nel Regno Unito ci si sta già muovendo verso questa direzione, come dimostra il documento dei "Gemini Principles" che abbiamo già citato. Questo documento è stato redatto con lo scopo di promuovere una visione comune nella creazione dei DT, in modo che gli sforzi convergano fin dall'inizio verso l'instaurazione di una rete a livello nazionale, chiamata "National Digital Twin" (NDT). Secondo il CDBB, l'instaurazione di questa rete è la risposta giusta nei confronti delle principali sfide che il continuo sviluppo della società umana sta portando, non solo ambientali, ma anche economiche e sociali.

Capitolo 3

Un caso di studio: Digital Twin di un sistema di riscaldamento

Come esposto nell'introduzione, lo scopo di questa tesi è esplorare l'applicazione del concetto di DT agli edifici abitativi. Nel capitolo precedente abbiamo osservato come i DT siano stati, almeno finora, applicati principalmente in ambiti industriali, manifatturieri e per costruzioni non di tipo abitativo. Esplorare un ambito diverso, con requisiti e scopi differenti, può portare vantaggi significativi allo sviluppo dei DT, anche per poter sondare quelli che potrebbero essere i limiti di questo strumento. Per farlo è stato creato un prototipo di DT di un'abitazione. In questo capitolo viene esposto il processo decisionale che ha portato alla scelta del contesto del caso di studio, la successiva modellazione dell'oggetto fisico e la progettazione dell'architettura della soluzione.

3.1 Scelta e analisi del caso di studio

Come abbiamo visto nella sezione 1.2, nella creazione di un DT è importante identificarne il contesto di realizzazione e lo scopo. Trattandosi della creazione di un prototipo, senza la pretesa di essere il DT completo di un'abitazione, è bene limitare il caso di studio a un sottoinsieme delle proprietà di un'abitazione. Considerato il focus sulla sostenibilità ambientale, espresso più volte finora, come caso di studio è stato scelto il sistema di riscaldamento di un'abitazione.

I sistemi di riscaldamento sono indubbiamente un'importante fonte di consumo nelle abitazioni, specialmente in inverno, e una scorretta gestione degli stessi può portare a ingenti sprechi. Monitorare i dati relativi a questi impianti potrebbe aiutare nella gestione degli stessi, rendendone più efficiente l'uso. Inoltre, a livello pratico, offre la possibilità di lavorare con una buona varietà di dati e modellare un sistema non banale.

Pertanto lo scopo prefissato per questo prototipo sarà quello di raccogliere i dati relativi al sistema di riscaldamento, in particolare temperature in vari punti della casa, consumi della caldaia e occupazione delle stanze per verificare una corretta accensione dei termosifoni. Inoltre, si vogliono offrire degli strumenti per monitorare tali dati in maniera agevole.

3.2 Modellazione del caso di studio

Il passo successivo nella creazione del prototipo è stato quello di identificare le entità coinvolte in questo sistema, identificando le relazioni fra esse e i dati a loro collegati, così da poter creare il modello digitale dell'impianto di riscaldamento. In questa fase della progettazione è importante tenere a mente le proprietà fondamentali di *reflection* e *rappresentatività e contestualizzazione*. I modelli creati dovranno essere coerenti con il contesto associato al DT, ossia il sistema di riscaldamento, e al tempo stesso riflettere nel modo migliore possibile la natura dell'oggetto fisico.

La prima entità identificabile è sicuramente quella della casa, che rappresenta il contenitore principale di tutte le altre entità. Aniché associare alla casa un'unica temperatura interna, risulta più utile ai fini dello scopo di questo prototipo monitorare le temperature all'interno di ciascuna stanza. Pertanto, alla casa sarà associata solo la temperatura esterna e la presenza di occupanti nella casa, potenzialmente utile per abilitare un'eventuale automatizzazione dell'accensione del riscaldamento.

Per monitorare la temperatura in ogni stanza è necessario modellare come entità le stanze stesse. In questo modo è possibile monitorarne anche l'uso in maniera indipendente l'una dall'altra, così da verificare l'efficienza dell'accensione dei termosifoni. La temperatura all'interno della stanza può essere rilevata in due modi differenti, vicino o lontano dalla finestra. In questo modo si possono ottenere informazioni sullo stato dell'isolamento termico di muri o vetri, o identificare comportamenti anomali, come una finestra aperta per troppo tempo.

La caldaia è l'entità centrale di un impianto di riscaldamento, pertanto è di fondamentale importanza mantenerne efficiente il funzionamento, per prevenire sprechi e guasti. Mancando una preparazione tecnica sul funzionamento delle caldaie, i modi più semplici per identificare malfunzionamenti e anomalie consistono nel rilevare il consumo di gas combustibile, monitorare la temperatura dell'acqua e inserire un allarme per valori fuori norma nei gas di scarico.

Altre entità importanti da modellare sono i termostati e i termosifoni. Sebbene all'interno di un prototipo semplice come questo non contribuiscano ad

apportare una grande quantità di dati, rendono comunque più completa e precisa la rappresentazione del PO, anche nell’ottica di future espansioni del progetto. Nei termostati verrà rilevata la temperatura desiderata impostata, mentre nei termosifoni verrà monitorato lo stato di accensione o spegnimento, valore che assume un’importanza maggiore nel caso in cui siano installati termosifoni indipendenti in ciascuna stanza.

Entità	Proprietà	Eventi	Relazioni
Casa	<ul style="list-style-type: none"> • Temperatura esterna; • Casa occupata. 	-	<ul style="list-style-type: none"> • Contiene stanza; • Contiene caldaia;
Stanza	<ul style="list-style-type: none"> • Temperatura interna; • Temperatura vicino alla finestra; • Stanza occupata. 	-	<ul style="list-style-type: none"> • Contiene termostato; • Contiene termosifone.
Caldaia	<ul style="list-style-type: none"> • Temperatura dell’acqua; • Temperatura desiderata dell’acqua; • Consumo gas combustibile. 	Allarme valori gas di scarico fuori norma.	<ul style="list-style-type: none"> • Connessa al termostato.
Termostato	<ul style="list-style-type: none"> • Temperatura desiderata. 	-	<ul style="list-style-type: none"> • Connesso al termosifone.
Termosifone	<ul style="list-style-type: none"> • Stato di accensione. 	-	-

Figura 3.1: Tabella rappresentante il metamodello del caso di studio. Vengono elencate le entità con le proprietà ed eventi a esse associati. Sono presenti anche le relazioni fra le entità identificate.

Il sistema di riscaldamento risulta modellato da una varietà di entità differenti, che abilita la creazione di più oggetti virtuali associati allo stesso oggetto fisico. Questa caratteristica ricade nella proprietà di *composability* dei DT, che dovrà essere tenuta a mente anche nella progettazione della soluzione, per permettergli di interagire con altri DT. Un altro vantaggio di questo tipo di modellazione, rispetto a una dove tutti i dati vengono raccolti e visualizzati in un’unica entità, è quello di permettere una maggiore flessibilità nella costruzione del DT. Tramite questo set di modelli si può, infatti, modellare il sistema di riscaldamento di una casa con un numero variabile di stanze, un condominio con più unità abitative e caldaie, una stanza con più termosifoni ecc. Teoricamente è possibile anche creare il DT dei sistemi di riscaldamento di tutte le case in una via o un isolato.

3.3 Una proposta architettonica

Una volta stabilite le entità attraverso le quali modellare il sistema di riscaldamento della casa, sono state analizzate le possibili scelte riguardanti l’architettura del DT stesso.

Per prima cosa, è stato necessario stabilire se progettare il sistema per essere hostato su un server locale o cloud. Un host locale del gemello digitale potrebbe fornire tempi di accesso tempestivi in caso di situazioni d'emergenza, ma in questo caso di studio le possibili situazioni d'emergenza non richiedono necessariamente tempi di risposta nell'ordine dei secondi. In situazioni come, ad esempio, il guasto di una caldaia, anche in caso di un avviso quasi in tempo reale, non sarebbe possibile risolvere il problema in tempi brevi, dovendo comunque coinvolgere l'intervento di un tecnico specializzato. Lo scopo principale di un DT in un'abitazione è quello di monitorare i dati in modo da renderne più efficiente l'utilizzo, e di individuare la necessità di una manutenzione possibilmente prima che avvenga il guasto. Pertanto i tempi di risposta dell'architettura cloud sono più che sufficienti per garantire il corretto funzionamento del sistema, e non minano l'*entanglement* fra l'oggetto fisico e gli oggetti logici nel DT. Inoltre, l'architettura cloud fornisce una serie di benefici che un host locale non potrebbe garantire. Innanzitutto una maggiore robustezza del sistema, in quanto un guasto della macchina host porterebbe sicuramente a uno spegnimento totale. Questo aspetto garantisce l'*accountability* e la *persistenza* del DT. Inoltre, anche il costo dell'implementazione risulta sensibilmente inferiore. Il pagamento sotto forma di abbonamento ai servizi cloud rappresenterebbe una spesa decisamente inferiore, soprattutto nel momento dell'installazione del sistema, non dovendo comprare la macchina host e i software fondamentali, come quelli per la protezione da attacchi informatici. Con l'architettura cloud il DT risulta quindi a tutto tondo più sicuro, robusto e conveniente, rendendone anche più facile l'utilizzo e l'accesso da remoto anche a utenti non esperti nel campo dell'informatica.

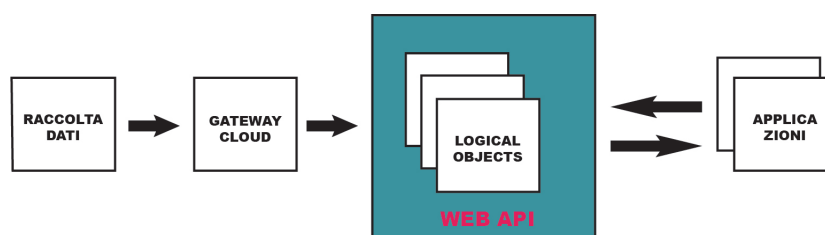


Figura 3.2: Schema dell'architettura della soluzione.

Stabilita la creazione di un DT su cloud, e alla luce di quanto appreso nello studio presentato nel capitolo 1, è stato possibile in fase di analisi identificare gli elementi chiave che avrebbero composto il progetto, delineando un'architettura di base da seguire nella fase implementativa. Come si può vedere in figura 3.2, il progetto sarà composto dai seguenti elementi principali:

1. Un sistema per generare e collezionare dati. Considerati i costi e il tempo che richiederebbe l'implementazione di una vera e propria WSN, questo elemento, ai fini dello scopo del prototipo, può essere implementato anche in forma di simulatore;
2. Un gateway cloud, che si occupi di ricevere i dati dall'elemento precedente e di gestirne l'invio ai LO. Questi due elementi incarnano il collegamento fra il PO e i LO del gemello digitale;
3. La virtualizzazione dell'oggetto fisico, basata sul modello creato nella sezione 3.2. Sarà composto da un insieme di LO che replicano lo stato dei vari elementi del PO;
4. Interfaccia API per l'interrogazione dei LO, sono un elemento fondamentale per permettere alle applicazioni di interrogare i LO e aggiungere funzionalità al DT;
5. Le applicazioni, che usano i dati ottenuti interrogando i LO tramite le API per fornire servizi.

Nonostante la WSN verrà solo simulata nell'ambito di questo progetto, si è ritenuto comunque utile ipotizzarne una possibile architettura, per facilitare la fase di implementazione del simulatore, senza tuttavia scendere nel dettaglio, in quanto lo studio della rete di sensori è del tutto secondario all'interno di questa tesi. La WSN ipotizzata presenta un'architettura a stella, con un nodo intermedio per ogni stanza che si occupi di registrare tutti i dispositivi al suo interno, e un gateway centrale che comunichi con i nodi intermedi e la parte cloud del DT.

Per quanto riguarda le API, dato che gli oggetti virtuali saranno salvati su cloud, devono necessariamente avere un'architettura compatibile con protocolli di comunicazione web. Dovranno permettere di leggere tutti i dati presenti nei LO, e anche di aggiornarne i valori, sia per poter caricare i valori inviati dal PO, sia per permettere alle applicazioni di influenzare il comportamento del PO, qualora sia previsto tra le funzionalità del DT che viene prodotto.

Infine, le applicazioni sono l'elemento che fornisce tutte le funzionalità che un DT possiede in aggiunta a quelle del PO. Senza di esse non sarebbe nemmeno possibile visualizzare lo stato dei LO. Le proprietà di *memorizzazione*, *augmentation*, *ownership*, *servitization* e *predictability* dipendono in larga parte, se non esclusivamente, dalle applicazioni che vengono sviluppate per il DT. Nel contesto di questo prototipo, l'obiettivo è di presentare una semplice applicazione che permetta di monitorare i dati in maniera facilmente leggibile. Grazie alla presenza delle API, però, è possibile creare qualsiasi applicazione inerente al contesto in cui il DT viene creato.

Capitolo 4

Implementazione del prototipo con Azure Digital Twins

In questo capitolo viene esposta l'implementazione finale del prototipo progettato nel capitolo precedente, partendo dalla scelta della piattaforma Azure per lo sviluppo del progetto, proseguendo con la modellazione in DTDL del caso di studio e la creazione con Azure Digital Twins Explorer dei LO. Viene infine esposta l'architettura finale del progetto e il suo funzionamento.

4.1 La piattaforma Microsoft Azure

Considerata l'architettura progettata nella sezione 3.2, è stata scelta Microsoft Azure come piattaforma cloud per lo sviluppo della soluzione. Azure offre centinaia di strumenti per la creazione, sviluppo e gestione di applicazioni cloud. In particolare, Azure è una delle piattaforme più utilizzate per la creazione di DT basati su cloud, in quanto offre guide, supporto e risorse specifiche per l'IoT e i Digital Twin. Tra queste vi è Azure Digital Twins (ADT), una platform-as-a-service [16] che fornisce API per caricare modelli, creare grafi di gemelli digitali e relazioni fra essi, interrogare il grafo con query SQL, leggere e modificare i valori delle proprietà dei gemelli e interagire con gli eventi generati dai gemelli stessi, chiamate da Azure "route eventi". Queste API seguono i principi architetturali noti come REST, in quanto permettono al client che ne fa richiesta l'accesso alle risorse (in questo caso i gemelli) a partire dall'URI dell'istanza di ADT, tramite connessioni stateless. In particolare ADT usa il protocollo HTTP per lo scambio di messaggi, scritti in formato JSON. Azure permette anche di gestire tutti gli elementi del progetto attraverso il Portale Azure. Il portale è accessibile tramite browser web e fornisce l'interfaccia utente per la gestione delle risorse, facilitando la connessione fra i vari elementi della soluzione e permettendo di monitorarne costantemente lo

status e i messaggi scambiati. Avendo scelto questa piattaforma per lo sviluppo del progetto, la sua implementazione verrà necessariamente influenzata dal funzionamento della piattaforma. All'interno delle prossime sezioni verranno evidenziati i punti in cui determinate scelte sono state effettuate per rispondere alle esigenze della piattaforma Azure.

4.2 Traduzione dei modelli in linguaggio DTDL

I modelli per i Gemelli Digitali di Azure vengono definiti tramite DTDL (Digital Twin Definition Language), linguaggio basato su JSON-LD. DTDL permette di rappresentare gli oggetti fisici come forme di dati che ne racchiudono il nome, i dati, gli eventi e i comandi a esso associati, in maniera analoga a una classe in un linguaggio orientato agli oggetti. DTDL non viene utilizzato esclusivamente nell'ambito di Azure Digital Twins, pertanto alcune delle funzionalità del linguaggio non sono attualmente supportate nella creazione di DT con Azure. Nell'ambito di questa tesi vengono trattati, quindi, solo gli aspetti del linguaggio rilevanti per l'attività svolta.

Gli elementi principali che costituiscono un modello in DTDL sono:

- **Proprietà:** dati che rappresentano lo stato di un'entità. Vengono archiviate direttamente su ADT e possono essere lette in qualsiasi momento;
- **Telemetrie:** dati che rappresentano, solitamente, la lettura di qualche sensore. Al contrario delle proprietà, non vengono salvati direttamente su ADT, ma sono degli eventi che devono essere gestiti nel momento in cui si verificano attraverso la creazione di event routes, per poter poi essere inviati alle applicazioni interessate ai dati in questione;
- **Relazioni:** permettono alla soluzione di essere rappresentata sotto forma di grafo, modellando le relazioni fra gli oggetti del contesto in cui viene creato il DT;
- **Componenti:** permettono la creazione di entità che non necessitano di essere definite in maniera indipendente, ma che sono comunque parte integrante della soluzione.

Questi elementi vengono inseriti nel campo `contents`, come matrice di definizioni di attributi. Gli altri campi importanti del modello sono `@id`, che identifica univocamente il modello, e `@type`, che definisce il tipo di appartenenza del modello. I modelli creati nell'ambito del progetto sono tutti di tipo *interface*, in quanto verranno usati per istanziare un qualsiasi numero di oggetti logici, ciascuno dei quali rappresenterà un oggetto fisico aderente al modello.

```
"@id": "dtmi:thesis:digital_house:room;1",
"@type": "Interface",
"displayname": "Room - Interface Model",
"@context": "dtmi:dtdl:context;2",
```

Listato 4.1: Esempio di intestazione del modello per l'entità Room

Tutte le entità identificate nel capitolo precedente (3.2) sono state modellate in DTDL identificando il `@type` dei dati di cui è stata prevista la raccolta. Dati come stato di occupazione della stanza o temperatura impostata sul termostato rappresentano più lo stato dell'entità di riferimento che letture di sensori, e sono stati, quindi, modellati come proprietà. Viceversa, valori come la temperatura esterna della casa o quella dell'acqua nella caldaia sono state identificate come telemetrie, in quanto ottenibili tramite letture di sensori e con valori altamente variabili. Unico caso particolare risiede nella classificazione dello stato di accensione dei termosifoni come telemetria anziché proprietà. Questa scelta è stata effettuata per permetterne la lettura a un'applicazione esterna al DT e quindi monitorare la sua variazione nel tempo, in particolare in associazione alla fluttuazione delle temperature. La sua classificazione come proprietà non avrebbe permesso questo confronto, in quanto solo il suo ultimo valore sarebbe stato salvato sul DT.

```
"contents": [
  {
    "@type": "Property",
    "name": "RoomId",
    "schema": "string",
    "writable": true
  },
  {
    "@type": "Property",
    "name": "floor",
    "displayName": "Floor",
    "schema": "integer",
    "writable": true
  },
  {
    "@type": "Property",
    "name": "room_occupied",
    "displayName": "Room occupied",
    "schema": "boolean",
    "writable": true
  }
],
```

```
{
  "@type": ["Telemetry", "Temperature"],
  "name": "room_temperature",
  "displayName": "Room Temperature",
  "schema": "double",
  "unit": "degreeCelsius"
},
{
  "@type": ["Telemetry", "Temperature"],
  "name": "window_temperature",
  "displayName": "Window Temperature",
  "schema": "double",
  "unit": "degreeCelsius"
},
}
```

Listato 4.2: Campo contents dell'entità Room

Infine, tutti i modelli, sempre all'interno del campo `contents`, prevedono l'inserimento di relazioni con altri modelli per permettere la costruzione di un grafo con le singole istanze delle entità. Le relazioni sono le stesse elencate in figura 3.1.

```
{
  "@type": "Relationship",
  "@id": "dtmi:thesis:digital_house:room:has_thermostat;1",
  "name": "has_thermostat",
  "displayName": "Contains ",
  "target": "dtmi:thesis:digital_house:thermostat;1"
},
{
  "@type": "Relationship",
  "@id": "dtmi:thesis:digital_house:room:has_heater;1",
  "name": "has_heater",
  "displayName": "Contains ",
  "target": "dtmi:thesis:digital_house:heater;1"
}
]
```

Listato 4.3: Esempio delle relazioni dell'entità Room

4.3 Creazione del grafo con Azure Digital Twins Explorer

Come esposto nella sezione 4.1, i LO vengono creati tramite ADT, istanziabile attraverso il Portale Azure. Qui vengono fornite le stringhe per connettersi alla risorsa e si organizzano gli endpoint per inoltrare i dati di telemetria a applicazioni esterne. Azure mette a disposizione molti modi per interagire con l'istanza di ADT tramite le sue API. Uno di questi è Digital Twins Explorer, un'interfaccia grafica accessibile tramite browser che si collega ad ADT e permette di eseguire quasi tutte le operazioni disponibili tramite le API, visualizzandone al tempo stesso gli effetti sul DT. Con DTE si possono caricare i modelli DTDL e generare gemelli a partire da essi. I gemelli possono essere collegati fra loro seguendo le relazioni definite nei modelli, andando a creare un grafo della virtualizzazione del PO. DTE permette anche di visualizzare le proprietà di ciascun gemello e i suoi valori attuali, o di inserirne di nuovi. Infine, DTE permette anche di eseguire query SQL sul grafo e salvarne un set desiderato, per poterle rieseguire a piacimento. Le query eseguite sul grafo permettono, ad esempio, di generare viste del grafo evidenziando solo le istanze di uno specifico modello, o solo i gemelli con un determinato valore di una delle loro proprietà, senza dover cambiare il grafo per farlo.

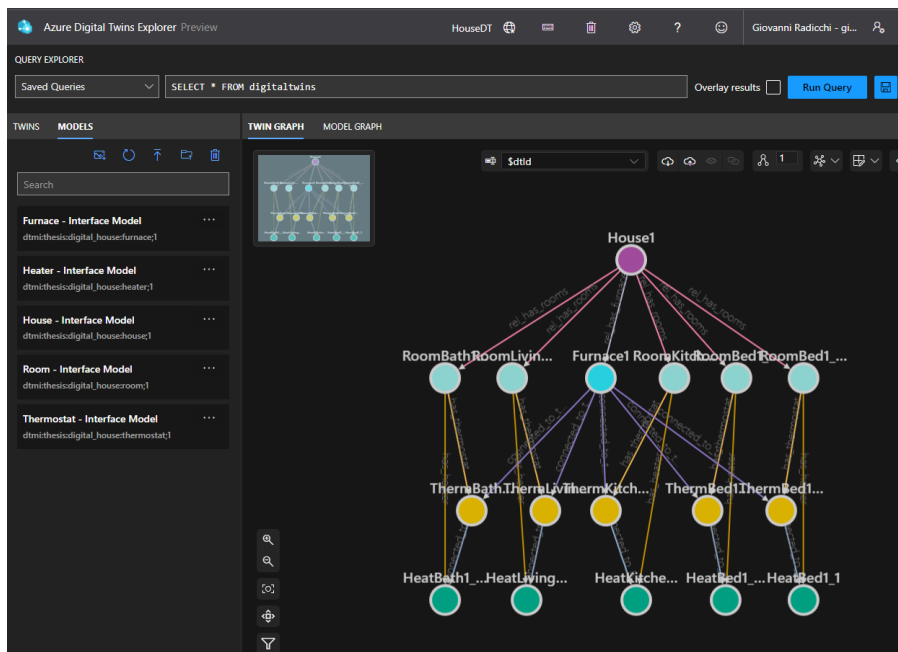


Figura 4.1: Grafo del gemello HouseDT come appare su Digital Twins Explorer

Il grafo mostrato in figura rappresenta il sistema di riscaldamento di una casa con 5 stanze. In esso sono presenti i gemelli della casa e della caldaia, delle 5 stanze e di un termostato e un termosifone per ciascuna stanza. Il loro insieme costituisce il gemello del sistema di riscaldamento che è stato simulato e monitorato in questo progetto.

4.4 Architettura della soluzione ed elementi principali

L'architettura proposta nella sezione 3.3 ha subito delle modifiche in fase implementativa, adattandosi alle risorse messe a disposizione da Azure, e al funzionamento delle stesse. L'architettura finale del prototipo è mostrata in figura 4.2.

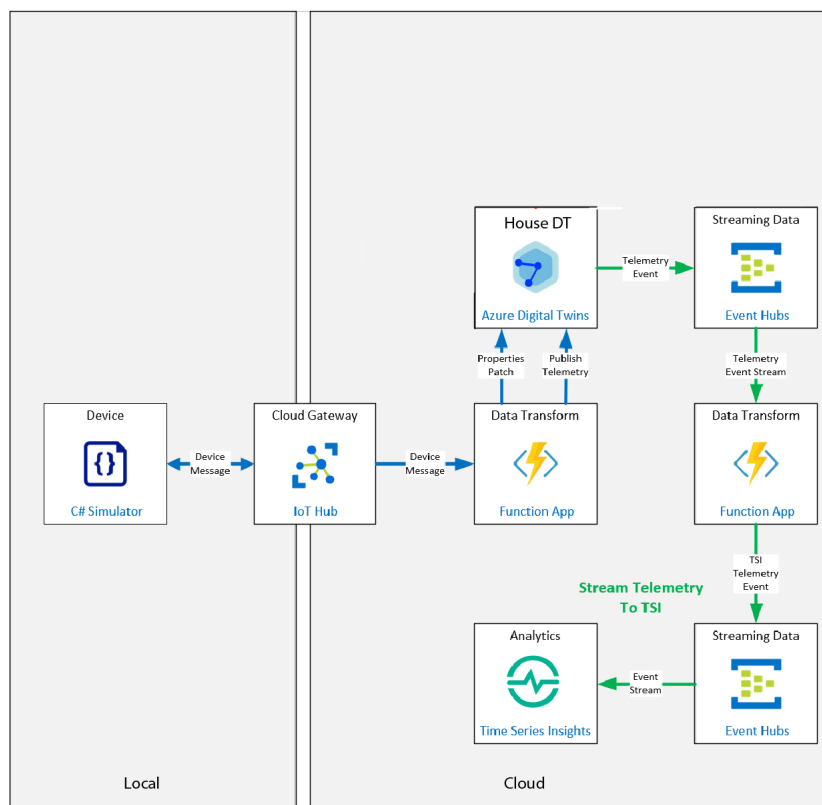


Figura 4.2: Schema dell'architettura della soluzione

Il progetto presenta un flusso lineare di dati che vanno dal programma in C#, che simula i dati che verrebbero invece raccolti da una rete di sensori

e device IoT, fino all'applicazione Time Series Insights, passando per l'istanza di ADT e una serie di altri elementi che svolgono compiti specifici nello smistamento e traduzione dei dati di proprietà e telemetrie.

4.4.1 Simulatore di rete di sensori e device IoT

Il simulatore si occupa di generare i dati che normalmente sarebbero rilevati dalla rete di sensori o da dispositivi come caldaia, termostati e termosifoni. Il programma simula anche i comportamenti che avrebbe un sistema dotato di smart things, accendendo e spegnendo i termosifoni quando le temperature raggiungono determinate soglie. Per l'invio dei dati, viene creato un oggetto `deviceClient` per ciascuno dei dispositivi di cui è stato creato un DT, che viene connesso al gateway cloud tramite una stringa di connessione specifica per ciascuno di essi. L'invio vero e proprio avviene lanciando un task che si riattiva ogni 5 secondi, raccogliendo i dati e spedendoli tramite il `deviceClient` corretto.

4.4.2 Gateway cloud

La funzione di gateway cloud viene svolta dall'IoT Hub, una risorsa di Azure che funge da hub centrale per i messaggi e le comunicazioni fra dispositivi IoT e applicazioni cloud. In questo progetto l'Hub non svolge nessun'altra funzione se non quella di inoltrare i messaggi ricevuti dai vari dispositivi, ma necessita comunque di una serie di impostazioni per svolgere il suo ruolo.

Per produrre le stringhe di connessione necessarie al simulatore per l'invio dei messaggi, devono essere registrati sull'Hub tutti i dispositivi che invieranno dati. La stringa è univoca per ciascun dispositivo, per cui l'Hub identifica il mittente grazie ad essa. Nel momento in cui i dati vengono inoltrati, l'ID dispositivo associato al messaggio sarà quello con cui il dispositivo è stato registrato nell'Hub.

Per l'inoltro dei messaggi nell'Hub deve essere registrato l'endpoint verso cui verranno spediti i dati. Si crea quindi una sottoscrizione di eventi che viene impostata per attivarsi ogni qualvolta il simulatore invia dei dati, e si associa una Function App come destinataria dei messaggi.

4.4.3 Funzione per la trasformazione dei dati

Le Function App di Azure sono una soluzione serverless per la gestione di eventi, di cui deve essere programmata solo la tipologia di evento trigger e la relativa risposta. Azure si occupa di fornire il server e tutti i servizi per mantenere il programma attivo e funzionante.

In questo progetto sono state create due Function App, e in questa sottosezione viene trattata quella che si occupa di reagire agli eventi dell'Hub IoT, trasmettendo i dati in arrivo ad ADT nella forma appropriata. Quando la funzione si attiva, crea un oggetto `DigitalTwinsClient` per gestire la connessione con ADT. Dopodiché, in base al tipo di dispositivo indicato dall'ID del mittente, trasforma i dati separando quelli relativi a proprietà del gemello da quelli delle telemetrie. I primi vengono scritti in un `JsonPatchDocument` in modo da rimpiazzare i dati già presenti e associando al dato il nome della proprietà corrispondente. Il `JsonPatchDocument` viene poi inviato al DT in maniera asincrona tramite il `DigitalTwinsClient` creato in precedenza. La sezione `body` del messaggio, che contiene i dati di telemetria, viene invece estratta e tradotta in un `JsonObject`, senza nessuna trasformazione. Questo perché l'istanza di ADT non salva i dati di telemetria, ma è impostato per inoltrarli ad applicazioni esterne, qualora li richiedessero. Se nessuna applicazione li richiede, i dati vanno perduti. Anche questi dati vengono inviati in maniera asincrona dal `DigitalTwinsClient`.

E' importante sottolineare che anche le Function App utilizzano le API REST per comunicare con ADT. I metodi della classe `DigitalTwinsClient` utilizzati incapsulano le chiamate HTTP e l'utilizzo delle API di ADT. Il comando corretto per l'inserimento dei dati deve essere incluso nel `JsonPatchDocument` che viene inviato. In questo caso si tratta del comando `replace`, che rimpiazza i dati presenti nei gemelli con i nuovi dati mandati dall'Hub IoT.

4.4.4 Elaborazione dei dati di telemetria

Azure mette a disposizione l'applicazione Time Series Insights per raccogliere, analizzare e visualizzare i dati delle telemetrie. Tramite TSI Explorer (accessibile via browser analogamente a DTE) permette di creare e salvare un set di serie temporali da visualizzare sotto forma di grafici. Nell'ambito di questo progetto, le serie temporali sono i flussi di dati in arrivo dai vari gemelli e ciascuna di esse corrisponde a un campo telemetria di uno dei gemelli. Questi set, chiamati viste, possono essere creati per soddisfare le esigenze di utenti differenti, come ad esempio il gestore dell'impianto di riscaldamento o il proprietario dell'abitazione.

Per far sì che i dati arrivino a questa applicazione è necessario impostare una pipeline di dati che prevede la realizzazione di una seconda Function App e di due Event Hub. Questi sono strumenti che permettono lo streaming di dati e la gestione di eventi in tempo reale. Il primo Event Hub viene impostato come endpoint per i dati di telemetria ricevuti dai gemelli. Le telemetrie vengono quindi inviate alla Function App, che in maniera simile a quella precedentemente analizzata, estrae i dati e li inserisce in un dizionario

che usa come chiave il nome delle telemetrie a cui i dati fanno riferimento. Il dizionario viene quindi serializzato in una stringa JSON e inviato al secondo Event Hub, che viene impostato come origine evento per la ricezione dei dati nell'applicazione TSI.

4.5 Test della soluzione e considerazioni sul suo funzionamento

Una volta completate tutte le componenti della soluzione, è stato possibile eseguire i test di funzionamento. Ogni volta che il simulatore viene lanciato i dati prodotti vengono inviati ai gemelli e all'applicazione TSI, permettendo di osservare e monitorare in qualsiasi momento lo stato del sistema.

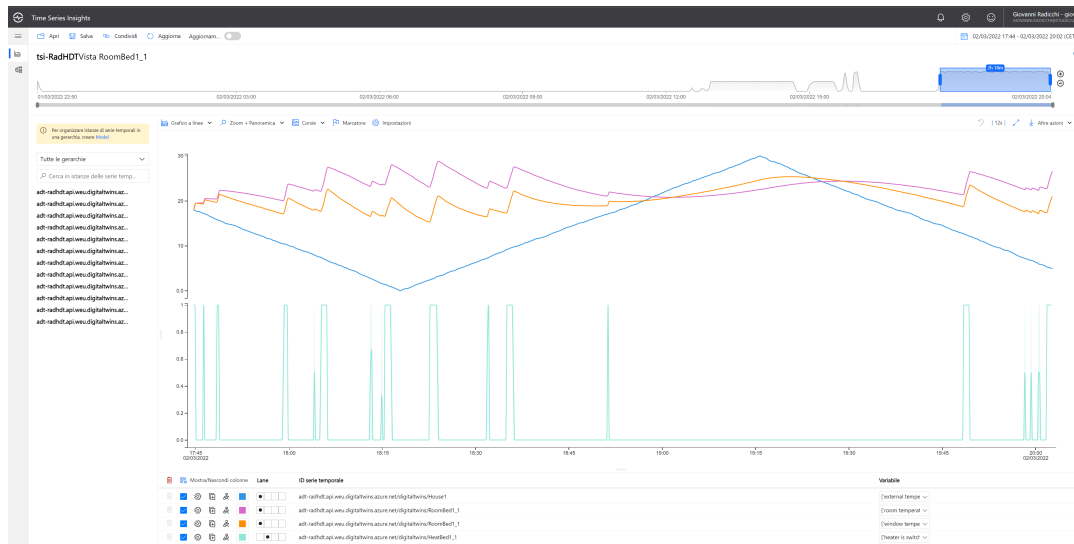


Figura 4.3: Vista di una delle stanze: in arancione la temperatura vicino alla finestra, in viola quella interna, in blu quella esterna alla casa. Il grafico sottostante indica i momenti in cui il termosifone nella stanza si è acceso

In figura 4.3 vengono mostrate le temperature relative a una delle stanze. Si può osservare l'aumento delle temperature interne alla stanza in corrispondenza dell'accensione dei termosifoni, o come i termosifoni non si accendano quando la temperatura esterna è più elevata. Trattandosi di dati simulati mancano di realismo, ma se fossero dati reali il DT avrebbe permesso di identificare una scarsa efficienza nell'isolamento termico dell'ambiente, evidenziata dalla grossa differenza di temperatura fra la zona attorno alla finestra e la zona interna della stanza. Questi risultati mettono in evidenza i potenziali benefi-

ci pratici che può portare la realizzazione di un DT per un'abitazione, ma si possono trovare vantaggi anche a livello di funzionalità.

La realizzazione della parte applicativa di questo progetto non è ottimizzata per l'utente finale, ma serve per mostrare principalmente le potenzialità delle applicazioni associate a un DT. In questo caso tutti i dati in uscita da ADT vengono inoltrati nella stessa pipeline, processati dalla stessa Function APP e mostrati in una sola applicazione. I dati potrebbero, però, essere inviati su più route differenti, ciascuna specifica per l'utente alla quale è destinata. Il gestore del DT potrebbe trovare utile visualizzare tutti i dati per accertarsi del funzionamento del sistema, mentre chi si occupa della manutenzione della caldaia potrebbe voler controllare solo i dati relativi a quell'oggetto, richiedendo la creazione di una pipeline specifica per i dati in arrivo dal suo gemello. Al tempo stesso l'utente della casa, sia esso un inquilino o il proprietario, potrebbe voler vedere comunque tutti i dati, ma su un'applicazione progettata ad hoc per essere user-friendly. Tutte queste richieste possono essere soddisfatte contemporaneamente, creando pipeline apposite. Il risultato è che un singolo DT può soddisfare un gran numero di richieste e necessità differenti, aumentando il proprio valore grazie alle applicazioni ad esso associate.

Conclusioni

Il prototipo realizzato all'interno di questa tesi risulta molto semplice e richiederebbe ancora molto lavoro per essere implementato efficacemente in un'abitazione reale, ma credo raggiunga appieno lo scopo prefissato. Gli strumenti forniti da Azure hanno permesso di implementare un sistema funzionante e robusto in maniera relativamente semplice e con dei costi accessibili. I concetti studiati nei primi capitoli hanno trovato la loro applicazione anche all'interno del contesto abitativo, che sembra prestarsi bene all'impiego dei DT. La creazione di una vera WSN e la realizzazione di un'applicazione per gli utenti in grado di visualizzare i dati a seconda del profilo inserito, insieme alla modellazione di più sistemi all'interno della casa, porterebbero a compimento la realizzazione di un DT partendo da questo prototipo. Il risultato sarebbe uno strumento utile e molto potente per ogni stakeholder associato a un'abitazione. Molti dei vantaggi che abbiamo visto associati ai DT di costruzioni più grandi possono essere applicati anche alle abitazioni: possibilità di programmare le manutenzioni e intervenire su guasti prima che si presentino; fornire servizi aggiuntivi che possano essere venduti agli utenti da chi costruisce o da chi possiede l'immobile; minori spese ed emissioni dovute ad un utilizzo più efficiente ecc. Soprattutto il risparmio energetico è un fattore importante. Considerato che gli edifici a scopo abitativo rappresentano una grossa fetta delle costruzioni all'interno delle città, rendere più efficiente ciascuno di essi rappresenterebbe un grosso passo avanti nella transizione verso città più verdi ed ecologiche.

D'altra parte, creare un DT per un'abitazione potrebbe non portare tutti i vantaggi ipotizzati. Il costo per l'utilizzo dei servizi Azure aumenta con volumi di dati maggiori, e per quanto risulti accessibile, si tratta comunque di una spesa consistente. Questo senza considerare il costo di mantenimento, anche dal punto di vista energetico, di una WSN e dei dispositivi IoT che verrebbero installati. Soprattutto in abitazioni piccole, come appartamenti o monolocali, la creazione di un DT potrebbe portare spese e consumi maggiori dei risparmi. Inoltre, gran parte dei vantaggi e delle funzionalità aggiuntive che un DT potrebbe portare potrebbero essere ottenute da una soluzione domotica, i cui costi potrebbero essere più contenuti. Anche dal punto di vista dei servizi offer-

ti, abbiamo visto come molti di essi siano più una prospettiva futura per i DT, che un risultato ottenibile a breve termine, fattore che potrebbe scoraggiare investimenti in questo campo.

In conclusione, ritengo che il breve studio e il prototipo realizzato offrano uno sguardo utile dell'applicazione dei DT in ambito abitativo. Questo campo potrebbe non essere il teatro principale di sviluppo dei DT, ma potrebbe comunque suscitare un discreto interesse qualora si tratti di edifici di grosse dimensioni, come ad esempio un condominio. Difatti le prospettive future dei DT sono senza dubbio promettenti. Non solo sono sistemi che diventano più efficaci e che generano più valore man mano che passa il tempo, ma il proseguo della ricerca e un mercato sempre più ampio porteranno alla creazione di piattaforme e soluzioni differenti, che potrebbero generare un abbassamento dei costi di realizzazione e un funzionamento migliore a tutto tondo. Una sempre più ampia diffusione dei DT permetterà la creazione di reti e interconnessioni fra essi, fornendo una base per lo sviluppo delle smart cities e altri sistemi avanzati.

Ringraziamenti

La stesura di questa tesi e la chiusura di questo percorso di laurea hanno rappresentato un ostacolo nella mia vita che, a tratti, sembrava quasi insormontabile. Non sarei mai riuscito ad arrivare alla fine senza tutte le persone che mi hanno accompagnato in tutti questi anni, ed è quindi il minimo che possa fare ringraziarle in questa occasione.

Desidero innanzitutto ringraziare il mio relatore professor Alessandro Ricci per tutto il supporto e la disponibilità mostrati, per aver fornito gli articoli e l'idea di dirigere la tesi sui Digital Twin, un argomento interessantissimo che non avrei esplorato senza il suo intervento.

Ringrazio inoltre tutti gli amici e i compagni di studio incontrati in questi anni. Ciascuno di voi ha svolto un ruolo importante nel rendere quest'esperienza indimenticabile ed edificante, soprattutto dal punto di vista umano. Anche se ci si perde o ci si perderà di vista, sono grato di aver conosciuto ciascuno di voi, di quello che abbiamo condiviso e di quello che mi avete dato come persone. In particolare ringrazio Marco, Giovanni, Marco, Lorenzo, Aurora, Piero, Giovanni e Andrea per la quantità incalcolabile di momenti condivisi, per le idee scambiate, per il supporto e il sostegno. Senza di voi non sarebbe valsa la pena arrivare a questo punto. Ringrazio inoltre Alessandra per aver condiviso moltissime idee e materiali sui BIM dal punto di vista di un architetto, che ha contribuito a dare solidità al suo confronto con i Digital Twin.

Ringrazio mia nonna Giancarla e mio zio Gilberto per aver revisionato la tesi e avermi messo in contatto con l'ingegner Giorgio Mazzola, che ringrazio per avermi fornito consigli sulla natura e le applicazioni dei BIM.

Infine, i ringraziamenti più importanti vanno ai miei genitori e a mia sorella. Più di chiunque altro siete stati presenti in ogni momento di difficoltà e incertezza, mai una volta è mancato il vostro sostegno, il vostro aiuto e il vostro incoraggiamento. Senza di voi, questa tesi non ci sarebbe, e di questo sappiate che avrete sempre la mia gratitudine.

Bibliografia

- [1] 01building. Il rapporto fra bim e digital twin. 10 2019.
- [2] A. Bolton, L. Butler, I. Dabson, M. Enzer, M. Evans, T. Fenemore, and F. Harradence. Gemini principles — centre for digital built britain, 2018.
- [3] Building Information Modeling. Building information modeling — Wikipedia, the free encyclopedia, 2022.
- [4] Digital Twin. Digital twin — Wikipedia, the free encyclopedia, 2022.
- [5] Edilportale. Smart building, oltre 8 miliardi investiti in italia nel 2019 ma solo 2 per soluzioni davvero ‘intelligenti’. 02 2021.
- [6] Edilportale. Smart building, polimi: ‘nel 2020 investimenti in calo dell’11% nonostante il superbonus’. 11 2021.
- [7] Mahmoud El Jazzar, Melanie Piskernik, and Hala Nassereddine. Digital twin in construction: An empirical analysis. 05 2020.
- [8] Simon Evans, Cristina Savian, Allan Burns, and Chris Cooper. Digital twins for the built environment - an introduction to the opportunities, benefits, challenges and risks. 10 2019.
- [9] Michael Grieves and John Vickers. *Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems*, pages 85–113. 08 2017.
- [10] Siavash Khajavi, Naser Hossein Motlagh, Alireza Jaribion, Liss Werner, and Jan Holmström. Digital twin: Vision, benefits, boundaries, and creation for buildings. *IEEE Access*, 7:147406 – 147419, 10 2019.
- [11] Qiuchen Lu, Ajith Kumar Parlikad, Philip Woodall, Gishan Ranasinghe, and James Heaton. Developing a dynamic digital twin at a building level: using cambridge campus as case study. pages 67–75, 01 2019.

-
- [12] Paola Mammarella. Domotica e impianti smart protagonisti della prestazione energetica degli edifici. 06 2020.
- [13] Alessandra Marra. Bim: a cosa serve e chi lo userà. lo stato dell'arte sul building information modeling, i vantaggi nell'utilizzarlo e i software compatibili. 10 2015.
- [14] Roberto Minerva, Gyu Myoung Lee, and Noel Crespi. Digital twin in the iot context: A survey on technical features, scenarios, and architectural models. *Proceedings of the IEEE*, PP:1–40, 06 2020.
- [15] Roland Rosen, Georg von Wichert, George Lo, and Kurt D. Bettenhausen. About the importance of autonomy and digital twins for the future of manufacturing. *IFAC-PapersOnLine*, 48(3):567–572, 2015. 15th IFAC Symposium on Information Control Problems in Manufacturing.
- [16] Sconosciuto. What is azure digital twins? 02 2022.
- [17] Mike Shafto, M Conroy, R Doyle, E Glaessgen, C Kemp, J LeMoigne, and L Wang. Modeling, simulation, information technology and processing roadmap, 05 2010.
- [18] Siemens. Digital twin – driving business value throughout the building life cycle, 09 2018.