

POLITECNICO DI TORINO

Corso di Laurea Magistrale
in Architettura Costruzione Città



Tesi di Laurea Magistrale

**Dal Building Information Modeling al
Building Energy Modeling:
VPL per la gestione dell'interoperabilità
mediante lo standard IFC**

Salvatore Magnano

Relatore

Prof.ssa Anna Osello

Correlatore

Ing. Matteo Del Giudice

Febbraio 2020

POLITECNICO DI TORINO

Corso di Laurea Magistrale
in Architettura Costruzione Città



Tesi di Laurea Magistrale

Dal Building Information Modeling al Building Energy Modeling: VPL per la gestione dell'interoperabilità mediante lo standard IFC

Relatore

Prof.ssa Anna Osello
firma del relatore

.....

Correlatore

Ing. Matteo Del Giudice
firma del correlatore

.....

Candidato

Salvatore Magnano
firma del candidato

.....

Febbraio 2020

“I try not to do anything that’s too close to what I’ve done before. And the nice thing is we have a big universe here. It’s filled with new ideas. All you have to do is grab them.”

Stan Lee

Ringraziamenti

Ringrazio la mia relatrice, la Professoressa Anna Osello per avermi dato l'opportunità di avventurarmi nel Building Information Modeling dandomi modo di migliorare le mie skills in questo ambito. Desidero ringraziare il mio correlatore Matteo Del Giudice, per essere stato sempre disponibile e per aver costantemente accolto le mie idee con entusiasmo. Un sentito ringraziamento va anche a Monica Dettori, per aver chiarito ogni mio dubbio le innumerevoli volte in cui ho avuto bisogno di un supporto, a Linda Falsini per avermi prestato un po' della sua scrupolosità in fase di stesura, e a tutto lo staff del laboratorio *DrawingTOthefuture*.

Ringrazio i miei amici e colleghi, quelli vicini e lontani. Voi, in un modo o nell'altro, avete reso migliori le mie giornate durante questi lunghi anni fatti di sforzi, notti e sacrifici. È anche grazie a voi se ho superato uno dei momenti più tristi e bui della mia vita, aiutandomi quando ne avevo più bisogno. Non smetterò mai di essere riconoscente a voi ed alla persona che, anche da lontano, mi è sempre stata vicina per tutti questi anni. Grazie per tutto quello che avete fatto per me, anche se in futuro le nostre strade si divideranno, io non vi dimenticherò mai.

Il ringraziamento più grande va ai miei due soli. È grazie ai miei genitori se ho potuto continuare a coltivare i miei sogni e quello in cui credo. Voi mi avete dato l'opportunità di sbagliare e rimediare ai miei errori, mi avete sempre sostenuto nelle difficoltà ed avete sempre dato il massimo per me. Per tutta la mia vita voi siete stati le mie certezze ed anche se uno di voi non potrà leggere queste parole, ci tengo a dirvi che a voi devo tutto.

Increasingly alarming data affected the climate of our day. A study carried out by the Intergovernmental Panel on Climate Change has confirmed that by 2050 we will see an increase in the average temperature of 1.5 ° C. This increase is mainly due to the increase in CO₂ levels in the atmosphere, which keeps trapping heat in the atmosphere, causing the temperature to rise. At present the world-building park is responsible for around 40% of greenhouse gas emissions, therefore it has now become a necessity to design new buildings that are much more perfect in thermal performance. In parallel, the construction sector may be available for new tools, which allow impossible processes and results until a few decades ago. In this sense, the new building design can be understood as a set of processes that puts various professional figures of the construction sector in close contact. The BIM methodology presents an innovative medium that optimizes the sharing of data and information between different actors involved. Thanks to this aspect, it is now possible to carry out various specific analyzes such as those of the energy type. This thesis offers an assessment of the current degree of interoperability between BIM and BEM, a comparison between the capacity of different software in the sector and a solution to possible interoperability problems using tools based on visual programming languages.

Dati sempre più allarmanti riguardano il clima dei nostri giorni. Uno studio svolto dall'Intergovernmental Panel on Climate Change ha confermato che entro il 2050 assisteremo ad un aumento della temperatura media complessiva pari ad 1,5 C°. Tale aumento è dovuto soprattutto all'innalzamento dei livelli di CO2 nell'atmosfera, che contribuisce ad intrappolare il calore nell'atmosfera provocando l'innalzamento delle temperature. Allo stato attuale il parco edilizio mondiale è responsabile del 40% circa delle emissioni di gas serra, ormai è quindi diventata una necessità progettare nuovi edifici che siano molto più performanti dal punto di vista energetico. In parallelo il settore edilizio può avvalersi di strumenti nuovi, che permettano processi e risultati impossibili fino a qualche decennio fa. In questo senso la nuova progettazione edilizia può essere intesa come un insieme di processi che mette a stretto contatto varie figure professionali del settore delle costruzioni. La metodologia BIM rappresenta un mezzo innovativo che ottimizza la condivisione dei dati e delle informazioni fra diversi attori coinvolti. Grazie a questo aspetto è adesso possibile effettuare diverse analisi specifiche come quelle di tipo energetico. Questa tesi offre una valutazione sul grado attuale dell'interoperabilità fra BIM e BEM, una comparazione fra le capacità di diversi software del settore ed una soluzione ad eventuali problemi di interoperabilità avvalendosi di strumenti basati su linguaggi di programmazione visuale.

1	Introduzione	21
1.1	Building Information Modeling	21
1.1.1	Definizione Building Information Modeling	21
1.1.2	Dal Computer Aided Design al Building Information Modeling	24
1.1.3	BIM to BEM per l'efficienza energetica degli edifici	26
1.2	Quadro Normativo BIM	28
1.2.1	ISO 19650	28
1.2.2	Normativa italiana in ambito BIM	34
1.3	Quadro Normativo Energia	39
1.3.1	Direttiva (UE) 2018/844	40
1.3.2	ISO 52016	40
1.3.3	Recepimento nazionale delle normative europee	42
1.3.4	UNI EN ISO 52016	44
1.3.5	UNI TS 11300	45
1.3.6	Decreto Ministeriale 26 Giugno 2015	47
1.4	Comitato Termotecnico italiano	49
1.4.1	Validazione dei software per la prestazione energetica degli edifici	49
1.5	Procedure di calcolo e formati di scambio	54
1.5.1	Dalla modellazione alle analisi tecniche	54
1.6	Green Building eXtensive Markup Language	56
1.6.1	Background	56
1.6.2	Extensive Markup Language	56
1.6.3	Funzionamento gbXML	57
1.7	Industry Foundation Classes	58

1.7.1	Background	58
1.7.2	ISO 16739:2013	59
1.7.3	Definizione IFC	59
1.7.4	Funzionamento IFC	60
1.8	Linguaggi di programmazione e programmazione visuale	68
1.8.1	VPL e progettazione parametrica in BIM	69
1.8.2	Linguaggio di programmazione	70
2	Metodologia	72
2.1	Caso Studio	72
2.1.1	Inquadramento caso studio	72
2.1.2	Modello BIM caso studio	73
2.1.3	Output caso studio: gbXML e IFC	73
2.2	Workflow BIM to BEM	75
2.2.1	Valutazione 1: Predisposizione all'importazione dei dati	75
2.2.2	Valutazione 2: Grado di interoperabilità formati di scambio	79
2.2.3	Valutazione 3: Conformità alla normative vigenti	100
2.3	Esportazione IFC e configurazione formato di scambio	101
2.3.1	Impostazioni per esportazioni personalizzate	102
2.3.2	Network IFC	107
2.4	VPL e codifica: realizzazione algoritmo	110
2.4.1	Obiettivi e funzioni algoritmo	111
2.4.1.1	Lettura File IFC	111
2.4.1.2	Lettura proprietà da modello Revit	114
2.4.1.3	Scrittura stringhe in conformità con lo standard IFC	116
2.4.1.4	Scrittura nuovo file IFC	118

2.4.2	Miglioramento algoritmo	119
2.4.2.1	Miglioramento prestazioni	119
2.4.2.2	Debugging algoritmo	120
2.4.2.3	Interfaccia Dynamo Player	122
3	Risultati e Criticità	124
3.1	Risultati	124
3.1.1	Interoperabilità BIM to BEM	124
3.1.2	Applicazioni algoritmo	125
3.2	Criticità	126
3.2.1	Flusso dati	126
3.2.2	Industry Foundation Classes	127
4	Conclusioni e Sviluppi Futuri	128
5	Bibliografia e sitografia	129
6	Allegati	135

Acronimo <i>Acronym</i>	Italiano	English
ACDat <i>CDE</i>	Ambiente Condivisione Dati	Common Data Environment
AEC		Architecture Engineering Construction
ANCC	Associazione Nazionale per il Controllo della Combustione	
ANTA	Associazione Nazionale per la Tutela dell'Ambiente	
<i>ASCII</i>		American Standard Code for Information Interchange
ATI	Associazione Termotecnica Italiana	
<i>BEM</i>		Building Energy Modeling
BIM	Building Information Modeling	Building Information Modeling
<i>CAD</i>		Computer Aided Design
CEN	Comitato Europeo di Normazione	European Committee for Standardization
CTI	Comitato Termotecnico Italiano	
<i>DOE</i>		Department Of Energy
<i>EP</i>		Energy Plus

Indice degli acronimi

gbXML		Green Building XML
<i>GUI</i>		Graphical User Interface
<i>IAI</i>		International Alliance for Interoperability
<i>IDE</i>		Integrated Development Environment
<i>IFC</i>		Industry Foundation Classes
<i>IPCC</i>	Gruppo Intergovernativo sul Cambiamento Climatico	Intergovernmental Panel on Climate Change
ISO	Organizzazione internazionale per la normazione	International Organization for Standardization
LOD	Livello di sviluppo degli Oggetti Digitali	Level Of Detail
<i>NIBS</i>		National Institutes of Building Science
<i>SaaS</i>		Software as a Service
<i>STEP</i>		Standard for The Exchange of Product model data
UNI	Ente Nazionale Italiano di normazione	
VPL	Linguaggio di Programmazione Visuale	Visual Programming Language
<i>XML</i>		eXtensible Markup Language

Figura 1.	Schema delle 7 dimensioni del BIM	21
Figura 2.	Grafico MacLeamy	23
Figura 3.	Confronto progettazione tradizionale e metodologia BIM	24
Figura 4.	Grafico Bew-Richards	25
Figura 5.	Schema normativa ISO 19650	29
Figura 6.	Gerarchia requisiti informativi	30
Figura 7.	Ciclo generico di gestione delle informazioni	31
Figura 8.	Ciclo informativo in corrispondenza di un <i>trigger event</i>	32
Figura 9.	Principio dell'ambiente di condivisione dati	33
Figura 10.	Schema del trattamento degli elaborati e delle informazioni	34
Figura 11.	Distinzione modelli ed elaborati	36
Figura 12.	Processo delle costruzioni	37
Figura 13.	Procedura di verifica CTI	52
Figura 14.	Aggiornamento 2019 della procedura di verifica CTI	53
Figura 15.	Logo gbXML	56
Figura 16.	Iter normativo	58
Figura 17.	Livelli di maturità del formato IFC	58
Figura 18.	Esempio confronto schema EXPRESS-G e schema EXPRESS	61
Figura 19.	Schema livelli IFC	64
Figura 20.	Programming & VPL in Dynamo	69
Figura 21.	Esempio nodi e collegamenti Dynamo	69
Figura 22.	Confronto linguaggi di programmazione	70
Figura 23.	Tools Python	70
Figura 24.	Esempio flusso codici da linguaggio Python a machine code (Windows)	71
Figura 25.	Implementation Python	71
Figura 26.	Vista complesso industriale FCA Mirafiori	72
Figura 27.	Modello BIM del caso studio	73
Figura 28.	Esportazione caso studio in formato gbXML	74
Figura 29.	Schema valutazione 1 software BEM	76
Figura 30.	Schema flusso dati da BIM a BEM	79
Figura 31.	Modello digitale BestTest 600	80
Figura 32.	Schema flusso dati Green Building Studio	81
Figura 33.	Creazione nuovo progetto Green Building Studio	81
Figura 34.	Risultati valutazioni BestTest 600 (Green Building Studio)	82
Figura 35.	Upload gbXML (FCA) in Green Building Studio	82
Figura 36.	Schema flusso dati Ecodesigner Star	83
Figura 37.	Plugin Archicad - Revit: Interfaccia e esportazione	83
Figura 38.	Importazione BestTest600 in Archicad	84
Figura 39.	Finestra Ecodesigner Star	84
Figura 40.	Stratigrafie e proprietà Ecodesigner Star	85
Figura 41.	Risultati valutazione BestTest Ecodesigner Star	85
Figura 42.	Problema importazione modello FCA in Ecodesigner Star	86
Figura 43.	Modello FCA importato in Archicad	86
Figura 44.	Zone termiche modello FCA in Ecodesigner Star	87
Figura 45.	Risultati valutazioni modello FCA in Ecodesigner Star	87
Figura 46.	Schema flusso dati EdilClima	88
Figura 47.	Scelta normative e impostazioni di calcolo in EdilClima	88
Figura 48.	Geolocalizzazione modello in EdilClima	89
Figura 49.	Associazione stratigrafie in EdilClima	89

Figura 50.	Controllo zone termiche in EdilClima	90
Figura 51.	Controllo stratigrafie in EdilClima	90
Figura 52.	Controllo verifiche di legge in EdilClima	91
Figura 53.	Input grafico EdilClima	91
Figura 54.	Schema flusso dati Design Builder	92
Figura 55.	Importazione modello BestTest in Design Builder	92
Figura 56.	Importazione modello FCA in Design Builder	93
Figura 57.	Importazione modello FCA in Design Builder	93
Figura 58.	Schema flusso dati Open Studio	94
Figura 59.	Visualizzazione grafica BestTest in Open Studio	95
Figura 60.	Errore importazione BestTest in Open Studio	95
Figura 61.	Zone termiche FCA importate in Open Studio	96
Figura 62.	Schema flusso dati Termolog	96
Figura 63.	Impostazione normative e importazione IFC Termolog	97
Figura 64.	Importazione modelli BestTest e FCA in Termolog	97
Figura 65.	Zone termiche importate Termolog	98
Figura 66.	Stratigrafie FCA importate in Termolog	98
Figura 67.	Schema flusso dati Termus	99
Figura 68.	Finestra importazione IFC in Termus	99
Figura 69.	Errore importazione file IFC in Termus	99
Figura 70.	Certificato di conformità rilasciato dal CTI	100
Figura 71.	Plugin AutodeskIFC2018	101
Figura 72.	Interfaccia esportazione ifc di Revit migliorata	102
Figura 73.	Modifica mappatura IFC in Revit	103
Figura 74.	Modifica file txt per mappatura IFC in Revit	103
Figura 75.	Modifica Parameter Mapping Table in Revit	103
Figura 76.	Schermata esportazione Revit Property Sets	104
Figura 77.	Schermata esportazione User Defined Property Sets	105
Figura 78.	User Defined Property Set di default in Revit	106
Figura 79.	Scrittura User Defined Property Sets	107
Figura 80.	Creazione parametro Revit	107
Figura 81.	Parametro personalizzato inserito correttamente in file ifc	107
Figura 82.	Mappatura classi ifc in Revit	108
Figura 83.	Problema mappa classi IFC	109
Figura 84.	Workflow BIM to BEM integrato a modifiche da VPL tool	110
Figura 85.	Matrice funzionamento algoritmo	111
Figura 86.	Nodo algoritmo: lettura file ifc	112
Figura 87.	Nodo algoritmo: lettura file ifc, processo di separazione per caratteri	112
Figura 88.	Nodo algoritmo: lettura file ifc, rimozione caratteri	113
Figura 89.	Nodo algoritmo: lettura file ifc, scrittura valore numerico	113
Figura 90.	Nodo algoritmo: lettura file ifc, ricerca classe	114
Figura 91.	Nodo algoritmo: lettura file ifc, valori numerici classe	114
Figura 92.	Nodo algoritmo: lettura proprietà, lista proprietà	114
Figura 93.	Nodo algoritmo: lettura proprietà, lettura modello BIM	115
Figura 94.	Nodo algoritmo: lettura proprietà, lista proprietà	115
Figura 95.	Nodo algoritmo: scrittura proprietà, iteratore	117
Figura 96.	Nodo algoritmo: scrittura nuovo file ifc	118
Figura 97.	Confronto dimensioni modelli	119
Figura 98.	Nodo algoritmo: sostituzione script classe materiali	120

Figura 99.	Nodo algoritmo: nodi filtro materiali	121
Figura 100.	Nodo algoritmo: nodi debugging materiali	121
Figura 101.	Nodo algoritmo: icona Dynamo Player	122
Figura 102.	Nodo algoritmo: Input & Output Dynamo Player	122
Figura 103.	Nodo algoritmo: Definizione input e creazione nuovo file	123
Figura 104.	Confronto IFC dopo modifiche	124
Figura 105.	Esempio differenti workflow	125
Figura 106.	Schema interpretazione dati IFC	127
Figura 107.	Edificio esempio in formato gbXML	135
Figura 108.	Estratto file IFC	135

1 Introduzione

1.1 Building Information Modeling

1.1.1 Definizione Building Information Modeling

Il Building Information Modeling è stato definito dal NIBS (National Institutes of Building Science) come la *“rappresentazione digitale di caratteristiche fisiche e funzionali di un oggetto”* [1]. Con l’acronimo BIM si intende una metodologia piuttosto che uno strumento, applicabile in progettazione, pianificazione, realizzazione e manutenzione di un edificio. L’elemento alla base di questa metodologia è il modello informativo, più comunemente chiamato modello digitale, ovvero un modello virtuale che contiene informazioni geometriche e non geometriche, riguardanti l’intero ciclo di vita di un edificio. BIM interessa le varie figure professionali coinvolte in un progetto, infatti questa metodologia apporta sostanziali benefici legati allo scambio di informazioni fra questi ultimi, ponendosi come soluzione innovativa che punta verso una progettazione collaborativa. Ogni professionista può infatti accedere, modificare, aggiungere o eliminare informazioni presenti nel modello digitale in ogni fase della progettazione. Quando si parla di progettazione *BIM-based* si intende quindi un approccio alla progettazione diverso da quello classico. In concreto, grazie al modello BIM che dispone di tutte le informazioni riguardanti il progetto, è possibile organizzare i punti e le modalità della progettazione in modo olistico, non tralasciando nessun aspetto del percorso progettuale. I vantaggi sono evidenti anche in aspetti pratici della progettazione, infatti secondo il Boston Consulting Group l’utilizzo di questa metodologia comporta una riduzione dei costi di progettazione pari al 15% ed una riduzione dei tempi di esecuzione pari al 30%.

L’adozione della metodologia BIM influisce attivamente su tutti i campi legati alla progettazione e alla gestione dell’edificio. Volendo schematizzare i campi d’applicazione di tale metodologia può essere utile descrivere

uno schema concettuale che riporta le principali attività della metodologia, possiamo quindi descrivere 7 dimensioni. Oltre alle classiche tre dimensioni usate per rappresentare un oggetto in 3D, il BIM è utile all’analisi dei tempi (4D), dei costi (5D), della gestione dell’edificio



Fig. 1: Schema delle 7 dimensioni del BIM
<http://biblus.acca.it/focus/le-7-dimensioni-del-bim/>

realizzato (6D) e della valutazione della sostenibilità (7D). [2]

- Modellazione (3D): il modello digitale risulta essere molto accurato a livello grafico, aspetto divenuto rilevante in tutte le discipline interessate al progetto. Grazie ad un modello 3D fedele alle geometrie dell'edificio reale è possibile valutare preventivamente problematiche legate al *code checking*, ovvero il controllo all'adeguatezza del progetto alle richieste progettuali e normative e al *clash detection* ovvero controllo dei conflitti geometrici fra le varie parti costituenti il modello.
- Analisi dei tempi (4D): monitorare l'avanzamento dei lavori servendosi di uno strumento efficiente che consente una precisa organizzazione fra le parti interessate è doveroso al giorno d'oggi. Il BIM consente di pianificare le azioni ed i tempi di esecuzione necessari al progressivo avanzamento dei lavori, dalla progettazione alla costruzione.
- Analisi dei costi (5D): la possibilità di ricavare le quantità dei materiali richiesti direttamente dal modello BIM dà modo ai professionisti di effettuare computi metrici accurati, evitando sprechi di materiali con conseguente aumento dei costi. L'aggiornamento automatico delle quantità durante la progettazione rende la stesura di nuovi computi metrici molto più rapida rispetto ai metodi tradizionali.
- Gestione dell'edificio (6D): un modello informativo fedele a quanto costruito (chiamato As-Built) risulta essere estremamente utile anche dopo la costruzione dell'edificio stesso, in quanto dispone delle informazioni necessarie alla manutenzione e futura dismissione dell'edificio.
- Sostenibilità (7D): lo sfruttamento delle risorse naturali, economiche e sociali è un aspetto che è ormai impossibile trascurare nella progettazione di nuovi edifici. L'uso del BIM come soluzione all'integrazione della sostenibilità nella progettazione dà la possibilità ai professionisti di adoperare soluzioni nuove e performanti che tengano conto di una quantità di dati altrimenti inverosimilmente considerabili. [2]

Vi è da evidenziare però, che l'utilizzo di questa metodologia non comporta nessuna semplificazione nella qualità del lavoro richiesto dai professionisti. L'utilizzo di funzioni e strumenti sempre più performanti permette di raggiungere risultati migliori, che tengono conto di una quantità di dati maggiore rispetto al passato. Tuttavia lo sforzo intellettuale richiesto alle figure professionali coinvolte continua ad essere un elemento indispensabile della progettazione. Il contributo al progetto da parte di questi ultimi viene solo riorganizzato secondo un approccio nuovo, appunto *BIM-oriented*. Appare quindi chiaro come un prerequisito fondamentale per l'ottenimento di un prodotto finale di qualità sia la preparazione

tecnica dei professionisti che deve comprendere l'utilizzo di strumenti innovativi.

Come è possibile constatare dal grafico proposto dall'architetto statunitense nonché presidente di BuildingSMART Patrick MacLeamy (Fig. 2), il quale mette a confronto la progettazione classica e la nuova progettazione *BIM-oriented*, la quantità di lavoro richiesto è pressoché uguale.

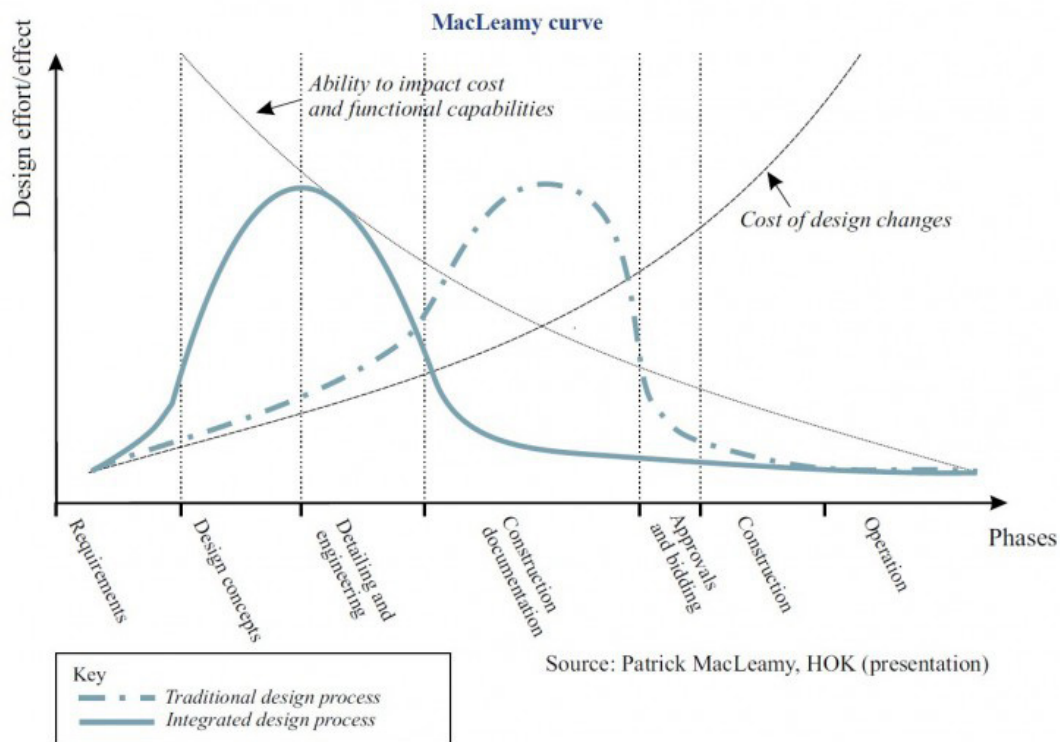


Fig. 2: Grafico MacLeamy

<http://biblus.acca.it/wp-content/uploads/Diagramma-di-MacLeamy-e1464776871198.jpg>

Ad essere rivoluzionata è la gestione del tempo, che vede nella metodologia BIM una prima fase densa di lavoro da svolgere. In un primo momento l'approccio alla metodologia potrebbe sembrare ostico e sconveniente rispetto alla metodologia tradizionale, poiché nella fase iniziale viene richiesto un impegno non indifferente da parte dei professionisti. In realtà è proprio grazie alla mole di lavoro svolto sin dalle prime fasi del progetto, che i professionisti hanno la possibilità di non incappare in errori che comporteranno rallentamenti e relativi aumenti dei costi. Nella peggiore delle ipotesi invece, giunti ad un certo livello nell'avanzamento dei lavori, alcuni errori diventano irrisolvibili causando spiacevoli risultati.

1.1.2 Dal Computer Aided Design al Building Information Modeling

La prima grande rivoluzione nelle attività progettuali è avvenuta intorno al 1985, quando dalla progettazione manuale, eseguita con l'ausilio di strumenti fisici, si è affermata la progettazione digitale, ovvero un tipo di progettazione realizzata avvalendosi di computer muniti di relativi software di disegno, il Computer Aided Design. Questa trasformazione ha portato ad una modifica riguardante l'esecuzione degli elaborati, ma le modalità di progettazione sono rimaste pressoché identiche. Quello che sta avvenendo negli ultimi anni, ovvero il passaggio dal CAD al BIM, è ben diverso, infatti la metodologia BIM, introdotta nel 2002, oltre ad offrire strumenti nuovi ed all'avanguardia, rivoluziona il metodo in cui le figure professionali collaborano e si relazionano al progetto. Uno dei punti fondamentali del BIM è infatti il concetto di progettazione condivisa ovvero l'insieme di attività svolte dai professionisti nei confronti delle altre figure coinvolte. Lo scambio di informazioni avviene infatti dalle prime fasi progettuali e rimane costante, se non crescente, durante l'intera progettazione e costruzione dell'edificio.

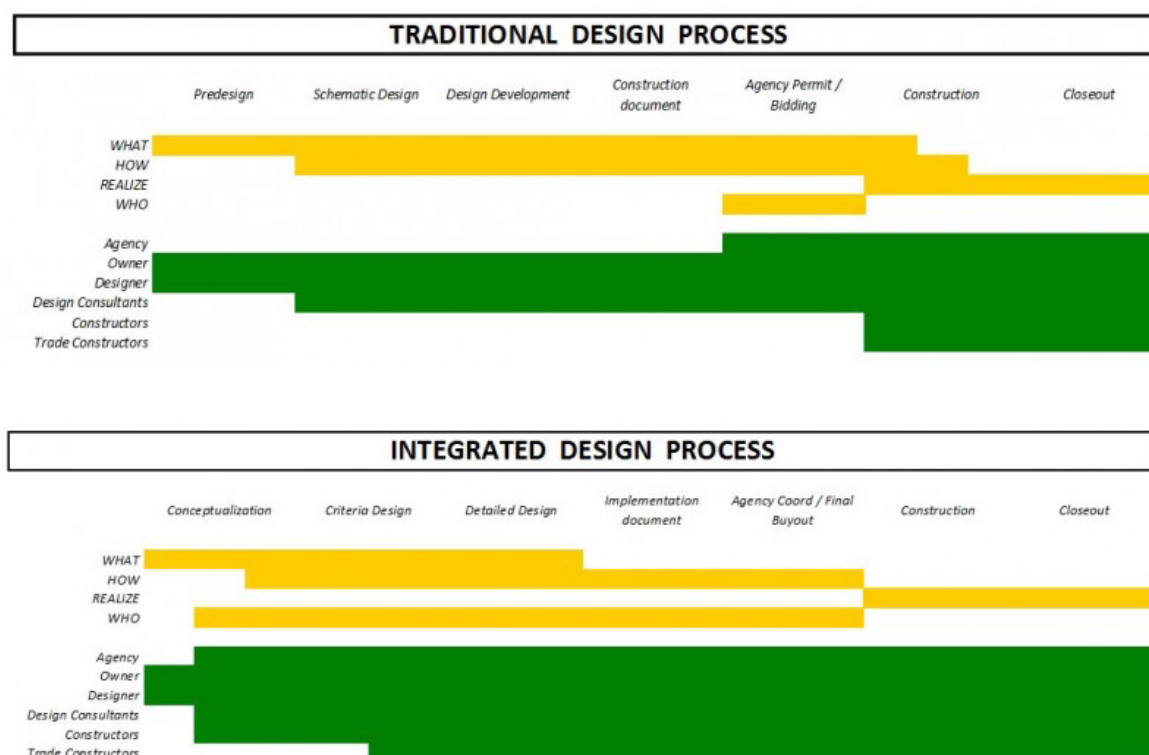


Fig. 3: Confronto progettazione tradizionale e metodologia BIM

<http://biblus.acca.it/come-il-bim-cambia-il-nostro-modo-di-lavorare/>

Il grafico mostra chiaramente come già dalle prime fasi progettuali è richiesta una collaborazione fra le varie figure coinvolte, aspetto fondamentale se si vuole raggiungere un risultato di ottima qualità con un rapporto costo/benefici soddisfacente. Tuttavia la transizione dalla metodologia tradizionale CAD alla nuova metodologia BIM non è un

passaggio semplice ed immediato, soprattutto per i professionisti impiegati nel settore AEC da diversi anni. Nel tempo si sta assistendo ad una graduale modifica ed “aggiornamento” da parte degli studi professionali verso la metodologia BIM. Il grado di aggiornamento può essere diviso per livelli, come illustrato nel grafico presentato dagli ingegneri Mark Bew e Mervyn Richards (Fig. 4). Il grafico Bew-Richards suddivide l'utilizzo del BIM in base a quattro livelli di maturità definiti in virtù della qualità dello scambio informativo ed al tasso di collaborazione. Per livello di maturità si intende dunque il grado di scambio di dati ed informazioni fra le parti, la tipologia di strumenti utilizzati e la qualità degli elaborati condivisi.

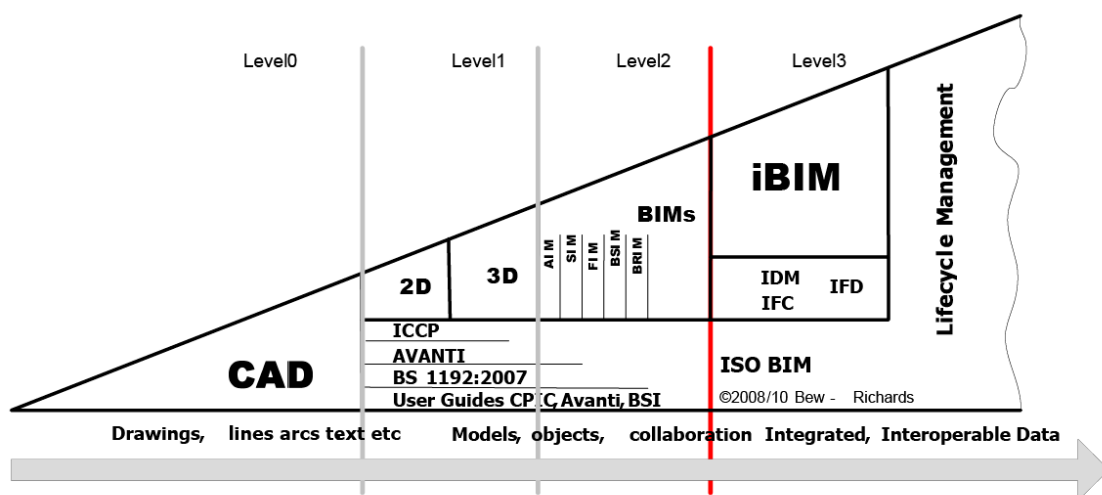


Fig. 4: Grafico Bew-Richards

<https://www.bimthinkspace.com/2015/02/index.html>

Il livello 0 è il livello più basso, nonché il più vicino alla metodologia tradizionale. In questo livello non vi è collaborazione fra le parti, lo scambio delle informazioni avviene tramite supporti cartacei o con l'ausilio di strumenti elettronici non interoperabili, principalmente file CAD.

Il livello 1 o livello di collaborazione parziale prevede l'utilizzo di un ambiente condivisione dati, definito in Italia dalla normativa UNI 11337, nel quale vengono caricati i dati sensibili del progetto. A questo livello l'organizzazione dei dati è gestita all'interno del ACDat dove vengono utilizzati strumenti CAD e occasionalmente strumenti BIM, tuttavia l'interoperabilità è minima e si limita alla definizione della progettazione in assenza di un modello condiviso. La differenza sostanziale fra il livello 1 ed il livello 2 risiede nella modalità di condivisione dei dati fra le parti operanti sul progetto. Ogni operatore lavora ad un suo modello che però viene sviluppato seguendo linee guida presenti in un file comune. In questo modo vi è collaborazioni fra le parti che hanno piena visione delle informazioni disponibili per il

progetto, aspetto fondamentale per l'unione degli elaborati al fine di creare un modello BIM unificato. I software utilizzati in questo livello devono essere adeguati all'interoperabilità in quanto è necessario importare ed esportare diversi formati comuni. Nel livello 2 viene presa in considerazione anche la gestione del tempo (4D) e l'analisi dei costi (5D).

Il livello 3 è il livello in cui avviene un'integrazione completa delle informazioni in un ambiente dati condiviso. Questo livello è caratterizzato dalla presenza di un modello unico condiviso accessibile da tutte le figure coinvolte, previo opportuni permessi di accessibilità, sul quale gli operatori potranno intervenire (modificare, aggiungere o rimuovere informazioni). L'interoperabilità dei dati contenuti nel modello è garantita dalla possibilità di conversione in formato IFC, uno standard ISO non proprietario, attualmente in continuo sviluppo. In questo livello vengono considerati anche aspetti legati al facility management (6D) e valutazione della sostenibilità (7D). [3]

1.1.3 BIM to BEM per l'efficienza energetica degli edifici

Secondo uno studio effettuato dal IPCC è stato stimato che entro il 2050 circa 10 miliardi di persone vivranno in grandi centri urbani, mentre nei prossimi 10 anni circa 2,5 miliardi di persone si sposteranno dalle aree rurali verso le città. Per gestire un tale flusso avremmo bisogno di costruire circa 13000 edifici al giorno, un'operazione per circa 3 trilioni di euro. Appare chiaro come lo sviluppo di una simile quantità di edifici avrà un considerevole impatto sull'ambiente. Nella situazione attuale, il parco edilizio moderno è responsabile del consumo di circa il 40% dell'energia totale consumata e della produzione del 30% dei rifiuti non riciclati che gravano sul territorio, prodotti in fase di costruzione. In uno scenario simile appare chiaro come rendere il settore AEC il quanto più veloce ed efficiente possibile, nonché costruire degli edifici che siano il quanto più performanti dal punto di vista energetico, sia diventata una necessità. L'utilizzo della metodologia BIM è senz'altro utile al raggiungimento di questi obiettivi poiché oltre ai vantaggi descritti precedentemente in questo elaborato, rende possibile l'integrazione del Building Energy Modeling già dalle prime fasi progettuali.

Per BEM si intende il processo di costruzione di modelli digitali che tengano conto delle caratteristiche utili ad effettuare delle valutazioni energetiche o analisi tecniche di un edificio. L'integrazione completa di BIM e BEM consente di creare modelli definiti *Digital Twins*, ovvero modelli digitali che hanno le caratteristiche dell'edificio reale, ne replicano il comportamento fisico-tecnico e consentono di monitorarne le performance. L'utilizzo di

un modello che tenga conto delle caratteristiche energetiche di un edificio oltre ad essere utile durante la fase di progettazione, rende possibile l'esecuzione di valutazioni energetiche parallelamente allo sviluppo del progetto diventando uno strumento di monitoraggio delle prestazioni dell'edificio che una volta costruito. Una volta completata la costruzione, la possibilità di equipaggiare l'edificio con sensori che comunicano dati al modello digitale, rende le valutazioni energetiche estremamente accurate e fedeli alla realtà, in questo modo è possibile gestire dinamicamente gli impianti ai fini dell'ottenimento di un'efficienza ottimale. [4]

1.2 Quadro Normativo BIM

Il quadro normativo è sicuramente molto vasto e complesso. Allo stato attuale l'Italia è soggetta principalmente a normative di diverso tipo, quali:

- Normative ISO: norme tecniche organizzate a livello mondiale.
- Normative CEN: norme (EN) redatte a livello europeo in collaborazione con enti nazionali e internazionali come ISO.
- Normative UNI: norme di giurisdizione italiana, in conformità con le norme dei livelli superiori.

Come stabilito dal cosiddetto “Accordo di Vienna” stipulato tra ISO e CEN a partire dal 1989 e rivisto fino al 2016, i livelli normativi fra i vari stati membri devono essere coordinati e allineati. Tale allineamento è possibile per mezzo di due modalità. Nel primo caso prevale il livello ISO quindi la normativa emanata da questo ente viene automaticamente recepita dal CEN e dagli stati membri, nel secondo caso invece prevale il livello CEN che si ripercuote alle norme locali per poi essere approvate a livello ISO. Nella votazione effettuata nel 2016 l'Italia è stata l'unico stato membro a votare a favore del livello CEN quindi, alla quasi unanimità, è stato adottato il procedimento ISO. Tuttavia vi è da notare che la linea ISO prevede, in caso di necessità da parte degli stati membri, la possibilità di sviluppare standard la cui utilità è stata riconosciuta a livello internazionale. [5]

Allo stato attuale la normativa internazionale di riferimento per il BIM è la ISO 19650 – *“Organizzazione delle informazioni sui lavori di costruzione – Gestione delle informazioni nell'uso del BIM”*. A quest'ultima si aggiunge in funzione di allegato nazionale per quanto riguarda l'Italia, la norma UNI 11337 che viene adottata unitamente alla normativa internazionale, previo riscrittura ed adeguamenti.

1.2.1 ISO 19650

La ISO 19650 – *“Organizzazione delle informazioni sui lavori di costruzione – Gestione delle informazioni nell'uso del BIM”* articolata in due parti:

- Parte 1: Concetti e principi (*“Concepts and Principles”*);
- Parte 2: Fase di progettazione e produzione dei beni immobili (*“Delivery phase of the assets”*)

Durante il 2020 è prevista l'entrata in vigore anche delle parti 3 e 5:

- Parte 3: Fase di gestione e manutenzione dei beni immobili (*“Operation phase of assets”*)
- Parte 5: Specifiche per il BIM orientato alla sicurezza, ambienti digitali costruiti,

gestione efficiente degli beni immobiliari

Come è possibile osservare in Fig. 5, la ISO 19650 deriva dalle più datate PAS inglesi, in particolare dalla norma PAS 1192 – 2:2013 ormai ritirata e sostituita con le più recenti ISO. La prima parte della norma ISO 19650 si articola in 13 capitoli e gli argomenti trattati

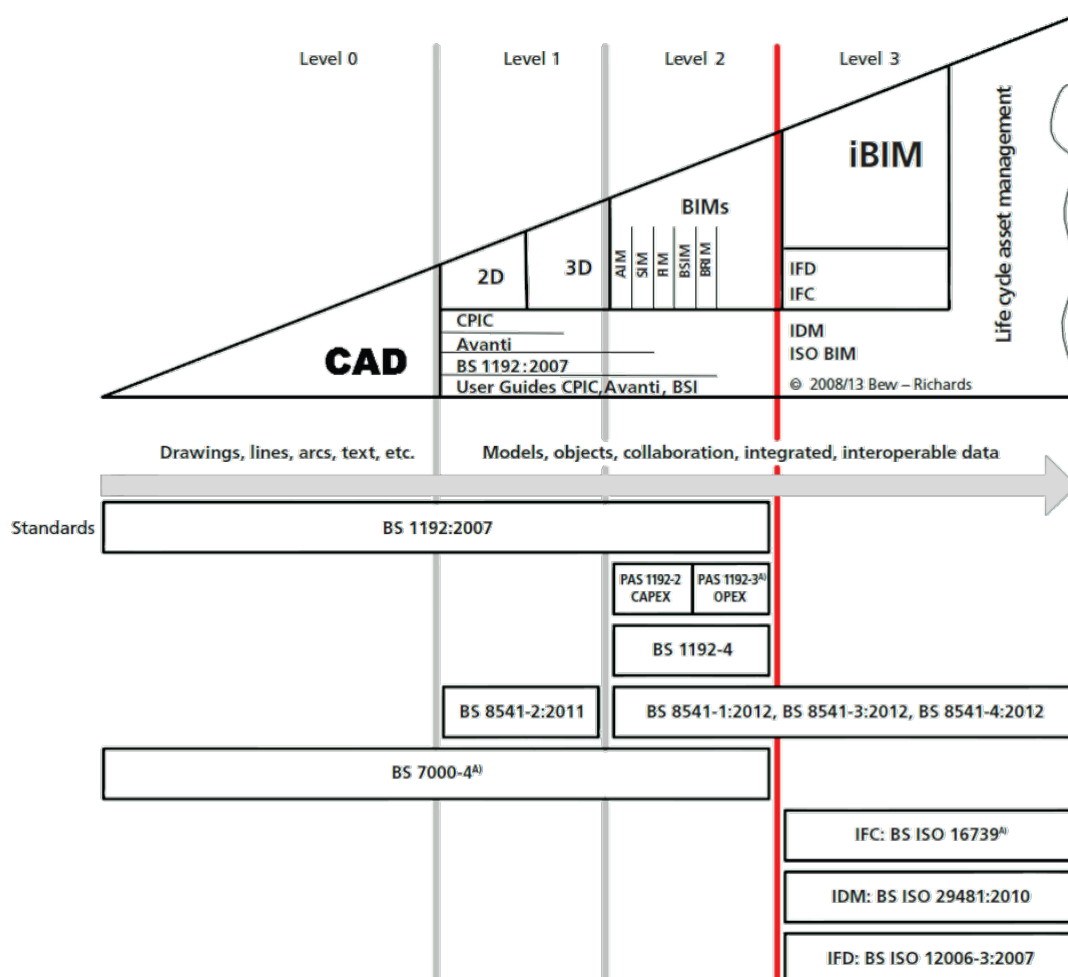


Fig. 5: Schema normativa ISO 19650

<http://www.bis-lab.eu/2017/04/23/the-road-to-ibim-il-bim-nel-regno-unito/>

ricoprono l'intero ciclo di vita di tutte le tipologie di edifici. Nel capitolo 4, la norma espone uno schema molto simile al precedentemente citato grafico a triangolo di Bew-Richards. Osservando lo schema appare chiaro come vi è una progressiva e crescente integrazione fra i dati. Nella fase 2 i dati vengono condivisi avvalendosi di strumenti come modelli e file, mentre nella fase 3 vi è un'interrogazione diretta dei megadati. Nel capitolo 5 viene fissata la gerarchia dei requisiti informativi. I vari soggetti interessati infatti elaborano i documenti necessari al fine di stabilire i requisiti che saranno perseguiti dagli incaricati. Questa parte della normativa corrisponde al Capitolato informativo della UNI 11337 – 5 e 6. Gli obbiettivi del committente e del progetto diventano quindi il punto di partenza nell'iter progettuale che

tiene conto delle specifiche tecniche e garantisce continuità fra i vari passaggi.

Lo schema in Fig. 6 è esplicativo per quanto viene trattato nel capitolo 5 della norma.

Nel dettaglio le varie voci riguardano:

Gerarchia dei requisiti informativi

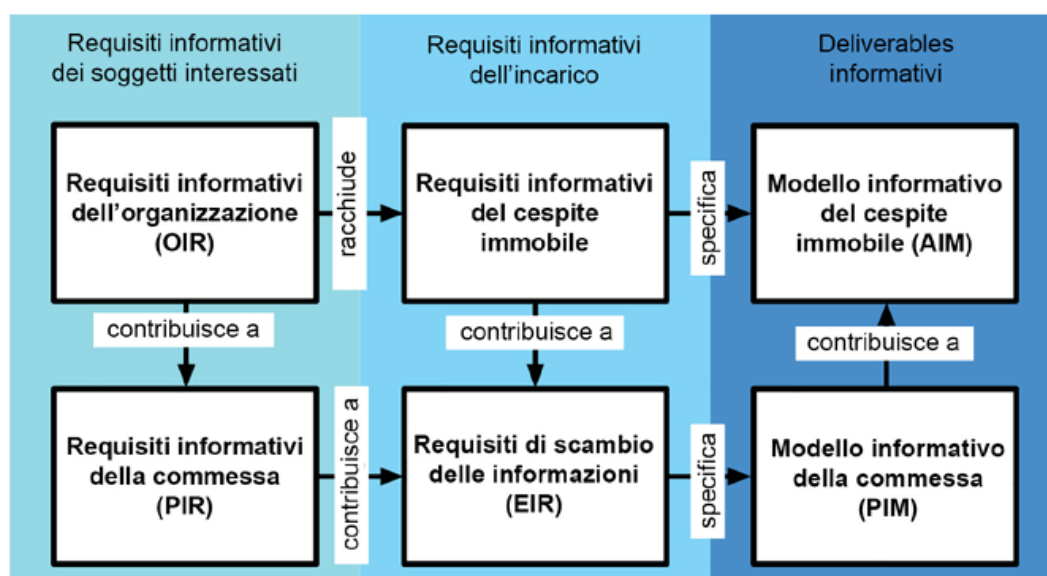


Fig. 6: Gerarchia requisiti informativi

<https://blog.archicad.it/tutto-quello-che-hai-sempre-voluto-sapere-sulla-iso-19650>

- *OIR: Organization Information Requirements* o Requisiti informativi dell'organizzazione. Hanno come fine l'individuazione degli obiettivi strategici del soggetto proponente.
- *PIR: Project Information Requirements* o Requisiti informativi della commessa. Sono utili al miglioramento degli obiettivi già fissati nell'OIR ma adattati ad un determinato caso.
- *AIR: Asset Information Requirements* o Requisiti informativi del cespite immobile. Definisce gli aspetti tecnici quali procedure e metodi legati ad un determinato caso.
- *EIR: Exchange Information Requirements* o Requisiti di scambio delle informazioni. Fissa gli aspetti tecnici quali i formati di scambio utilizzati, coordinate del progetto, impostazioni dell'edificio ed altri dati correlati.
- *PIM: Project Information Model* o Modello informativo della commessa. È il modello digitale utile al reperimento delle informazioni per la messa in opera dell'immobile.
- *AIM: Asset Information Model* o Modello informativo del cespite immobile. È il modello digitale utile al reperimento delle informazioni relative alla gestione ed alla manutenzione dell'immobile, nonché informazioni riguardanti i dispositivi installati ed

i piani di manutenzione.

Il capitolo 6 tratta il ciclo di consegna delle informazioni che viene pianificato a monte della fase progettuale. Secondo quanto stabilito ogni incaricato dovrà rispettare quattro principi:

- La gestione delle informazioni deve essere eseguita durante tutto l'iter progettuale
- Lo sviluppo delle informazioni deve essere pianificato
- I requisiti informativi devono poter essere trasferiti alle vari ed eventuali parti interessate o aggregate
- Lo scambio delle informazioni deve avvenire all'interno di un ACDat il quale dovrà essere utilizzato durante l'intero ciclo di vita dell'edificio.

Nello schema riportato (Fig. 7) vengono illustrati i passaggi dalla fase gestionale, quindi di consegna e nuovamente gestionale.

Punto A: trasferimento di informazioni al modello *PIM*; Punto B: modello sviluppato fino

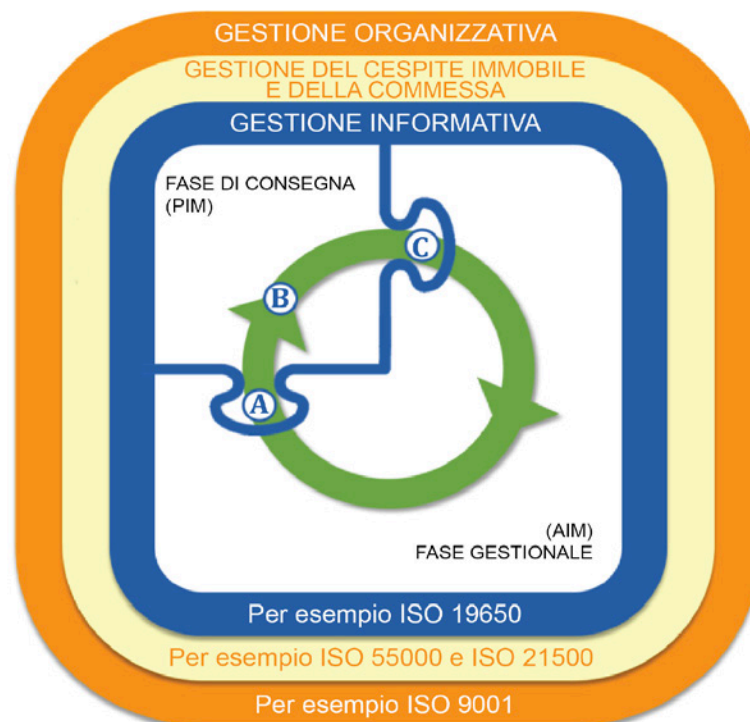


Fig. 7: Ciclo generico di gestione delle informazioni

<https://blog.archicad.it/tutto-quello-che-hai-sempre-voluto-sapere-sulla-iso-19650>

alla fase di consegna; Punto C: nuovo passaggio di informazioni dal modello *PIM* a *AIM* per la fase gestionale. Il ciclo informativo si avvia per ogni evento chiamato *trigger event*, pianificato e non pianificato che modifica l'immobile in una qualsivoglia parte del suo ciclo di vita. I *trigger event* pianificati devono essere specificati nel *PIR* e nel *OIR*.

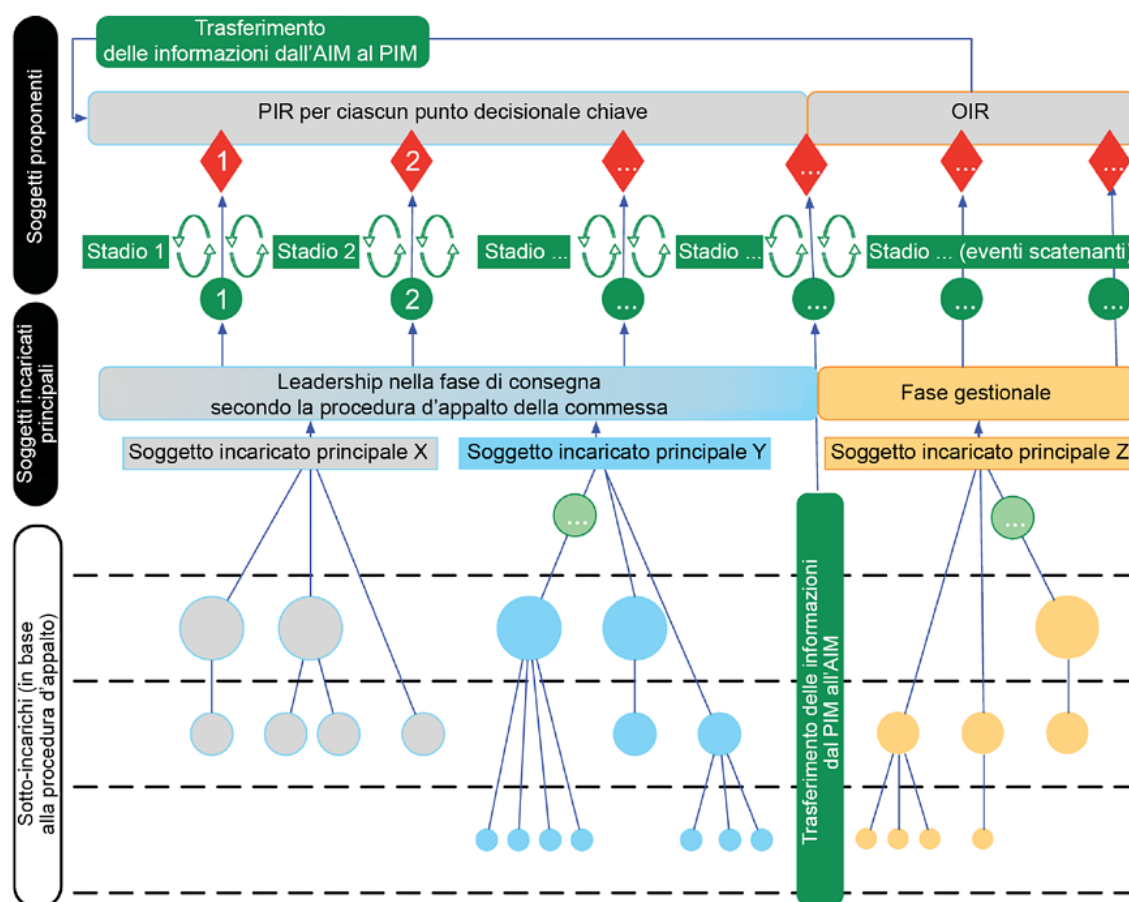


Fig. 8: Ciclo informativo in corrispondenza di un *trigger event*

<https://blog.archicad.it/tutto-quello-che-hai-sempre-voluto-sapere-sulla-iso-19650>

Lo schema (Fig. 8) esplicita le modalità di attuazione riguardanti un *trigger event* nel quale vengono definiti uno o più soggetti interessati ed elementi quali punti decisionali e scambio di informazioni.

- Riquadri grigi, azzurri e gialli: Soggetti o gruppo di soggetti incaricati
- Rombi rossi: punti decisionali; Pallini verdi: scambio di informazioni
- Frecce verdi: verifica e validazione delle informazioni
- Freccia blu verticale: flusso di contenuti informativi

Ogni *trigger event* costituisce una consegna per la quale non vi è un numero prestabilito, a differenza della UNI 11337 che ne prefissa otto.

Nel capitolo 10 vengono stabilite e pianificate le consegne che devono trattare:

- La verifica dei requisiti informativi dell'AIM e dell'EIR
- Tempi, tipologia e qualità del dato consegnato
- Identificazione dei responsabili della consegna
- Identificazione dei destinatari delle informazioni consegnate

Le consegne avvengono tramite determinati elaborati quali:

- Piano di consegna
- Matrice della responsabilità
- Strategia di aggregazione della struttura

Il capitolo 11 tratta la gestione e la produzione delle informazioni introducendo i concetti di interferenza informativa, chiamata *clash detection*, e di incongruenza informativa, chiamata *model checking*.

L'incongruenza informativa o *model checking* è divisa in due tipologie:

- Incongruenza di tipo *hard*: quando due oggetti occupano lo stesso spazio fisico.
- Incongruenza di tipo *soft*: un oggetto non dispone di adeguata area di utilizzo o le aree di utilizzo fra due oggetti si sovrappongono.

In questa parte della norma vengono fissati i LOD che regolano la qualità, la quantità e la granularità delle informazioni riguardanti gli oggetti.

Il capitolo 12 definisce le caratteristiche e gli scopi dell'ambiente di condivisione dati. La norma impone che ogni modello ed ogni informazione all'interno dell'ACDat abbiano un proprio codice di revisione e un codice di stato degli usi ammessi, suddiviso in quattro tipologie quali:

- Stato di elaborazione
- Stato di condivisione
- Stato di pubblicazione
- Stato di archiviazione

Tra uno stato ed il successivo è necessario applicare un protocollo di verifica ai requisiti informativi prefissati (Fig. 9).

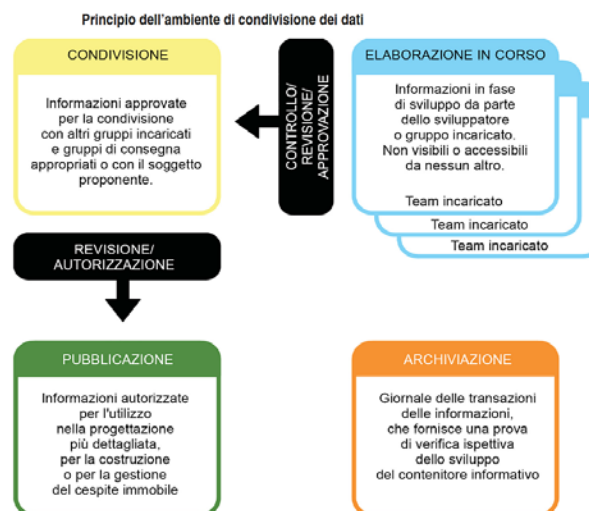


Fig. 9: Principio dell'ambiente di condivisione dati

<https://blog.archicad.it/tutto-quello-che-hai-sempre-voluto-sapere-sulla-iso-19650>

Il capitolo 13 stabilisce una distinzione fra produzione di elaborati e gestione dell'informazione anche se queste due tipologie rimangono ad ogni modo vincolate fra loro (Fig. 10).

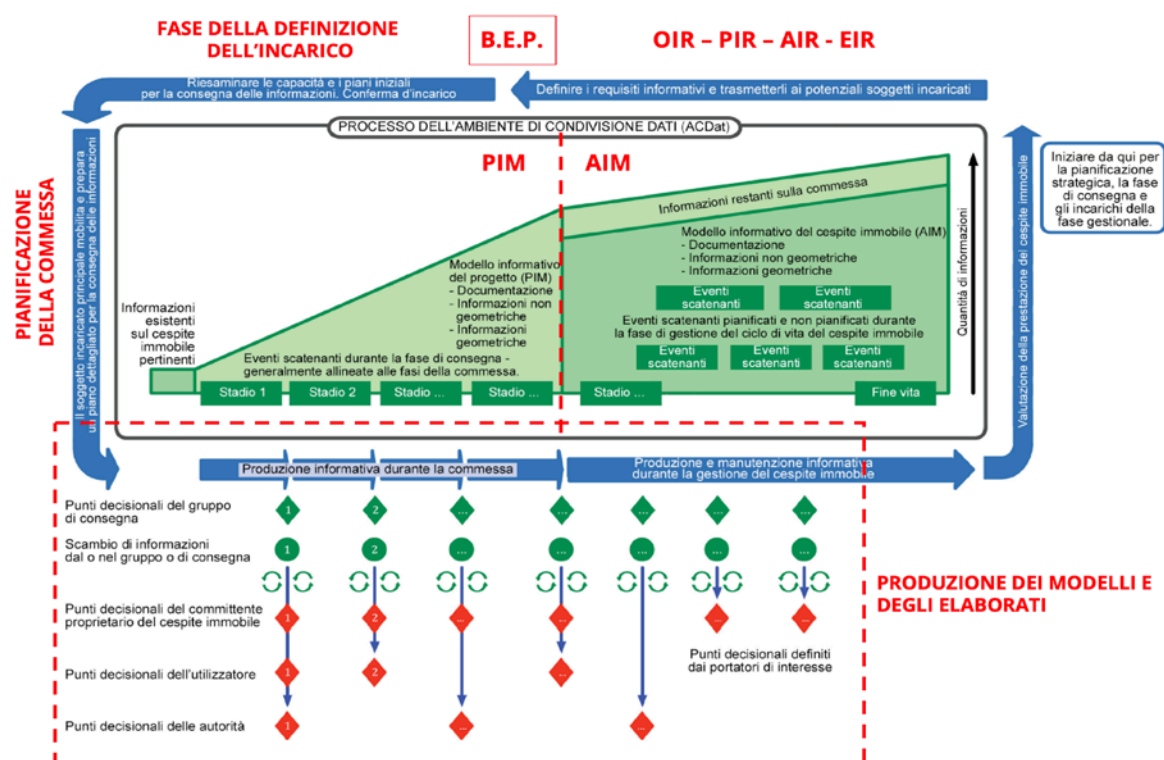


Fig. 10: Schema del trattamento degli elaborati e delle informazioni
<https://blog.archicad.it/tutto-quello-che-hai-sempre-voluto-sapere-sulla-iso-19650>

1.2.2 Normativa italiana in ambito BIM

In un primo momento la metodologia BIM è stata utilizzata solo su iniziativa di singoli e da privati, tuttavia negli ultimi tempi il suo utilizzo ha visto e continua a vedere una forte crescita. Il graduale adeguamento al BIM da parte di molti esperti nel settore è sicuramente dovuto anche all'introduzione di una nuova legislazione e di una normativa tecnica. Le normative che regolano il BIM in Italia sono principalmente il Decreto Ministeriale 560/2017 ed la Normativa UNI 11337. Esse costituiscono un importante elemento normativo che ha come fine quello di disciplinare la filiera di progettazione per quanto riguarda l'utilizzo di metodi e strumenti elettronici quali la metodologia BIM.

Il Decreto Ministeriale 560/2017, entrato in vigore da Gennaio 2018, stabilisce in che modo ed entro quale tempistica dovrà avvenire una graduale introduzione degli strumenti elettronici ed i relativi metodi per quanto riguarda la modellazione per l'edilizia e per le infrastrutture. Gli articoli del suddetto decreto fissano delle linee guida riguardanti adempimenti preliminari fissati dalle stazioni appaltanti, quali la formazione del personale, il piano di acquisizione o

di manutenzione degli strumenti hardware e software, il piano per il controllo della gestione dei dati e degli eventuali irregolarità di questi ultimi, l'utilizzo di piattaforme interoperabili per mezzo di formati aperti non proprietari. In dettaglio, l'articolo 6 del decreto fissa dei termini riguardanti i concorsi pubblici entro i quali i vincitori dovranno obbligatoriamente avvalersi di metodi e strumenti elettronici. [6]

- a) “ 1. Le stazioni appaltanti richiedono, in via obbligatoria, l'uso dei metodi e degli strumenti elettronici di cui all'articolo 23, comma 1, lettera h), del codice dei contratti pubblici secondo le seguente tempistiche:

Per i lavori complessi relativi a opere di importo a base di gara pari o superiore a 100 milioni di euro, a decorrere dal 1° gennaio 2019;

- b) *per i lavori complessi relativi a opere di importo a base di gara pari o superiore a 50 milioni di euro, a decorrere dal 1° gennaio 2020;*

- c) *per i lavori complessi relativi a opere di importo a base di gara pari o superiore a 15 milioni di euro, a decorrere dal 1° gennaio 2021;*

- d) *per le opere di importo a base di gara pari o superiore alla soglia di cui all'articolo 35 del codice dei contratti pubblici, a decorrere dal 1° gennaio 2022;*

- e) *per le opere di importo a base di gara pari o superiore a 1 milione di euro, a decorrere dal 1° gennaio 2023;*

- f) *per le opere di importo a base di gara inferiore a 1 milione di euro, a decorrere dal 1° gennaio 2025;” [6]*

Un altro documento fondamentale che riguarda la normazione della metodologia BIM in Italia è la norma UNI 11337:2017. Questa norma italiana costituisce al momento il documento più dettagliato che regola, dividendo gli argomenti in 10 parti, gli aspetti principali riguardanti questa nuova metodologia. [7]

Si riportano brevemente le 10 parti costituenti la norma:

- “*Parte 1: Modelli, elaborati e oggetti informativi per prodotti e processi*”. Questa parte della norma chiarisce definizioni e concetti riguardanti le modalità di trasmissione delle informazioni; definizione di oggetti digitali; definizione di strutture informative e dei processi. In questa parte vengono classificati i dati secondo la loro densità in un insieme gerarchico:
 - Dato: si indica il singolo elemento conoscitivo.
 - Informazione: si indica un insieme di dati.
 - Contenuto Informativo: si indica un insieme di informazioni.

Per quanto riguarda la trasmissione delle informazioni, quali esse siano grafiche,

documentate o multimediali, la norma definisce come “veicoli informativi”:

- Elaborato: indicato come veicolo di rappresentazione, digitale o non digitale.

Modello: indicato come veicolo di virtualizzazione, esclusivamente digitale.

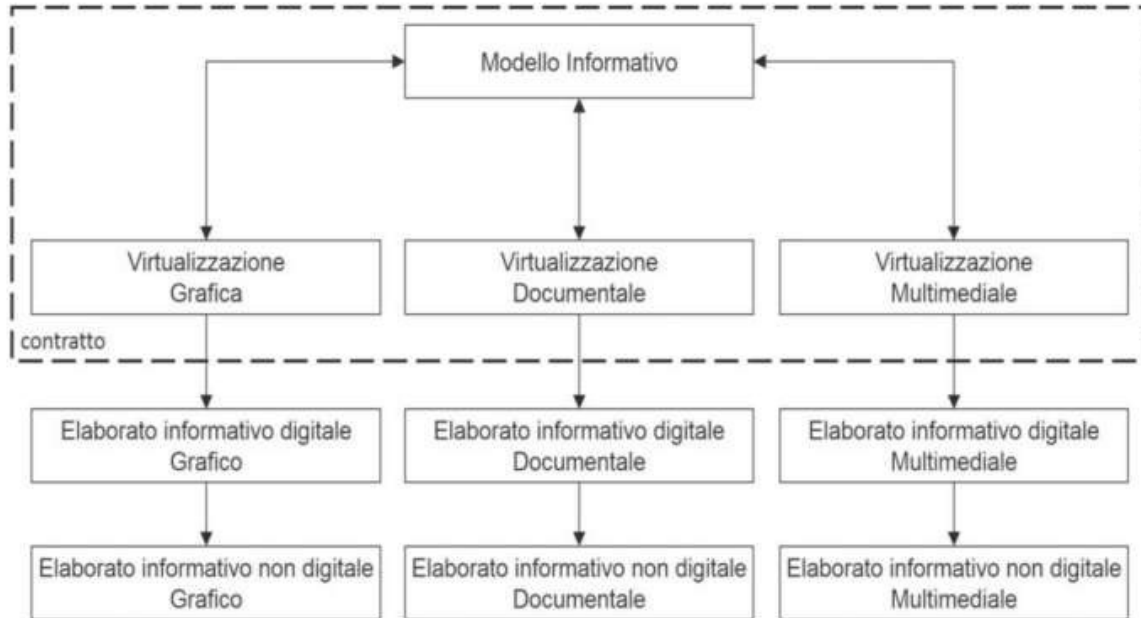


Fig. 11: Distinzione modelli ed elaborati

<https://www.01building.it/bim-bim-normativa-uni-11337>

La norma individua quindi il “grado di maturità digitale” del processo delle costruzioni diviso in cinque livelli:

- Livello 0: non digitale
- Livello 1: base
- Livello 2: elementare
- Livello 3: avanzato (Questo livello corrisponde al Livello 2 della PAS 1192-2 inglese)
- Livello 4: ottimale

La norma fissa anche una struttura riguardante il processo delle costruzioni, divisa in stati a loro volta costituiti da diverse fasi. (Fig. 12)

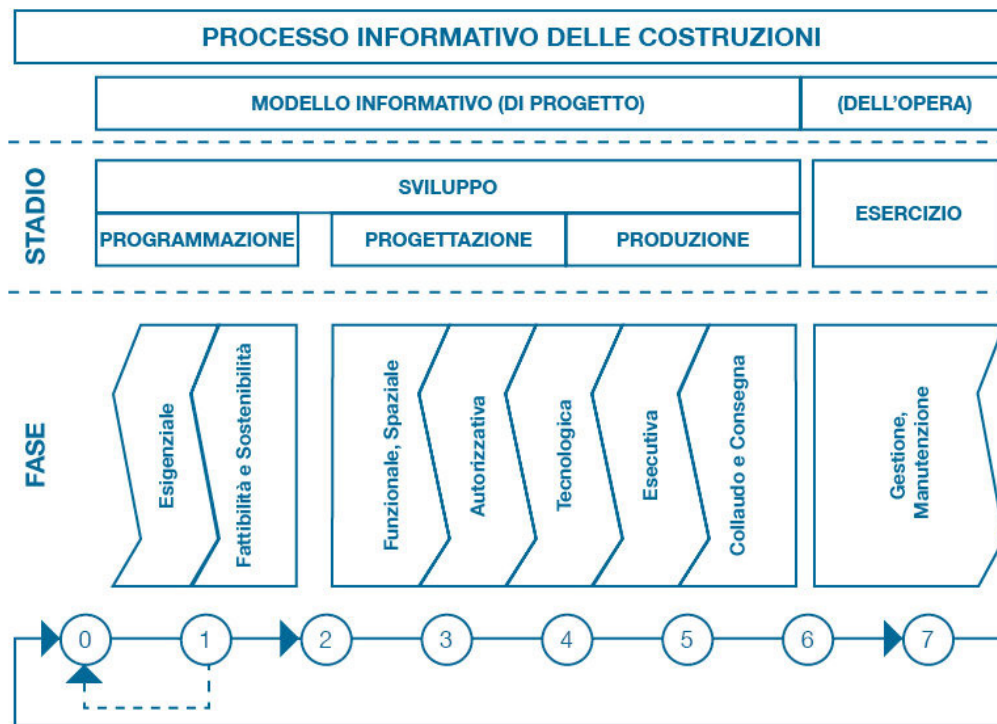


Fig. 12: Processo delle costruzioni

<http://biblus.acca.it/il-bim-in-italia-nelle-nuove-uni-11337-1-la-parte-prima-della-norma/>

- *“Parte 2: Criteri di denominazione e classificazione di modelli, prodotti e processi”*. Questa parte della norma è ancora in fase di elaborazione. Una volta ultimata avrà lo scopo di definire gli attributi identificativi utili a stabilire se vi è adeguatezza fra modelli ed elaborati, nonché metodi per ampliare tali attributi.
- *“Parte 3: Modelli di raccolta, organizzazione e archiviazione dell’informazione tecnica per i prodotti da costruzione”*. Indica come raccogliere ed archiviare i dati e le informazioni tecniche dei prodotti da costruzione. Le descrizioni di tali prodotti sono suddivise secondo due categorie rispettivamente qualitative o quantitative. La specifica introduce i concetti di:
 - *LOG*: attributi informativi geometrici
 - *LOI*: attributi informativi non geometrici
- *“Parte 4: Evoluzione e sviluppo informativo di modelli, elaborati e processi”*. Regola la gestione digitalizzata del processo informativo delle costruzioni, si prefissa come fini:
 - Specificare gli obiettivi per ogni fase del progetto, come specificato nella Parte 1 della stessa norma.
 - Definire una scala comune riguardante gli oggetti relativi ai modelli.

- Definire una scala comune per gli stati di lavorazione e approvazione dei contenuti.

Posta particolare attenzione sulla definizione dei *LOD* che, nella norma, vengono chiamati “*Livello di sviluppo degli Oggetti Digitali*” classificati secondo una scala da “*A*” a “*G*”, come alternativa della scala britannica da “*1*” a “*6*” e della scala statunitense da “*100*” a “*500*”. Introdotti anche dei *LOD* specifici riguardanti il restauro.

- “*Parte 5: Flussi informativi nei processi digitalizzati*”. La specifica introduce i concetti di “*Capitolato Informativo*”, “*Offerta di Gestione Informativa*” e “*Piano di Gestione Informativa*”. In questa parte vengono regolamentati i modelli grafici e l’analisi dei *clash detection*. Fornisce infine i requisiti necessari per la definizione del ACDat.
- “*Parte 6: Linee Guida per la redazione del capitolato informativo*”. Specifica che viene introdotta come supporto per parti interessate, scritta in forma di rapporto tecnico. Utile ad evitare incomprensioni e rallentamenti nella stesura di un corretto capitolato informativo.
- “*Parte 7: Requisiti di conoscenza, abilità e competenza delle figure coinvolte nella gestione e nella modellazione informativa*”. Nella specifica regola la definizione delle figure professionali impiegate in ambito BIM, suddivise in:
 - *BIM Manager*: gestore delle informazioni.
 - *CDE Coordinator*: gestore del ACDat
 - *BIM Coordinator*: coordinatore dei flussi informativi
 - *BIM Specialist*: operatore avanzato della gestione e della modellazione informativa
- “*Parte 8: Processi integrati di Gestione delle Informazioni e delle Decisioni*”. La specifica si presenta come unione tra la norma nazionale, il *project management* e la EN ISO 19650, ovvero gestisce l’integrazione fra attività e figure professionali tradizionali del settore e attività e figure informative.
- “*Parte 9: Gestione informativa in fase di esercizio*”. Non ancora pubblicata, questa specifica svilupperà il fascicolo digitale del fabbricato, elemento utile come strumento di garanzia della qualità del bene immobiliare.
- “*Parte 10: Organizzazione delle figure coinvolte nella gestione e nella modellazione informativa*”. Non ancora pubblicata, regolerà i criteri generali per la gestione delle informazioni in forma digitale. Attuerà quindi una standardizzazione delle procedure di scambio dei dati. [8]

1.3 Quadro Normativo Energia

L'attuale politica energetica adottata dall'Unione Europea impone agli stati membri un sempre più crescente utilizzo dell'energia proveniente da fonti rinnovabili ed una relativa diminuzione dell'energia proveniente da fonti fossili, al fine di rendere l'Unione indipendente dalle tradizionali fonti di energia che al momento vengono quasi interamente importate da Paesi esteri.

Il pacchetto clima-energia 20-20-20 impone agli stati membri una riduzione del 20% delle emissioni di gas serra, un aumento del 20% dell'utilizzo di energia da fonti rinnovabili (17% per quanto riguarda l'Italia) entro la fine del 2020. [9]

Le principali direttive europee trattanti il tema energia si dividono in:

- Direttive che regolano il mercato interno dell'elettricità:
 - Direttiva 5 giugno 2019/944/UE
 - Regolamento 2019/943/UE
- Regolamenti volti alla prevenzione rischio di blackout e cooperazione fra enti regolatori dell'energia:
 - Regolamento 2019/941/UE
 - Regolamento 2019/942/UE
- Direttiva sulle energie rinnovabili:
 - Direttiva 2018/2001/UE
- Direttiva sull'efficienza energetica:
 - Direttiva 2018/2002/UE
- Regolamento sulle procedure di gestione dell'energia:
 - Regolamento 2018/1999/UE
- **Direttiva sull'efficienza energetica in edilizia:**
 - **Direttiva 2018/844/UE**
- Direttiva e regolamenti sull'efficienza dei prodotti ad energia elettrica:
 - Direttiva 2009/125/Ce
 - Regolamento 2017/1369/UE
- Direttiva sui biocarburanti:
 - Direttiva 2015/1513/UE

1.3.1 Direttiva (UE) 2018/844

La direttiva UE 30 Maggio 2018/844 tratta argomenti in materia di prestazione energetica ed efficienza energetica degli edifici, nello specifico modifica le seguenti direttive:

- Direttiva 2010/31/UE riguardante la prestazione energetica degli edifici (riscrittura della Direttiva 2002/91/CE EPBD)
- Direttiva 2012/27/UE riguardante l'efficienza energetica (DEE)

La direttiva ha come fine il graduale abbassamento delle emissioni di gas a effetto serra per almeno il 40% entro il 2030 (del 80-95% se si considerano i livelli di emissioni del 1990) ed il miglioramento del parco edilizio al fine di ottenere un sistema energetico sostenibile e decarbonizzato entro il 2050 (l'iter prevede step intermedi nel 2030 e nel 2040). Ogni stato membro è tenuto ad adeguarsi alla direttiva entro il 20 Marzo 2020.

Nel dettaglio ogni stato avrà l'obbligo di migliorare la prestazione energetica di tutti gli edifici, adottare strategie nazionali per la riqualificazione degli immobili volte ad incentivare la realizzazione di edifici ad energia quasi zero (NZEB), pur mantenendo un buon rapporto costi/benefici.

La direttiva introduce un nuovo concetto chiamato “indicatore di predisposizione degli edifici all'intelligenza” come strumento per “*sensibilizzare i proprietari e gli occupanti sul valore dell'automazione degli edifici e del monitoraggio elettronico dei sistemi tecnici per l'edilizia e dovrebbe assicurare gli occupanti circa i risparmi reali di tali nuove funzionalità migliorate*”. [10]

Si dovranno inoltre ampliare le infrastrutture per la ricarica dei veicoli elettrici. Ogni edificio non residenziale, che esso sia nuovo o oggetto di ristrutturazione e che abbia almeno 10 posti auto, dovrà disporre di almeno un punto di ricarica per veicoli elettrici. [11]

1.3.2 ISO 52016

All'inizio del 2017 il CEN ha approvato il pacchetto normativo per l'applicazione della direttiva EPDB (adesso Direttiva UE 2018/844) ed in parallelo avviene il passaggio di tutto il pacchetto dall'ambito CEN europeo all'ambito ISO mondiale con numerazione riservata 52000.

Il 27 Giugno 2017 è stata pubblicata la ISO EN 52016 in sostituzione della più anziana UNI EN 13790 dalla quale deriva la UNI TS 11300. Le principali novità riguardano:

- Abolizione del metodo stagionale
- Revisione del metodo di calcolo mensile

- Integrazione con il gruppo delle norme EPB (normativa ISO 52000-1)
- Integrazione per il calcolo dei carichi termici estivi ed invernali
- Introduzione del metodo di calcolo dinamico orario

Al fine di chiarire il significato dei metodi di calcolo, si riporta uno schema per i significati dei termini utilizzati. Per metodo orario si indica l'intervallo di tempo compreso nel calcolo, esso si divide in :

- Annuale o stagionale: ormai abbandonato. In Italia non è mai stato utilizzato per le verifiche di legge.
- Mensile: segue gli intervalli definiti dal calendario. Tipologia usata nelle UNI/TS 11300.
- Orario: L'intervallo di tempo utilizzato per il calcolo è a base oraria. Metodo introdotto nelle nuove norme EN.

Per quanto riguarda i termini “stazionario” e “dinamico” si definisce:

- Stazionario: chiamato anche “quasi stazionario” o “statico”. Il calcolo effettuato su ogni intervallo di tempo e indipendentemente dagli altri calcoli.
- Dinamico: il calcolo su un intervallo di tempo tiene conto dei risultati ottenuti in tutti i calcoli svolti su gli intervalli di tempo precedenti.

È possibile svolgere calcoli combinando qualsiasi termine della prima tipologia con uno della seconda, in ogni caso quello che sta avvenendo adesso, con l'introduzione delle nuove norme, è il passaggio da “Calcolo Mensile Stazionario” a “Calcolo Orario Dinamico” per il riscaldamento ed il raffrescamento degli ambienti. [12]

Il nuovo metodo di calcolo garantisce risultati più precisi rispetto al precedente inoltre è di facile lettura. I precedenti metodi statici (usati nelle UNI/TS 11300), pur essendo efficaci ai fini del calcolo dei fabbisogni estivi ed invernali, presentano dei limiti che diventano evidenti nel calcolo in edifici con sistemi impiantistici vasti e complessi, carichi termici variabili, presenza di diverse attività degli occupanti, diverse condizioni d'uso degli impianti. Situazione simile anche nel caso del calcolo del fabbisogno di energia per il raffrescamento, a causa delle rapide variazioni delle condizioni esterne che rendono poco affidabili i risultati ottenuti dai metodi statici su base mensile. Queste carenze vengono colmate nel metodo dinamico orario che tiene conto delle variazioni orarie del clima e delle azioni degli occupanti. A differenza dei precedenti metodi, dove la zona termica veniva “semplificata” in un unico modello RC (Resistenza – Capacità), nel nuovo metodo dinamico orario ogni elemento costruttivo, sia opaco (pareti, solai, pavimenti) che trasparente (serramenti, coperture vetrate), viene modellato separatamente. Da notare che la mole di dati da inserire non si discosta molto da

quella richiesta da i precedenti metodi. Il nuovo metodo può essere applicato sia su edifici residenziali che non residenziali, in entrambi i casi gli edifici dovranno essere scomposti in zone termiche e su queste ultime è possibile calcolare:

- *“Fabbisogno energetico sensibile orario per riscaldamento e raffrescamento”*
- *“Fabbisogno energetico latente orario (per umidificazione/deumidificazione)”*
- *“Valore orario della temperatura dell’aria dell’ambiente”*
- *“Carico termico sensibile e latente su base oraria per riscaldamento e raffreddamento per determinare i valori massimi ai fini del dimensionamento di terminali e centrali”*
- *“Condizioni termoigrometriche dell’aria da immettere per garantire la necessaria umidificazione o deumidificazione.” [13]*

Il metodo di calcolo orario consente quindi di ottenere con precisione il fabbisogno di energia di un edificio, ponendo attenzione anche all’inerzia termica e proprietà termoisolanti dell’edificio. La possibilità di valutare con precisione i carichi termici di picco estivi ed invernali rende il dimensionamento degli impianti più efficiente evitando sovradimensionamenti indesiderati, inoltre è possibile monitorare le variazioni di temperatura e umidità all’interno degli ambienti. I risultati ottenuti saranno quindi molto più vicini alla condizione reale dell’edificio rispetto a quelli ricavati con i metodi classici. [13]

1.3.3 Recepimento nazionale delle normative europee

Le normative europee riguardanti le performance energetiche degli edifici sono state redatte in modo da prestare particolare attenzione alla modalità di scrittura degli algoritmi di calcolo. Queste norme andranno man mano a sostituire le vigenti normative nazionali, tuttavia questo passaggio non è immediato e viene effettuato per mezzo di una procedura di integrazione. Le normative europee dispongono infatti delle appendici nazionali (o allegati nazionali), ovvero una serie di allegati e tabelle caratterizzate dalla presenza di variabili riguardanti dati di contesto o scelte sull’utilizzo dell’algoritmo di calcolo, che saranno fissate dai vari enti normatori. In questo modo gli algoritmi fissati dalla normativa europea non vengono mai modificati, pur conservando una flessibilità tale da poter essere utilizzata dai vari enti in base alle proprie esigenze. Nello specifico, per il calcolo delle prestazioni energetiche degli edifici, la norma internazionale in vigore è la EN ISO 52016, tuttavia questa norma in Italia non sarà utilizzata fin quando non verrà redatta la versione italiana, ovvero la UNI EN ISO 52016 con relative appendici nazionali. Gli allegati nazionali vengono redatti dalle varie commissioni tecniche del Comitato Termotecnico Italiano. Una volta ultimato l’adeguamento normativo,

le norme utilizzate finora quali l'UNI/TS 11300 non saranno più utilizzate e sostituite con delle altre che faranno da linee guida per l'applicazione in Italia delle norme europee e conterranno una raccolta di moduli aggiuntivi a supporto di queste ultime. [14]

1.3.4 UNI EN ISO 52016

La norma ha come obiettivo la determinazione del carico sensibile di riscaldamento e raffrescamento, nonché valutare la temperatura interna invernale ed estiva e calcolare il fabbisogno di energia per il riscaldamento e raffrescamento degli edifici. Pubblicata nel 1° Marzo 2018, UNI EN ISO 52016 sostituisce completamente le procedure di calcolo riguardanti le analisi dei fabbisogni di energia termica. In particolare la nuova normativa sostituisce completamente le precedenti:

- UNI EN 15265 del 2008: *“Calcolo del fabbisogno di energia per il riscaldamento e il raffrescamento degli ambienti mediante metodi dinamici – Criteri generali e procedimenti di validazione”*
- UNI EN 15255 del 2008: *“Calcolo del carico sensibile di raffrescamento di un ambiente – Criteri generali e procedimenti di validazione”*
- UNI EN ISO 13790 del 2008: *“Prestazione termica degli edifici – Calcolo del fabbisogno di energia per il riscaldamento e il raffrescamento”*
- UNI EN ISO 13791 del 2012: *“Prestazione termica degli edifici – Calcolo della temperatura interna estiva di un locale in assenza di impianti di climatizzazione – Criteri generali e procedure di validazione”*
- UNI EN ISO 13792 del 2012: *“Prestazione termica degli edifici – Calcolo della temperatura interna estiva di un locale in assenza di climatizzazione – Metodi semplificati”*

Tuttavia ad oggi dovranno ancora essere utilizzate le normative precedenti, l'UNI EN ISO 52016 non potrà essere utilizzata fin quando sono saranno stati ultimati i lavori per la redazione della propria appendice nazionale ed il relativo adeguamento delle UNI TS 11300 al nuovo testo normativo. [15]

1.3.5 UNI TS 11300

Fino al completamento dell'appendice nazionale della UNI EN ISO 52016, la normativa tecnica di riferimento in Italia rimane la UNI/TS 11300 che si presenta come strumento per la stima delle prestazioni energetiche degli edifici ed è costituita da 6 parti. In dettaglio:

- *“Parte 1: Determinazione del fabbisogno di energia termica dell'edificio per la climatizzazione estiva ed invernale”;*

In vigore dal 02 Ottobre 2014

“La specifica tecnica definisce le modalità per l'applicazione nazionale della UNI EN ISO 13790:2008 con riferimento al metodo mensile per il calcolo dei fabbisogni di energia termica per riscaldamento e per raffrescamento. La specifica tecnica è rivolta a tutte le possibili applicazioni previste dalla UNI EN ISO 13790:2008 calcolo di progetto (design rating), valutazione energetica di edifici attraverso il calcolo in condizioni standard (asset rating) o in particolari condizioni climatiche e d'esercizio (tailored rating).” [16]

- *“Parte 2: Determinazione del fabbisogno di energia primaria e dei rendimenti per la climatizzazione invernale, per la produzione di acqua calda sanitaria, per la ventilazione e per l'illuminazione in edifici non residenziali”.*

In vigore dal 07 Febbraio 2019

“La specifica tecnica fornisce dati e metodi di calcolo per la determinazione dei fabbisogni di energia termica utile per il servizio di produzione di acqua calda sanitaria, nonché di energia fornita e di energia primaria per i servizi di climatizzazione invernale e acqua calda sanitaria. Essa fornisce inoltre il metodo di calcolo per la determinazione del fabbisogno di energia primaria per il servizio di ventilazione e le indicazioni e i dati nazionali per la determinazione dei fabbisogni di energia primaria per il servizio di illuminazione in accordo con la UNI EN 15193. La specifica tecnica fornisce dati e metodi per il calcolo dei rendimenti e delle perdite dei sottosistemi di generazione alimentati con combustibili fossili liquidi o gassosi. La specifica tecnica si applica a sistemi di nuova progettazione, ristrutturati o esistenti: per il solo riscaldamento, misti o combinati per riscaldamento e produzione acqua calda sanitaria, per sola produzione acqua calda per usi igienico-sanitari, per i sistemi di sola ventilazione, per i sistemi di ventilazione combinati alla climatizzazione invernale, per i sistemi di illuminazione negli edifici non residenziali.” [16]

- *“Parte 3: Determinazione del fabbisogno di energia primaria e dei rendimenti per la climatizzazione estiva”*

In vigore da 25 Marzo 2010

“La specifica tecnica fornisce dati e metodi per la determinazione: dei rendimenti e dei fabbisogni di energia dei sistemi di climatizzazione estiva; dei fabbisogni di energia primaria per la climatizzazione estiva. La specifica tecnica si applica unicamente ad impianti fissi di climatizzazione estiva con macchine frigorifere azionate elettricamente o ad assorbimento. La specifica tecnica si applica a sistemi di nuova progettazione, ristrutturati o esistenti: per il solo raffrescamento; per la climatizzazione estiva. La specifica tecnica non si applica ai singoli componenti dei sistemi di climatizzazione estiva per i quali rimanda invece alle specifiche norme di prodotto.” [16]

- *“Parte 4: Utilizzo di energie rinnovabili e di altri metodi di generazione per la climatizzazione invernale e per la produzione di acqua calda sanitaria”*

In vigore da 31 Marzo 2016

“La specifica tecnica calcola il fabbisogno di energia primaria per la climatizzazione invernale e la produzione di acqua calda sanitaria nel caso vi siano sottosistemi di generazione che forniscono energia termica utile da energie rinnovabili o con metodi di generazione diversi dalla combustione a fiamma di combustibili fossili trattata nella UNI/TS 11300-2.

Si considerano le seguenti sorgenti di energie rinnovabili per produzione di energia termica utile: solare termico; biomasse; fonti aerauliche, geotermiche e idrauliche nel caso di pompe di calore per la quota considerata rinnovabile; e per la produzione di energia elettrica: solare fotovoltaico.” [16]

- *“Parte 5: Calcolo dell’energia primaria e della quota di energia da fonti rinnovabili”*

In vigore da 31 Marzo 2016

“La presente specifica tecnica fornisce metodi di calcolo per determinare in modo univoco e riproducibile applicando la normativa tecnica citata nei riferimenti normativi il fabbisogno di energia primaria degli edifici sulla base dell’energia consegnata ed esportata; la quota di energia da fonti rinnovabili. La presente specifica tecnica fornisce inoltre precisazioni e metodi di calcolo che riguardano, in particolare: 1) le modalità di valutazione dell’apporto di energia rinnovabile nel bilancio energetico; 2) la valutazione dell’energia elettrica esportata; 3) la definizione delle modalità di compensazione dei fabbisogni con energia elettrica attraverso energia elettrica prodotta da rinnovabili; 4) la valutazione dell’energia elettrica prodotta da unità cogenerative.” [16]

- *“Parte 6: Determinazione del fabbisogno di energia per ascensori e scale mobili”*

In vigore da 31 Marzo 2016

“La presente specifica tecnica fornisce dati e metodi per la determinazione del fabbisogno di energia elettrica per il funzionamento di impianti destinati al sollevamento e al trasporto di persone o persone accompagnate da cose in un edificio, di seguito detti impianti, sulla base delle caratteristiche dell’edificio e dell’impianto. I suddetti metodi di calcolo tengono in considerazione solo il fabbisogno di energia elettrica nei periodi di movimento e di sosta della fase operativa del ciclo di vita.

La presente specifica tecnica si applica ai seguenti impianti: ascensori; montascale; piattaforme elevatrici; montacarichi e montauto; scale mobili; marciapiedi mobili.

La presente specifica tecnica si applica alle seguenti tipologie di edifici: edificio residenziale; albergo; ufficio; ospedale; edificio adibito ad attività scolastiche e ricreative; centro commerciale; edificio adibito ad attività sportive; edificio adibito ad attività industriali e artigianali; edificio adibito al servizio di trasporto pubblico (stazione, aeroporto, ecc.).

La presente specifica tecnica può essere applicata a tipologie di edifici diverse da quelle indicate, previa una valutazione preliminare specifica del traffico richiesto. L’Appendice A fornisce, a titolo informativo, esempi di calcolo del fabbisogno energetico.” [16]

1.3.6 Decreto Ministeriale 26 Giugno 2015

Il 26 Giugno 2015 sono stati emanati tre decreti che completano il quadro normativo in efficienza energetica degli edifici. In particolare i decreti sono:

- Decreto su i requisiti minimi
- Decreto sulle linee guida del nuovo APE 2015
- Decreto sulla relazione tecnica di progetto

Il decreto requisiti minimi prevede un differente approccio in base alla tipologia di edificio ed intervento da eseguire. In base alla tipologia di intervento vengono stabiliti diversi requisiti e verifiche di legge. Le tipologie di intervento previste si dividono in:

- Nuova costruzione
- Demolizione e ricostruzione
- Ampliamento e sopraelevazione con nuovo impianto o estensione di impianto
- Ristrutturazione importante

- di primo livello
- di secondo livello
- Riqualificazione energetica
 - Riguardante l'involucro
 - Installazione nuovo impianto
 - Ristrutturazione di impianto
 - Sostituzione del generatore

[17]

1.4 Comitato Termotecnico italiano

Il CTI affonda le sue radici nel lontano 1918 quando venne costituita l'Associazione Termotecnica Italiana. Nel 1933 viene fondato il Comitato Termotecnico Italiano dall'Associazione Nazionale per il Controllo della Combustione. Nel 1950 ad iniziativa dell'ANCC e dell'ATI viene ricostituito in associazione il "Comitato Termotecnico Italiano C.T.I.", organismo autonomo con attività di normazione volontaria. Nel 1998 viene aggiunta la dizione "Energia e Ambiente" e nel 1999 l'ente ha ottenuto il riconoscimento della personalità giuridica con decreto del Ministero che corrisponde all'attuale Ministero dello Sviluppo Economico. [18]

Il C.T.I. Energia e Ambiente è un Ente federato all'UNI che svolge le proprie funzioni in ambito nazionale ed internazionale:

- Ambito nazionale: funzione di supporto normativo a leggi e regolamenti in cambio termo-energetico
- Ambito internazionale: partecipazione ai lavori del CEN e dell'ISO

Nello specifico il CTI Energia e Ambiente:

- Svolge collaborazioni con Organi Pubblici, Enti, Aziende e Associazioni del settore
- Collaborazione con Organismi Europei internazionali di normazione
- Promozione e partecipazione a studi, pubblicazioni e discussioni a carattere scientifico e tecnico
- Attività generali utili al raggiungimento degli scopi sociali [19]

1.4.1 Validazione dei software per la prestazione energetica degli edifici

Oltre alle funzioni sopracitate, ai sensi dell'art.7 DM 26 giugno 2015 il CTI svolge *"un'importante attività di verifica dei software commerciali e degli strumenti di calcolo della prestazione energetica degli edifici"* [20]

I dettagli sulle modalità di verifica possono essere consultati nel:

- DM 26 Giugno 2015
- D.Lgs 192/05

L'8 Giugno 2016 è entrato in vigore il regolamento utilizzato tutt'oggi per la verifica degli strumenti di calcolo e dei software commerciali ed il 29 Giugno dello stesso anno il CTI ha pubblicato l'elenco degli applicativi informatici certificati.

Dal 29 Giugno 2016 potranno quindi essere utilizzati, come strumento di calcolo della prestazione energetica degli edifici, solo i software presenti nell'elenco del CTI consultabile

alla pagina web cti2000.it (che costituisce il riferimento ufficiale del CTI “Registro dello strumento nazionale” per quanto concerne la conformità degli applicativi informatici)

Il “regolamento per la verifica di strumenti di calcolo e software commerciali ai fini del rilascio della dichiarazione CTI” [21] in conformità con il Decreto del Ministero dello Sviluppo Economico 26 Giugno 2015 stabilisce le modalità di verifica che i software devono rispettare, tra cui:

- Gli indici di prestazione energetica calcolati dal software o dallo strumento di calcolo devono avere uno scostamento massimo del $\pm 5\%$ rispetto ai corrispondenti dati determinati con l'applicazione dello strumento nazionale.
- Il software o lo strumento di calcolo che utilizza metodi semplificati deve riportare uno scostamento massimo compreso fra il $+20\%$ e il -5% rispetto ai corrispondenti dati determinati con l'applicazione dello strumento nazionale.

“Alla data di entrata in vigore del presente Regolamento sono in vigore le norme tecniche nazionali, predisposte in conformità allo sviluppo delle norme EN a supporto della direttiva 2010/31/UE, riportate nell'allegato D.” [21]

Il regolamento inoltre fornisce delle definizioni, vengono citate le più significative per quanto concerne la procedura di validazione:

- *“Applicativo informatico: Strumento di calcolo o software commerciale che implementa i metodi di calcolo definiti dal presente Regolamento al fine di produrre un risultato (indici di prestazione energetica) a seguito dell'imputazione di dati e di informazioni in ingresso.” [21]*
- *“Caso studio: esempio di edificio facente parte integrante dello strumento nazionale di riferimento e nel quale sono definiti i dati di ingresso e le informazioni necessarie per il calcolo delle prestazioni energetiche secondo le metodologie di cui al comma 1 dell'art. 3 del Decreto e sono forniti in uscita i risultati intermedi e finali (indici di prestazione energetica) funzionali alla verifica.” [21]*
- *“Comitato di verifica: commissione costituita da almeno tre componenti, nominati dalla Presidenza del CTI, di cui uno nominato responsabile del CdV. La commissione è supportata da un funzionario tecnico del CTI che ha seguito il processo di verifica e che svolge la funzione di segretario. I componenti del comitato di verifica sono tenuti a sottoscrivere un accordo di riservatezza con il CTI e una dichiarazione di assenza di conflitto di interessi.” [21]*
- *“Dichiarazione CTI: attestazione di terza parte rilasciata dal CTI in attuazione all'art. 7 del Decreto: - previa verifica del rispetto delle condizioni dal presente*

Regolamento, - mediante Certificato di garanzia di conformità.” [21]

- *“Indici di prestazione energetica: indici definiti alla Tabella 3 del punto 3.3 dell’Allegato I al Decreto, come “indice di prestazione energetica”.” [21]*
- *“Metodo di calcolo: algoritmo, stabilito dalle metodologie di cui al comma 1 dell’art. 3 del Decreto e definito dal punto 4 delle Linee Guida, utilizzato per calcolare gli indicatori numerici di prestazione energetica richiesti a partire dagli opportuni dati di ingresso. In riferimento alle tipologie di valutazione energetica indicate nella UNI/TS 11300, ai fini del presente Regolamento si prendono in considerazione le tipologie A1 (sul progetto) e A2 (standard).” [21]*
- *“Metodo di calcolo semplificato: metodo di calcolo contenente ulteriori semplificazioni rispetto al metodo di calcolo di cui al punto 4.2.1 (metodo di calcolo da rilievo sull’edificio) delle Linee Guida. Nota: ai sensi dell’art. 6, comma 12, lettera a) del D.lgs 192/05 le metodologie di calcolo semplificate sono finalizzate a ridurre i costi a carico dei cittadini.” [21]*
- *“Registro dello strumento nazionale: elenco degli applicativi informatici per i quali è stato rilasciato il riferimento alla richiesta di verifica di cui al successivo punto 6.6 e, se rilasciata, la Dichiarazione CTI, in applicazione al presente Regolamento. Nota: il Registro dello strumento nazionale è depositato presso la Segreteria CTI ed è reso pubblico in una apposita sezione del sito internet del CTI.” [21]*

La procedura di validazione avviene come descritto in Fig. 13.

Il CTI fornisce dei casi studio che saranno utilizzati in fase di verifica preliminare del software, per stabilire se vi è la presenza di criticità o scostamenti significativi dei risultati attesi. In dettaglio, come previsto dal DM 26 giugno 2015, tabella 3 del punto 3.3, non vi deve essere una variazione importante degli indici (riferiti alla quota non rinnovabile e totale dell’energia primaria):

- EPH [kWh/m²]: indice di prestazione energetica per la climatizzazione invernale;
- EPW [kWh/m²]: indice di prestazione energetica per la produzione di acqua calda sanitaria;
- EPV [kWh/m²]: indice di prestazione energetica per la ventilazione.
- EPC [kWh/m²]: indice di prestazione energetica per la climatizzazione estiva.
- EPL [kWh/m²]: indice di prestazione energetica per l’illuminazione artificiale.
- EPT [kWh/m²]: indice di prestazione energetica per il servizio per il trasporto di persone e cose (impianti ascensori, marciapiedi e scale mobili).
- EPgl [kWh/m²]: indice di prestazione energetica globale dell’edificio. [21]



Fig. 13: Procedura di verifica CTI
allegato A - Regolamento per la procedura di verifica

Il 02 Dicembre 2019 è stata pubblicata una integrazione al regolamento di strumenti di calcolo e software commerciali. In sintesi questa integrazione prevede un controllo del mantenimento dei requisiti necessari all'ottenimento del certificato di garanzia di conformità del CTI da effettuare a cadenza biennale.

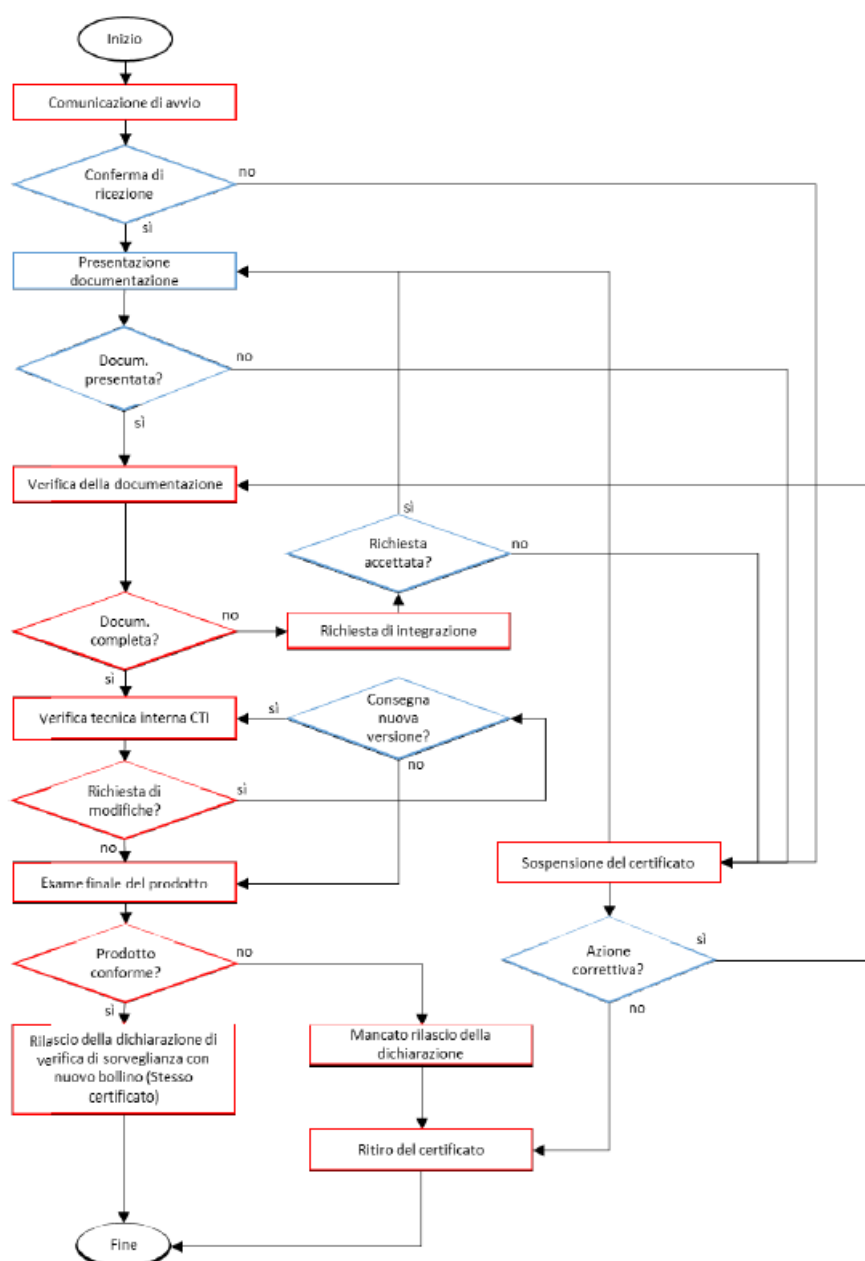


Fig. 14: Aggiornamento 2019 della procedura di verifica CTI
allegato A - Regolamento per la procedura di verifica

1.5 Procedure di calcolo e formati di scambio

I calcoli effettuati attualmente basati sulla norma UNI TS 11300 fanno parte dei metodi statici. Questa tipologia di calcolo è efficace per edifici semplici, ma al moltiplicarsi dei dati da trattare i risultati possono essere non del tutto soddisfacenti. Nel caso del calcolo per il raffrescamento ad esempio, i risultati sono poco affidabili, a causa delle variazioni continue e repentine delle condizioni esterne. D'altra parte metodi di calcolo semplificati risultano essere necessari data la complessità di equazioni come quella del calore in regime variabile. Nel tempo sono stati definiti diversi metodi, come il *Transfer Function Method* implementato dal software libero *Energy Plus* del DOE, il metodo *Heat Balance* o il *Radiative Time Series*. Una caratteristica che accomuna tutti i metodi appena citati è il rilevante impiego di risorse di calcolo e l'utilizzo di software specifici. [13]

1.5.1 Dalla modellazione alle analisi tecniche

Come anticipato, servirsi di strumenti che permettano di effettuare analisi energetiche accurate è ormai una necessità. Allo stato attuale vi sono però delle limitazioni nell'utilizzo della piattaforma BIM per quanto riguarda il settore energia. È prassi diffusa infatti affidarsi a strumenti che siano specializzati nelle analisi energetiche, i BEM software (o BEM tools). Questi software principalmente basano le loro analisi su caratteristiche del modello ben precise quali:

- Clima a cui una struttura è soggetta
- Involucro dell'edificio
- Zone termiche di cui l'edificio è composto
- Impianti e generatori adoperati

È chiaro come l'interoperabilità fra BIM software e BEM tool è un aspetto fondamentale che non va trascurato. L'edificio moderno infatti è soggetto ad un mole di dati molto vasta, che in alcuni casi, come accade spesso per edifici pubblici o industriali di estese dimensioni, rende indispensabile l'automazione nel trattamento di tali dati. L'interoperabilità è resa possibile attraverso l'utilizzo di formati di scambio non proprietari che le varie *software house* devono tenere in considerazione nello sviluppo dei propri software. Al momento i software BIM offrono diverse opzioni di esportazione dei dati presenti nel modello digitale, tuttavia i formati che più si addicono alle valutazioni energetiche sono il Green Building XML e l'Industry Foundation Classes.

Alla luce di questo scenario appare chiaro come l'utilizzo di strumenti altamente efficienti

dotati di funzioni interoperabili, che tengano conto delle vigenti normative è decisamente indispensabile. Citando le parole del presidente dell'ANTA, Socal Laurent il quale sottolinea come *“lo sviluppo del software applicativo è infatti il momento della verità per le norme di calcolo; i linguaggi di programmazione non consentono l'uso di forme ed espressioni ambigue o “politiche”, una decisione si deve prendere, sempre.”* [12]

1.6 Green Building eXtensive Markup Language



Fig. 15: Logo gbXML

<https://www.gbxml.org>

Lo schema Green Building XML, o più comunemente chiamato gbXML è stato ideato per facilitare il trasferimento delle informazioni degli edifici fra diversi software, che essi siano BIM software o strumenti per analisi specifiche. Attualmente è uno dei formati di scambio più utilizzati e diverse *software house* dispongono di prodotti in grado di esportare ed importare informazioni organizzate secondo lo schema gbXML. [22]

1.6.1 Background

Lo sviluppo di Green Building XML iniziò nel 1999, finanziato dalla California Energy Commission PIER Program, Pacific Gas and Electric e Green Building Studio. Lo schema è stato presentato nel mese di giugno del 2000 dalla compagnia Green Building Studio e in poco tempo divenne largamente utilizzato come lo schema utile ad effettuare analisi delle prestazioni degli edifici. Nel 2002 viene lanciato un sito internet dedicato, ovvero gbXML.org al fine di pubblicizzare lo schema. Nel 2009 l'azienda Autodesk acquista Green Building Studio, compagnia che aveva creato e implementato lo schema gbXML. Nel 2009 gbXML diventa indipendente da Green Building Studio ed assume il nome ufficiale di “*Open Green Building XML Schema, Inc*”. Lo schema continua ad essere implementato e nel 2013 riceve dei finanziamenti dall'U.S. DOE. Nel 2016 venne realizzato Open Studio, ovvero il primo software conforme al formato gbXML. Nel 2019 sono circa 50 i software che supportano il formato gbXML. [22]

1.6.2 Extensive Markup Language

Il formato gbXML basa il suo funzionamento su quello che in programmazione viene chiamato Extensible Markup Language. Quest'ultimo rappresenta un insieme di regole per la codifica delle informazioni, scritte in modo tale da rendere il file leggibile dagli utenti e dai computer. XML è un metalinguaggio, ovvero un linguaggio ben definito basato su regole precise, realizzato allo scopo di definire altri linguaggi. La parola *markup* (da *marking up*) indica la definizione di un *layout* del testo organizzato secondo convenzioni standard. “*Extensible*” indica la possibilità di estendere tale linguaggio con *tag* (etichette) personalizzate. I *tag* sono l'elemento che consente di assegnare un significato specifico a parti del testo come ad esempio l'indicazione di parametri o racchiudendo dell'altro testo informativo. Data la sua

struttura, XML può essere utilizzato per diversi campi d'applicazione quali la definizione della struttura di un documento, la riproduzione di elementi multimediali o lo scambio di informazioni. [23]

1.6.3 Funzionamento gbXML

Generalmente un file gbXML contiene le informazioni del progetto riguardanti la geometria e le proprietà di quest'ultimo. Se il programma utilizzato esporta correttamente le informazioni, queste dovrebbero riguardare:

- Blocchi e zone termiche
- Geolocalizzazione e descrizione dell'edificio
- Proprietà termiche dei materiali
- Stratigrafie delle strutture
- Caratteristiche geometrie e posizione nello spazio delle strutture opache
- Caratteristiche e posizione nello spazio delle strutture trasparenti

Data la natura rigida dei file XML non sono concessi errori di sintassi e le informazioni devono essere organizzate in modo gerarchico, seguendo regole ferree.

Si riporta un esempio di file XML fornito dal sito internet ufficiale gbXML.org.

La prima riga del file indica sempre la versione XML utilizzata e lo standard *Unicode* di riferimento. Ogni file gbXML in conformità con le regole XML deve inoltre avere un *root element* ovvero un elemento da cui dipendono tutti i successivi *tag* inseriti nel file. I *tag* sono contraddistinti da parentesi angolari, "<" per apertura e ">" per la chiusura del *tag*, mentre per indicare la fine del *tag* viene utilizzato il segno "/". Viene riportata una informazione come esempio:

```
<Latitude>33.748310</Latitude>
```

Lo stesso principio basato su *tag* viene applicato anche per la definizione della gerarchia fra le informazioni, come ad esempio:

```
<Location>  
  <Latitude>33.748310</Latitude>  
  <Elevation>994.09449</Elevation>  
</Location>
```

È necessario seguire alcune regole di scrittura quali:

- Scrittura corretta dei *tag*. Da notare che le lettere in maiuscolo vengono considerate differenti dall'equivalente minuscola.
- Bilanciamento fra *tag*. Ad ogni *tag* di apertura deve corrispondere un *tag* di chiusura.

1.7 Industry Foundation Classes

1.7.1 Background

Lo standard IFC affonda le sue radici nel 1994 quando l'Industry Alliance for Interoperability, un consorzio industriale formato da dodici società statunitensi, iniziò ad investire nella realizzazione di un codice informatico costituito da un insieme di classi C++ “in grado



Fig. 16: Iter normativo

<https://www.ibimi.it/partecipare-alle-attivita-buildingsmart/>

standard IFC. Adesso l'organizzazione è conosciuta come BuildingSMART International definita come “open, neutral and international not-for-profit organization” [24]. Dal 2005 ad oggi buildingSmart International pubblica il proprio operato tramite capitoli nazionali chiamati buildingSMART (IBIMI è il Capitolo Italiano di buildingSMART) inoltre di recente l'organizzazione ha definito degli standard per qualificare delle figure professionali in ambito BIM. È importante sottolineare l'accordo fra BuildingSMART International ed ISO tramite la norma ISO 16739, la quale stabilisce che ogni aggiornamento allo standard IFC da parte di buildingSMART viene automaticamente recepito come nuova versione riconosciuta dalla norma ISO. Questo accordo influenza direttamente ogni stato membro, in quanto ogni

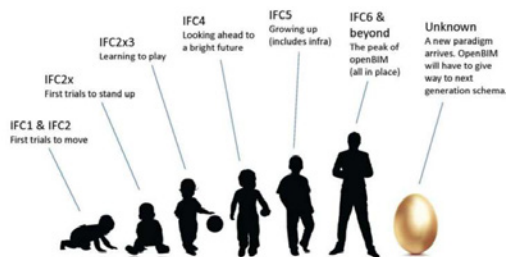


Fig. 17: Livelli di maturità del formato IFC

<https://www.01building.it/bim/standard-buildingsmart/>

di supportare lo sviluppo di applicazioni integrate” [1].

A partire dal 1995 furono aperte le iscrizioni alle parti interessate e nel 1997 il nome dell'alleanza cambiò in International Alliance for Interoperability, riconosciuta come organizzazione no profit istituita con l'obiettivo di sviluppare e promuovere lo

norma ISO viene automaticamente recepita dal CEN il quale si limita a verificare che non vi siano conflitti fra queste ultime e le norme comunitarie. Di conseguenza, per via dell'accordo di Vienna, ogni norma CEN viene automaticamente recepita dagli stati membri, tramite azione dell'UNI per quanto riguarda Italia. [25]

1.7.2 ISO 16739:2013

Come detto in precedenza, occorre precisare che IFC è uno standard. Gli standard in generale sono documenti contenenti linee guida, caratteristiche, specifiche tecniche e requisiti riguardanti un determinato servizio o processo, redatti in modo da garantirne il corretto uso, nonché qualità nel prodotto finale. Per quanto riguarda il processo, uno standard è uno strumento utile al fine di evitare errori e ridurre i costi. IFC è divenuto uno standard ISO, ovvero la sua approvazione è stata valutata da un gruppo di esperti di un comitato tecnico. Il progetto riguardante l'IFC è stato esaminato e condiviso dai membri ISO, una volta ottenuto il consenso dei membri, il progetto è stato riconosciuto come standard. [26]

Oltre allo standard IFC sono stati riconosciuti altri due standard con le relative norme ISO che li regolano:

- ISO 12006-3: questa norma definisce lo standard *IFD (Information Framework for Dictionary)*
- ISO 29481: questa norma definisce lo standard *IDM (Information Delivery Manual)*

1.7.3 Definizione IFC

Lo standard IFC nasce allo scopo di facilitare l'interoperabilità fra i vari operatori nel settore AEC, evitando che in questo passaggio si verifichino perdite o modifiche indesiderate dei dati. L'IFC è un formato aperto non proprietario, ovvero non è controllato da un singolo produttore software e il suo campo d'applicazione comprende l'intero ciclo di vita dell'edificio, dalle prime fasi di progettazione fino alla pianificazione della realizzazione ed alla futura manutenzione. BuildingSMART identifica lo standard IFC come il principale strumento per l'interscambio di informazioni in ambito OPEN BIM, definendolo come uno strumento fine ad un “*approccio universale alla collaborazione per la progettazione e la costruzione degli edifici basati su standard e flussi di lavoro aperti*”. [1]

La struttura dell'IFC è pensata allo scopo di descrivere ogni componente dell'edificio con informazioni di tipo geometrico e non geometrico come proprietà specifiche dell'elemento.

IFC codifica i dati in tre possibili formati:

- IFC-SPF: definito dalla ISO 10303-21 (STEP-File), è il formato predefinito. In questo tipo di formato ogni riga consiste in un oggetto singolo registrato. Il principale vantaggio di questo formato sono le dimensioni compatte e la possibilità di poter essere letto personalmente dall'utente. L'estensione del file è .ifc
- IFC-XML: definito dalla ISO 10303-28 (STEP-XML), è adatto all'interoperabilità di

strumenti XML e lo scambio di modelli riguardanti parti dell'edificio. Meno utilizzato rispetto al formato predefinito, l'estensione del file è .ifcXML

- IFC-ZIP: versione archivio compresso dei formati precedenti, la caratteristica interessante è la possibilità di poter contenere anche materiale aggiuntivo. L'estensione del file è .ifcZIP

Le versioni IFC sono state costantemente aggiornate sin dal 1996, anno in cui fu pubblicata la prima versione (IFC1.0).

1.7.4 Funzionamento IFC

L'IFC basa il suo funzionamento su EXPRESS il quale è un linguaggio per la modellazione dati riguardanti un prodotto. Quando si parla di linguaggi di modellazione dei dati o *data modeling language* si intende procedure utili al fine di esprimere e riportare informazioni e concetti organizzati secondo delle regole precise. Queste ultime vengono disposte secondo uno schema che costituisce il modello dati o *data model* di un prodotto. EXPRESS è diventato uno standard da quando è stato approvato e standardizzato da ISO con la norma ISO 10303-11 il cui titolo ufficiale è “*Industrial Automation system and integration – Product data representation and exchange – Part 11: Description methods: The EXPRESS language reference manual*” [27], informalmente conosciuta come “*STEP*”, acronimo che indica “*Standard for The Exchange of Product model data*” [27]. La norma riguardante la struttura a cui un file STEP deve attenersi è la ISO 10303-21 “*Implementation methods: Clear text encoding of the structure*” [28]. Come descritto dalla norma, EXPRESS pur non essendo un linguaggio di programmazione, il suo sviluppo è stato influenzato da diversi linguaggi come ADA, Algol, C, C++, Euler, Modula-2, Pascal, PL/I, SQL. Il funzionamento di EXPRESS si basa sulla definizione di uno schema che identifica un dominio, quindi è possibile valutare se una data istanza risiede o meno nel medesimo dominio. Questo concetto è molto simile a quello su cui si basa XML, tuttavia le possibilità in EXPRESS risultano superiori in quanto quest'ultimo include nel suo funzionamento diversi operatori matematici e logici e consente l'uso di procedure e funzioni come parti integranti dello schema.

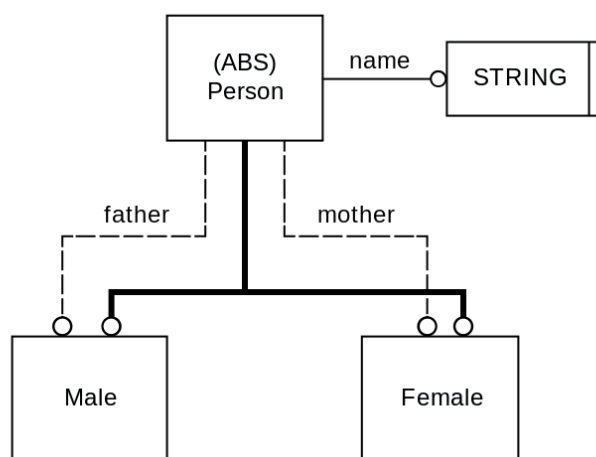
La norma stabilisce come obiettivi di EXPRESS:

- Il linguaggio deve essere comprensibile sia da umani che da computer
- Il linguaggio deve essere divisibile in diverse parti
- Il linguaggio deve definire delle entità che hanno proprietà e soggetti a cui essere applicate

- Il linguaggio deve essere neutro per implementazioni o contesto, tuttavia deve poter supportare modelli per particolari intenti.

L'elaborato principale dello schema EXPRESS è costituito da un file di testo formato da *physical lines*. Per *physical line* si intende una linea di testo comprendente caratteri e numeri che si conclude con un invio a capo o *Linefeed*, punto di partenza per la *physical line* successiva. Ad oggi è consentito usare in EXPRESS solo i caratteri definiti in ISO/IEC 10646 (*Unicode*) nell'intervallo compreso tra 20 e 7E, ovvero i caratteri *ASCII*, 09 (*Tab*), 0A (*Linefeed*), 0D (*Carriage Return*) e il carattere speciale “\n” che indica *newline*. *ASCII* è quindi un codice per la codifica di caratteri, pubblicato nel 1968. Oggi ne esistono diverse varianti tra cui *extended ASCII* chiamato anche *high ASCII* che è un'estensione dello standard principale in quanto include altri caratteri quali vocali accentate, simboli semigrafici e altri simboli usati meno frequentemente.

La norma ISO 10301-11 descrive anche un'altra modalità di visualizzazione dati chiamata *EXPRESS-G* (“G” indica l'iniziale della parola *Graphic*). Questo elaborato risulta molto utile al fine della comprensione rapida di uno schema. Caratterizzato da facile lettura e simboli specifici per ogni azione, *EXPRESS-G* consente agli operatori di individuare facilmente le connessioni fra le varie entità presenti in uno schema EXPRESS. Tuttavia vi è da sottolineare che questa tipologia di elaborato non è in grado di esporre la totalità delle informazioni presenti nell'equivalente file di testo, nonché formato ufficiale di *EXPRESS*.



```

SCHEMA Family;

ENTITY Person
  ABSTRACT SUPERTYPE OF (ONEOF (Male,
  Female));
  name: STRING;
  mother: OPTIONAL Female;
  father: OPTIONAL Male;
END_ENTITY;

ENTITY Female
  SUBTYPE OF (Person);
END_ENTITY;

ENTITY Male
  SUBTYPE of (Person);
END_ENTITY;

END_SCHEMA;
  
```

Fig. 18: Esempio confronto schema EXPRESS-G e schema EXPRESS

https://upload.wikimedia.org/wikipedia/commons/thumb/c/c3/EXPRESS-G_diagram_for_Family_schema.svg/710px-EXPRESS-G_diagram_for_Family_schema.svg.png

Si riporta un breve esempio di modello dati in cui l'entità *Person* è la *Supertype entity* a cui

appartengono due *Subtype entity* che sono *Male* e *Female*. Il collegamento fra *supertype* e *subtype* in questo caso è obbligatorio direzionale di tipo *ONEOF* quindi può essere vera solo una delle due *subtype entity*. L'entità *Person* inoltre dispone di un collegamento obbligatorio direzionale *Name* verso una stringa che ne indica il nome e due collegamenti opzionali direzionali che sono *Father* e *Mother* anch'essi collegati con *Male* e *Female*. L'esempio riportato in Fig. 18 mette a confronto uno schema EXPRESS-G ed il suo equivalente EXPRESS. Appare chiaro come la comprensione della versione grafica sia molto più immediata rispetto alla corrispondente di testo.

EXPRESS dispone di diverse tipologie di dati, ognuna delle quali ha un simbolo corrispondente in EXPRESS-G.

Si riportano le tipologie di dati presenti in EXPRESS:

- *Entity data type*: tipologia principale in EXPRESS, organizzate secondo una gerarchia tra *Supertype* e *Subtype entity* e con i loro attributi.
- *Enumeration data type*: rappresentano degli insiemi finiti di identificatori specificati utili alle enumerazioni di dati, possono essere estesi in altri schemi
- *Defined data type*: indicano ulteriori specifiche riguardanti la tipologia dei dati
- *Select data type*: definiscono una scelta fra diverse opzioni, possono essere estesi ad altri schemi.
- *Simple data type*: la forma più comune di dati che comprende diverse tipologie:
 - *String*: dati che contengono caratteri compresi nella ISO 10646/Unicode, può avere una lunghezza qualsiasi
 - *Binary*: usata raramente, di dimensioni limitate a 32 bit
 - *Boolean*: comprende una scelta fra due valori che sono *True* o *False*
 - *Logical*: similmente a *Boolean* comprende una scelta fra tre opzioni che sono *True*, *False* e *Unknown*
 - *Number*: riferito ai valori numerici, comprende la tipologia *Integer* per i numeri interi e *Real* per i numeri in virgola mobile.
- *Aggregation data type*: descrive il tipo di aggregazione fra i dati che può essere di tipo *Set*, *Bag*, *List* o *Array*:
 - *Set*: insieme di valori che non segue un ordine preciso. Ogni valore è presente una sola volta nell'insieme
 - *Bag*: insieme di valori che non segue un ordine preciso. Ogni valore può essere presente svariate volte nell'insieme
 - *List*: insieme di valori disposti in un ordine ben preciso. Ogni valore presente

nell'insieme deve avere un riferimento assegnato

- *Array*: insieme di valori disposti in un ordine ben preciso. I valori non devono avere obbligatoriamente un riferimento assegnato.

Tutte le entità possono essere un *subtype* o un *supertype* di un'altra entità, non vi è alcun limite al numero di collegamenti possibili. Le entità possono essere collegate a degli attributi e in questo modo ne assimilano il contenuto. I collegamenti possono essere di tre tipi:

- *Explicit*: collegamenti con gli attributi direttamente visibili nel file STEP
- *Derived*: gli attributi sono ricavati da una espressione. Altri attributi possono fare parte della stessa espressione
- *Inverse*: non vengono aggiunte informazioni ma dei vincoli a degli altri collegamenti entità-attributo

Principalmente lo standard ifc segue le regole di EXPRESS e l'organizzazione del suo schema si basa su una ripartizione delle parti stabilita secondo una precisa gerarchia. La lingua di riferimento per nozioni e nomenclatura dei dati è la lingua l'inglese ed in base a quest'ultima vengono definiti *types* (tipi), *entities* (entità), *rules* (regole) e *function* (funzioni). Per questi elementi prima di ogni parola è presente il prefisso “Ifc” e tutte le parole successive al prefisso devono essere unite fra di esse con la prima lettera in maiuscolo (tipo di notazione chiamata *CamelCase*). I nomi degli attributi riguardanti le varie entità non hanno il prefisso, ma anch'essi sono scritti in *CamelCase*. I *property set* (set di proprietà) hanno il prefisso “Pset_” e vengono scritti anch'essi in *CamelCase*. Le *quantity* (quantità) hanno il prefisso “Qto_” e come per i precedenti casi, viene usato il *CamelCase* per le parole successive.

Nel dettaglio lo schema gerarchico ifc definisce quattro livelli (Fig. 19).

- *Domain layer*: è il livello più alto dello schema. Questo livello contiene le entità che racchiudono concetti riguardanti una determinata disciplina. Le entità presenti in questo livello sono autonome e non possono essere *subtype* di nessun'altra entità.
- *Shared layer*: chiamato anche *interoperability level*, in questo livello vi sono le entità che vengono usate per elementi in comune fra diverse costruzioni o per il *facility management*. Queste entità possono avere come riferimento le entità presenti nel *Domain layer*.
- *Core layer*: in questo livello viene definita la struttura di base di un file ifc. Le entità possono avere come riferimento le entità presenti ai livelli superiori dello schema. In questo livello l'ifc viene organizzato secondo una struttura ben precisa principalmente suddivisa in:
 - *Control Extension*: riguarda le entità per il controllo e l'assegnazione degli

oggetti

- *Product Extension*: riguarda le entità che descrivono i prodotti. Possono essere descritti sia i singoli prodotti che le intere categorie.
- *Process Extension*: vi fanno parte le entità che racchiudono le informazioni riguardanti processi, sequenze, pianificazione, e controllo dei lavori.
- *Kernel*: è la parte più astratta del *Core schema*. Racchiude entità che descrivono aspetti generali degli oggetti, proprietà e relazioni fra di essi, sequenza dei processi e raggruppamenti. Le entità presenti nel *Kernel* fanno da collegamento con le entità del *Resource layer*.
- *Resource layer*: questa parte dello schema comprende le entità non radicate, che quindi esistono solo se vi è un collegamento diretto o indiretto con una delle entità presenti nelle parti superiori dello schema. È importante sottolineare che le entità presenti nel *Resource layer* non hanno un'identità specifica quindi possono essere associate a

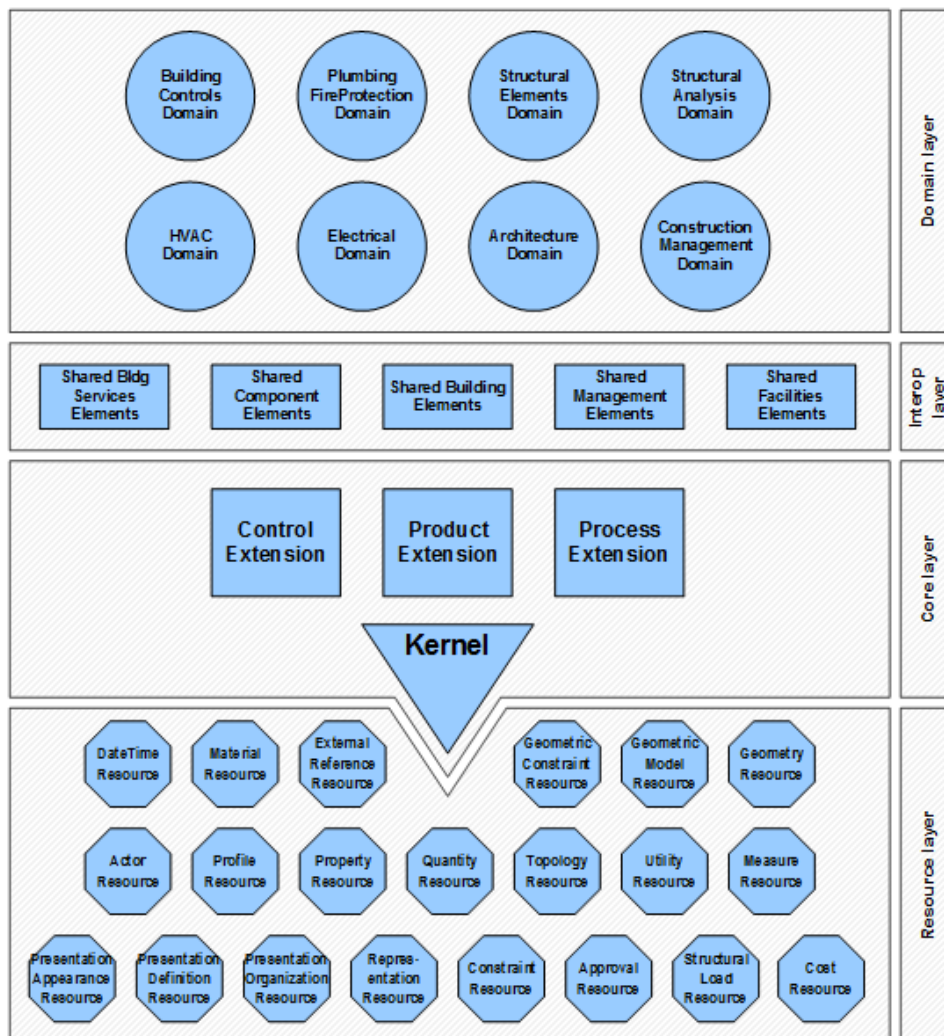


Fig. 19: Schema livelli IFC

https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD2_TC1/HTML/

molteplici entità in base al dato che contengono, ogni qualvolta il loro contenuto sia adatto ed utile ad una ulteriore specifica di un'entità radicata.

Riassumendo possiamo dire che è quindi possibile suddividere le entità presenti in un file ifc in due tipologie distinte, ovvero:

- Entità radicate: sono le entità che hanno come riferimento l'entità *IfcRoot*, hanno una propria identità e attributi come nome e descrizione.
- Entità non radicate: sono le entità che non dispongono di una loro identità ed esistono solo se vi è un riferimento diretto o indiretto, con un'entità radicata.

IfcRoot è definita da buildingSMART come “... the most abstract and root class for all entity definitions that roots in the kernel or in subsequent layers of the IFC specification. It is therefore the common supertype of all IFC entities, beside those defined in an IFC resource schema. All entities that are subtypes of *IfcRoot* can be used independently, whereas resource schema entities, that are not subtypes of *IfcRoot*, are not supposed to be independent entities.” [24]

IfcRoot è quindi la classe principale a cui tutte le entità definite dal *Kernel* o nel *Resource layer* devono fare riferimento. *IfcRoot* è a sua volta diviso in tre parti:

- *IfcObjectDefinition*: riguarda la presenza dei vari tipi di oggetti materiali
- A sua volta diviso in due principali tipologie:
 - *IfcObject*: riguarda i singoli oggetti e racchiude le informazioni che lo costituiscono.
 - *IfcTypeObject*: riguarda un'intera tipologia di oggetti che condividono le stesse informazioni.

Ognuno dei due tipi dispone di informazioni riconducibili a sei tipologie:

- *IfcActor*: riguarda le organizzazioni o le persone.
- *IfcControl*: riguarda le regole sui costi, tempi e ordine delle lavorazioni.
- *IfcGroup*: riguarda interi gruppi di oggetti che condividono la stessa finalità d'uso.
- *IfcProduct*: classe base per tutti gli oggetti fisici e spaziali. I prodotti hanno materiali, forme e descrizione dello spazio che occupano associati a loro. I prodotti possono essere di tipo spaziale come *IfcSite*, *IfcBuilding*, *IfcBuildingStorey*, *IfcSpace*; elementi fisici come *IfcWall*, *IfcDoor*, *IfcStair*.
- *IfcProcess*: riguarda le informazioni legate alle procedure, tempi d'attuazione, eventi particolari. Questa entità contiene informazioni riguardanti tempo e durata di un qualsivoglia intervento o processo, rendendo possibile una

pianificazione programmata, nonché ordine cronologico delle azioni eseguibili solo in un certo ordine prefissato.

- *IfcResource*: riguarda elementi a disponibilità limitata, che essi siano oggetti fisici o capacità funzionali. In questa classe vengono raccolte informazioni riguardanti risorse come materiali, manodopera e attrezzature. Le risorse possono essere assegnate a vari processo e possono essere nidificate fra loro stabilendo quindi un ordine gerarchico.

- *IfcRelationship*: riguarda le relazioni fra gli oggetti

Anche questa tipologia può essere divisa in base al tipo di relazione in:

- *IfcRelDecomposes*: definisce la connessione come parte di un oggetto frammentato, come ad esempio la suddivisione di un edificio in diversi piani e questi ultimi in molteplici ambienti
- *IfcRelAssigns*: definisce una relazione fra un oggetto impiegante ed una attività impiegata sull'oggetto.
- *IfcConnects*: indica la modalità di collegamento fra due o più oggetti
- *IfcRelAssociates*: indica dei collegamenti con elementi esterni all'IFC
- *IfcRelDefines*: indica la relazione su un oggetto o una parte di esso con il suo intorno.

- *IfcPropertyDefinition*: definisce le proprietà applicabili sugli oggetti.

Le proprietà applicate agli oggetti possono essere di diverso tipo quali numeri, stringhe, unità di misura, range di valori, elenchi, tabelle, valori limite o un dato strutturato. Questa tipologia principalmente può essere divisa in:

- *IfcPropertySet*: rappresenta le proprietà associate ad un evento, un oggetto o ad una tipologia di oggetti.
- *IfcPropertySetTemplate*: è stata introdotta dalla versione IFC4, definisce proprietà relative ai tipi di dati

Attualmente l'ultima versione rilasciata è la IFC 4.2, tuttavia quest'ultima si trova ancora in fase di valutazione, quindi l'ultima versione ufficiale rimane ancora la IFC 4.1. Dalla versione IFC4 è stata inserita una classe chiamata *IfcProjectLibrary* con la quale è possibile collegare progetti esterni con cui si condividono o vengono importate determinate informazioni. Il collegamento fra la più anziana classe *IfcProject* e la nuova classe è un chiaro esempio di collegamento indiretto. *IfcProject* comprende infatti una vasta rete di collegamenti diretti ed indiretti essendo una delle classi principali. Essa comprende informazioni di progetto quali nome e descrizione, coordinate, unità di misura e altre informazioni riguardanti il contesto,

infatti ogni altra classe presente nel file deve essere collegata ad essa in modo diretto o indiretto. Un altro esempio di relazioni fra classi è dato nel caso della classe *IfcMaterial*. Questa classe indica un materiale specifico ed ha come riferimenti le proprietà contenute in altre classi opzionali quali *texture*, colori, proprietà termiche o meccaniche per citarne alcune. [26]

1.8 Linguaggi di programmazione e programmazione visuale

I linguaggi di programmazione possono essere intesi come un insieme di regole redatte seguendo uno specifico linguaggio formale. Tramite queste regole è possibile dare istruzioni alla macchina per il conseguimento di una determinata operazione. Essenzialmente ogni istruzione parte da uno o più dati di input i quali vengono processati tramite una funzione o un'insieme di funzioni (espressione) per la produzione di dati di output. Durante la scrittura di tali regole è possibile distinguere:

- Parole chiave: determinate parole appartenenti al linguaggio di programmazione associate ad una istruzione specifica. Ad ogni parola chiave quindi corrisponde un'azione specifica.
- Variabili: parole alla quale viene attribuito un significato dall'utente durante la redazione di un codice.
- Caratteri speciali: simboli con significati particolari. Ad ogni simbolo corrisponde una funzione ben precisa.
- Regole sintattiche: relazione fra le varie parole componenti un codice. Le regole sintattiche devono essere osservate scrupolosamente e variano in ogni linguaggio di programmazione. [29]

Questi concetti riguardano sia i vari linguaggi di programmazione che linguaggi di programmazione visuale. Per linguaggio di programmazione visuale o “*Visual Programming Language*” si intende un tipo di linguaggio che converte il tradizionale codice formato da varie righe di comandi in un codice formato da “blocchi funzionali”. I blocchi funzionali sono degli elementi che contengono una data informazione quale essa sia funzione o dato. Durante la creazione di un codice tramite VPL una rete di blocchi funzionali vengono collegati tra loro tramite un “ponte”, ovvero un elemento che crea collegamenti fra due o più blocchi funzionali. Questa tipologia di codifica per certi versi semplifica la creazione di un codice, per altri rende alcuni aspetti più ostici da affrontare. Fra i pro vi è senz'altro la ridotta e in alcuni casi assente necessità di seguire una sintassi precisa, infatti una volta collegati due blocchi funzionali, se questi ultimi sono collegabili, non è necessario specificare alcun simbolo particolare. Fra i contro invece vi è il reperimento dei blocchi funzionali, infatti per realizzare un'espressione più o meno complessa si dovranno avere una serie di blocchi funzionali precisi. L'assenza di uno di questi blocchi rende impossibile il completamento della funzione. Dal punto di vista grafico, se da una parte un codice in programmazione visuale risulta estremamente semplice da leggere, dall'altra è facile capire come le dimensioni di tale “mappa” aumentino velocemente per codici complessi, aspetto che rende la lettura del

codice non immediata.

1.8.1 VPL e progettazione parametrica in BIM

In questo elaborato verrà approfondito l'utilizzo di Dynamo, un software di programmazione visuale *Open Source* fornito da Autodesk che si interfaccia direttamente con il software BIM Revit. Una volta avviato, le operazioni che si andranno ad effettuare in Dynamo si ripercuoteranno nel modello corrente in Revit. Con questo software è possibile modellare parametricamente oggetti dalle forme complesse tramite funzioni specifiche e organizzare le informazioni contenute nel modello, produrre elaborati quali grafici, abachi e file di vario tipo. Inoltre è possibile automatizzare operazioni ripetitive che, se eseguite manualmente, richiederebbero molto tempo. La programmazione in Dynamo avviene, come negli altri software di programmazione visuale, tramite “nodi”, ovvero dei blocchi informativi contenenti un dato di input, output o una funzione, e dei collegamenti utili per collegare due o più nodi.

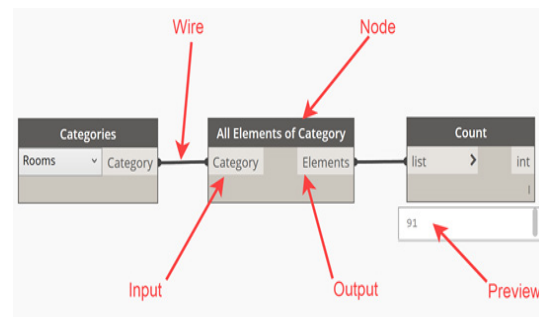


Fig. 20: Esempio nodi e collegamenti Dynamo
<https://archsmarter.com/what-is-dynamo-revit/>

Quando si intende inserire manualmente informazioni in Dynamo occorre utilizzare il linguaggio di programmazione *Python*. Infatti Dynamo basa il suo funzionamento sull'*implementation* di Python chiamata *IronPython*. Attualmente IronPython riconosce i codici realizzati con sintassi e parole chiave appartenenti a Python versione 2.

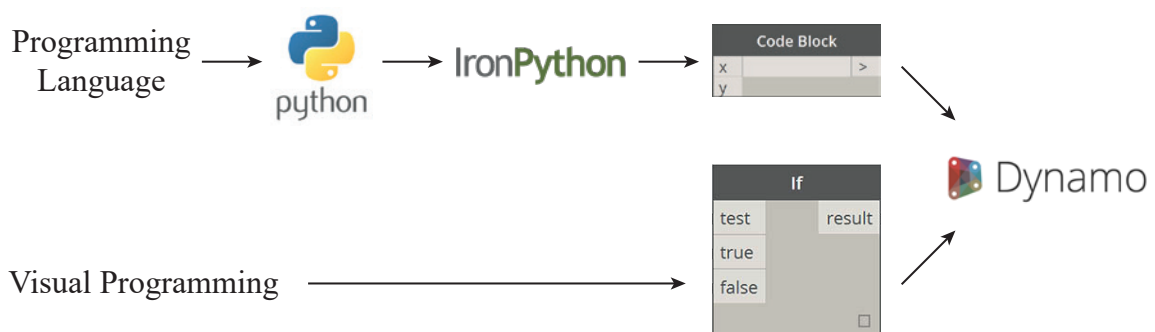


Fig. 21: Programming & VPL in Dynamo

Un elemento estremamente utile in Dynamo è la libreria dei nodi. Quest'ultima è un archivio contenente i nodi supportati da Dynamo che possono quindi essere richiamati facilmente e inseriti nel proprio algoritmo. È inoltre possibile scaricare nodi personalizzati eseguiti da

altri utenti e poi caricati nel sito ufficiale a cui Dynamo è collegato di default. In questo modo è facilmente possibile creare un algoritmo per i propri scopi pur non avendo esperienze avanzate in programmazione. [30]

1.8.2 Linguaggio di programmazione

In questo elaborato verrà approfondito l'utilizzo di Python, linguaggio di programmazione che si sta affermando nel mercato mondiale come uno dei linguaggi più diffusi. Le

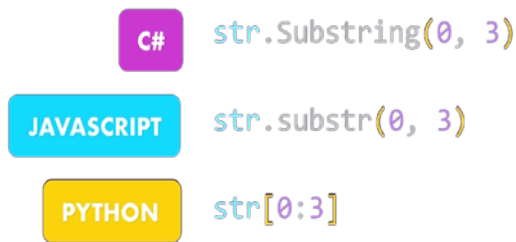


Fig. 22: Confronto linguaggi di programmazione
<https://www.codewithmosh.com>

motivazioni di tale crescita risiedono nella capacità di Python di risolvere problemi complessi in poco tempo per mezzo di codici composti da poche linee di codice. Al momento molte compagnie di spessore hanno scelto di utilizzare Python come linguaggio di programmazione principale per i loro prodotti

per tale ragione.

Un altro punto forte di questo linguaggio è la sua versatilità. Python si presta a lavori legati ad analisi di dati, intelligenza artificiale e *machine learning*, automazione, realizzazione di applicazioni web, *mobile* e desktop inoltre può essere usato per *testing* e *hacking*. Python fa parte dei linguaggi *high-level*, ovvero linguaggi che si avvalgono di codici facilmente comprensibili dall'utente ed è un linguaggio *cross-platform* ovvero può essere usato in Windows, Mac e Linux. Python dispone inoltre di svariati strumenti e librerie nonché ambienti virtuali per condividere informazioni con gli altri utenti utilizzando lo stesso linguaggio.

Attualmente esistono due versioni di Python:

- Python 2: è la versione *Legacy*, ovvero la vecchia versione di Python che non sarà più implementata a partire dal 2020



Fig. 23: Tools Python
<https://www.codewithmosh.com>

- Python 3: è la versione più recente. Gli ultimi e i futuri aggiornamenti di questo linguaggio riguardano prevalentemente questa versione.

Per la scrittura di un codice possono essere usate due tipologie di tool:

- *Editor*: tipologia di software che consente la scrittura in uno o più linguaggi di

programmazione

- IDE: software simile ad un *editor* ma con delle funzioni integrate utili a velocizzare il lavoro dell'operatore, quali autocompletamento del codice, *debugging* e *testing*.

Python inoltre dispone di diverse *implementation* ognuna avente le proprie funzioni ed in alcuni di esse fungono da convertitore per altri linguaggi di programmazione.

Possiamo identificare il linguaggio Python come l'insieme di regole che costituiscono il linguaggio stesso. Le *implementation* sono dei software che comprendono tali informazioni e le eseguono tramite un codice. Quando si installa Python in una macchina utilizzando Windows si sta installando una *implementation* di Python chiamata *CPython*. Quest'ultima è la *implementation* di default per Windows,

scritta in linguaggio C da cui ne trae il prefisso. Attualmente esistono varie *implementation* scritte in linguaggi differenti.

Ogni aggiornamento del linguaggio di programmazione viene inizialmente recepito da *CPython* (in quanto *implementation* di default) e gradualmente viene recepito dalle altre *implementation*. Queste ultime, oltre alle funzioni comuni, hanno funzioni specifiche, differenti o esclusive in alcuni casi. [31]



Fig. 24: Implementation Python
<https://www.codewithmosh.com>

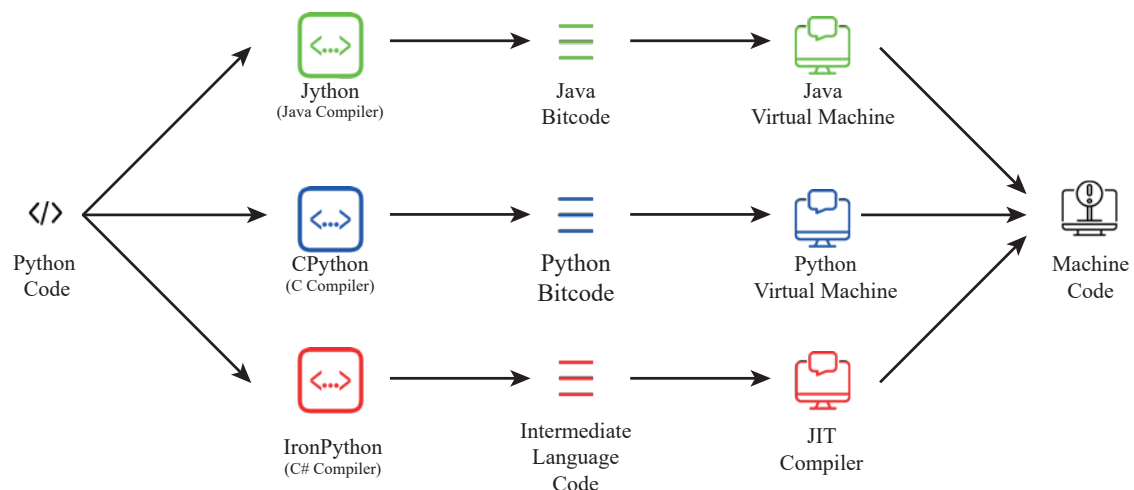


Fig. 25: Esempio flusso codici da linguaggio Python a machine code (Windows)

2 Metodologia

2.1 Caso Studio

2.1.1 Inquadramento caso studio

Il caso studio in esame è un edificio parte dello stabilimento industriale FCA sito nel quartiere Mirafiori di Torino. L'intero complesso industriale si estende per circa 550000 m² ed al suo interno vengono attualmente svolte attività quali lastratura, verniciatura e montaggio, inoltre vi sono locali quali ex magazzini ed ex fucine. Lo stabilimento venne inaugurato nel 1939 e nel 1956 fu ampliato con una nuova serie di fabbricati. A causa degli ingenti danni riportati durante la seconda guerra mondiale furono effettuati interventi di ristrutturazione sulle strutture interessate. Ad oggi è stato impossibile effettuare un sopralluogo, quindi risulta difficile differenziare con esattezza le parti originali del 1939 da quelle ricostruite nel dopoguerra.

L'edificio in esame riguarda una parte dell'intero stabilimento e più precisamente si tratta della parte adibita alla lastratura.

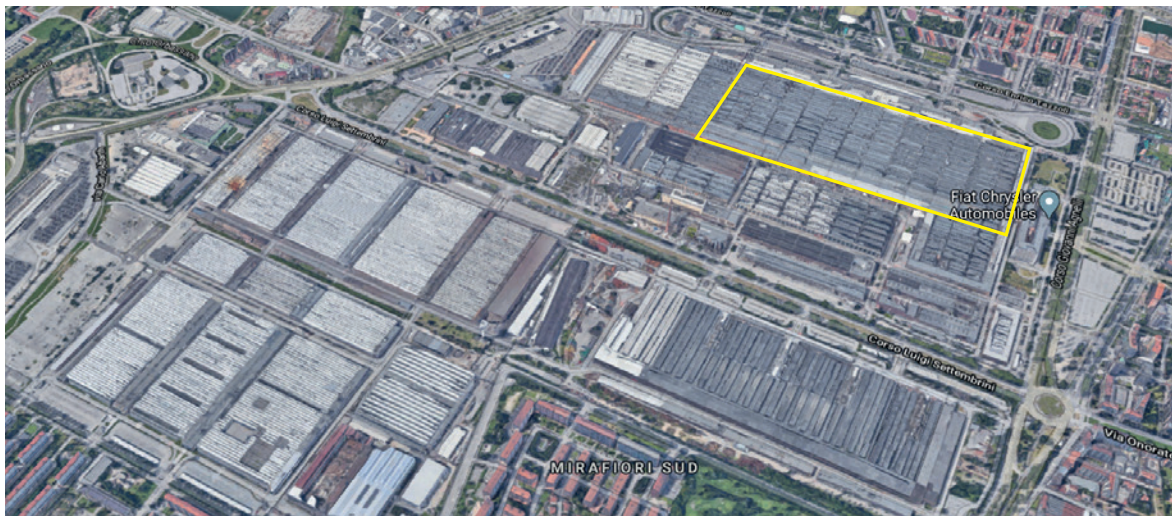


Fig. 26: Vista complesso industriale FCA Mirafiori

<https://www.google.com/maps>

Il materiale informativo riguardante l'intero stabilimento è difficile da reperire per motivi legati alla privacy aziendale. Gli elaborati forniti da FCA riguardano le planimetrie e gli impianti di climatizzazione dell'edificio lastratura. [32]

2.1.2 Modello BIM caso studio

Il modello utilizzato per lo svolgimento della presente valutazione è stato realizzato da Rodolfo Rotti durante la sua tesi di laurea magistrale nel corso di laurea in Ingegneria Energetica e Nucleare.

Come accennato in precedenza, il modello riguarda una parte dell'intero stabilimento FCA di Mirafiori. Il modello BIM è stato modellato sulla base di due file AutoCAD riguardanti la planimetria e la schematizzazione dell'impianto di climatizzazione del singolo edificio. In base i dati forniti, la modellazione dell'impianto è stata semplificata e non risulta essere conforme all'edificio reale. [32]

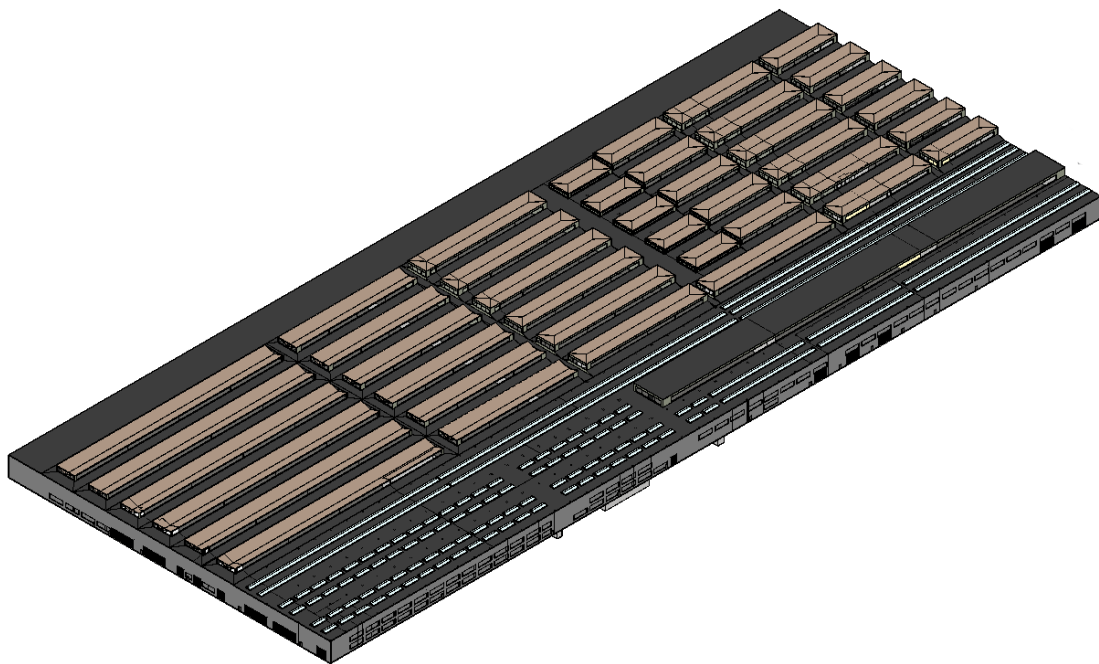


Fig. 27: Modello BIM del caso studio
Modello a cura di Rodolfo Rotti

2.1.3 Output caso studio: gbXML e IFC

Sulla base del lavoro svolto sono stati esportati due file rispettivamente in formato .gbXML e .ifc. In una prima fase riguardante la valutazione preliminare, sono state mantenute le impostazioni di base già salvate nel modello BIM. In questo modo si è cercato di ridurre al minimo le modifiche effettuate a quest'ultimo cercando comunque di estrarre le informazioni necessarie per le valutazioni da effettuare. In una seconda fase si è proceduto alla personalizzazione delle impostazioni cercando di ottimizzare il più possibile la qualità dei dati di output allo scopo di esportare un file che si prestasse adeguatamente alla valutazione

energetica da parte dei software BEM.

Il file .gxbml è stato esportato utilizzando le impostazioni mostrate in Fig. 28. Inoltre è possibile notare come le zone termiche siano state già definite e riportino le proprietà adeguate alla valutazione da effettuare.

Il file .ifc è stato realizzato utilizzando un'impostazione di Revit che consente di esportare il maggior numero possibile di proprietà. In questa fase il file non è infatti stato personalizzato al fine di valutare la quantità di informazioni esportate dalla piattaforma BIM.

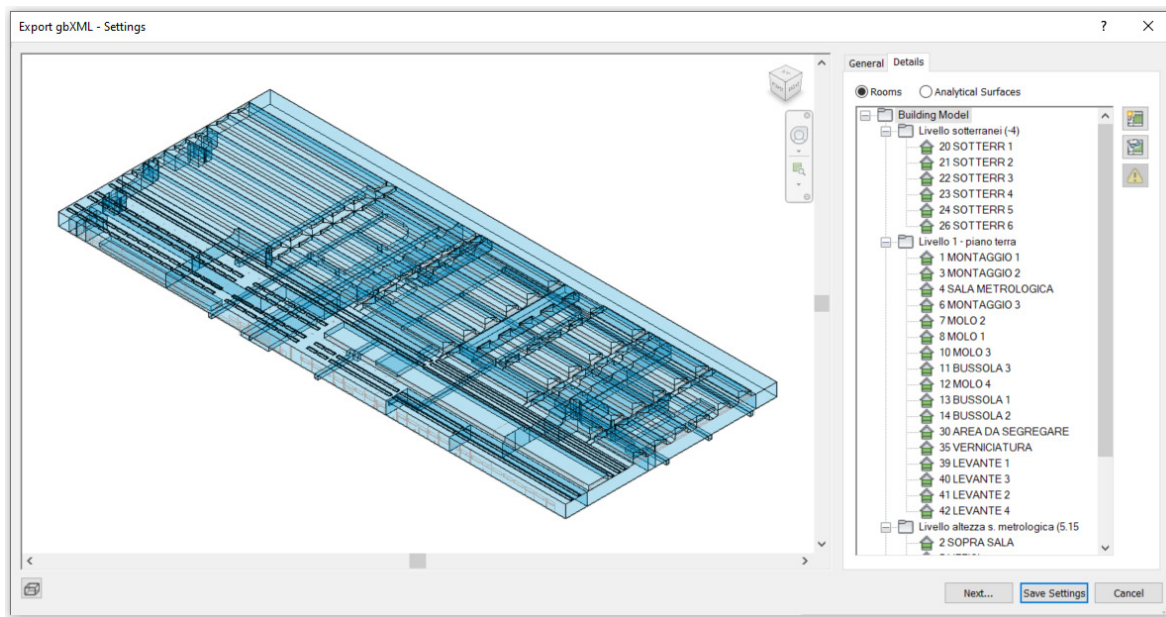


Fig. 28: Esportazione caso studio in formato gbXML

Modello a cura di Rodolfo Rotti

2.2 Workflow BIM to BEM

La valutazione è stata suddivisa per step in modo da valutare a grandi linee un modesto numero di software, quindi procedere in modo sempre più restrittivo al fine di individuare un software con cui saranno effettuare le prove di interoperabilità. Vi è da specificare che la valutazione del processo di interoperabilità richiede il controllo sia della lettura dei dati in entrata da parte dei software BEM, che il controllo della corretta scrittura delle informazioni da parte dei software BIM. Per quanto riguarda il settore energia, l'interoperabilità fra software avviene principalmente in due modi:

- Attraverso *plugin* specifici che dovrebbero garantire un'interoperabilità ottimale fra due o più software, avvalendosi o meno di un formato aperto non proprietario.
- Attraverso formati di scambio aperti non proprietari: al momento i formati principalmente usati sono .gbXML e .ifc

Attualmente esistono centinaia di software che svolgono analisi e simulazioni nei settori riguardanti le analisi energetiche, acustiche, illuminotecniche o svolgono test di idoneità alle verifiche di legge. In questo scenario sono stati selezionati alcuni fra i software disponibili, scelti in base alle esigenze del caso studio ed alla tipologia di simulazione da effettuare. I software infatti spesso tengono conto di variabili specifiche dettate dalle esigenze della *software house* o del territorio in cui saranno applicati. In primo luogo sono stati quindi scartati i software che svolgono compiti specifici e difficilmente applicabili a casi generali. I software selezionati saranno valutati per step fino alla scelta di un solo software per il *workflow* BIM to BEM legato al presente caso studio. L'analisi è divisa principalmente in due parti. La prima parte dei test riguarderà essenzialmente le potenzialità dei software legati alla capacità di importare ed esportare file nei formati più utilizzati attualmente, quali .gbXML e .ifc. Il secondo step valuterà il grado di interoperabilità legato alla qualità dei dati letti ed alla risposta agli input dell'utente da parte del software una volta importati i dati in entrambi i formati di scambio utilizzati.

2.2.1 Valutazione 1: Predisposizione all'importazione dei dati

Il questo primo step sono stati passati in rassegna i software selezionati. Viene riportata quindi la loro predisposizione all'interoperabilità nonché eventuali note di merito o demerito rilevate. A seguire uno schema dei risultati riscontrati (Fig. 29) e note aggiuntive riguardanti peculiarità rilevate nei singoli software.

Valutazione 1: Predisposizione all'importazione dei dati

Software	Software House	Motore di Calcolo / Tipologia Software	gbXML		IFC	
			Import	Export	Import	Export
Revit	Autodesk	BIM	✓	✓	✓	✓
Archicad	Graphisoft	BIM	✓	✓	✓	✓
Ecodesigner Star	Graphisoft	VIPCORE Calculation Engine	✓	✓	✗	✗
Green Building Studio	Autodesk	Saas Autodesk	✓	✗	✗	✗
Insight	Autodesk	Saas Autodesk	✓	✓	✗	✗
Open Studio	Alliance for Sustainable Engine	Energy Plus; Radiance	✓	✓	✓	✗
Design Builder	Design Builder Software	Energy Plus	✓	✓	✗	✗
Simergy	Digital Alchemy	Energy Plus	✓	✗	✓	✓
Energy Plus	USDOE	Engine autonomo open source	✓	✗	✗	✗
EdilClima 700	EdilClima S.r.l.	Proprietario	✗	✗	✓	✗
Termolog EpiX 10	Logical Soft S.r.l.	Proprietario	✗	✗	✓	✓
Termus	Acca Software S.p.a.	Proprietario; Energy Plus	✗	✗	✓	✓
Cypertherm	Cype Ingenieros S.A.	Proprietario	✓	✗	✓	✗
Blumatica Energy	Blumatica	Proprietario	✗	✗	✗	✗
Docet	ENEA e ITC-CNR	Proprietario	✗	✗	✗	✗
IDA ICE	Equa	Proprietario	✗	✗	✓	✗
Termo	Namirial; Università delle marche	Proprietario	✗	✗	✗	✗

✓ Consentito

✗ Non consentito

✓ Necessita un software specifico

Fig. 29: Schema valutazione 1 software BEM

Valutazione 1: Predisposizione all'importazione dei dati

Note sui singoli software:

- Revit: software BIM fra i più diffusi nonché software utilizzato per la modellazione del caso studio.
- Green Building Studio: SaaS di Autodesk. Una volta caricato il file gbXML sul relativo sito internet, il servizio elabora i dati e genera delle possibili soluzioni. È quindi possibile modificare i singoli elementi in base alle proprie preferenze ed esigenze o sfruttare una fra le soluzioni proposte dall'algoritmo del servizio.
- Insight: SaaS di Autodesk. Servizio che riporta grafici e dati di facile lettura, è possibile valutare facilmente e congiuntamente performance energetiche e valutazioni legate all'esposizione solare.
- Archicad: software BIM fra i più diffusi. Capacità notevole di importazione ed esportazione dei file in formato .ifc. Grazie ad un modulo integrato (EcoDesigner Star) raggiunge la piena interoperabilità per il formato gbXML.
- Ecodesigner Star: modulo integrato di Archicad. Basandosi sul modello corrente sul software BIM, crea un modello energetico per le valutazioni e simulazioni del caso. Il modello è esportabile in formato gbXML, inoltre le modifiche effettuate al modello energetico vengono applicate in automatico al modello BIM.
- Open Studio: GUI di Energy Plus, nonché primo software a supportare tale motore di calcolo. L'importazione dei file in formati ifc non è aggiornata e si hanno difficoltà nelle operazioni. Buona l'importazione dei file in formato gbxml, tuttavia la macchina rallenta bruscamente per modelli ricchi di informazioni.
- Design Builder: GUI di Energy Plus. È possibile importare file in formato gbXML e valutarli, interessante la presenza di *shortcut* verso CasaClima e Energy Plus e Revit. Fra le GUI di Energy Plus sembra essere la più performante in termini di sfruttamento della macchina.
- Simergy: GUI di Energy Plus. Basa le proprie analisi sul modello energetico ma è possibile importare file in formati ifc. Una volta importato il file ifc, il software ne ricava il modello energetico.
- Energy Plus: Le simulazioni avvengono attraverso compilazione tabellare. I risultati sono molto dettagliati tuttavia non vi è alcun metodo input grafico. Essendo un motore di calcolo ampiamente studiato i risultati ottenuti sono molto precisi, tuttavia occorre compilare minuziosamente ogni variabile richiesta. La maggiore pecca di questo motore di calcolo è l'interoperabilità, praticamente nulla per i formati gbXML e ifc.
- EdilClima 700: svolge le simulazioni basandosi su modellazione o importazione IFC.

Valutazione 1: Predisposizione all'importazione dei dati

Interessante la presenza di uno *shortcut* per il collegamento diretto con Revit.

- Termolog EpiX 10: software italiano che svolge le simulazioni basandosi su modellazione grafica o metodo tabellare, è inoltre possibile importare file in formato ifc.
- Termus: svolge le simulazioni basandosi su modellazione o importazione ifc. Presenta un'interfaccia intuitiva ed una funzione molto utile riguardante l'individuazione automatica dei ponti termici, tuttavia l'affidabilità di quest'ultima funzione non è stata valutata nel presente elaborato.
- Cypertherm: la stessa *software house* dispone di un altro software basato su EP, tuttavia il grado di interoperabilità, sia in formato gbXML che ifc è limitata.
- Blumatica energy: Interessante una funzione che suggerisce in automatico le modifiche da apportare al progetto per rispettare i requisiti normativi, purtroppo l'importazione dei dati è possibile solo con formati tecnici quali dwg e dxf o immagini png e pdf.
- Docet: strumento di calcolo molto semplice basato su foglio di calcolo. Rispetta i requisiti di legge tuttavia vi è interoperabilità nulla per tutti i formati.
- IDA ICE: basa l'importazione su formato ifc. Interessante la funzione che permette di consultare la mappa ifc durante l'importazione con la possibilità di importare solo le classi ritenute utili ai fini del calcolo. In questo modo si può alleggerire il modello importato rendendo più rapide le operazioni all'interno del software.
- Termo: semplice software di analisi. Purtroppo il grado di interoperabilità è limitato ai formati dwg, dxf, bitmap e jpeg.

2.2.2 Valutazione 2: Grado di interoperabilità formati di scambio

Come già anticipato, è di fondamentale importanza utilizzare strumenti che garantiscano il mantenimento e la corretta lettura dei dati contenuti nel modello originale. Su questo principio sono state fatte delle prove di interoperabilità, ovvero test riguardanti importazione ed esportazione dei dati. I test seguono uno schema logico utile tracciare un percorso dei dati ottimale per ogni software. Considerando che il modello BIM a disposizione è stato modellato sul software BIM Revit di Autodesk, quest'ultimo sarà ritenuto il punto iniziale dello schema logico. È stato inoltre modellato un modello semplificato chiamato *BestTest* utile alla valutazione delle potenzialità dei software con il quale viene comparato il funzionamento di un modello complesso (caso studio FCA) con un modello di facile lettura.

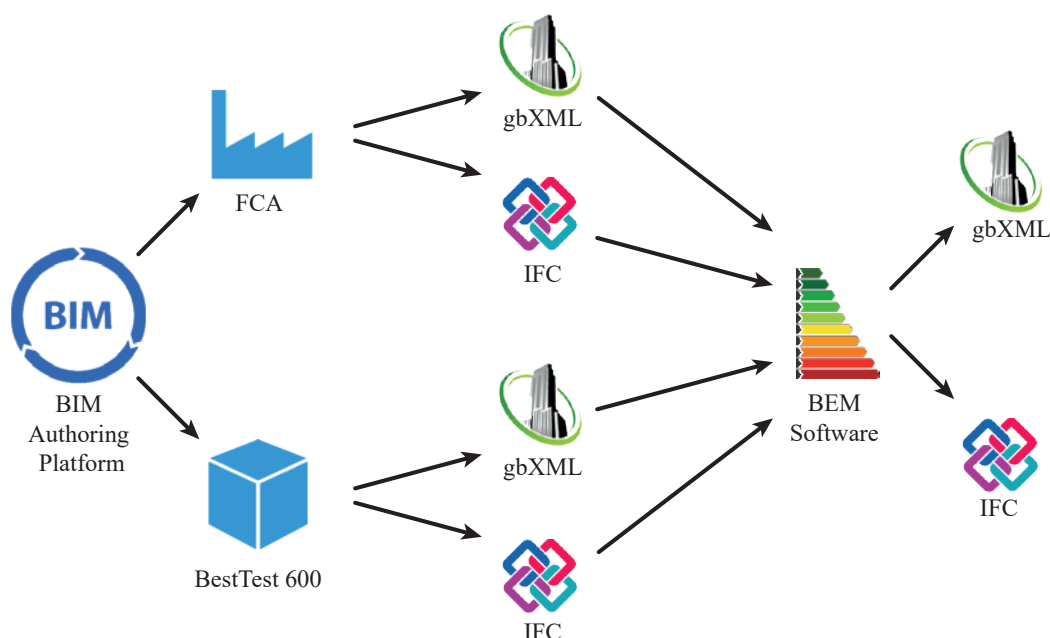


Fig. 30: Schema flusso dati da BIM a BEM

Il modello semplificato è costituito da un ambiente di piccole dimensioni, all'interno non vi sono tramezzi e vi è un ambiente unico. L'involucro è costituito da quattro mura costituite da una stratigrafia personalizzata dall'utente e modellata nel software BIM Revit. Tre muri esterni non dispongono di aperture mentre su un unico muro vi sono due aperture finestrate. L'ambiente è geolocalizzato in posizione simile rispetto al caso studio e la finalità d'uso è stata impostata come ambiente per uffici.

Ulteriori valutazioni sono state effettuate su software aventi un livello di interoperabilità adeguato. Per ogni software sono state fatte delle prove di importazione con i formati gbXML e ifc rispettivamente per i file del modello *BestTest* come modello semplificato e del modello FCA come modello complesso. Si riporta uno schema esplicativo dei risultati ottenuti e commenti relativi alle operazioni effettuate.

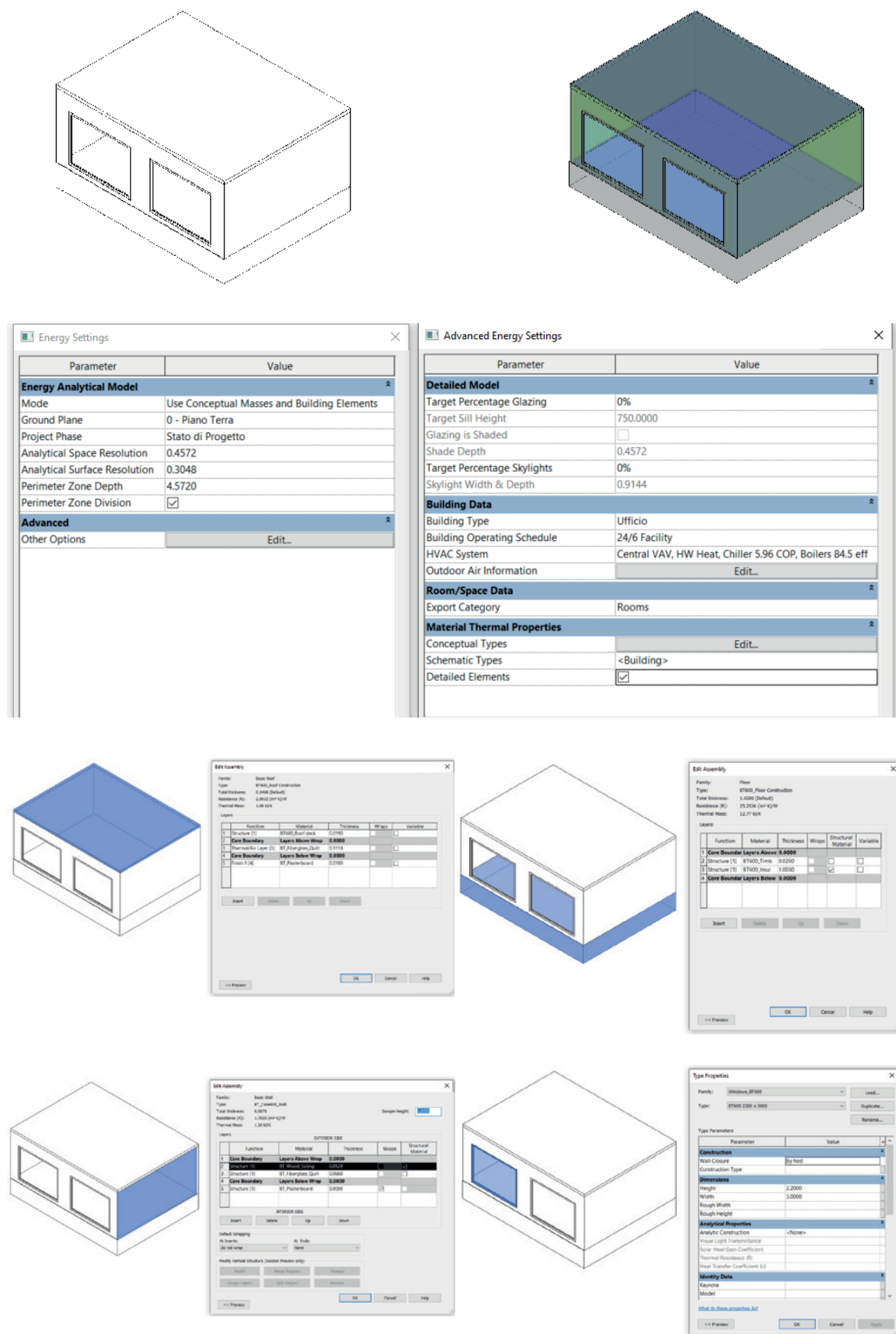


Fig. 31: Modello digitale BestTest 600

Valutazione 2: Grado di interoperabilità formati di scambio

Green Building Studio

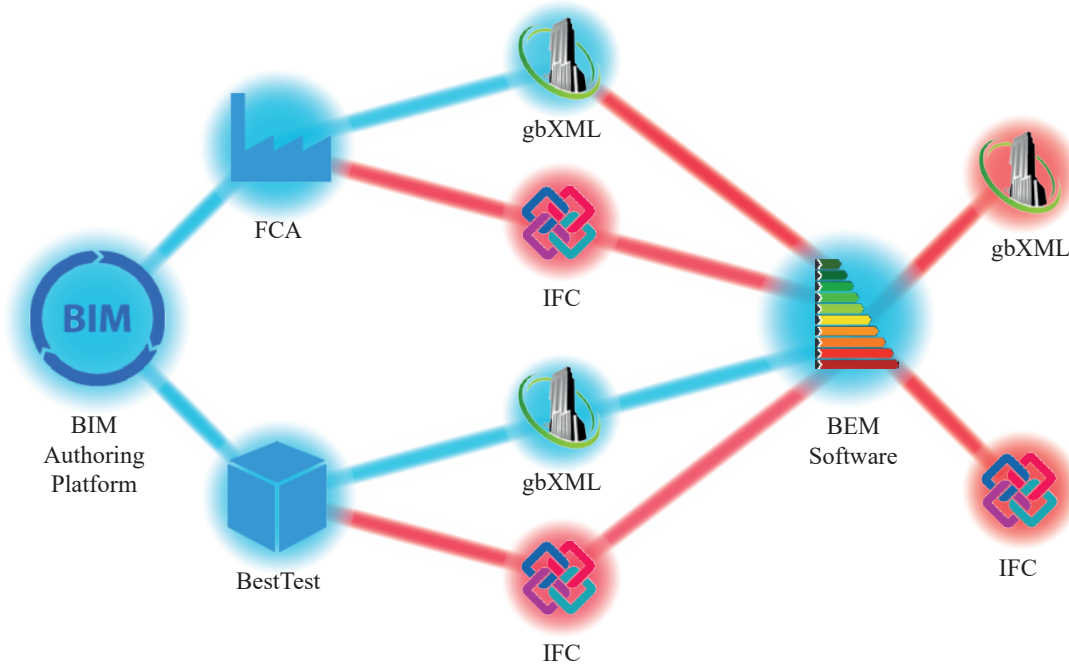


Fig. 33: Schema flusso dati Green Building Studio

Green Building Studio è un SaaS di Autodesk. Per procedere con la valutazione occorre effettuare il login sul sito del software e creare un nuovo progetto, durante la creazione è necessario impostare la geolocalizzazione di quest'ultimo. (Fig. 32)

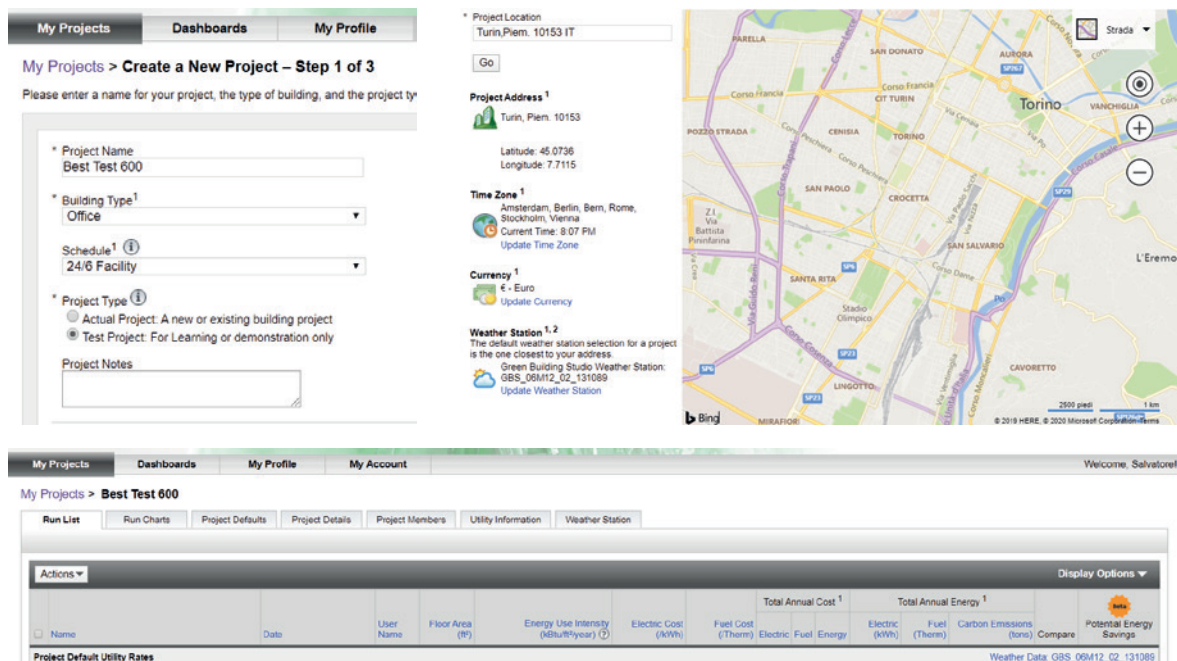


Fig. 32: Creazione nuovo progetto Green Building Studio

Una volta creato il nuovo progetto è possibile caricare il file gbXML

Valutazione 2: Grado di interoperabilità formati di scambio

Il primo test è stato effettuato utilizzando il modello BestTest. Ad upload ultimato è stato possibile visualizzare informazioni del modello quali proprietà degli elementi che definiscono l'involucro edilizio, profili d'uso e consumi in termini di energia o economici.



Fig. 35: Risultati valutazioni BestTest 600 (Green Building Studio)

Per quanto riguarda il modello FCA non è invece stato possibile effettuare alcuna valutazione poiché il software non riconosce il file gbXML fornito riportando un messaggio di errore che non permette di proseguire. La valutazione è stata quindi interrotta.



Fig. 34: Upload gbXML (FCA) in Green Building Studio

Ecodesigner Star

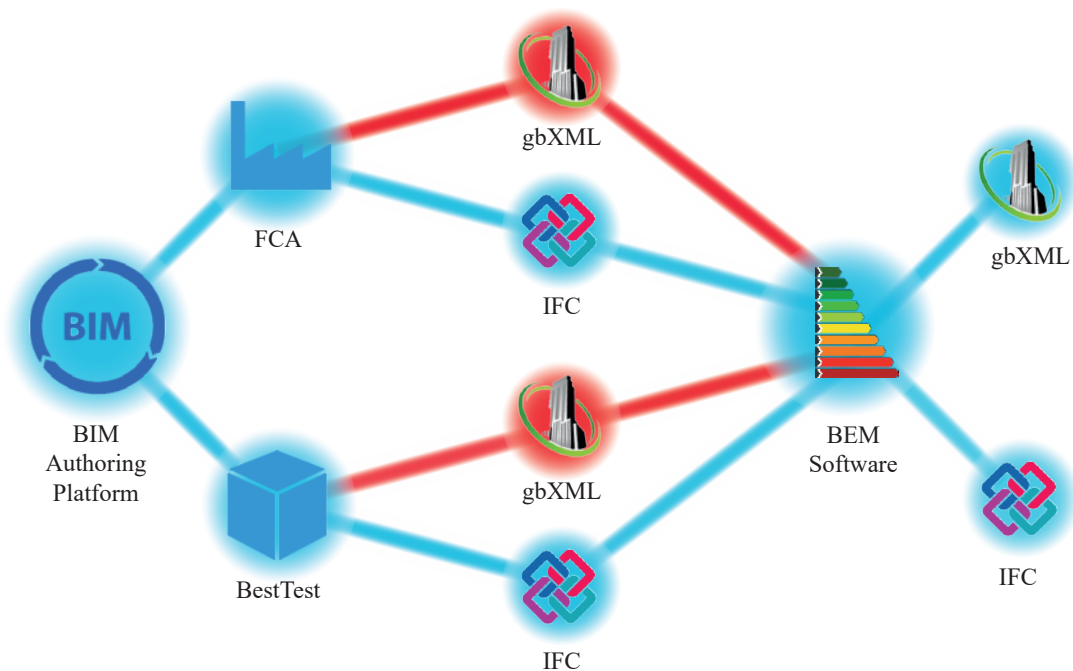


Fig. 36: Schema flusso dati Ecodesigner Star

Ecodesigner star è un software integrato in Archicad che consente di effettuare valutazioni utilizzando il modello BIM corrente come base per la realizzazione del modello energetico utile per le valutazioni. Il primo passo è stato quindi quello di utilizzare il file ifc esportato da Revit ed importarlo in Archicad. In questa fase sono state fatte diverse prove e modifiche per la creazione del file ifc. Nello specifico si è esportato un primo file utilizzando le impostazioni di Revit (utilizzando il *plugin* AutodeskIFC2018, fondamentale per l'esportazione dei file ifc), ed un secondo file utilizzando un *plugin* di Graphisoft (ARCHICAD_Connection_For_Revit_2018) che crea un collegamento diretto fra Archicad e Revit, quindi è possibile esportare da Revit un file ifc ottimizzato per la lettura in Archicad. Per la prosecuzione dei test si è optato per il secondo file, poiché a differenza del primo test, è stato possibile mantenere con facilità la corrispondenza fra i materiali costituenti gli elementi costruttivi e questi ultimi, tuttavia utilizzando questo *plugin* è possibile creare un file ifc esclusivamente in schema IFC2x3, che attualmente non rappresenta l'ultimo schema rilasciato da BuildingSMART.

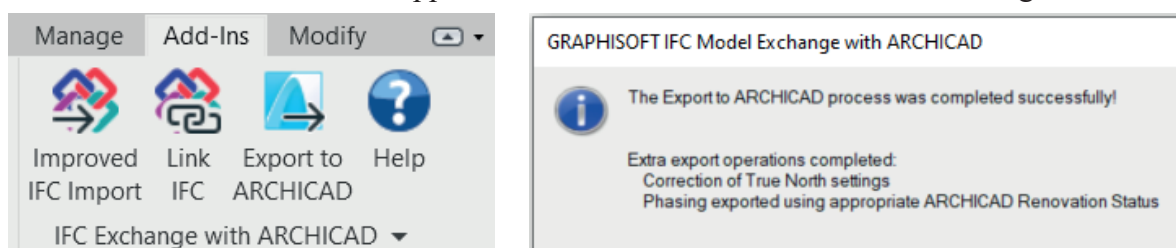


Fig. 37: Plugin Archicad - Revit: Interfaccia e esportazione

Durante il primo test riguardante il modello BestTest non si sono riscontrati particolari problemi. Il file è stato letto correttamente e le geometrie del modello risultano fedeli all'originale. Analizzando i singoli elementi è possibile constatare come le proprietà assegnate ed essi siano presenti, tuttavia sono assenti le proprietà termiche dei materiali (problema riguardante la fase di esportazione di Revit come approfondito nel capitolo 2.3.1).

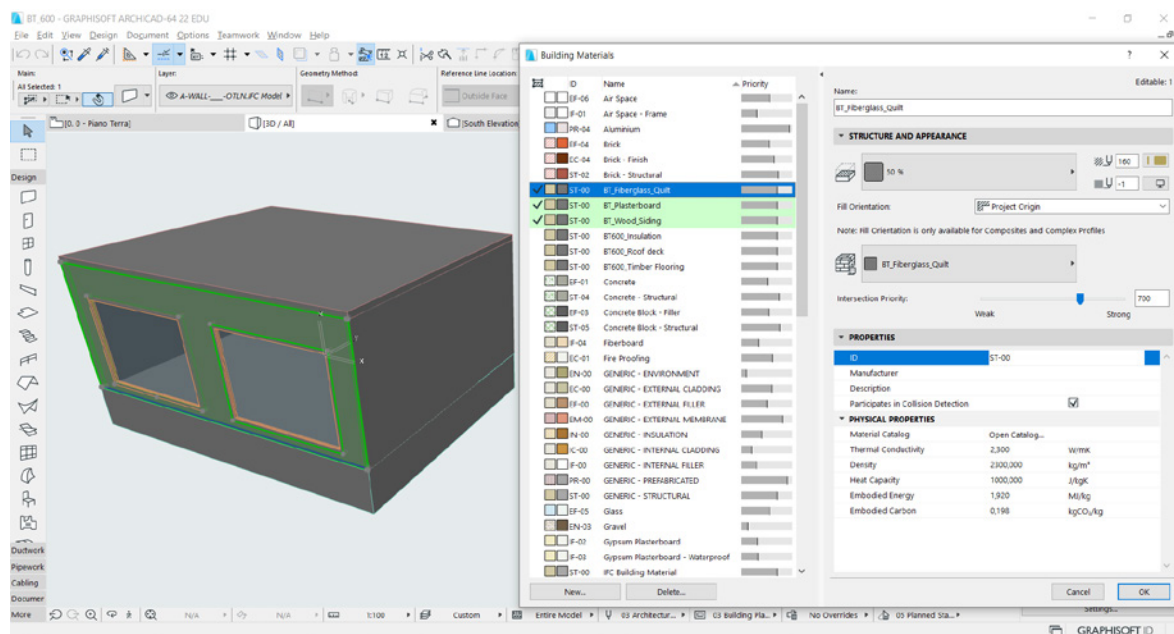


Fig. 38: Importazione BestTest600 in Archicad

Per accedere ad Ecodesigner Star occorre accedere alla finestra *Design* di Archicad quindi selezionare *Energy Evaluation*. Una volta aperto il software EcoDesigner Star i locali presenti nel modello BIM dovranno essere associati a delle zone termiche create esclusivamente

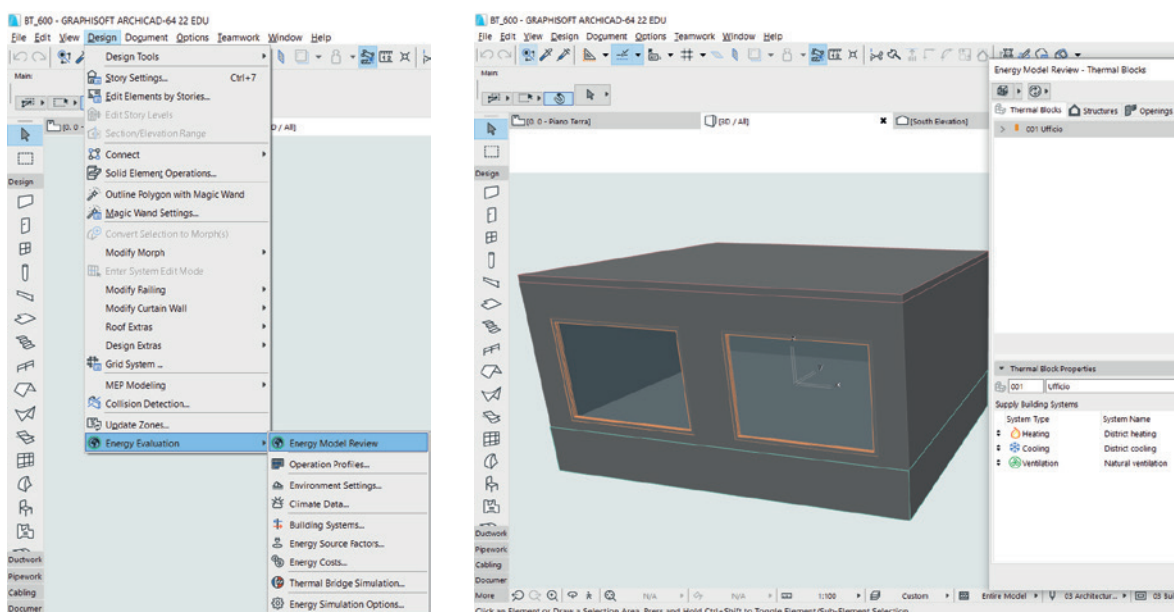


Fig. 39: Finestra Ecodesigner Star

Valutazione 2: Grado di interoperabilità formati di scambio

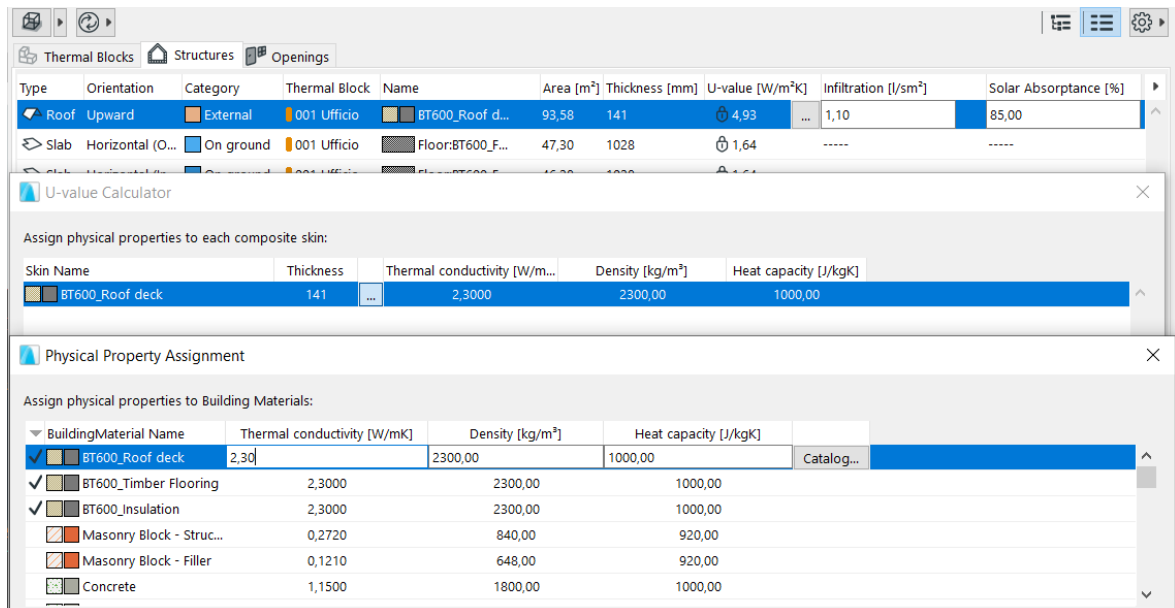


Fig. 40: Stratigrafie e proprietà Ecodesigner Star

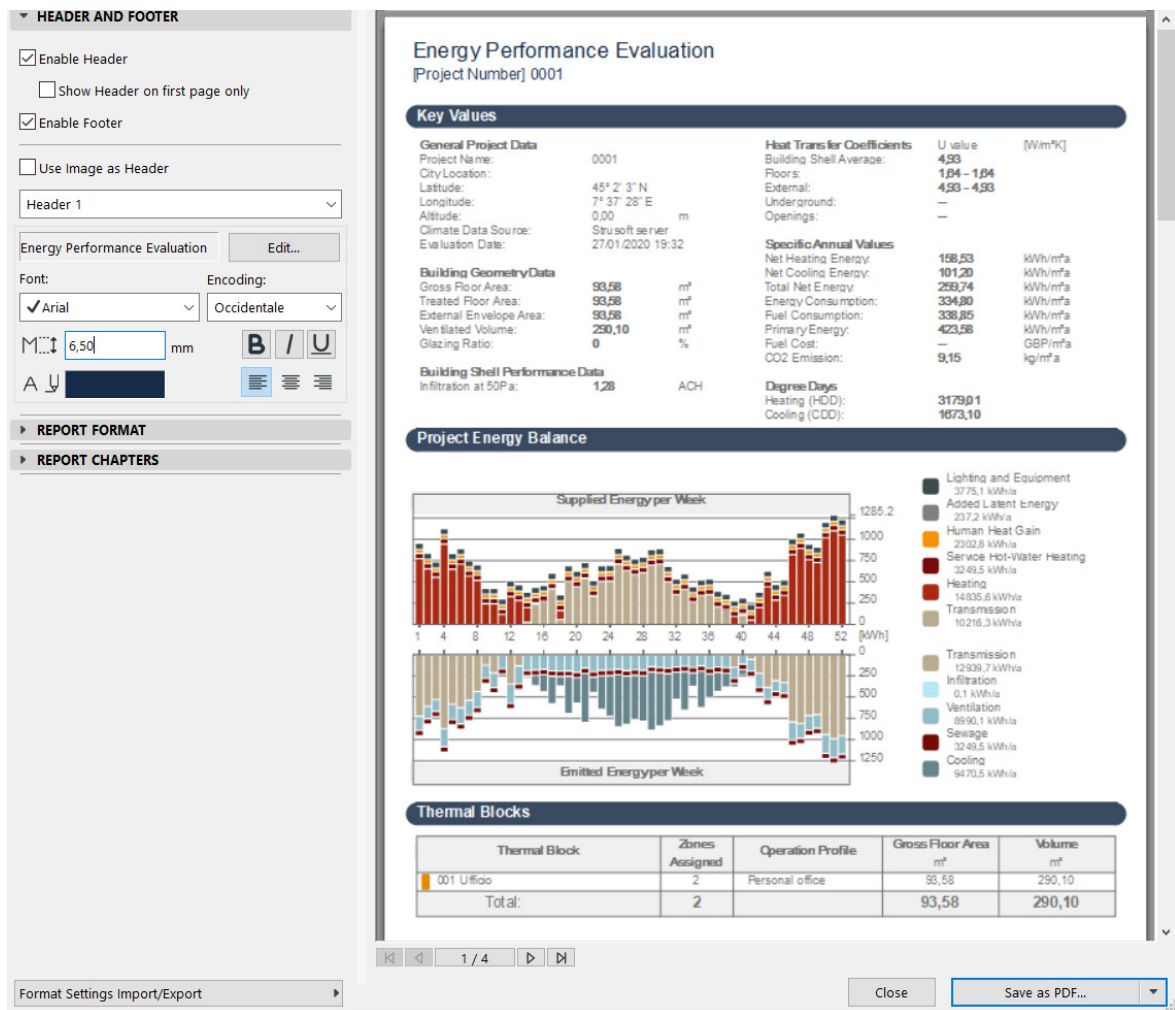


Fig. 41: Risultati valutazione BestTest Ecodesigner Star

Valutazione 2: Grado di interoperabilità formati di scambio

sul software corrente, tuttavia le proprietà del modello BIM legate agli ambienti vengono riconosciute in automatico. In fase di creazione delle zone termiche è possibile specificare la tipologia di impianti presenti e la destinazione d'uso dei locali (Fig. 39). Prima di effettuare la simulazione è possibile visualizzare le strutture che definiscono l'involucro dei locali e le relative proprietà termiche, editabili in base ad eventuali modifiche (Fig. 40). Una volta lanciata la simulazione viene restituito un documento PDF contenente grafici ed informazioni numeriche tra cui i consumi di energia primaria suddiviso per i relativi settori di utilizzo. È possibile inoltre esportare un file gbXML contenente le informazioni del modello energetico appena realizzato (Fig. 41).

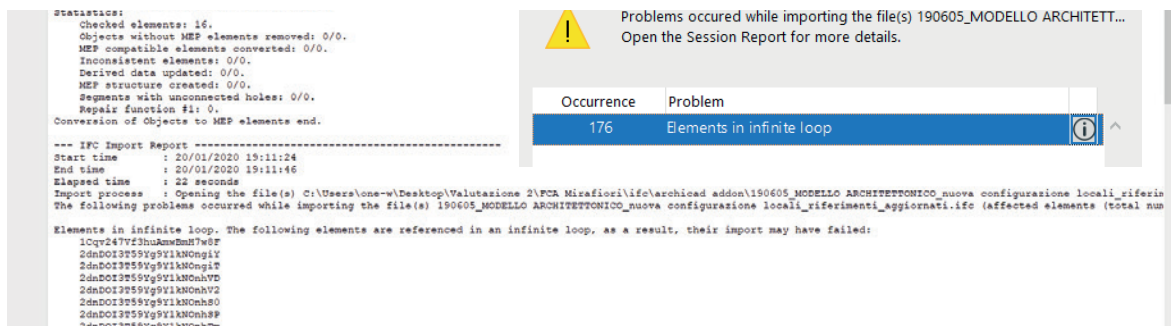


Fig. 42: Problema importazione modello FCA in Ecodesigner Star

Per quanto riguarda il test del modello FCA l'importazione del file ha riportato un messaggio di errore con la conseguente eliminazione degli elementi "non funzionanti".(Fig. 42)

Il test prosegue con la valutazione del modello come effettuato precedentemente per il test del modello BestTest, infatti nonostante alcuni elementi siano stati eliminati, l'involucro risulta ancora integro ed utilizzabile nel test. (Fig. 43)

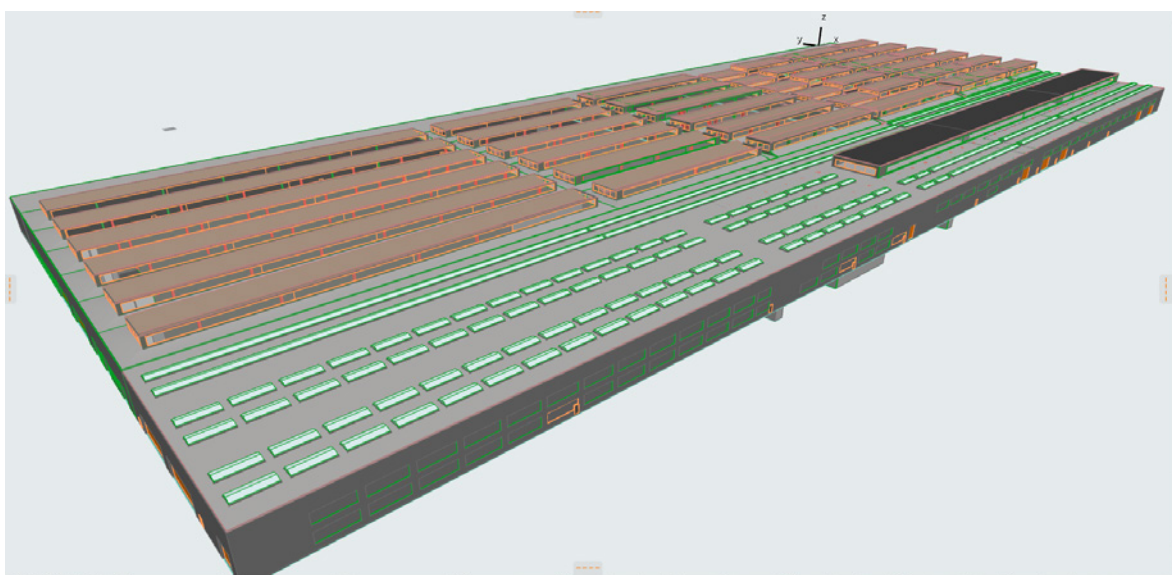


Fig. 43: Modello FCA importato in Archicad

Valutazione 2: Grado di interoperabilità formati di scambio

Come effettuato precedentemente è necessario creare dei nuovi blocchi termici, (o zone termiche) che conterranno gli ambienti costituenti il modello (Fig. 44). Una volta ultimato questo passaggio è possibile lanciare la simulazione come fatto in precedenza. (Fig. 45)

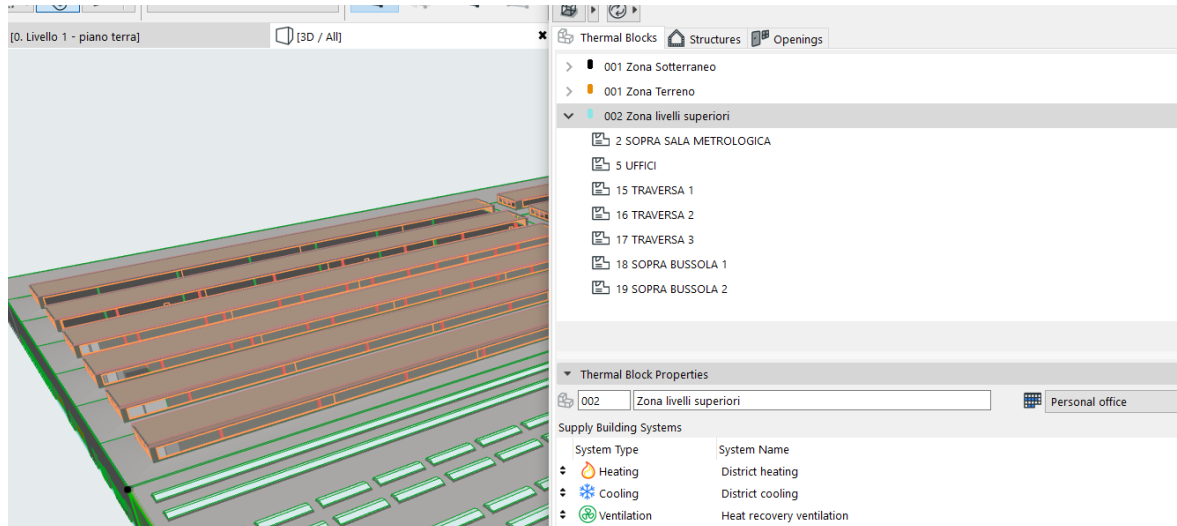


Fig. 44: Zone termiche modello FCA in Ecodesigner Star

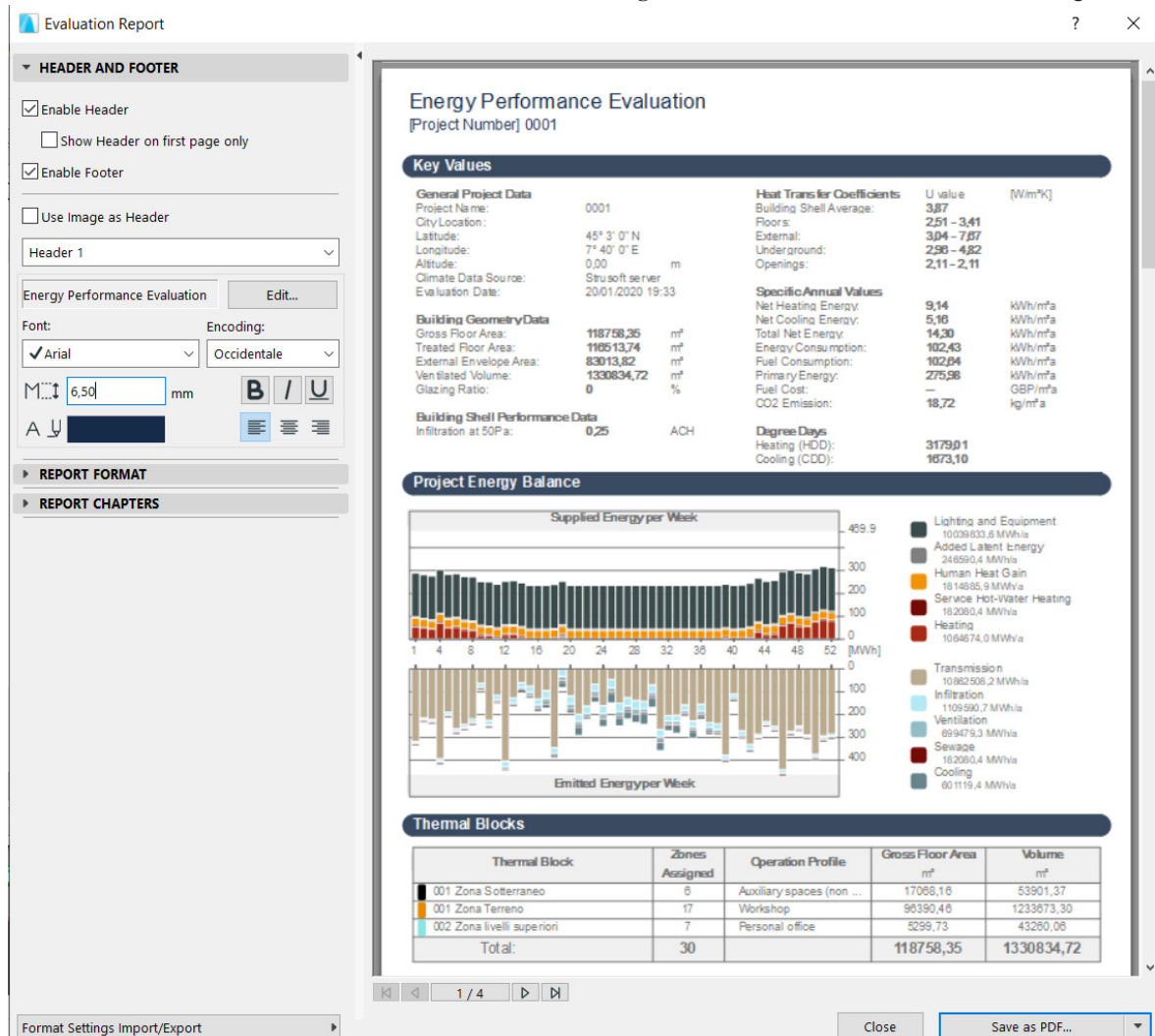


Fig. 45: Risultati valutazioni modello FCA in Ecodesigner Star

EdilClima

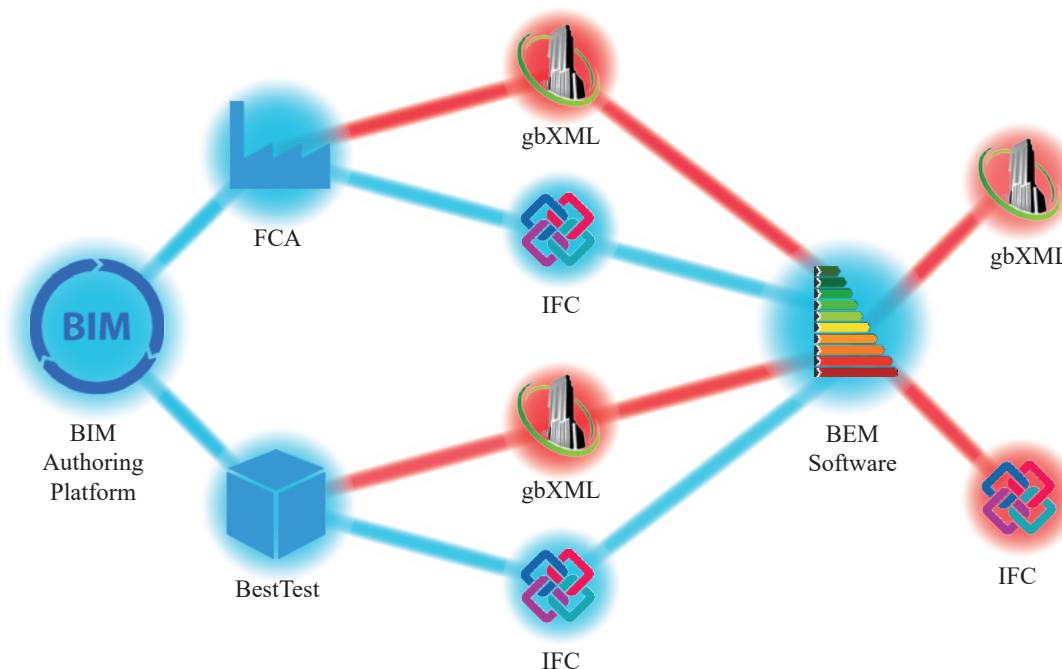


Fig. 46: Schema flusso dati EdilClima

Edilclima EC700 è uno dei software per il calcolo delle prestazioni energetiche degli edifici approvato dal CTI. Il software esegue calcoli in conformità alla UNI/TS 11300, inoltre consente di effettuare il calcolo dinamico orario in conformità alla nuova UNI EN ISO 52016-1:2018. Il software è costituito da una struttura modulare che prevede come base il modulo del motore di calcolo, a cui vengono aggiunti diversi moduli riguardanti specifiche funzioni e condizioni.

Per effettuare le analisi è possibile scegliere il tipo di calcolo da effettuare e la normativa di riferimento (Fig. 47).

L'interfaccia mostra diverse sezioni di configurazione:

- Verifiche di legge e relazione tecnica:**
 - ☒ secondo D.Intem. 26.06.15
 - ☒ Abilita verifiche secondo DLgs 03.03.2011, n. 28
 - Data titolo edilizio: dal 1 gennaio 2018
 - ☐ secondo CAM - Criteri Ambientali Minimi (DM 11.10.2017)
- Attestati energetici:**
 - secondo D.Intem. 26.06.15
 - ☐ Edificio privo di impianto riscaldamento
 - ☐ Edificio privo di impianto ACS
- Opzioni lavoro:**
 - Ponti termici: Calcolo analitico
 - Resistenze linari: Appendice A UNI EN ISO 6946
 - Serre e locali non climatizzati: Calcolo semplificato
 - Capacità termica: Calcolo semplificato
 - Definizione ombreggiamenti: Calcolo automatico
 - Radiazione solare: Calcolo con angolo di Azimut
- Opzioni normative:**
 - Rendimento globale medio stagionale: DM 26.06.15 ed UNI/TS 11300 (calcolo "fisico")
 - Verifica di condensa interstiziale: UNI EN ISO 13788
 - Regime normativo: UNI/TS 11300-4 e 5:2016

Fig. 47: Scelta normative e impostazioni di calcolo in EdilClima

Valutazione 2: Grado di interoperabilità formati di scambio

The screenshot displays the EdilClima software interface with the following sections:

- Dati progetto**: Regime normativo (UNI 10349:2016 selected).
- Dati mensili**: Comune (Torino), Provincia (Torino), Distanza dal mare (> 40 km), Regione di vento (A), Direzione preval. vento (NE), Velocità vento media (1.40 m/s), Velocità vento max (2.80 m/s), Codice ISTAT (1272).
- Dati geografici**: Gradi giorno DPR 412/93 (2617 gg), Altitudine s.l.m. (239 m), Latitudine Nord (45° 7'), Longitudine Est (7° 43'), Codice Catastale (L219), CAP (10100).
- Dati invernali**: Stazione di rilevazione per (TO - Bauducchi), Temperatura (-8.0 °C), Irraggiamento (0.0 °C), Ventosità (-8.0 °C), Periodo convenzionale riscaldamento (Zona climatica E, Durata 183 giorni, Dal giorno 15 ottobre, Al giorno 15 aprile).
- Dati estivi**: Località riferimento estiva (Torino), Temperatura bulbo secco (31.0 °C), Temperatura bulbo umido (22.7 °C), Umidità relativa (50.0 %), Umidità assoluta (14.4 g/kg), Escursione termica giornaliera (11.0 °C).

Fig. 48: Geolocalizzazione modello in EdilClima

Una volta impostati questi parametri e geolocalizzato il modello (Fig. 48) il software effettuerà i calcoli e le verifiche in conformità alle impostazioni.

Il primo test, come per i casi precedenti, è stato effettuato con il modello del BestTest. Una volta aperto, è facilmente possibile importare un file ifc, tuttavia per completare l'importazione occorre riassociare i materiali presenti nel file con i materiali presenti nell'archivio di EC700. (Fig. 49). I materiali presenti nell'archivio del software sono stati inseriti utilizzando i dati

IFC			EdilClima		
Materiale	Spessore [mm]	Codice	Materiale	Spessore [mm]	
BT_Fiberglass_Quilt	66.00	e611	Pannello in lana di vetro	0.00	
BT_Plasterboard	9.00	e1002	Intonaco di gesso	0.00	
BT_Wood_Siding	12.00	e1109	Pannelli di fibra di legno duri e extraduri	0.00	
BT600_Insulation	1003.00	e723	Pannello in lana di roccia a doppia den...	0.00	
BT600_Roof deck	140.80	e1131	Legname (20°C e 65% umidità)	0.00	
BT600_Timber Flooring	25.00	e1132	Pannello in legno compensato	0.00	

EdilClima									
Barriere al vapore Calcestruzzi Intonaci Isolanti Laterizi Solette Vari Pannelli Intercapedini									
Tipologia materiale	Codice	Descrizione	Sp	M.V.	Cond	R.V.	C.T.	Nome	
Barriere al vapore	e101	Barriera vapore in bitume puro	0	1050	0.170	50000	1	UNI EN ISO ...	
Impermeabilizzazioni	e102	Barriera vapore in carta o cartone ...	0	1100	0.230	2500	1	UNI 10351 - ...	
Impermeabilizzazioni in bitume-A.B. ISOL...	e104	Barriera vapore in bitume feltro /fo...	0	1100	0.230	50000	1	UNI EN ISO ...	
	e106	Barriera vapore foglio di alluminio (...)	0	2700	220.000	9999999	0.88	UNI 10351 - ...	
	e109	Barriera vapore in fogli di P.V.C.	0	1390	0.160	50000	0.9	UNI EN ISO ...	
	e110	Barriera vapore in fogli di polietilene	0	920	0.330	100000	2.2	UNI EN ISO ...	
	e111	Barriera vapore in fogli di polietilene	0	980	0.500	100000	1.8	UNI EN ISO ...	

Fig. 49: Associazione stratigrafie in EdilClima

Valutazione 2: Grado di interoperabilità formati di scambio

riportati nelle apposite normative.

Una volta associati i materiali occorre controllare che tutti i locali appartengano a determinate zone termiche ed in caso positivo è possibile importare il file. (Fig. 50)

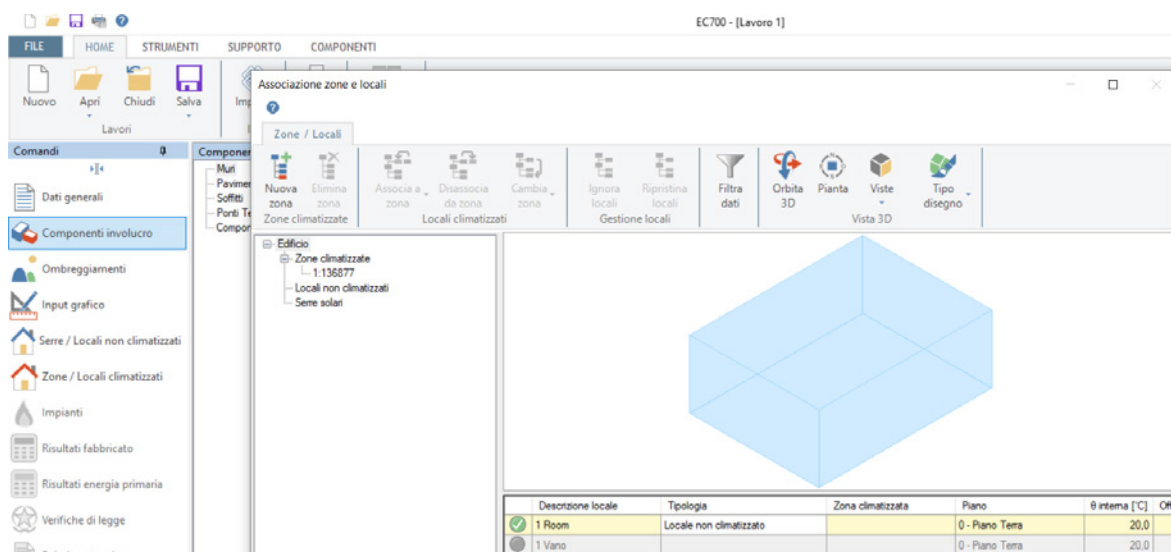


Fig. 50: Controllo zone termiche in EdilClima

Appena importato il file è possibile visualizzare e modificare elementi costituenti l'involucro edilizio. Parallelamente alla modifica di un elemento, i dati riguardanti la struttura vengono aggiornati simultaneamente (Fig. 51). In questa fase il software offre anche la possibilità di valutare simultaneamente il rispetto delle verifiche di legge per i singoli elementi. (Fig. 53)

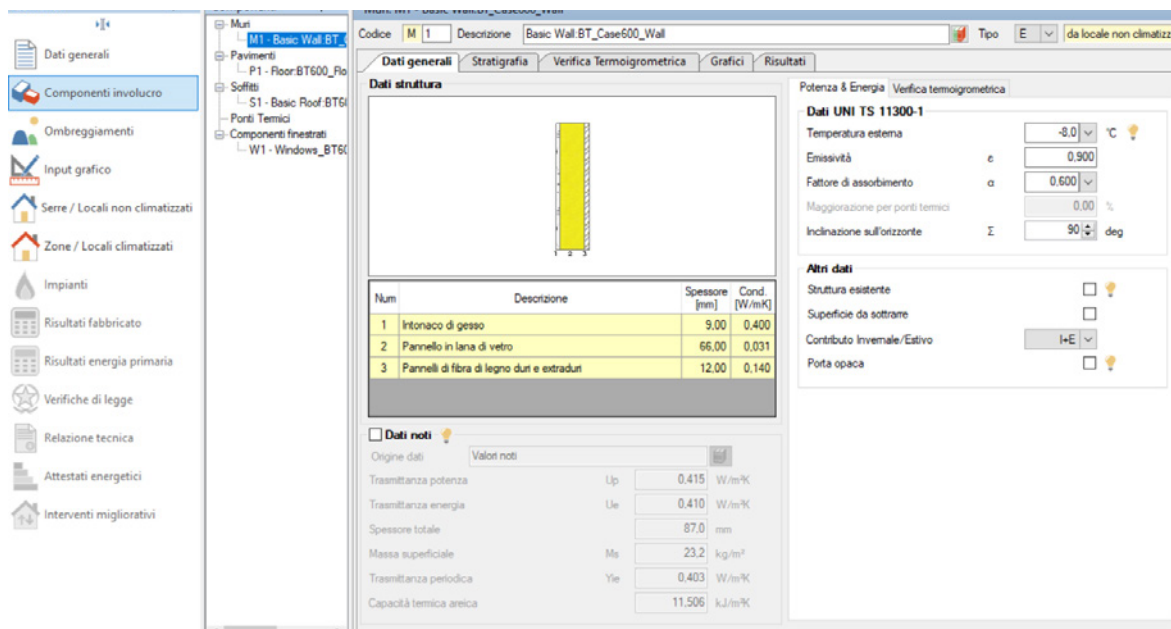


Fig. 51: Controllo stratigrafie in EdilClima

Anche il test riguardante il modello FCA è stato positivo. Il file IFC è stato letto e, dopo aver riassociato i materiali e definito a quali zone termiche appartenessero i locali non correttamente assegnati, il modello è stato correttamente importato.

Valutazione 2: Grado di interoperabilità formati di scambio

Fig. 53: Controllo verifiche di legge in EdilClima

L'input grafico consente di verificare ed apportare modifiche al modello creato dal software sulla base dei dati inseriti nel file ifc importato. È inoltre possibile valutare eventuali incongruenze geometriche causate dall'importazione di un file incorretto. (Fig. 52)

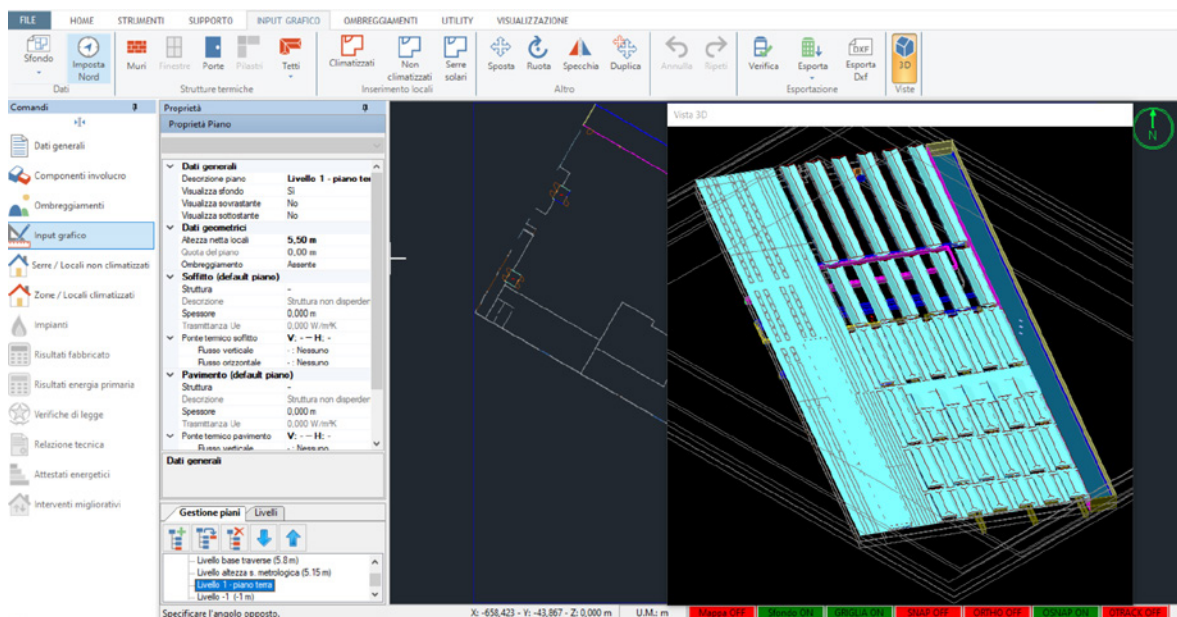


Fig. 52: Input grafico EdilClima

Design Builder

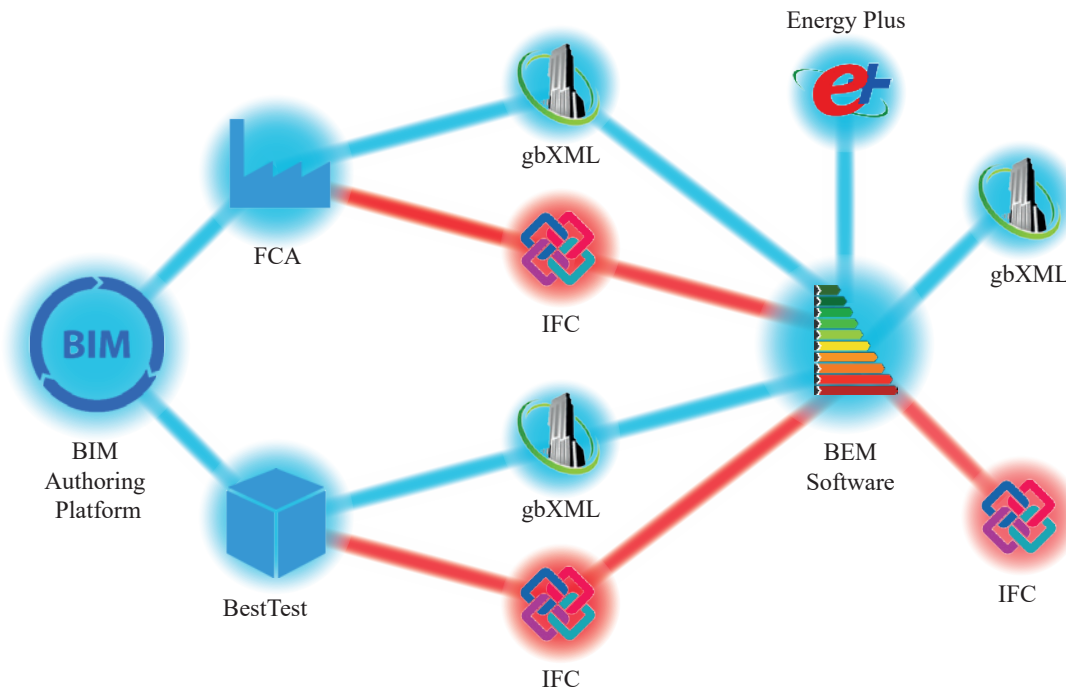


Fig. 54: Schema flusso dati Design Builder

Le valutazioni sono state eseguite con la versione 4.5 del software. Tuttavia è da specificare la presenza dell'ultima versione del modulo chiamato DesignBuilder UNI 11300. Tramite quest'ultimo è possibile esportare il modello ottenuto in Design Builder in un software esterno chiamato MasterClima 11300, il quale consente di effettuare calcoli in conformità con la normativa UNI TS 11300.

Il test riguardante il modello BestTest, in questo caso è avvenuto tramite l'importazione del formato .gbXML poiché quest'ultimo è l'unico formato riconosciuto dal software. L'importazione avviene selezionando il file da importare e non vi è bisogno di effettuare

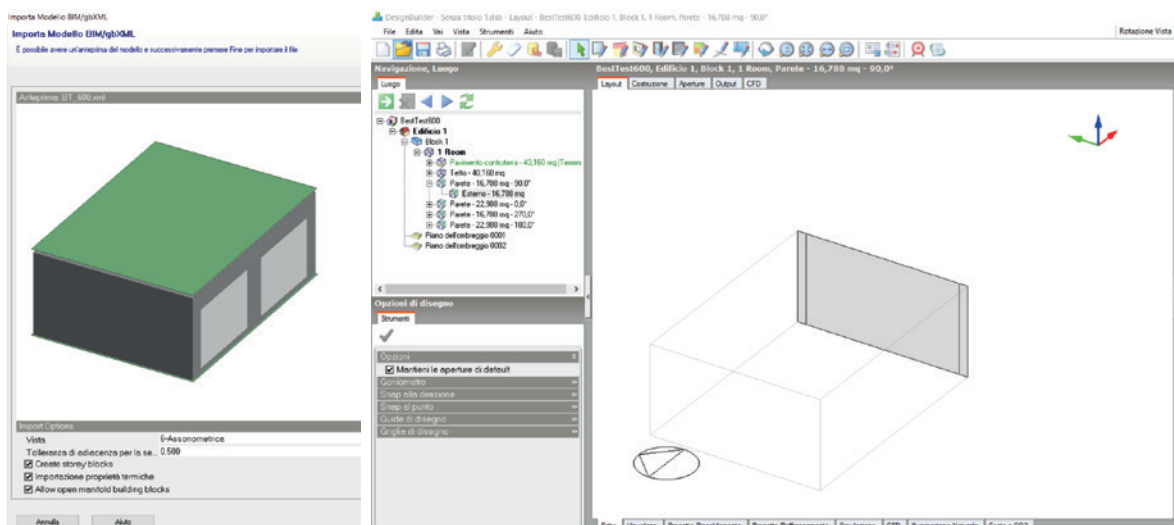


Fig. 55: Importazione modello BestTest in Design Builder

Valutazione 2: Grado di interoperabilità formati di scambio

alcuna modifica preliminare, il file viene quindi caricato (Fig. 55).

Nel caso del modello BestTest non ci sono state particolari incorrettezze nell'importazione delle geometrie, tuttavia vi è da considerare che il modello originale è stato trasformato in un modello energetico provvisto solo di superfici bidimensionali.

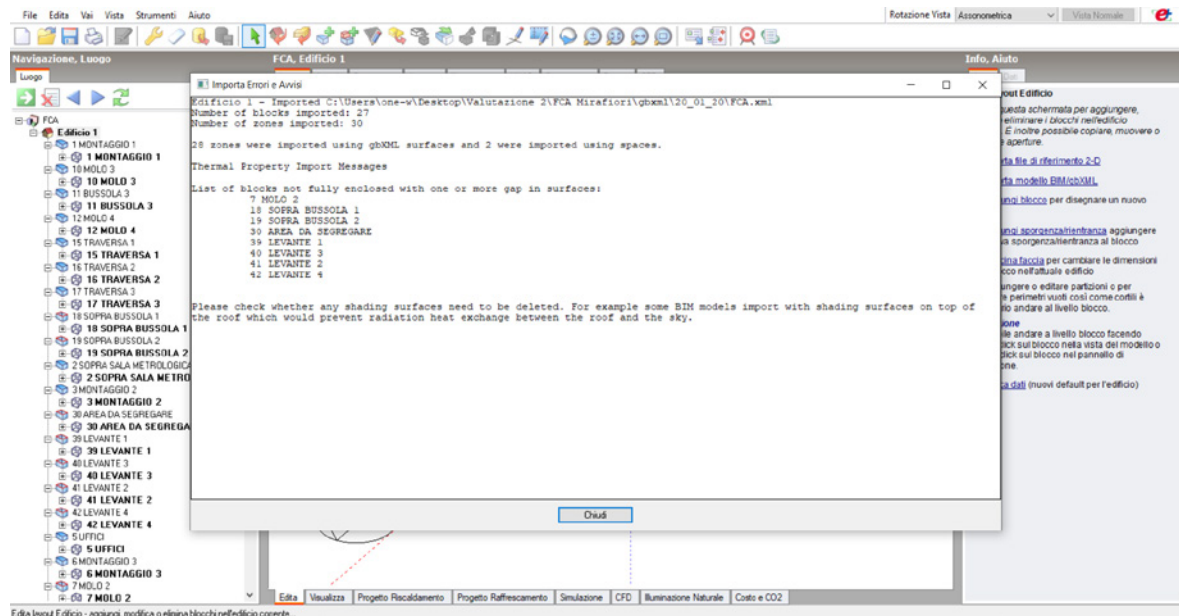


Fig. 56: Importazione modello FCA in Design Builder

Durante l'importazione del modello FCA è stato visualizzato un messaggio di avvertenza segnalante delle irregolarità nella struttura importata (Fig. 56).

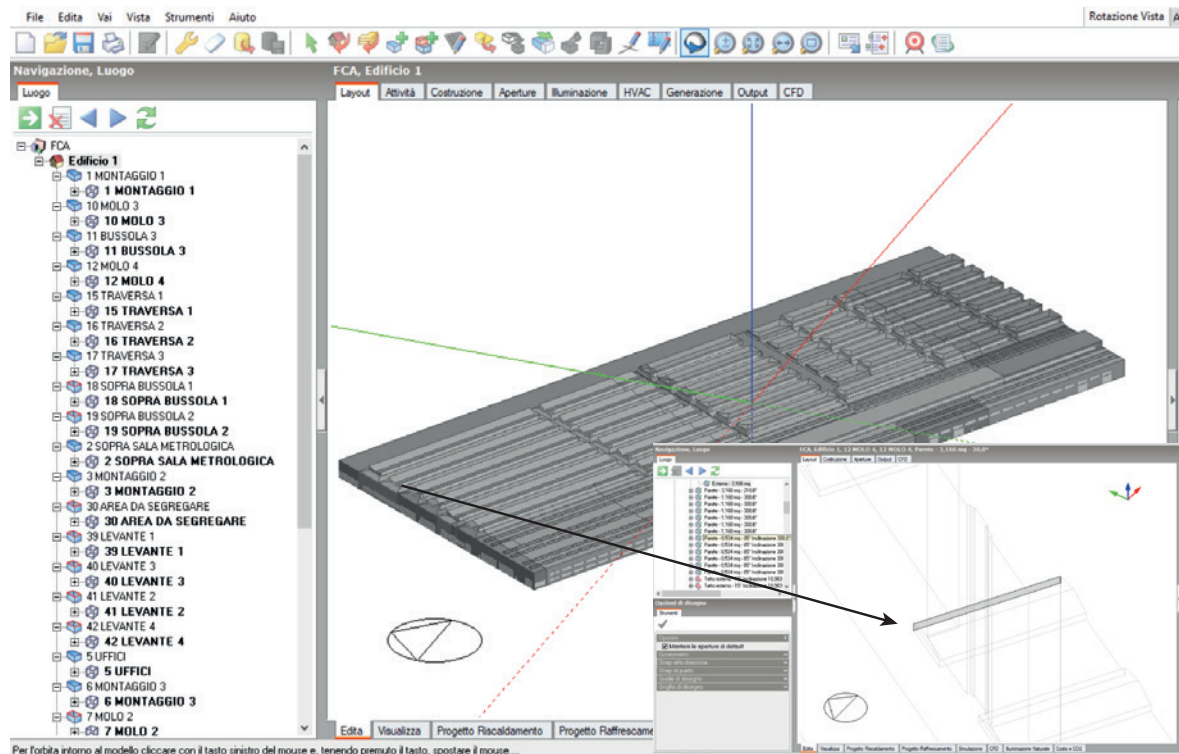


Fig. 57: Importazione modello FCA in Design Builder

Tralasciando questa avvertenza, si può dire che il modello è stato importato correttamente. Durante questa valutazione si riscontra un rallentamento importante della macchina. Ulteriori test non sono stati effettuati considerando che la versione valutata non è l'ultima disponibile, quindi sicuramente arretrata per molti aspetti.

Open Studio

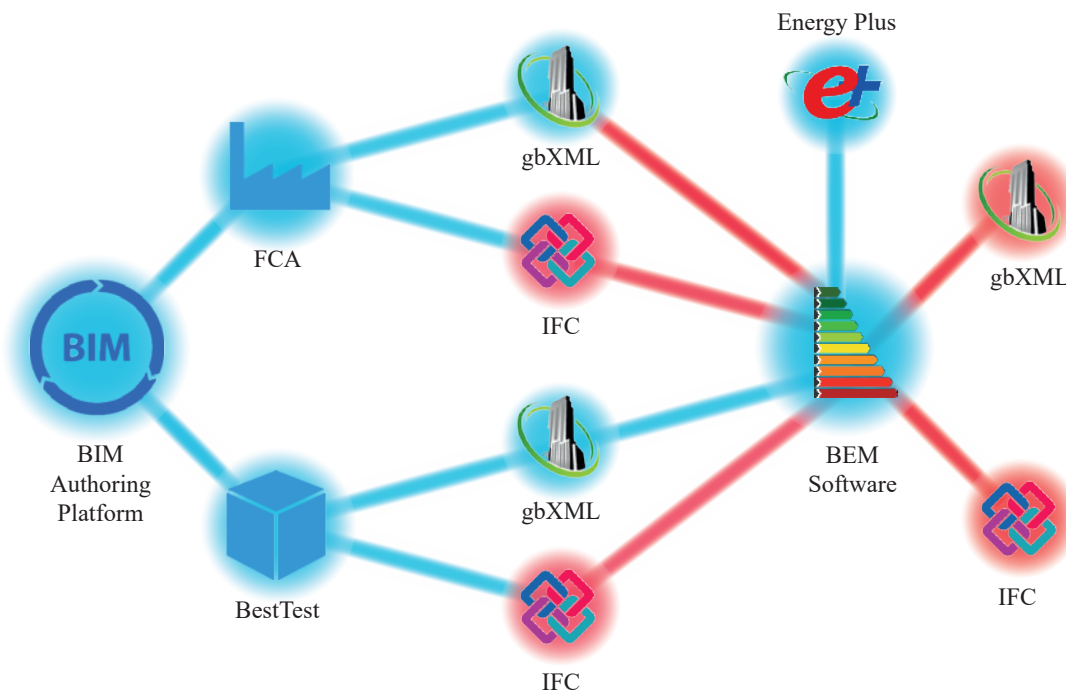


Fig. 58: Schema flusso dati Open Studio

Open studio è un software open source utile a svolgere analisi energetiche degli edifici, nonché prima GUI (Graphical User Interface) conforme al motore di calcolo Energy Plus. Attualmente supporta i motori di calcolo Energy Plus e Radiance. Open Studio si divide in una applicazione e un *plugin* per il software di modellazione 3D SketchUp. L'applicazione Open Studio rappresenta il nucleo del software ed è quest'ultimo che svolge i calcoli legati alle analisi tecniche. Il *plugin* per Sketchup è invece utile a creare le geometrie del modello energetico che può essere esportato per in formato IDF (Input Data File) Energy Plus.

I test sono stati eseguiti utilizzando l'applicazione di Open Studio importando su quest'ultima il file .gbXML. Open Studio dovrebbe comprendere la possibilità di importare file in formato ifc, tuttavia quest'ultima opzione non è disponibile per i nuovi utenti. Il problema che impedisce l'importazione dei file ifc riguarda la configurazione del BIM server di Open Studio. Per effettuare la configurazione occorre installare nella propria macchina il software Java versione 7 ma sfortunatamente quest'ultima è difficile da reperire in quanto versione

Valutazione 2: Grado di interoperabilità formati di scambio

ormai superata e non più implementata dalla *software house* (il download è disponibile solo per gli utenti in possesso di Java da prima del mese di Aprile 2015).

Il primo test è stato effettuato utilizzando il file .gbXML del modello BestTest. In questa fase viene visualizzato un messaggio di avvertenza riguardante le geometrie (Fig. 59), (messaggio non visualizzato negli altri test), tuttavia l'importazione avviene e le geometrie vengono rappresentate correttamente e gli elementi importati dispongono delle informazioni riguardanti le stratigrafie.

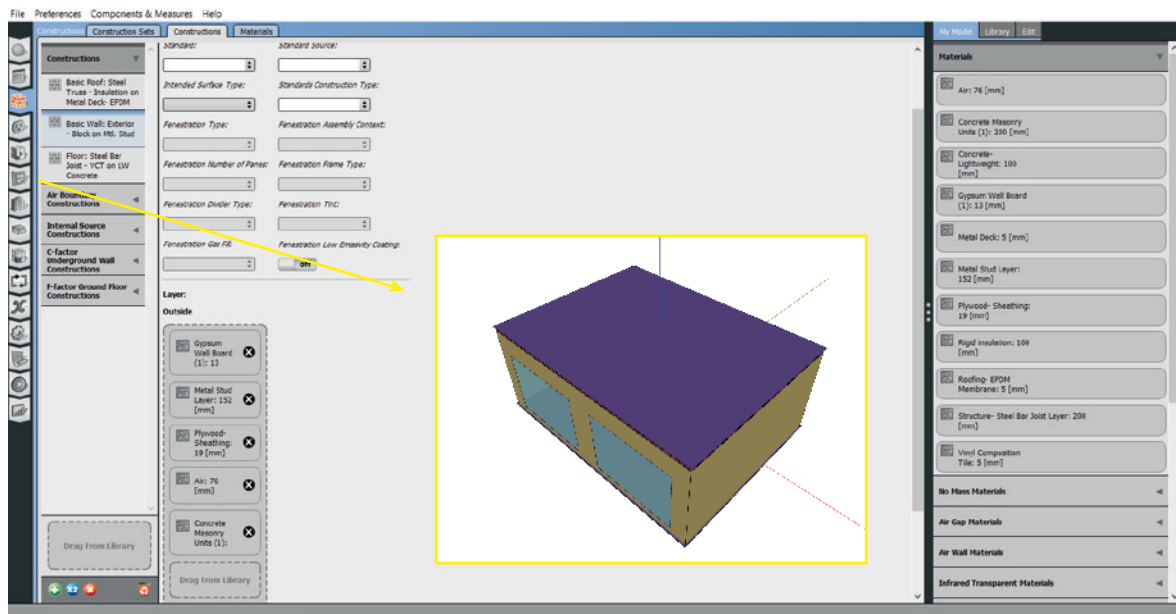


Fig. 60: Visualizzazione grafica BestTest in Open Studio

Il test riguardante il modello FCA purtroppo non ha prodotto risultati soddisfacenti. Quest'ultimo infatti è stato letto ed anche in questo caso, in fase di importazione, sono state visualizzate delle avvertenze riguardanti le geometrie. Il modello è stato tuttavia importato e le zone termiche riconosciute, così come le geometrie e le proprietà di queste ultime. Non vi sono stati particolari problemi che non siano stati riscontrati nella valutazione BestTest. Tuttavia la valutazione non è stata approfondita a causa di un malfunzionamento, molto probabilmente legato alle dimensioni elevate del modello. Una volta caricato il modello, è stato possibile visualizzare diverse finestre che riportavano i dati corretti, tuttavia era altresì impossibile aprire altre finestre. Non appena si interrogava una finestra "non funzionante" il software andava in *crash*. Anche dopo svariati tentativi il risultato non mutava ed il software

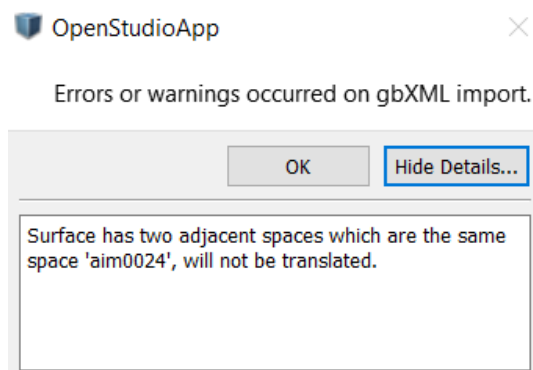


Fig. 59: Errore importazione BestTest in Open Studio

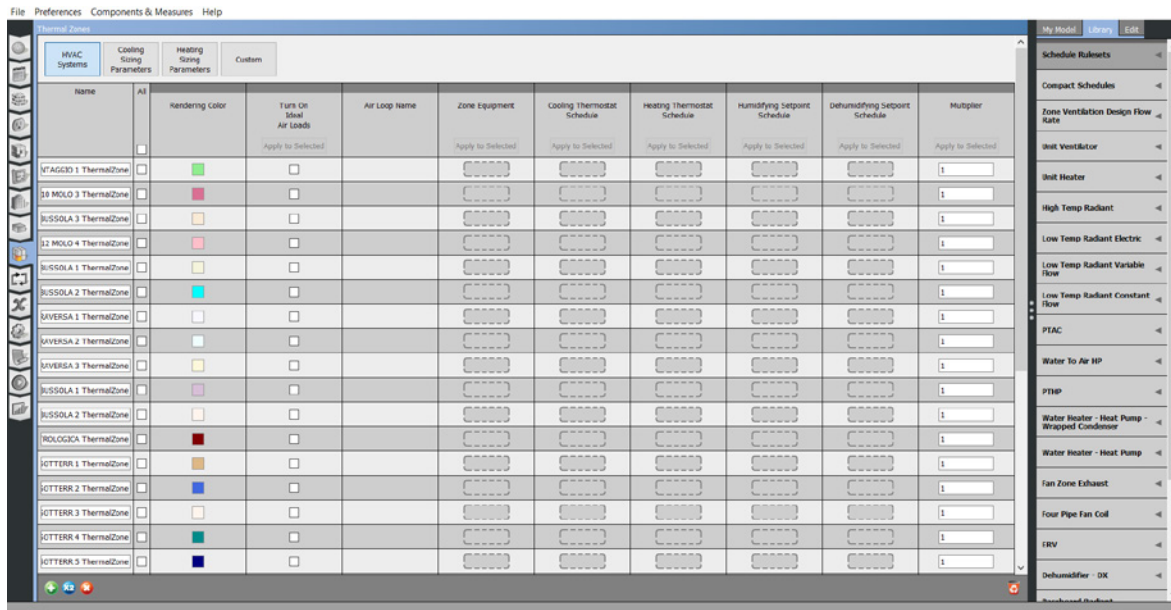


Fig. 61: Zone termiche FCA importate in Open Studio

smetteva di rispondere. Per le finestre funzionanti il software risultava molto lento nella risposta ai comandi dell'operatore. Data l'impossibilità alla lettura agevolata del modello si è optato per l'interruzione della valutazione.

Termolog EpiX 10

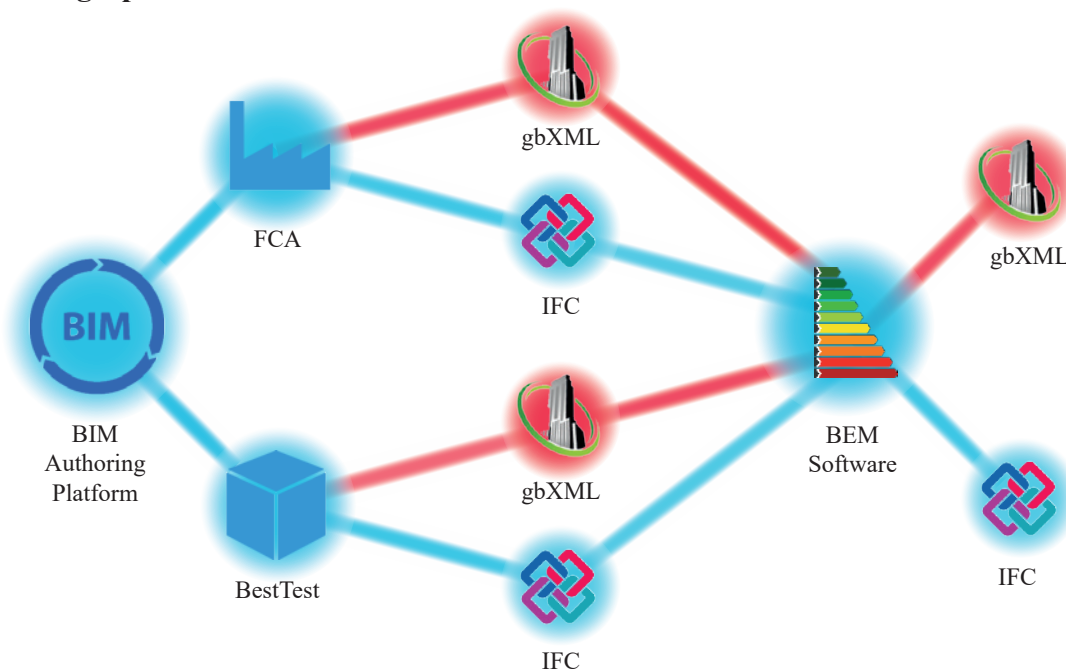


Fig. 62: Schema flusso dati Termolog

Termolog è uno dei software approvati dal CTI. I calcoli eseguiti sono conformi alle norme UNI TS 11300 ed è anche possibile effettuare diagnosi energetiche e calcoli con metodo

Valutazione 2: Grado di interoperabilità formati di scambio

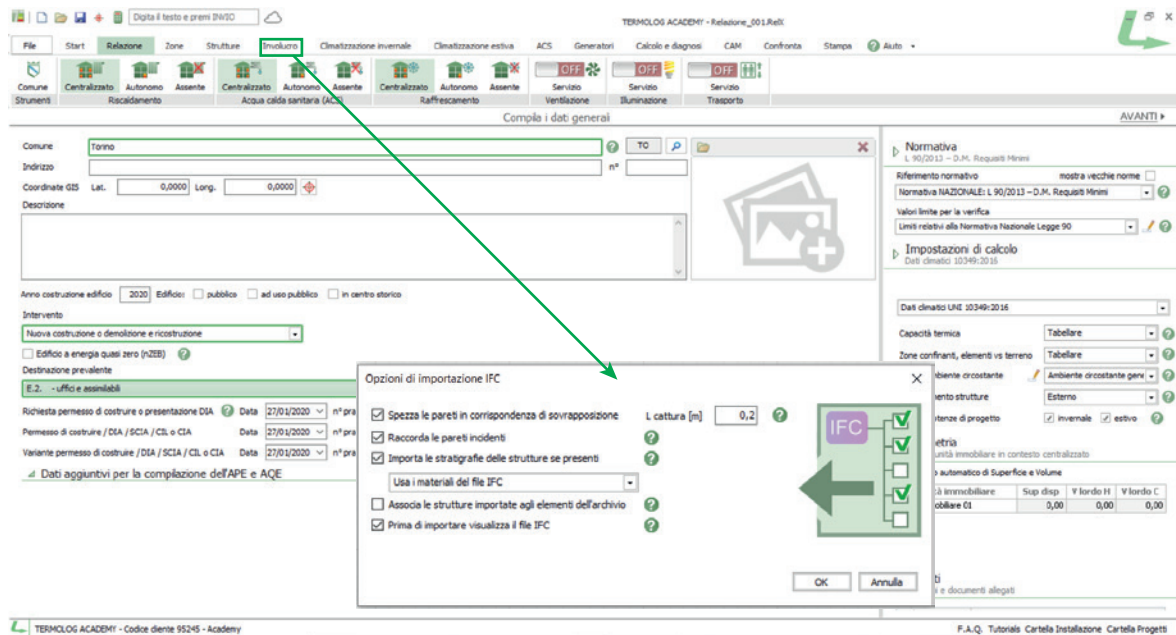


Fig. 63: Impostazione normative e importazione IFC Termolog

dinamico orario. Il software può importare le informazioni relative ai materiali direttamente dal file ifc o vi è la possibilità di riassociare questi ultimi con i materiali presenti in archivio, redatti utilizzando i dati forniti dalle normative tecniche.(Fig. 63)

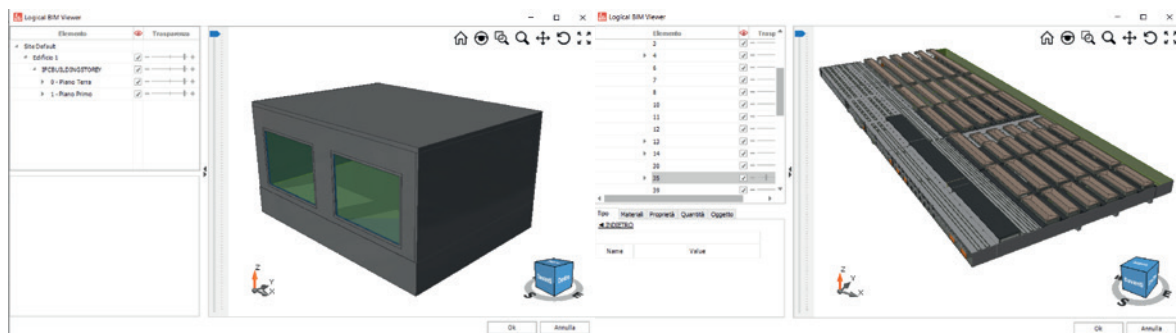


Fig. 64: Importazione modelli BestTest e FCA in Termolog

I test sono stati effettuati importando i file ifc dei modelli in quanto unico formato riconosciuto tra i due valutati, è inoltre possibile importare file in formato .dwg e .dxf. Il test riguardante il modello BestTest è andato a buon fine. Le geometrie sono state riconosciute perfettamente così come le stratigrafie di queste ultime.

Il test riguardante il modello FCA, come per il precedente, è andato a buon fine. Le geometrie sono state riconosciute, anche se con alcune imperfezioni.

Le zone termiche sono state correttamente importate dal file ifc ed ad esse sono stati associati gli elementi costituenti l'involucro edilizio con le relative stratigrafie. (Fig. 65; Fig. 64)

Valutazione 2: Grado di interoperabilità formati di scambio

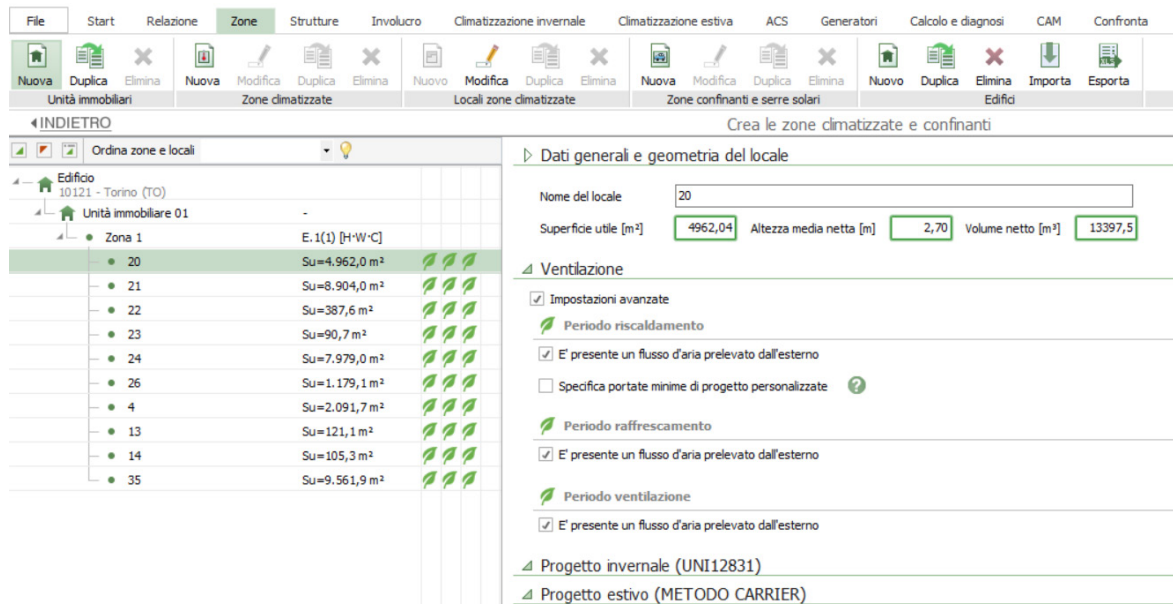


Fig. 65: Zone termiche importate Termolog

Seleziona o crea le strutture dell'edificio

Struttura	Tipo	Verso ...	S	A	U/ψ	C	Uso	U
Pareti								
IFC								
Muro di base:Generico - 150 mm 2	≡ Parete	Interno	15,0	-	2,127	72,288	5.679,38	✓
Muro di base:Generico - Muratura 150 mm	≡ Parete	Esterno	20,0	-	1,754	69,873	14.484,17	✓
Materiale	S [mm]	λ [W/mK]	C [KJ/KgK]					
• Adduttanza interna (flusso orizzontale)	0	7,690	0,000					
• Calcestruzzo - Leggero	200	0,500	1,000					
• Adduttanza esterna (flusso orizzontale)	0	25,000	1,000					
Muro di base:Generico - Muratura 150 mm 2	≡ Parete	Interno	20,0	-	1,754	69,873	1.649,49	✓
Muro di base:Interno - Parete 100 mm 2	≡ Parete	Interno	10,0	-	2,924	64,053	347,44	✓
Materiale	S [mm]	λ [W/mK]	C [KJ/KgK]					
• Adduttanza interna (flusso orizzontale)	0	7,690	0,000					
• Intonaco	14	1,000	1,000					
• Unità di muratura in calcestruzzo	72	0,500	1,000					
• Intonaco	14	1,000	1,000					
• Adduttanza esterna (flusso orizzontale)	0	25,000	1,000					
Muro di base:Interno - Parete 120 mm 2	≡ Parete	Interno	8,0	-	3,999	24,869	9,96	✓
Muro di base:Interno - Parete 120 mm 3	≡ Parete	Interno	12,0	-	2,618	70,290	445,76	✓
Materiale	S [mm]	λ [W/mK]	C [KJ/KgK]					

Fig. 66: Stratigrafie FCA importate in Termolog

Valutazione 2: Grado di interoperabilità formati di scambio

Termus

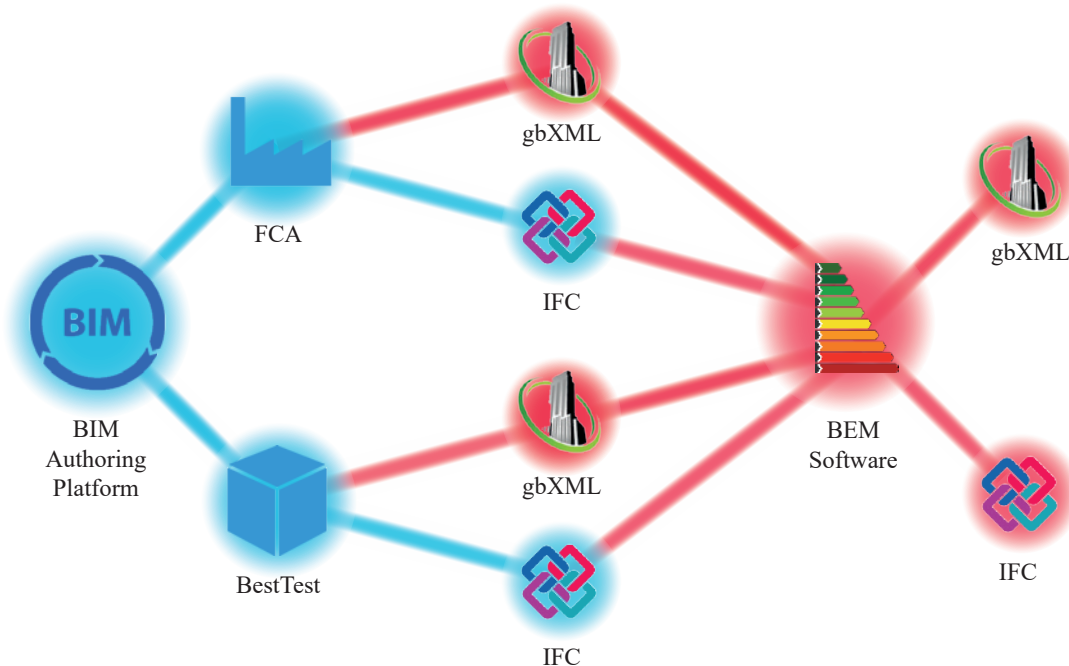


Fig. 67: Schema flusso dati Termus

Termus è uno dei software approvati dal CTI. Con Termus è possibile effettuare calcoli conformi alla normativa UNI TS 11300 ed è in grado di svolgere calcoli in regime dinamico orario. Il software crea un modello energetico basandosi su input grafico o importando un file ifc. Pur presentandosi come uno strumento molto valido non è stato possibile procedere con le valutazioni a causa di un errore inaspettato. Vi sono motivi per credere che l'errore sia generato per una qualche incompatibilità fra la macchina utilizzata ed il software. In ogni caso non è stato possibile eseguire alcuna valutazione.

Nuovo



Fig. 68: Finestra importazione IFC in Termus

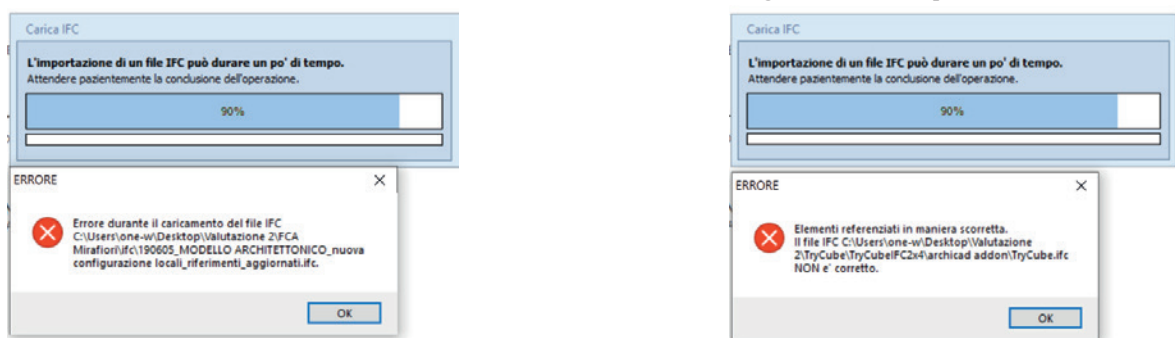


Fig. 69: Errore importazione file IFC in Termus

2.2.3 Valutazione 3: Conformità alla normative vigenti

In base ad i test effettuati ed in considerazione alla normativa attuale, sono stati selezionati solo i software che rispondono positivamente al recepimento normativo. A tal proposito è possibile consultare il sito del CTI, il quale dispone di una pagina riportante l'elenco dei software abilitati a rilasciare certificati di prestazione energetica degli edifici. La pagina in questione è raggiungibile all'indirizzo internet <https://www.cti2000.it>, che è lo strumento ufficiale di consultazione riconosciuto da normativa.



Fig. 70: Certificato di conformità rilasciato dal CTI
allegato A - Regolamento per la procedura di verifica

Tra i software valutati solo Edileclima EC700, Termolog EpiX 10 e Termus risultano attualmente riconosciuti dal CTI. I tre software sono strumenti estremamente validi per quanto riguarda valutazioni di prestazione energetica degli edifici e permettono di raggiungere un livello di accuratezza dei risultati abbastanza soddisfacente. Tuttavia per un ulteriore

approfondimento sull'analisi dell'interoperabilità BIM to BEM si è scelto di proseguire la valutazione utilizzando il software Termolog EpiX 10. Questa scelta è dovuta sostanzialmente ai problemi tecnici riscontrati durante l'uso di Termus ed alla presenza della funzione forzata da parte di Edileclima riguardante la associazione dei materiali presenti nel file ifc importato con i materiali previsti da normativa. Utilizzare materiali da normativa ha infatti degli aspetti positivi e negativi. Uno dei pro di quest'ultima funzione è sicuramente la sicurezza di utilizzo di materiali che abbiano proprietà termiche adeguate e verificate, evitando così possibili errori da parte degli utenti durante la compilazione di materiali personalizzati. Un aspetto negativo invece è dato dalla impossibilità di importare automaticamente le informazioni presenti nel file ifc riguardanti le proprietà termiche dei materiali costituenti l'involucro disperdente. Le proprietà termiche in questione in molti casi sono specifiche dell'edificio valutato e difficilmente si riscontra una corrispondenza diretta e perfetta con i materiali presenti in normativa. L'inesattezza di tali proprietà termiche comporta delle ripercussioni nel grado di accuratezza della valutazione, soprattutto per quanto riguarda i materiali isolanti termici ed acustici che spesso hanno proprietà specifiche fornite dai produttori.

2.3 Esportazione IFC e configurazione formato di scambio

Autodesk Revit è uno dei software BIM in grado di leggere ed importare informazioni da un file ifc, così come è in grado di esportare informazioni presenti nel modello utilizzando il medesimo formato. Tuttavia attualmente la capacità di Revit di default nella realizzazione di file ifc è alquanto ridotta. Tale lacuna viene colmata dall'installazione di un *plugin* chiamato AutodeskIFC20** (dove per ** si intende il valore numerico riferito all'anno della versione del software Revit, ad esempio Revit 2018 avrà come *plugin* AutodeskIFC2018). Come descritto dal sito Autodesk *“The IFC for Autodesk® Revit® 2018 contains up-to-date improvements on the default IFC import and export capabilities of Revit contributed by Autodesk and our Open Source contributors.”* [33], ovvero è utile al miglioramento della qualità dei dati importati ed esportati in Revit tramite ifc. Per utilizzare tale standard è quindi doveroso installare questo *plugin*, al fine di raggiungere una qualità soddisfacente dei dati in uscita dal software BIM. Il *plugin* è scaricabile dagli utenti registrati dal sito ufficiale di Autodesk. (Fig. 71)

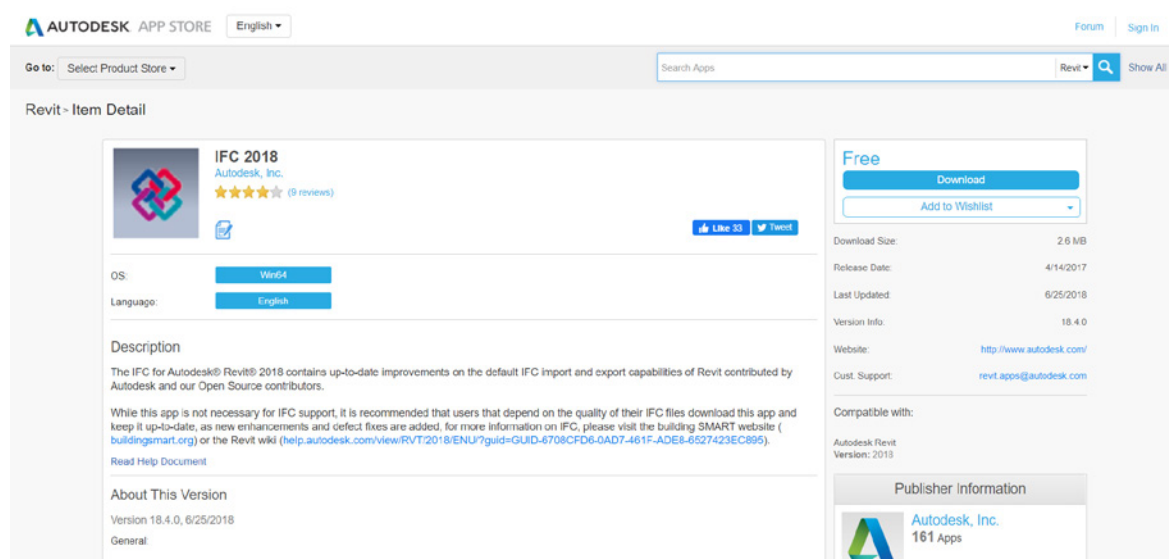


Fig. 71: Plugin AutodeskIFC2018

Una volta installato il *plugin* in Revit è possibile notare come la finestra di esportazione IFC dispone di diverse opzioni precedentemente assenti. Adesso è possibile scegliere fra diversi schemi preimpostati e personalizzare dei parametri in base alle proprie esigenze. (Fig. 72) Tuttavia limitarsi a scegliere fra le opzioni presenti spesso non è sufficiente. Nella maggioranza dei casi infatti, i software che esportano informazioni in formato IFC non riescono a mappare file in modo efficiente. È facile incappare in mancate corrispondenze, bug o assenze dei dati desiderati. Premesso ciò, è facilmente intuibile quanto sia importante effettuare un check del file IFC creato al fine di verificare la presenza, nonché la correttezza, dei dati desiderati. Per

Impostazioni per esportazioni personalizzate

svolgere questa operazione esistono diversi software chiamati ifc viewer, in questo caso è stato utilizzato il software IFC Quick Browser che consente di leggere le stringhe presenti nel file IFC fornendo i collegamenti fra le varie classi presenti.

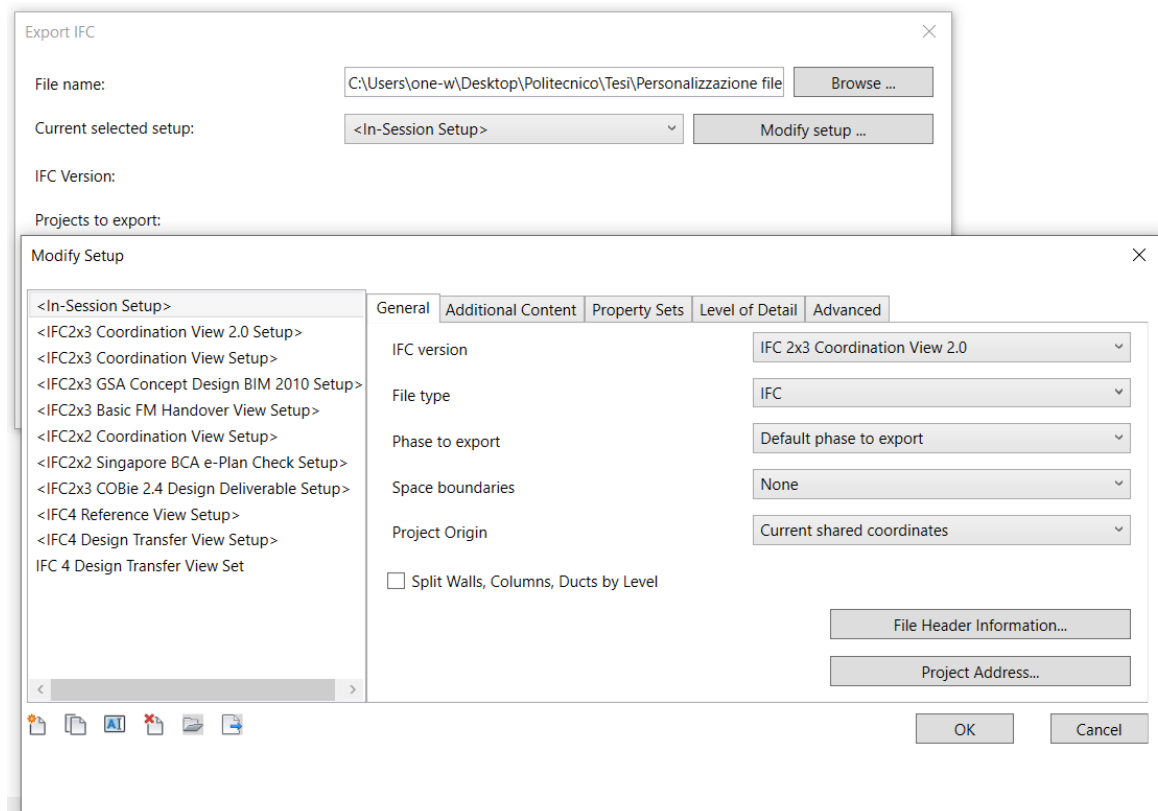


Fig. 72: Interfaccia esportazione ifc di Revit migliorata

2.3.1 Impostazioni per esportazioni personalizzate

Convenzionalmente per quando riguarda il software Revit esistono diverse operazioni che l'operatore può effettuare per rendere il più efficiente possibile l'esportazione dei dati in formato IFC. Attualmente le procedure da seguire non vengono descritte chiaramente così come è difficile trovare manuali, che dovrebbero essere forniti dalla *software house*, che spieghino in modo esaustivo e dettagliato cosa fare per effettuare tali operazioni per ogni classe IFC.

Le operazioni eseguibili si dividono principalmente in:

- *Mapping table*: accessibile da *File > Export > Options > IFC Options*. Si tratta di una tabella che crea delle corrispondenze dirette fra le categorie presenti in Revit e le classi IFC. Da questa tabella è possibile visualizzare le informazioni che verranno esportate nel file IFC, è inoltre possibile modificare tali informazioni indicando il nome della classe IFC da inserire nel file in corrispondenza del relativo parametro o settare con *Not Exported* i parametri da non esportare.

Impostazioni per esportazioni personalizzate

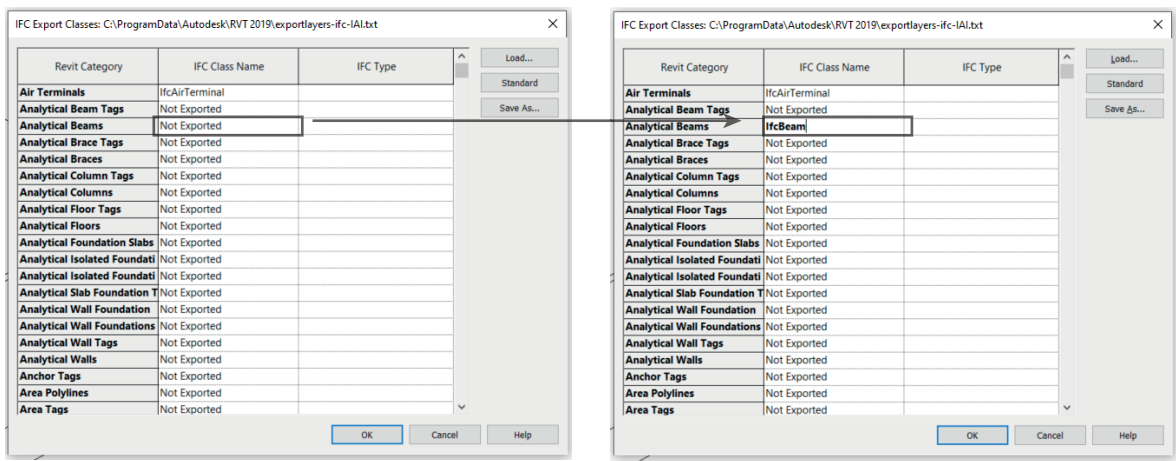


Fig. 73: Modifica mappatura IFC in Revit

- In alternativa è possibile editare il file .txt utilizzato dal software inserendo manualmente le informazioni desiderate. È necessario utilizzare l’input tab fra una informazione e l’altra.

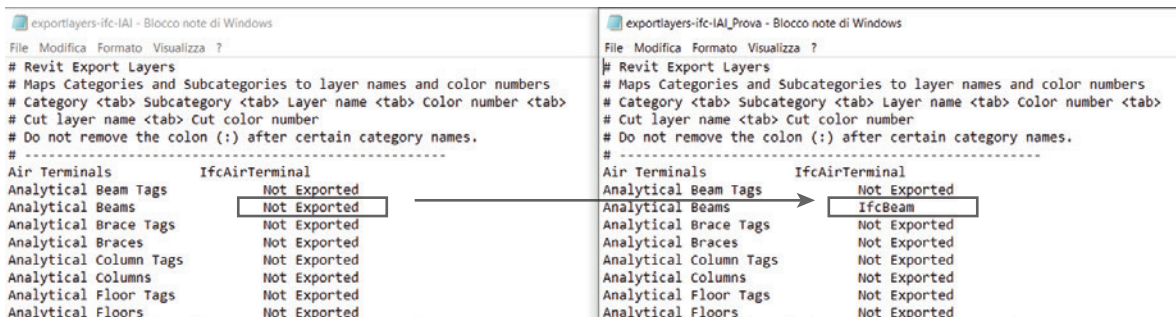


Fig. 74: Modifica file txt per mappatura IFC in Revit

- *Parameter Mapping Table*: Si tratta di una tabella che crea delle corrispondenze fra i parametri presenti in Revit con i parametri definiti dall’utente in un file .txt. Accessibile dalla finestra *Property Sets* in *Export IFC*, il file txt personalizzato può essere caricato nel software dal comando *Export parameter mapping table*. Una volta caricato il file txt, sarà creata una corrispondenza fra i parametri Revit ed i parametri IFC indicati nel file. Il file di testo deve seguire una sintassi ben precisa altrimenti non verrà letto dal software. Per inserire un parametro con questo metodo occorre creare un file txt e inserire informazioni nel seguente ordine:

IFC_Common_PropertySet <tab> IFC_Property_Name <tab> Revit_Property_Name

Pset_WallCommon	Compartmentation	CustomParameter1
Pset_WallCommon	Combustible	CustomParameter2

Fig. 75: Modifica Parameter Mapping Table in Revit

Impostazioni per esportazioni personalizzate

Secondo l'esempio indicato in un manuale tecnico fornito da Autodesk (Fig. 75):

Ogni informazione è divisa dalla seguente tramite l'input tab, quindi seguendo lo schema si ha:

Pset_WallCommon che equivale a *IFC_Common_PropertySet*
tab

Compartmentation che equivale a *IFC_Property_Name*
tab

CustomParameter1 che equivale a *Revit_Property_Name*

Quindi rinvio a capo per eventuali altri parametri.

- *PropertySet* (comando *Export Revit property sets*): con questa operazione i parametri presenti nel modello Revit vengono copiati come parametri IFC. Con questa funzione è possibile esportare tutte le proprietà legate ad un elemento. Questa funzione è utile in fase di valutazione di un modello, ma sconsigliata nell'uso comune. Anche se potrebbe sembrare vantaggioso avere un comando che esporti tutte le proprietà presenti, in pratica non è così. Attivare questa funzione infatti rende il file esportato più pesante in termini di memoria per circa il 70% (nella maggioranza dei casi), inoltre è consigliato non inserire in un file ifc informazioni superflue o non richieste. (Fig. 76)

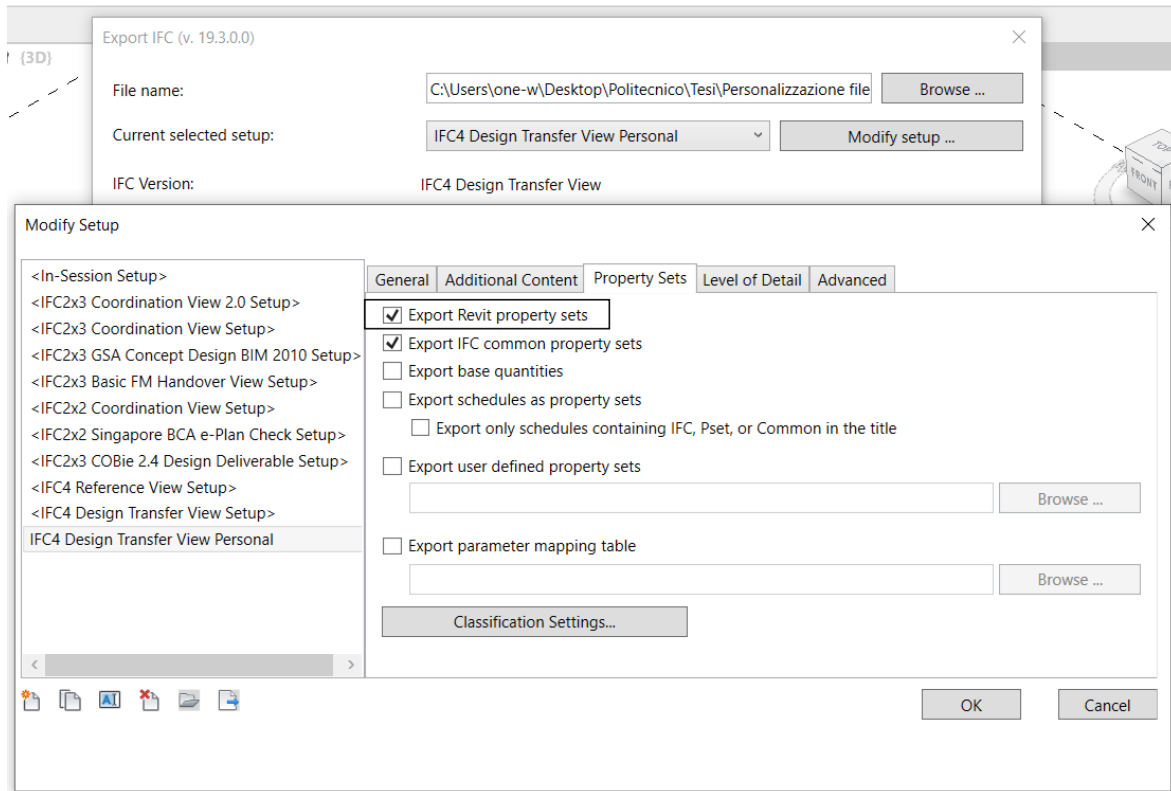


Fig. 76: Schermata esportazione Revit Property Sets

- *User Defined PropertySet*: questa funzione può essere intesa come una specifica

della funzione *Export Revit property sets* e offre un buon grado di libertà all'utente nella scelta delle informazioni da esportare. Questa operazione consente di esportare proprietà ben precise definite nel modello Revit. Principalmente vi sono due modi per utilizzare questa funzione. La prima opzione riguarda la creazione in Revit di *schedule* (abachi) che contengono le informazioni da esportare, queste vengono quindi lette in fase di esportazione ed inserite nel file IFC. Questa opzione non è la scelta migliore in termini di memoria della macchina, infatti il modello viene appesantito dagli abachi che hanno il solo scopo di fornire le informazioni da esportare, inoltre in questo modo avviene una duplicazione di dati già presenti, operazione sconsigliata in tutti i casi. La seconda opzione riguarda la creazione di un file txt che racchiude le informazioni necessarie per l'esportazione delle informazioni desiderate. (Fig. 77)

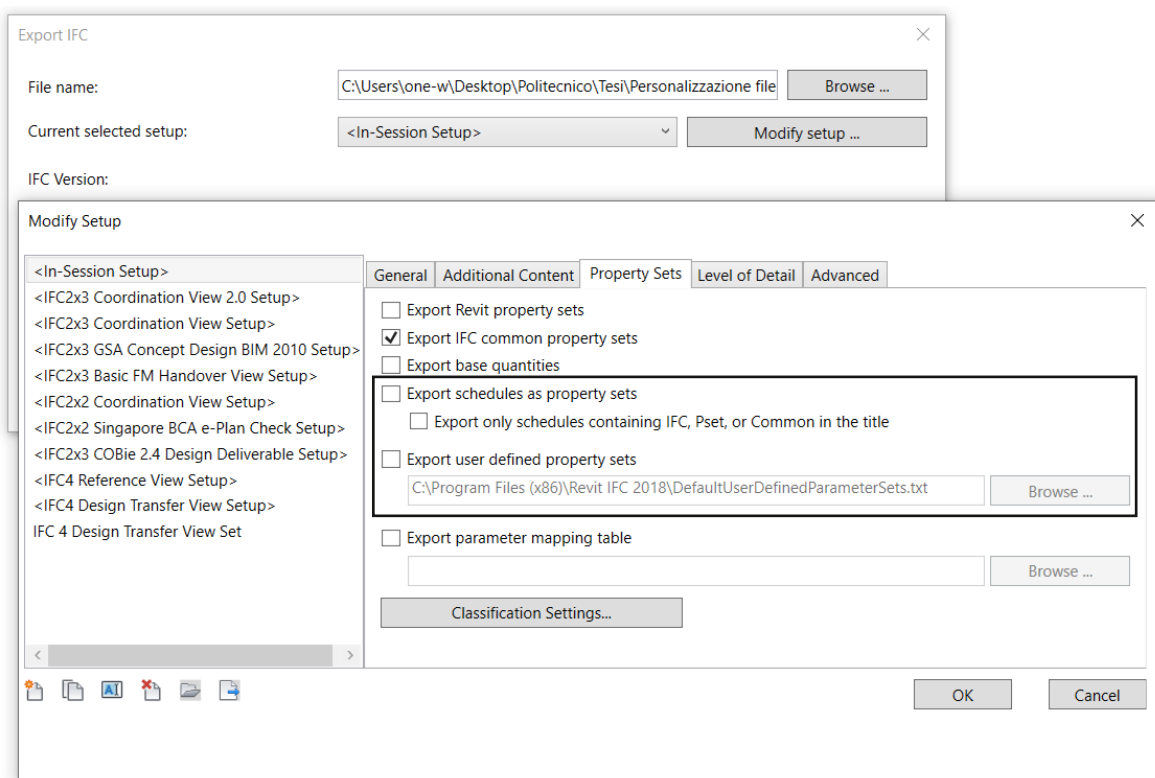


Fig. 77: Schermata esportazione User Defined Property Sets

Una versione di default file .txt può essere trovata nella cartella di installazione del *plugin AutodeskIFC2018*.

Questo metodo oltre ad evitare di appesantire il file del modello con svariati abachi, dà la possibilità di definire un unico *Property Set* definito dall'utente riferito a parametri comuni a più categorie di Revit quindi a diverse classi ifc. Questa funzione è accessibile dalla finestra *Property Sets* in *Export IFC*, spuntando l'opzione *Export user defined property set* e specificando il percorso di un file txt. Tuttavia durante la

Impostazioni per esportazioni personalizzate

creazione del *User Defined Property Set* è importante non commettere nessun errore di scrittura, sia per il riferimento alla classe, sia per il tipo di dato da esportare (ad esempio *real* o *integer* per valori numerici, *string* per una successione di caratteri o *boolean* per dati del tipo *True/False*). Il file in questione deve essere redatto seguendo una sintassi precisa con la quale l'utente specifica quali parametri devono essere inseriti nel file IFC.

```
#
# User Defined PropertySet Definition File
#
# Format:
#   PropertySet:      <Pset Name>      I[nstance]/T[ype]      <element list separated by ','>
#   <Property Name 1> <Data type>      <[opt] Revit parameter name, if different from IFC>
#   <Property Name 2> <Data type>      <[opt] Revit parameter name, if different from IFC>
#   ...
#
# Data types supported: Area, Boolean, ClassificationReference, ColorTemperature, Count, Currency,
#   ElectricalCurrent, ElectricalEfficacy, ElectricalVoltage, Force, Frequency, Identifier,
#   Illuminance, Integer, Label, Length, Logical, LuminousFlux, LuminousIntensity,
#   NormalisedRatio, MassDensity, PlaneAngle, PositiveLength, PositivePlaneAngle, PositiveRatio,
#   Power, Pressure, Ratio, Real, Text, ThermalTransmittance, ThermodynamicTemperature, Volume,
#   VolumetricFlowRate
#
# Example property set definition for COBie:
#
#PropertySet:  COBie_Specification      T      IfcElementType
#   NominalLength      Real      COBie.Type.NominalLength
#   NominalWidth       Real      COBie.Type.NominalWidth
#   NominalHeight      Real      COBie.Type.NominalHeight
#   Shape              Text      COBie.Type.Shape
#   Size               Text      COBie.Type.Size
#   Color              Text      COBie.Type.Color
#   Finish             Text      COBie.Type.Finish
#   Grade              Text      COBie.Type.Grade
#   Material           Text      COBie.Type.Material
#   Constituents       Text      COBie.Type.Constituents
#   Features           Text      COBie.Type.Features
#   AccessibilityPerformance      Text      COBie.Type.AccessibilityPerformance
#   CodePerformance     Text      COBie.Type.CodePerformance
#   SustainabilityPerformance      Text      COBie.Type.SustainabilityPerformance
#
```

Fig. 78: User Defined Property Set di default in Revit

- *IFCExportAs*: questa funzione non aggiunge classi o informazioni al file IFC ma ne modifica alcune. Con questa funzione è infatti possibile specificare che un determinato parametro venga esportato con una classe ifc definita. È necessario conoscere il nome della classe ifc e identificarla dopo aver creato il parametro.

2.3.2 Network IFC

Durante le simulazioni si è posta particolare attenzione alla presenza dei dati riguardanti le classi dell'involucro edilizio, nonché i singoli componenti, le zone termiche e le proprietà ad esse collegate. Dai test effettuati è emerso che non tutte le classi IFC vengono correttamente esportate dai software BIM, questo comporta una perdita dei dati necessari alle analisi. L'esempio riportato nelle figure (da Fig. 79 a Fig. 81) descrive come un parametro personalizzato può essere aggiunto ed ispezionato successivamente in un file IFC. Questo

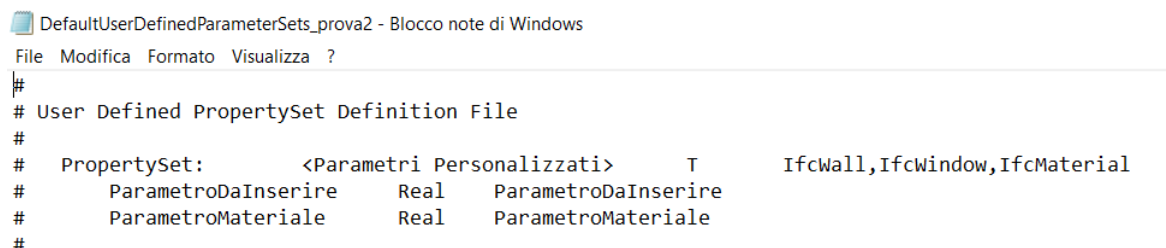


Fig. 79: Scrittura User Defined Property Sets

processo di personalizzazione riguarda parametri personalizzati e parametri già presenti nel software BIM. Il parametro di prova chiamato *ParametroDaInserire* viene correttamente visualizzato nel file IFC, tuttavia se viene seguita la stessa procedura per alcune tipologie di parametri essi non vengono esportati in ogni caso.

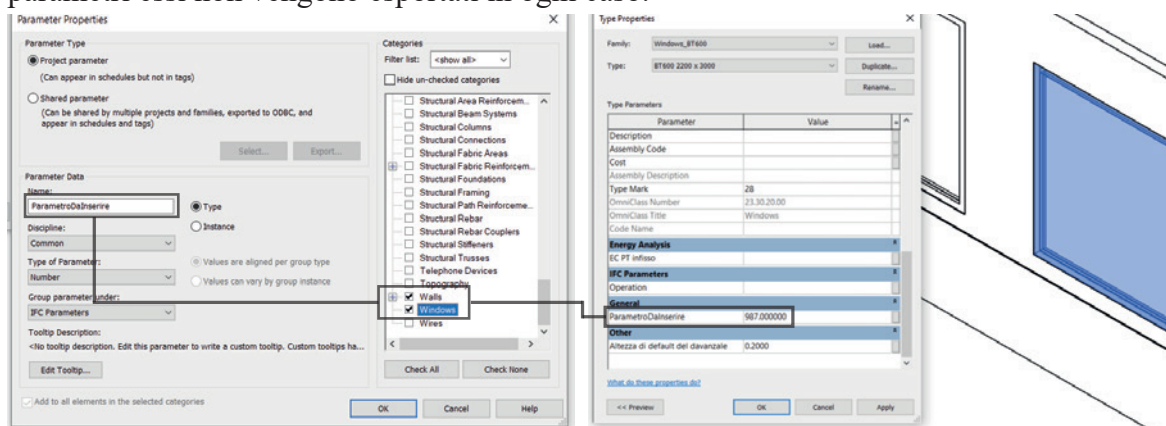


Fig. 80: Creazione parametro Revit

```
#1616= IFCPROPERTYSINGLEVALUE('Heat Transfer Coefficient (U)',$,IFCREAL(0.105562160638337),$);
#1617= IFCPROPERTYSINGLEVALUE('Roughness',$,IFCINTEGER(1),$);
#1618= IFCPROPERTYSINGLEVALUE('Thermal mass',$,IFCREAL(329579.5176),$);
#1619= IFCPROPERTYSINGLEVALUE('Thermal Resistance (R)',$,IFCREAL(9.47309143686502),$);
#1620= IFCPROPERTYSINGLEVALUE('ParametroDaInserire',$,IFCREAL(9999.12345),$);
#1621= IFCPROPERTYSINGLEVALUE('Structural Material',$,IFCLABEL('Metal Stud Layer'),$);
#1622= IFCPROPERTYSINGLEVALUE('Coarse Scale Fill Color',$,IFCINTEGER(0),$);
#1623= IFCPROPERTYSINGLEVALUE('Function',$,IFCIDENTIFIER('Exterior'),$);
#1624= IFCPROPERTYSINGLEVALUE('Width',$,IFCLENGTHMEASURE(460.),$);
#1625= IFCPROPERTYSINGLEVALUE('Wrapping at Ends',$,IFCIDENTIFIER('None'),$);
#1626= IFCPROPERTYSINGLEVALUE('Wrapping at Inserts',$,IFCIDENTIFIER('Do not wrap'),$);
```

Fig. 81: Parametro personalizzato inserito correttamente in file ifc

Un problema rilevante riscontrato durante i test di personalizzazione riguarda la perdita dei dati relativi alle proprietà termiche dei materiali. Tali proprietà sono necessarie alle analisi

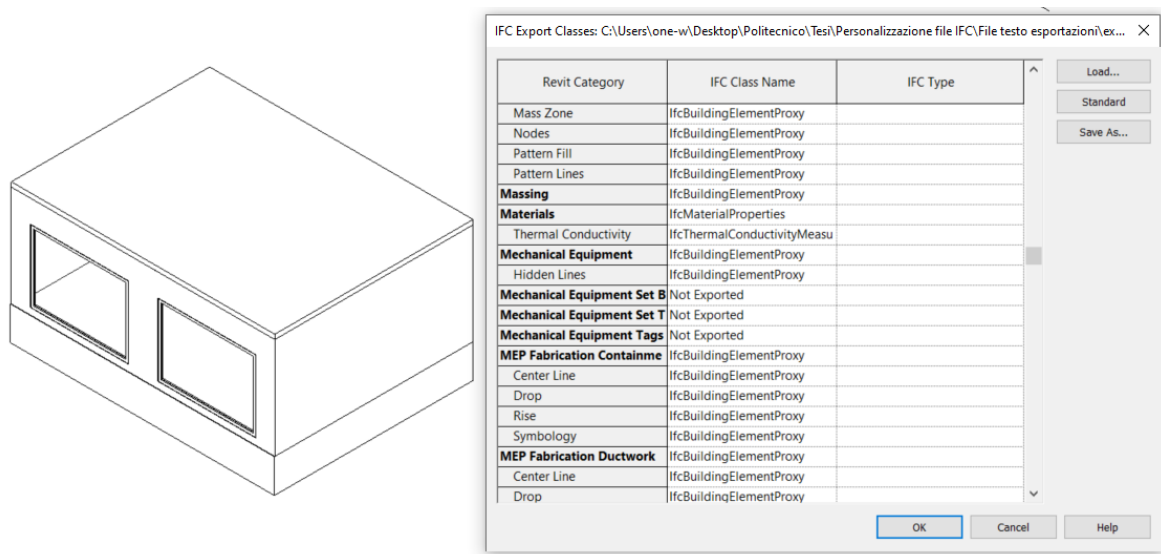


Fig. 82: Mappatura classi ifc in Revit

energetiche e non possono essere trascurate. Lo schema logico in Fig. 83 riporta una parte semplificata del network IFC e l'insieme di classi non trattate durante l'esportazione. L'assenza dei dati sembra dovuta infatti ad un problema relativo alla classe *IfcMaterialProperties* (Fig. 82). Questa classe anche se opportunamente mappata nelle impostazioni del software BIM, allo stato attuale del software, non viene esportata. L'assenza di questa classe si ripercuote su tutte le sotto entità che da essa dipendono. Sono stati effettuati test di esportazione per proprietà dei materiali quali conducibilità termica, il calore specifico, la densità, la permeabilità al vapore e la porosità.

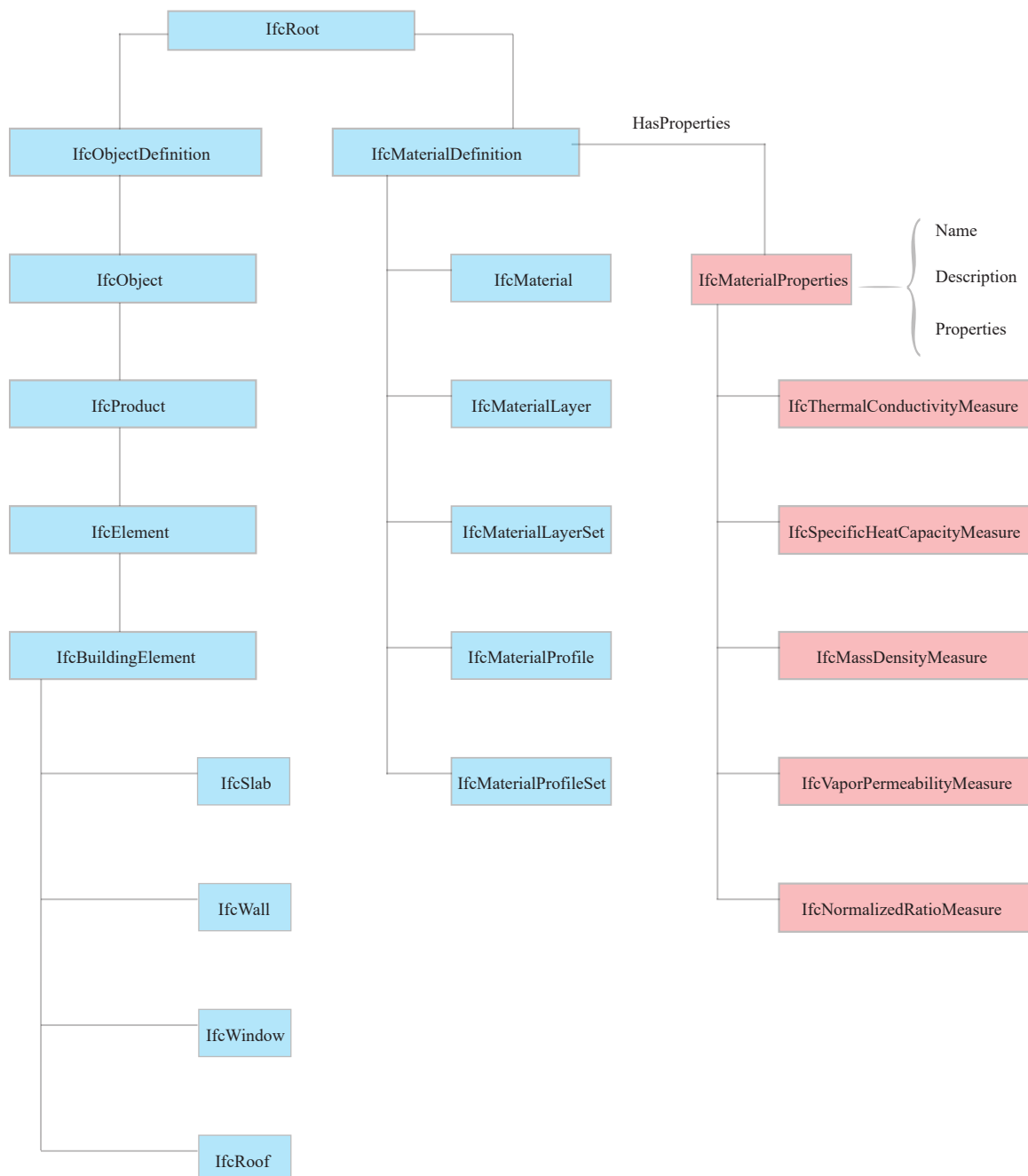


Fig. 83: Problema mappa classi IFC

2.4 VPL e codifica: realizzazione algoritmo

In questa parte dell'elaborato viene descritto il metodo e gli strumenti adoperati al fine di migliorare la qualità delle informazioni esportate dal software BIM tramite il formato IFC. Gli strumenti utilizzati sono principalmente il software VPL Dynamo nella versione 2.0.3 integrato in BIM Revit e delle linee di codice in linguaggio Python integrate a quest'ultimo. Il flusso dati inizia dalla modellazione BIM quindi viene esportato un primo file IFC ottimizzato utilizzando le impostazioni dello stesso software in funzione delle analisi da eseguire. Il file viene quindi controllato con un normale visualizzatore di testo o con specifici tool di controllo (ad esempio *IFC Quick Browser*). Dal controllo effettuato, a discrezione dell'utente, è possibile continuare normalmente importando il file IFC nel tool di valutazione energetica desiderato o personalizzarlo con le appropriate modifiche. Nel caso si voglia procedere con le modifiche è possibile utilizzare tool VPL con l'ausilio di un linguaggio di programmazione compatibile allo scopo di automatizzare l'inserimento delle informazioni richieste, quindi ripetere il controllo del file IFC. (Fig. 84)

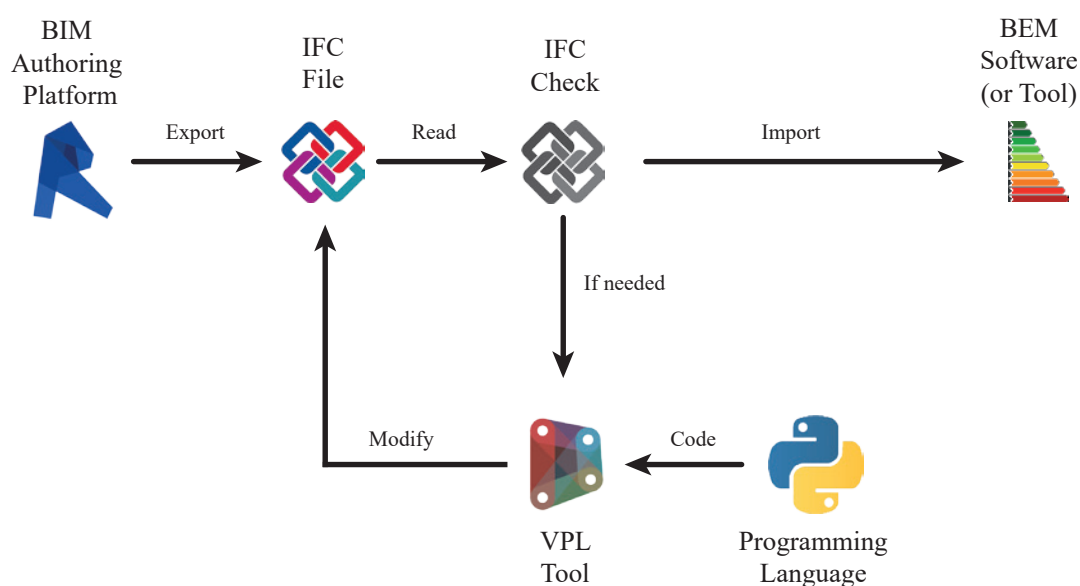


Fig. 84: Workflow BIM to BEM integrato a modifiche da VPL tool

In questo caso è stato realizzato un algoritmo che aggiunge le informazioni e le relative classi mancanti nel file IFC esportato da Revit. Le informazioni da inserire vengono lette dal modello corrente nel software BIM e confrontate con quelle già presenti nel file IFC esportato precedentemente. A questo punto le informazioni mancanti vengono inserite secondo uno schema a matrice ed aggiunte secondo una sintassi rigida in un nuovo file IFC. La formattazione delle nuove informazioni deve seguire gli standard definiti da normativa, in questo modo il nuovo file IFC soddisfa i requisiti di legge ed ha le informazioni necessarie per svolgere valutazioni energetiche servendosi dei software BEM. (Fig. 85)

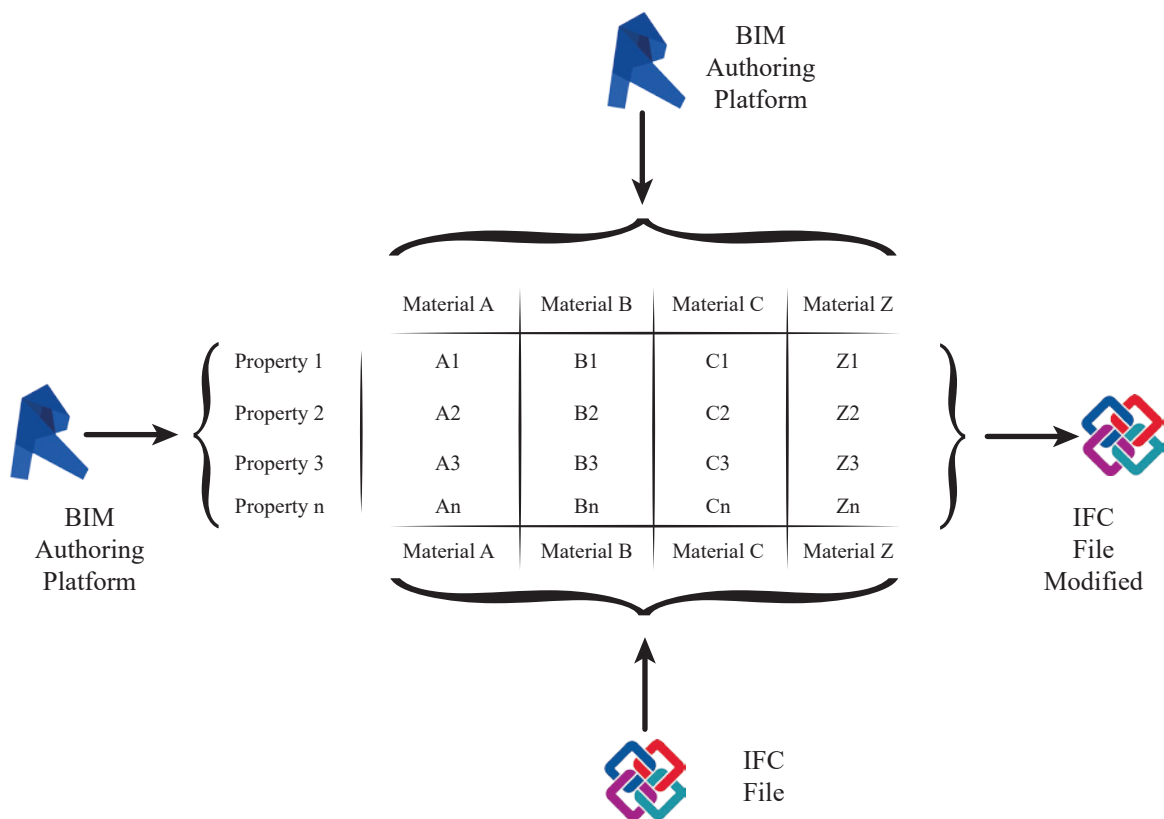


Fig. 85: Matrice funzionamento algoritmo

2.4.1 Obiettivi e funzioni algoritmo

Al fine di garantire la corretta esportazione del file IFC modificato nonché la correttezza delle informazioni inserite, l'algoritmo svolge diverse funzioni in ordine di sequenza:

- Lettura del file IFC esportato da Revit
- Lettura proprietà termiche da modello Revit
- Scrittura stringhe in conformità con lo standard IFC
- Unione file IFC originale e stringhe eseguite

2.4.1.1 Lettura File IFC

La lettura del file IFC avviene inserendo un nodo presente in Dynamo chiamato *File Path* che consente di specificare il percorso di un file di vario tipo, in questo caso il file IFC creato da Revit. Il file IFC viene collegato quindi ad un secondo nodo. Inizialmente è stato utilizzato un nodo scaricabile dal sito *dynamopackages.com* appartenente al pacchetto di nodi *Bakery* chiamato *Text File to List of Strings* (Fig. 86) legge il dato di input e restituisce una lista di stringhe dove gli elementi contenuti sono separati fra di loro ogni qualvolta è stato inserito un invio a capo nel file di input. Il metodo di separazione delle linee di testo del file IFC

può sembrare banale nonché rischioso, tuttavia in base alle informazioni contenute nella normativa di riferimento, ogni linea di testo deve obbligatoriamente essere suddivisa dalla successiva da un rinvio a capo (unico caso in cui è consentito l'utilizzo di questo comando).

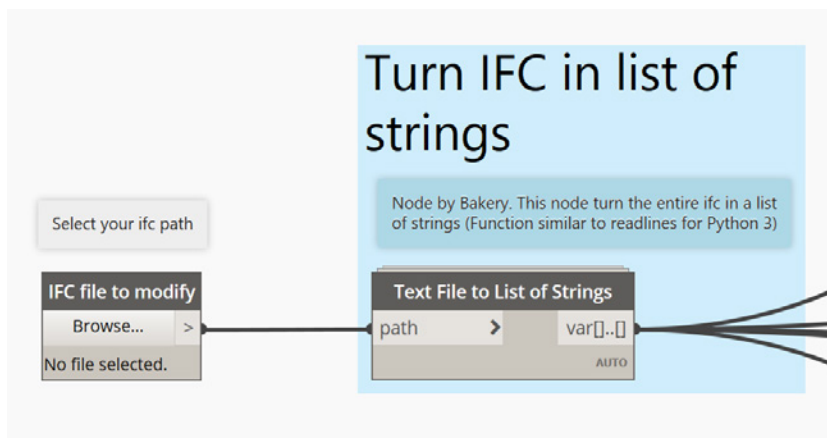


Fig. 86: Nodo algoritmo: lettura file ifc

Successivamente il contenuto del file suddiviso in stringhe contenenti ognuna una *physical line* viene letto nuovamente da un blocco di nodi il quale isola le sole *physical lines* aventi il carattere “#”. In questo modo vengono individuate e restituite le sole *physical lines* contenenti le classi IFC. (Fig. 87)

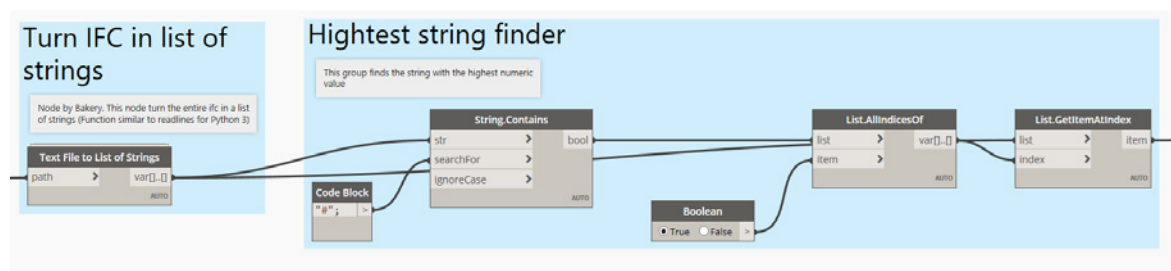


Fig. 87: Nodo algoritmo: lettura file ifc, processo di separazione per caratteri

Lo schema in figura Fig. 87 riporta un primo blocco in cui viene indicato il carattere da ricercare. Successivamente un secondo blocco chiamato *Strings.Contains* che legge il dato di input e restituisce un elenco contenente dei valori *True/False* riferiti ad ogni stringa presente nella lista input. I valori *True* indicano che nella posizione corrispondente si trova una stringa che contiene il valore richiesto. Il nodo successivo ovvero *List.AllIndicesOf* fornisce una lista avente gli indici delle sole stringhe contrassegnate con *True* dal nodo precedente. Il nodo *List.GetItemAtIndex* restituisce una lista avente come contenuto le stringhe corrispondenti agli indici forniti.

Il blocco di nodi successivo è stato scaricato dal sito *dynamopackages.com* e fa parte del pacchetto di nodi *Clockwork*. Il nodo *String.ReplaceMultiple* legge la lista di stringhe inserita come dati di input e sostituisce un carattere a scelta dell'utente ogni qualvolta quest'ultimo

viene letto nel dato di input. (Fig. 88)

In questo caso il carattere “#” è stato sostituito con un carattere nullo “”. In questo modo è stata ottenuta una lista contenente le classi IFC riportanti il numero della *physical line* corrispondente e le relative informazioni riguardanti la classe.

La lista ottenuta viene quindi inviata ad un nodo di tipo *Python Script*. Questa tipologia

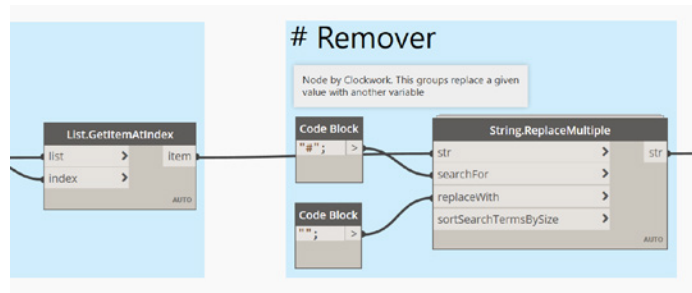


Fig. 88: Nodo algoritmo: lettura file ifc, rimozione caratteri

di nodi inizialmente non possiede alcuna funzione ma permette all’utente di scrivere una funzione personalizzata tramite un codice in linguaggio Python 2. In questo caso il codice redatto individua la posizione del carattere “=” in ogni stringa e ne isola l’indice (*index*), quindi legge il contenuto della stringa dall’indice zero, ovvero il primo carattere della stringa, fermandosi all’indice individuato. In questo modo vengono isolati i valori numerici di ogni stringa relativi al codice identificativo della *physical line* IFC. La seconda parte del codice legge i valori numerici appena ricavati e crea un nuovo valore numerico equivalente al numero maggiore addizionato ad una unità. Il valore ottenuto funge da punto di partenza per la scrittura delle stringhe aggiuntive. Ai fini esplicativi è riportato in Fig. 89 lo schema dei nodi ed il relativo codice Python presente nel *Python Script* visualizzato, si rimanda agli allegati per maggiori informazioni su altri codici realizzati.

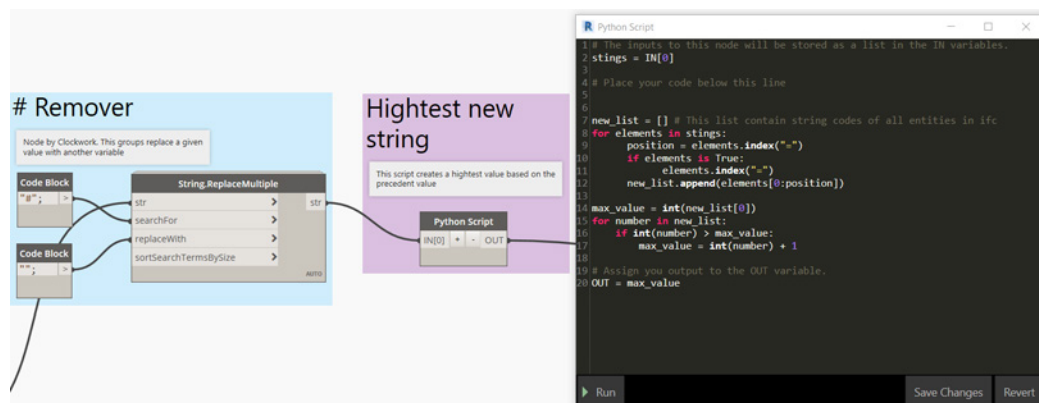


Fig. 89: Nodo algoritmo: lettura file ifc, scrittura valore numerico

La seconda operazione effettuata durante la lettura del file IFC riguarda l’individuazione delle stringhe contenenti la classe *IfcMaterial*. In modo analogo al precedente nodo, il file IFC viene letto e suddiviso in *physical lines*, le quali vengono analizzate e vengono restituite

Obiettivi e funzioni algoritmo

solo le stringhe che contengono i caratteri in sequenza “*IFCMATERIAL*”((Fig. 90)

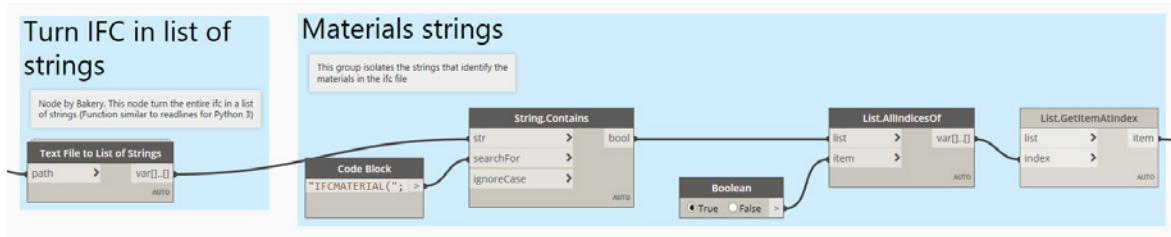


Fig. 90: Nodo algoritmo: lettura file ifc, ricerca classe

In modo analogo al precedente, un altro Python Script viene utilizzato per isolare i valori numerici identificativi delle *physical lines* contenente le classe *IfcMaterial*. (Fig. 91)

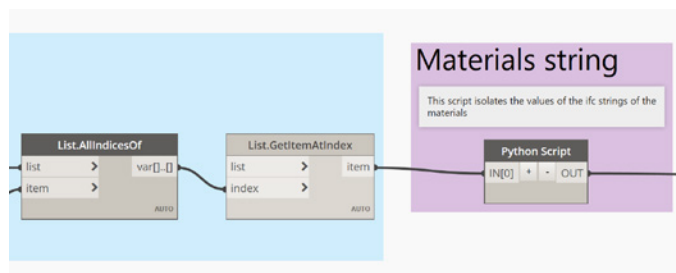


Fig. 91: Nodo algoritmo: lettura file ifc, valori numerici classe

2.4.1.2 Lettura proprietà da modello Revit

La seconda funzione dell'algoritmo riguarda la lettura delle proprietà da inserire nel file IFC dal modello corrente in Revit. Per definire i dati di input dell'utente si è scelto di creare una serie di nodi *Boolean* riguardanti ognuno una data proprietà termica. In questo modo vengono evitati errori di scrittura da parte dell'operatore. Una volta impostato un nodo sul valore *True* questo input viene letto dal successivo *Python script* il quale restituisce una lista contenente il nome della proprietà inserita rinominandola in modo conforme alla stessa presente in Revit (Fig. 92). In questa fase è infatti importante scrivere il nome delle proprietà desiderate in modo corretto, ovvero usare il nome esatto utilizzato dal software BIM, in caso

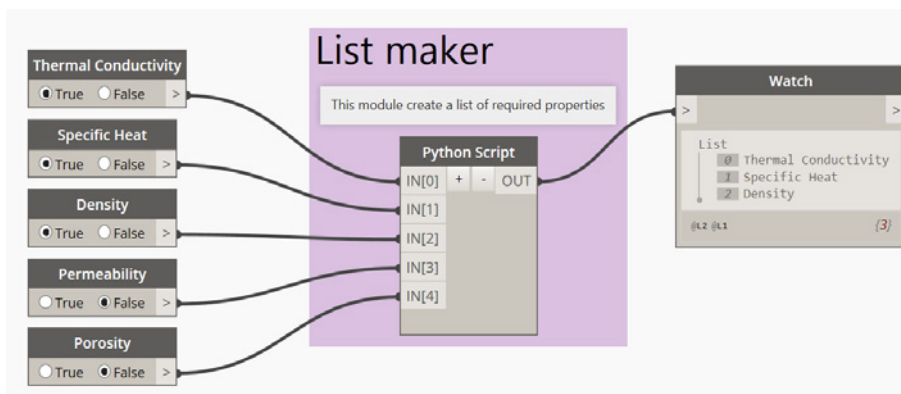


Fig. 92: Nodo algoritmo: lettura proprietà, lista proprietà

contrario esse non verranno visualizzate nel VPL tool.

La lista ottenuta diventa l'input per un nodo successivo il quale estrae le proprietà richieste da una lista contenente la totalità dei parametri riguardanti un elemento.

In parallelo al nodo appena descritto un altro gruppo di nodi ha la funzione di interrogare il modello corrente su Revit ed estrarre alcune specifiche informazioni. (Fig. 93)

Il primo nodo è di tipo *Code Block* ovvero una tipologia di nodi dove è possibile inserire

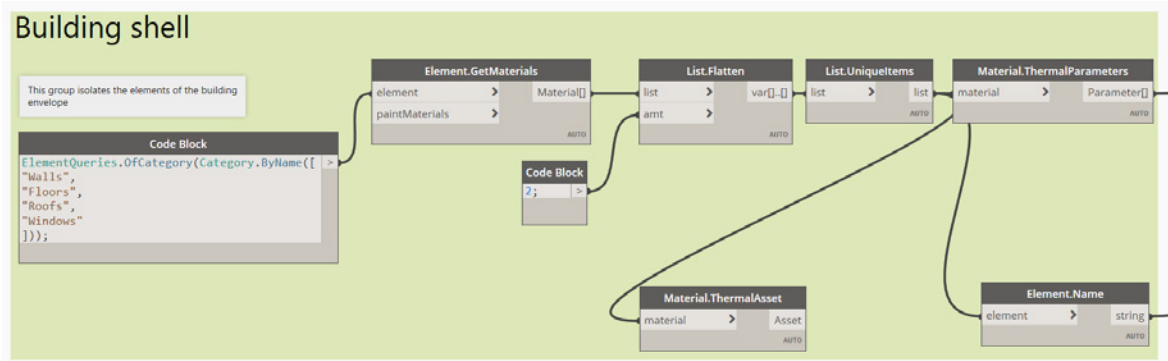


Fig. 93: Nodo algoritmo: lettura proprietà, lettura modello BIM

variabili o funzioni scrivendo manualmente l'informazione desiderata. In questo caso è stata inserita una funzione in linguaggio Python 2, che richiama gli oggetti presenti nelle categorie inserite (in questo caso *Walls*, *Floors*, *Roofs*, *Windows*). Il nodo restituisce quindi quattro liste (una per ogni categoria) contenenti gli elementi presenti nel modello facenti parte la categoria richiesta. Il nodo *Element.Get.Materials*, processa le liste fornite come dato di input e restituisce i materiali che compongono ogni elemento delle liste. Il nodo *List.Flatten* elimina la divisione in più liste e crea una lista unica contenente i materiali utilizzati. Il blocco *List.UniqueItems* quindi realizza una nuova lista inserendo ogni materiale utilizzato una sola volta. In questo modo i materiali utilizzati vengono analizzati una sola volta evitando

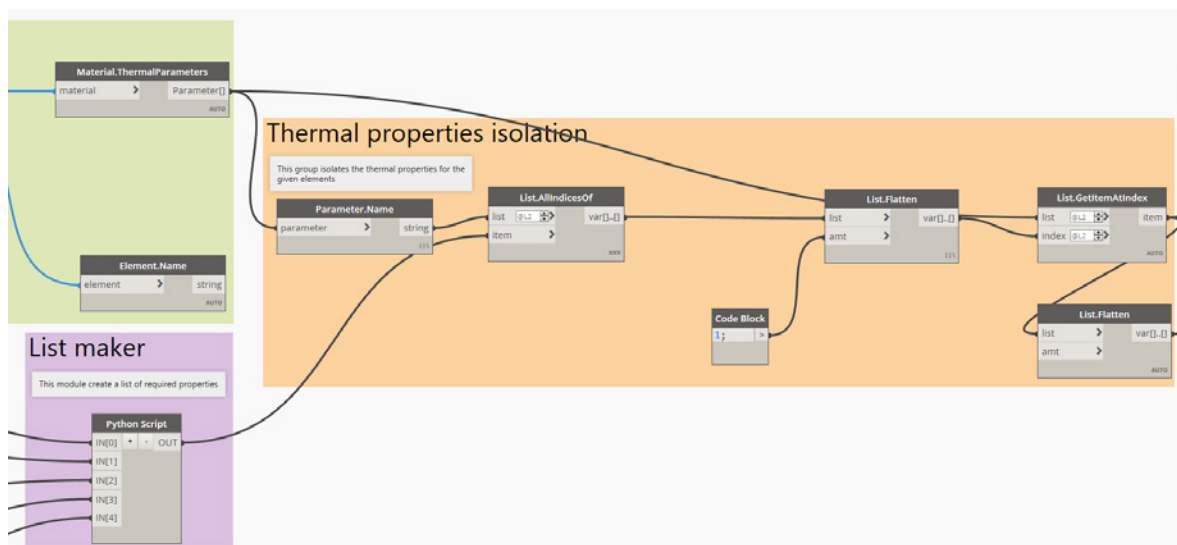


Fig. 94: Nodo algoritmo: lettura proprietà, lista proprietà

sprechi di tempo e memoria della macchina. L'ultimo nodo di questo blocco legge la lista dei materiali fornita come input e, consultando le informazioni presenti nel software BIM, restituisce diverse liste (una per ogni materiale), contenenti le proprietà termiche riguardanti gli elementi delle varie liste.

Il gruppo di nodi successivo (Fig. 94) ha la funzione di estrarre le proprietà richieste dall'utente ed organizzarle in liste. Il primo nodo chiamato *Parameter.Name*, riceve come dato di input le liste delle varie proprietà presenti in ogni materiale e restituisce una lista di stringhe contenenti il nome delle proprietà individuate. Il nodo *List.All.IndicesOF* confronta il nome delle proprietà richieste dall'utente con i nomi delle proprietà presenti nelle liste dei materiali e restituisce una lista contenente gli indici delle stringhe in cui i nomi confrontati risultano essere uguali. Il nodo *List.Flatten* raggruppa i risultati in un'unica lista quindi il nodo *List.GetItemAtIndex* produce una lista contenente le proprietà con posizione specificata dagli indici forniti dove ogni stringa visualizzata nella lista contiene il nome ed il valore della proprietà individuata. Per isolare il solo valore numerico le liste vengono lette dal nodo successivo *Parameter.Value* che isola il valore numerico della proprietà. In questo esempio sono state fornite due liste, il primo caso riporta una lista principale contenente delle sub liste aventi i valori numerici organizzati in ordine per materiale, il secondo caso invece riporta un'unica lista che contiene tutti i valori ricavati.

2.4.1.3 Scrittura stringhe in conformità con lo standard IFC

Questa parte dell'algoritmo è stata interamente codificata utilizzando un nodo di Dynamo del tipo *Python script*. Le ragioni di tal scelta sono legate al fatto che ogni stringa da realizzare, oltre a dover rispettare la sintassi come previsto da normativa ISO, deve includere diverse variabili che seguono andamenti differenti o mutano in base al modello utilizzato. Ottenere un risultato simile servendosi unicamente di un VPL tool avrebbe prodotto una mappa nodi estremamente estesa e difficile da gestire, inoltre alcune funzioni facilmente richiamabili tramite un linguaggio di programmazione, non lo sono in egual modo nella metodologia VPL.

Questo nodo riceve le informazioni ricavate dal software Revit e le informazioni provenienti dal file IFC, quindi itera i valori numerici e crea delle stringhe in conformità allo standard IFC (Fig. 95). In questo nodo le informazioni vengono organizzate secondo una matrice avente i valori numerici disposti per materiale e proprietà termica selezionata (Fig. 85). La prima tipologia di stringa ha dunque un nuovo valore numerico preceduto da un "#", quindi il testo in conformità con lo standard IFC riportante la classe della proprietà termica selezionata, infine dei valori numerici tra parentesi che riportano il valore della proprietà.

Si riporta una stringa tipo a scopo esplicativo:

- #9999= IFCPROPERTYSINGLEVALUE('NomeProprietà',\$,ClasseIfcProprietà(valorenumerico),\$);

Dopo la prima tipologia di stringa, viene quindi iterata una seconda tipologia di stringhe riportante la classe *IfcMaterialProperties*. In questo modo viene creato un collegamento fra i singoli valori delle varie proprietà termiche ed i materiali a cui fanno riferimento. In questa parte dello script viene letta la dimensione della matrice di riferimento generata per la tipologia di stringhe precedente che quindi viene suddivisa ad intervalli regolari in base al numero delle proprietà termiche richieste. In questo modo gli argomenti di ogni intervallo vengono inseriti in ogni stringa alla quale viene infine aggiunto il codice numerico identificativo del materiale presente nel file IFC.

Stringa esempio

- #9999= IFCMATERIALPROPERTIES('Pset_MaterialThermal',\$,(\$argomento1,\$argomento2),#ValoreNumericoMateriale);

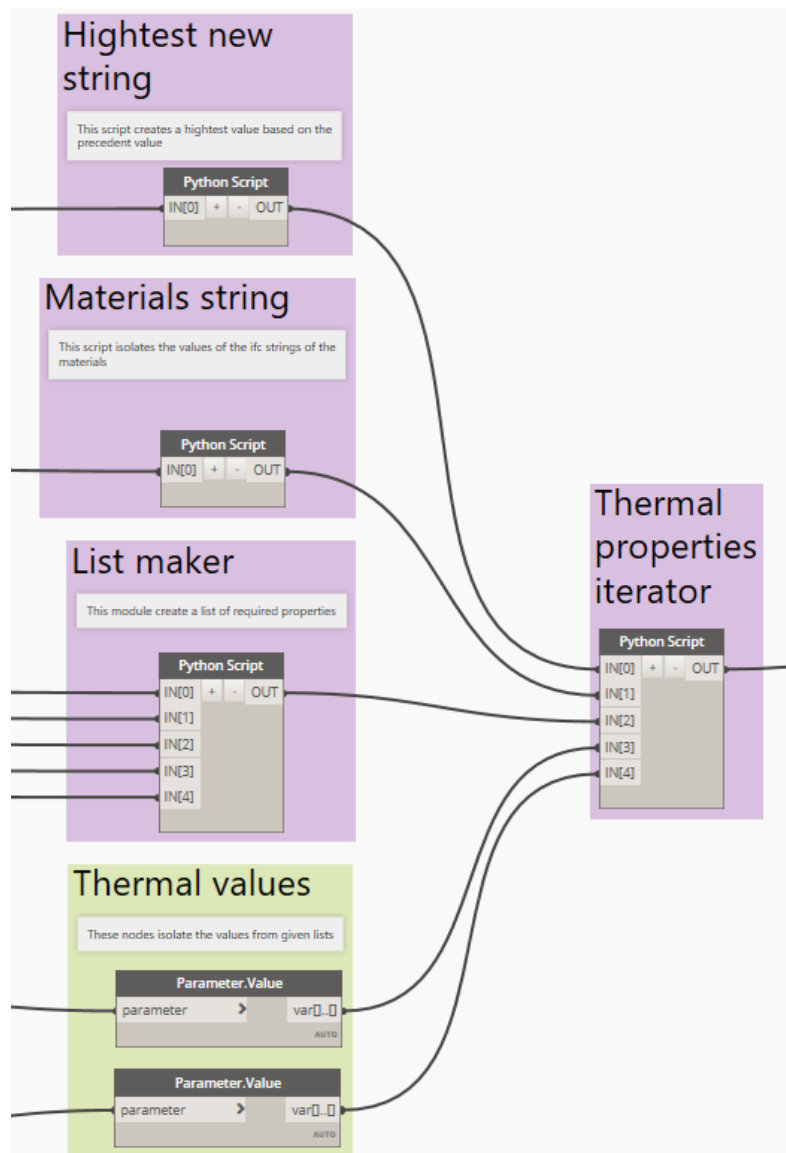


Fig. 95: Nodo algoritmo: scrittura proprietà, iteratore

2.4.1.4 Scrittura nuovo file IFC

L'ultima parte dello script unisce le nuove stringhe iterate con il contenuto del file IFC originale. In questa fase occorre inserire le stringhe nella posizione corretta o il file IFC non sarà letto da alcun software. I file IFC sono organizzati secondo uno schema a rete, questa organizzazione consente al file di leggere informazioni presenti in tutto il file a prescindere dall'ordine in cui queste vengono inserite. Basandosi su questo concetto le nuove stringhe saranno inserite alla fine del file ma prima delle stringhe di chiusura. La normativa di riferimento stabilisce infatti che le ultime tre stringhe devono obbligatoriamente riportare la scritta “ENDSEC;” succeduta da una stringa vuota per avere poi una terza stringa riportante “END-ISO-10-303-2I” in riferimento alla normativa in vigore.

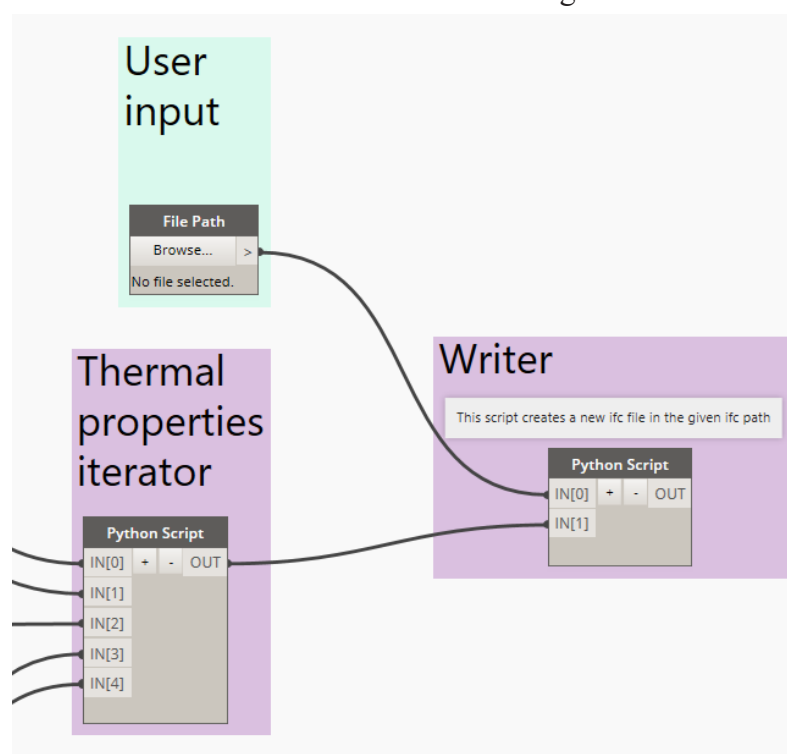


Fig. 96: Nodo algoritmo: scrittura nuovo file ifc

Anche questo nodo è stato realizzato con un *Python Script* (Fig. 96). Per la realizzazione di questo script sono state importate nel codice realizzato, delle funzioni presenti nella libreria di *IronPython* versione 2.7. In quest'ultima parte, il file contenuto nel percorso indicato dall'utente in fase iniziale viene letto e suddiviso in due parti. La prima parte del file comprende l'intestazione ed il corpo del file ifc (la parte contenente le classi), mentre la seconda parte riporta le ultime tre stringhe previste da normativa. Fra queste due parti viene collocato il contenuto dell'output prodotto dal nodo iteratore precedentemente descritto. Una volta unite le informazioni viene in automatico creato un nuovo file nella stessa destinazione del file originale.

Il nuovo file appena creato riporta il nome originale dell'ifc seguito dall'acronimo *MPI* (*Material Properties Iterator*). In questo modo si ha a disposizione il file IFC originale che non viene in alcun modo modificato ed il file ifc contenente le informazioni mancanti.

2.4.2 Miglioramento algoritmo

Ad algoritmo ultimato sono stati effettuati diversi test che hanno portato alla modifica di quest'ultimo al fine velocizzare le operazioni di lettura e scrittura dei dati. Inoltre sono stati inseriti dei nodi precauzionali alla sola funzione di evitare malfunzionamenti dell'algoritmo. Infine è stata prevista l'impostazione di *Dynamo Player* che consente all'utente di utilizzare l'algoritmo direttamente dal software BIM evitando i passaggi previsti per Dynamo.

2.4.2.1 Miglioramento prestazioni

I primi test di funzionamento sono stati effettuati utilizzando il modello BestTest. Durante queste prove l'algoritmo ha svolto le funzioni richieste con esito positivo ed è stato generato il nuovo file IFC. L'operazione è stata conclusa in circa 10 secondi. Una volta caricato il nuovo file nel software BEM non si riscontrano irregolarità e le informazioni inserite vengono correttamente visualizzate. I test sul modello FCA invece non sono stati altrettanto soddisfacenti. Una volta lanciato l'algoritmo, la macchina inizia ad iterare le informazioni, tuttavia il processo non giunge a conclusione. Il problema molto probabilmente è legato all'esaurimento della memoria RAM che una volta occupata interamente, smette di funzionare correttamente bloccando la macchina. Questo inconveniente riscontrato nel test sul modello FCA non è anomalo, data la differenza di dimensioni dei due file IFC.

Un file ifc realizzato da un modello semplice come nel caso BestTest, si sviluppa in circa 2500 *physical lines*. La

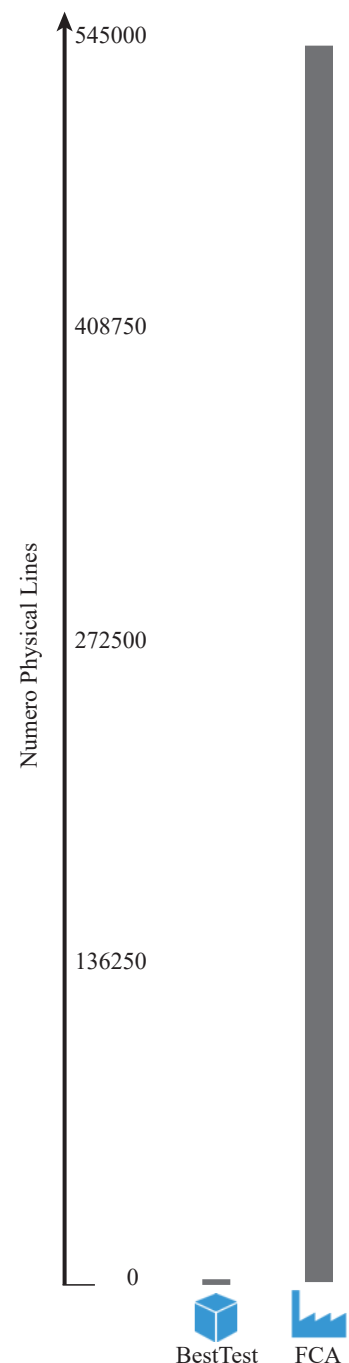


Fig. 97: Confronto dimensioni modelli

Miglioramento algoritmo

stessa mappatura utilizzata ed applicata al modello FCA produce un file di circa 545000 *physical lines*, dimensione che rende impossibile l'iterazione con questa configurazione dell'algoritmo. Per risolvere questo problema è stato necessario effettuare delle modifiche all'algoritmo. Le modifiche riguardano la parte adibita alla lettura ed all'estrazione delle informazioni dal file IFC importato. La nuova configurazione utilizza un solo nodo *Python Script* (Fig. 98) il quale legge il file ed estrae le informazioni necessarie durante la lettura. In questo modo la macchina legge e processa le informazioni in modo estremamente veloce e non occupa memoria RAM aggiuntiva ad ogni iterazione. A modifiche ultimate il modello BestTest è stato processato istantaneamente mentre il modello FCA è stato letto e processato in circa 3 secondi.

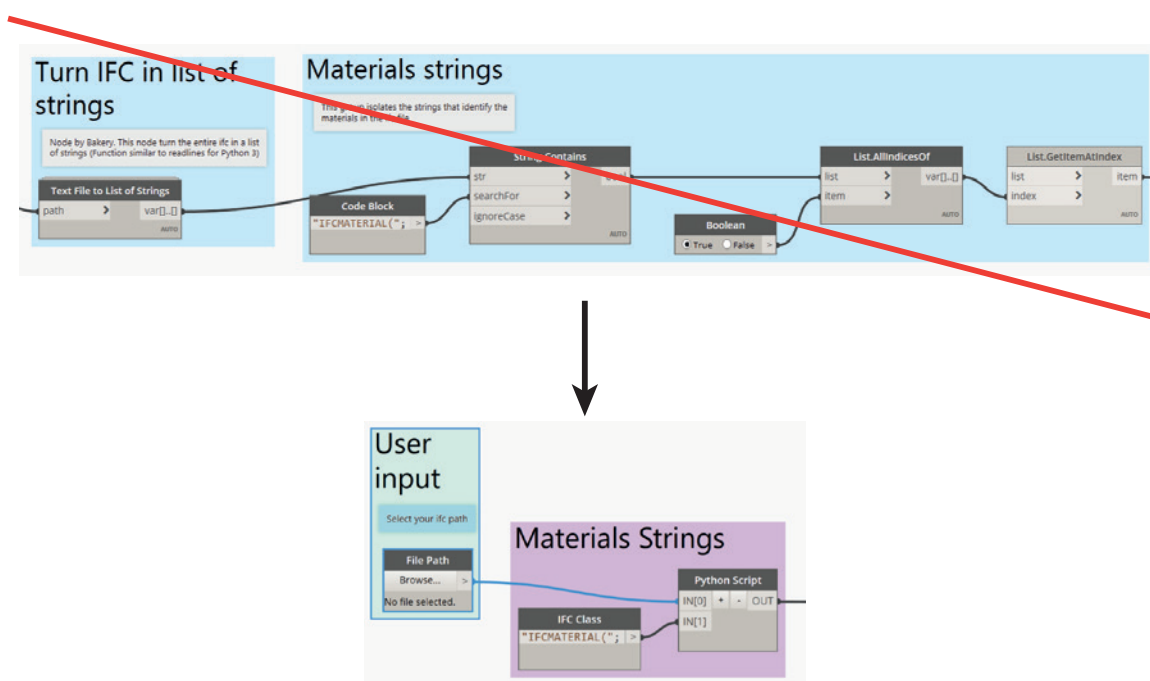


Fig. 98: Nodo algoritmo: sostituzione script classe materiali

2.4.2.2 Debugging algoritmo

Durante i test successivi sono state riscontrate alcune irregolarità nel funzionamento dell'algoritmo, quindi sono stati aggiunti dei nodi al solo scopo di evitare tali malfunzionamenti qualora si presentassero.

La prima modifica riguarda l'inserimento di un nodo "filtro" che legge i materiali presenti nel file IFC importato e li confronta con i materiali presenti nel software BIM (Fig. 99). In alcuni casi, il nome del materiale inserito nel file IFC dal software BIM, non è corretto e al suo posto vengono visualizzati una serie di caratteri illeggibili. Questa problematica è

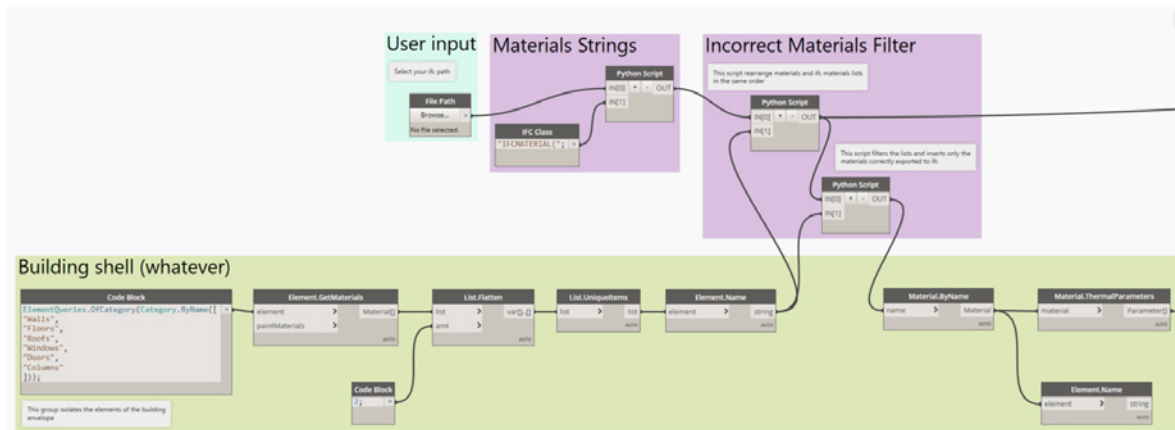


Fig. 99: Nodo algoritmo: nodi filtro materiali legata al fatto che il formato IFC supporta solo caratteri *ASCII*. Se durante la creazione di un nuovo materiale, nel software BIM viene inserito un carattere non compreso in questa tipologia, come ad esempio le vocali accentate, tale carattere non sarà riscritto una volta creato il file IFC. Al fine di evitare questo tipo di malfunzionamento il nodo inserito legge e confronta i materiali presenti, quindi restituisce una lista dei materiali che soddisfano la corrispondenza con il loro equivalente nel software BIM. La lista generata segue lo stesso ordine di inserimento eseguito nel file IFC. Script completo allegato al documento.

Un altro problema riscontrato riguarda il reperimento delle proprietà desiderate dall'utente. In alcuni casi non tutti i materiali presenti dispongono della totalità delle proprietà termiche richieste. Questo script inserisce un valore nullo (*None*) ogni qualvolta viene richiesta una proprietà non ancora inserita nel software BIM. Successivamente la stringa contenente il valore nullo viene eliminata e quindi non inserita nel nuovo file IFC.

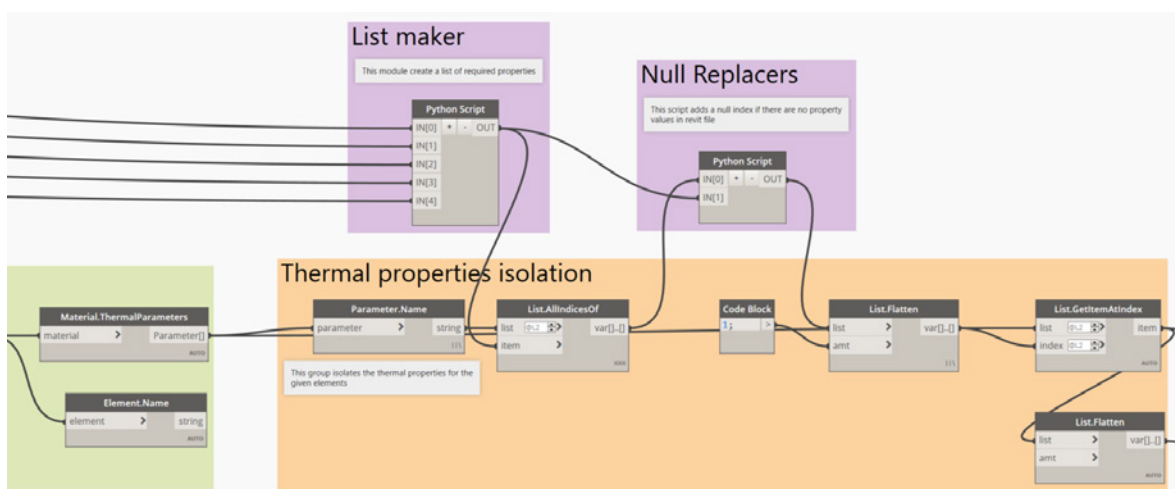


Fig. 100: Nodo algoritmo: nodi debugging materiali

2.4.2.3 Interfaccia Dynamo Player

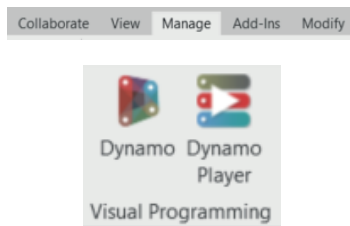


Fig. 102: Nodo algoritmo: icona Dynamo Player

Una volta ultimato l'algoritmo, è stata configurata l'interfaccia *Dynamo Player* al fine di un utilizzo rapido delle funzioni inserite. In questo modo anche gli utenti meno esperti possono usufruire di algoritmi preimpostati non dovendo aprire ed impostare alcun tool VPL. La configurazione avviene in Dynamo nel quale occorre definire dei dati di input che saranno visualizzati nel software BIM e dei dati di output che fungono da riferimento automatico utilizzato dal software per stabilire le operazioni da eseguire (Fig. 101). Una volta impostata l'interfaccia, per accedervi occorre utilizzare la finestra *Manage > Dynamo Player* e selezionare l'algoritmo desiderato. (Fig. 102; Fig. 103)

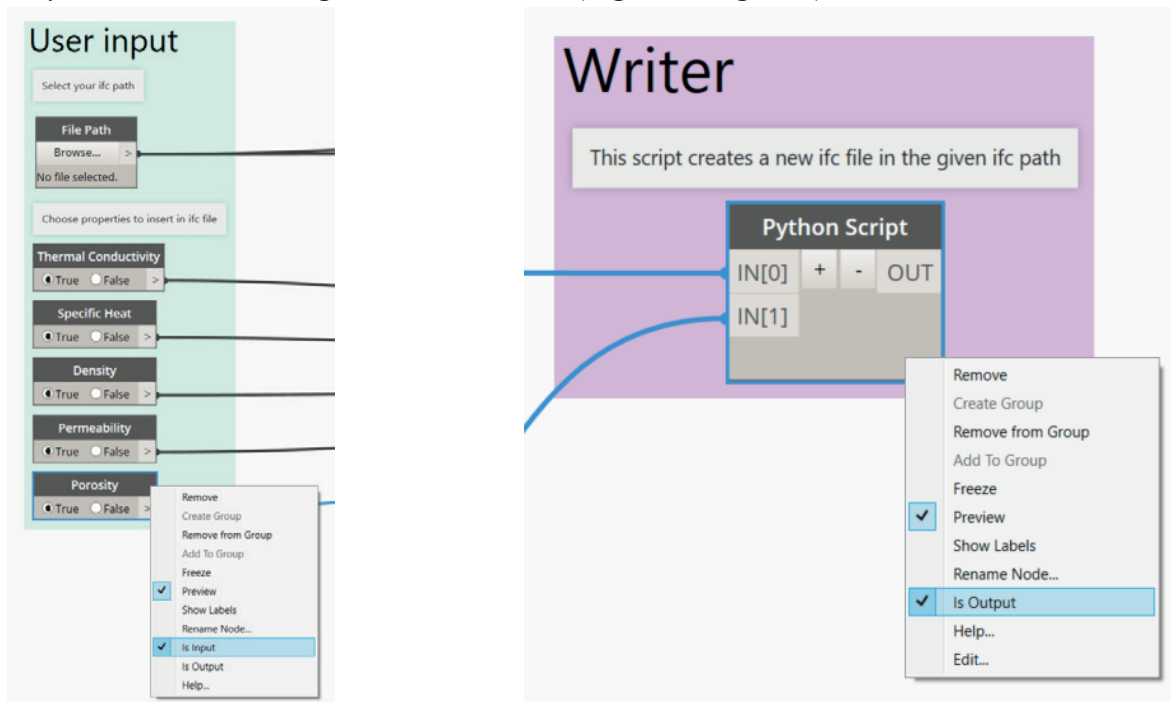


Fig. 101: Nodo algoritmo: Input & Output Dynamo Player

Una volta selezionato l'algoritmo è possibile definire i dati di input in base alle proprietà termiche richieste e il file IFC di riferimento, quindi è possibile lanciare l'algoritmo.

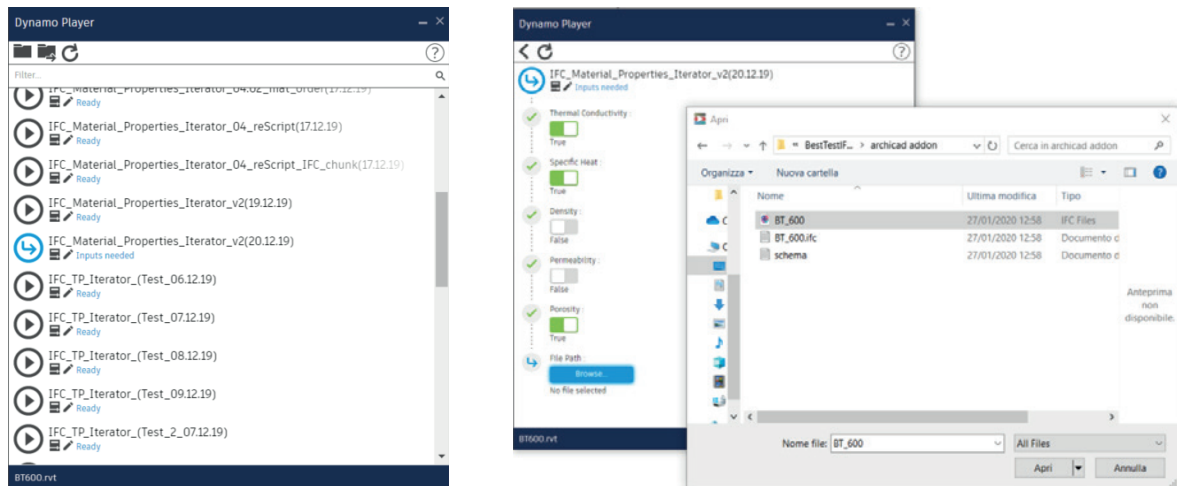


Fig. 103: Nodo algoritmo: Definizione input e creazione nuovo file

3 Risultati e Criticità

3.1 Risultati

3.1.1 Interoperabilità BIM to BEM

Il lavoro svolto ha prodotto come risultato un algoritmo che legge informazioni contemporaneamente da un modello BIM e da un file IFC. Le informazioni vengono inserite in un nuovo file IFC che replica i dati letti dal file originale aggiungendo le informazioni ricavate dal modello BIM. Nella configurazione attuale l'algoritmo itera informazioni relative alle proprietà termiche dei materiali (conducibilità termica, calore specifico, densità, permeabilità al vapore, porosità). Le informazioni inserite sono state scelte al fine di testare l'interoperabilità con software legati alle valutazioni di efficienza energetica degli edifici. Il file IFC ottenuto è stato quindi importato in un software di valutazione energetica certificato dal CTI. È possibile notare come le informazioni aggiunte per mezzo dell'algoritmo siano state correttamente lette e visualizzate (Fig. 104). Nel caso del modello FCA sono state iterate ed aggiunte al file IFC originale oltre 150 proprietà termiche relative a 32 materiali da costruzione. Il processo di scrittura è avvenuto in circa 4 secondi e non sono stati riscontrati errori. L'operazione di modifica del file IFC originale può sembrare superflua

File IFC originale

• Adduttanza interna (flusso orizzontale)	0	7,690	0,000
• Pannello in gesso	16	1,000	1,000
• Pannello in gesso	16	1,000	1,000
• Strato montante in metallo	92	1,000	1,000
• Pannello in gesso	16	1,000	1,000
• Adduttanza esterna (flusso orizzontale)	0	25,000	1,000

↓

File IFC modificato

• Adduttanza interna (flusso orizzontale)	0	7,690	0,000
• Pannello in gesso	16	0,650	0,840
• Pannello in gesso	16	0,650	0,840
• Strato montante in metallo	92	0,333	1,004
• Pannello in gesso	16	0,650	0,840
• Adduttanza esterna (flusso orizzontale)	0	25,000	1,000

Fig. 104: Confronto IFC dopo modifiche

per piccoli modelli, tuttavia risulta essere indispensabile per modelli di grandi dimensioni. File così ricchi di informazioni, come nel caso del file IFC di FCA rendono la lettura e la modifica manuale operazioni estremamente ostiche da affrontare. Servirsi di mezzi di automazione come in questo caso, dà la possibilità ai professionisti di risparmiare ore, se non giorni di lavoro, che sarebbero dovuti essere impiegati nello svolgimento ripetitivo di azioni elementari, quali la scrittura manuale delle informazioni mancanti o l'immissione di esse nel nuovo software interrompendo così il flusso dati automatizzato ed aumentando esponenzialmente la possibilità di errore.

3.1.2 Applicazioni algoritmo

In questo caso l'algoritmo realizzato è stato utilizzato per inserire delle proprietà mancanti nel file IFC originale relative ai materiali da costruzione, necessarie per una corretta valutazione energetica degli edifici.

Tuttavia è possibile utilizzare lo stesso algoritmo, con le dovute modifiche, per l'esportazione di altri dati come ad esempio le proprietà strutturali dei materiali o altre informazioni appartenenti ad una categoria totalmente differente. Conoscendo lo schema IFC utilizzato è infatti possibile ricostruire i collegamenti fra la classi IFC mancanti in modo da riprodurre la totalità delle informazioni necessarie, quali esse siano,

salvo una corretta scrittura secondo quanto previsto dallo standard ISO.

La diffusione del linguaggio di programmazione Python ha reso possibile il suo utilizzo in diversi software del settore. Sono molteplici i software riguardanti modellazione digitale, realtà virtuale e rendering che utilizzano il linguaggio Python per consentire ai propri utenti di codificare delle funzioni personalizzate in modo da velocizzare e migliorare la qualità del proprio lavoro. Un algoritmo simile può essere quindi realizzato su tools appartenenti ad altre *software house* ed il risultato raggiunto sarebbe il medesimo, in alternativa una configurazione simile potrebbe essere utilizzata come punto di partenza per la codifica di un altro algoritmo.

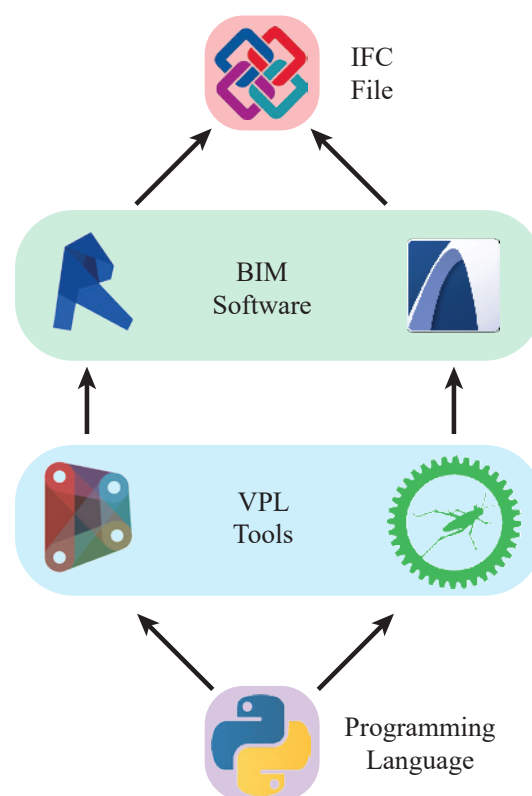


Fig. 105: Esempio differenti workflow

3.2 Criticità

3.2.1 Flusso dati

Il passaggio di informazioni dal software BIM ai software di valutazione energetica al momento avviene in diversi modi. Da una parte si ha il modello energetico, ovvero un modello digitale semplificato che riassume le informazioni necessarie alle valutazioni energetiche, il quale viene esportato generalmente tramite il formato .gbXML. Vi sono alcuni tools e *plugin* che, in alcuni casi specifici, migliorano la qualità del flusso. Tuttavia è anche vero che esportando il solo modello energetico dal software BIM applichiamo a tutti gli effetti una divisione dei dati. Per certi versi avere un modello digitale semplice, di facile lettura e comprendente gli elementi necessari alla valutazione da svolgere può sembrare un vantaggio, ma in realtà quello che può essere inteso come un punto di forza, diventa al contempo una criticità. La semplificazione forzata dei dati, in special modo per modelli complessi, può generare degli errori di valutazione da parte degli utenti, tipici del tradizionale processo di progettazione. Invero quando si lavora su un modello semplificato, non si ha la totale visione dell'edificio (ed anche del modello digitale) e ciò potrebbe condurre a delle scelte che risultino ottimali dal punto di vista energetico ma in conflitto con la condizione reale e integrale dell'edificio e nei casi più gravi, la semplificazione delle informazioni può portare ad un contrasto normativo. Il processo alternativo vede l'utilizzo del formato IFC come mezzo per lo scambio di informazioni BIM to BEM. In questo caso il formato di scambio risulta essere personalizzabile per quanto riguarda le informazioni da inserire, tuttavia questa assenza di confini rende il processo di interoperabilità rischioso. Se da un lato non avviene una semplificazione delle informazioni del modello BIM (a meno che l'operatore non la attui di propria spontanea volontà) d'altro canto le informazioni inserite spesso vengono lette in modo singolare in base al software di destinazione. Si necessita quindi di un controllo minuzioso delle informazioni inserite da parte dell'operatore, a condizione che esse vengano correttamente importate. Un'ulteriore criticità riguarda il confronto dati tra il modello *BIM As Built* ed i dati riguardanti l'edificio reale. Nell'ipotesi di un flusso dati BIM to BEM perfetto, i dati inseriti in un modello BIM (il quale dispone del 100% dei dati riguardanti un edificio realmente costruito) vengono importati in un software di simulazione energetica che svolge analisi (considerando la tipologia di analisi più accurata prevista da normativa) e restituisce dei dati che dovrebbero replicare perfettamente il comportamento dell'edificio reale. In pratica, anche riuscendo a realizzare un flusso dati a regola d'arte, il risultato dell'output non sarà altrettanto preciso. Ciò è dovuto al fatto che il comportamento prestazionale di ogni edificio è fortemente influenzato dall'utenza. Per quanto ci si possa sforzare di prevedere

il comportamento degli attori che usufruiscono di un edificio, la valutazione effettuata potrà simularne i comportamenti ma in ogni caso il risultato ottenuto sarà simile ma non coincidente perfettamente con la realtà.

3.2.2 Industry Foundation Classes

IFC è attualmente utilizzato come mezzo per la condivisione dei dati fra diversi software. Il flusso dati avviene per mezzo di file realizzati secondo uno standard ISO che stabilisce come tali file debbano essere utilizzati. Teoricamente ogni software che importa ed esporta informazioni per mezzo di questo standard dovrebbe quindi tradurre le informazioni presenti all'interno del proprio ambiente dati in una "lingua" (quale il formato IFC) comprensibile a tutti gli altri software. Di conseguenza un software che importa informazioni ritraduce il contenuto del file IFC in conformità al proprio ambiente dati e restituisce all'utente l'equivalente delle informazioni originali. Allo stato attuale questo processo non è così perfetto, anzi è colmo di imperfezioni e carenze. Ciò è dovuto al fatto che ogni software adatta le proprie informazioni al formato IFC in modo singolare basandosi su quale sia il modo migliore per tradurre i propri dati. Questo è possibile a causa della complessa rete di collegamenti che caratterizzano le classi IFC, difatti ogni elemento dispone di molteplici entità che lo descrivono ed ognuna di esse dispone di diverse tipologie di collegamenti. Una traduzione ottimale delle informazioni dovrebbe avere una corrispondenza diretta fra informazione nel software BIM ed informazione inserita nel file IFC ma questo non avviene e per compensare le differenze fra i linguaggi i software applicano delle semplificazioni delle informazioni che spesso generano un'inevitabile perdita di dati.

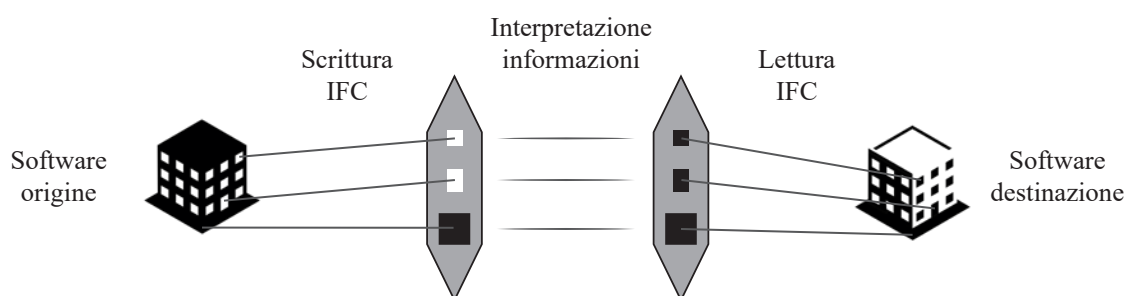


Fig. 106: Schema interpretazione dati IFC

4 Conclusioni e Sviluppi Futuri

La metodologia BIM sta conducendo ad una rivoluzione del settore AEC. In un nuovo scenario che vede un parco edilizio sempre più performante ed una mole di dati da gestire mai riscontrata in passato, avvalersi di strumenti che garantiscano la piena collaborazione fra le parti è divenuto una necessità. Il fulcro della metodologia, ovvero il modello BIM, racchiude in sé l'intero processo e la totalità delle informazioni di un edificio. Modellare un edificio oggi non comprende solo la definizione delle forme geometriche, ma è più uno strutturare le informazioni in modo grafico e testuale. Inoltre quanto modellato dovrà essere accessibile a tutte figure professionali in modo da non tralasciare nessuno degli aspetti previsti dalla progettazione. Ogni singola disciplina diviene parte di un unico grande processo integrale che inizia con l'istituzione di un progetto e termina con la demolizione dell'edificio costruito. In tal senso la metodologia BIM ha fatto, e continua a fare tuttora, passi da gigante per il miglioramento della qualità degli strumenti interoperabili e la gestione delle informazioni. Il formato IFC, che attualmente rappresenta il mezzo il quale le informazioni vengono condivise fra i diversi software, dal 1994 ad ora, è stato svariate volte implementato ed è tuttora in fase di sviluppo. Il tempo porterà sicuramente ad un continuo miglioramento dei software BIM grazie al quale sarà implementata la capacità di interoperabilità fra le parti. In parallelo il formato IFC continuerà a rilasciare schemi sempre più dettagliati che si presteranno, in un futuro prossimo, alla gestione della totalità delle informazioni in modo dinamico, adattandosi perfettamente ad ogni realtà progettuale. L'ultima barriera che verrà del tutto infranta sarà il confine tra modello digitale ed edificio reale. L'utilizzo di sensori installati nell'edificio invieranno dati sempre più precisi ad un modello digitale che sarà utilizzato dagli operatori o da algoritmi per gestire al meglio l'edificio negli usi prestabiliti.

5 Bibliografia e sitografia

- [1] Acca Software, «ifc cos'è e quali sono i vantaggi,» Acca Software, [Online]. Available: http://biblus.acca.it/focus/ifc-cose-e-quali-sono-i-vantaggi/#Il_BIM. [Consultato il giorno 10 01 2020].
- [2] N. Furcolo, «Le 7 dimensioni del BIM,» Biblus.Acca, [Online]. Available: <http://biblus.acca.it/focus/le-7-dimensioni-del-bim/>. [Consultato il giorno 29 12 2019].
- [3] BiblusBIM, «Livelli di maturità BIM,» Acca Software, [Online]. Available: <http://bim.acca.it/livelli-di-maturita-bim/>. [Consultato il giorno 10 01 2020].
- [4] B. Lack and S. Butler, “Bridging BIM and BEM: the path forward to more efficient building design and operations,” 25 04 2019. [Online]. Available: https://go.schneider-electric.com/WW_201903_BIM-BEM-white-paper_SF-LP.html?source=Advertising-Online&sDetail=BIM-BEM-WP_WW. [Accessed 14 01 2020].
- [5] H. Bourg, «blog.archicad,» 02 08 2019. [Online]. Available: <https://blog.archicad.it/tutto-quello-che-hai-sempre-voluto-sapere-sulla-iso-19650>. [Consultato il giorno 29 12 2019].
- [6] Ministero delle infrastrutture e dei trasporti, «Decreto Ministro MIT n. 560 del 1.12.2017,» Ministero delle infrastrutture e dei trasporti, 07 12 2017. [Online]. Available: <http://www.mit.gov.it/sites/default/files/media/normativa/2018-01/Decreto%20Ministro%20MIT%20n.%20560%20del%201.12.2017.pdf>. [Consultato il giorno 10 01 2020].
- [7] Edil Tecnico, «UNI 11337 Standard italiani BIM,» Edil Tecnico, 21 05 2018. [Online]. Available: <https://www.ediltecnico.it/63404/uni-11337-standard-italiani-bim/>. [Consultato il giorno 19 01 2020].
- [8] BIM Idea, «UNI 11337,» BIM Idea, 2020. [Online]. Available: <https://www.bimidea.it/uni11337/>. [Consultato il giorno 10 01 2020].
- [9] Rete Ambiente, «Normativa energia,» Rete Ambiente Srl, 01 2020. [Online]. Available: <http://www.reteambiente.it/normativa/energia/info/quadro/>. [Consultato il giorno 07 01 2020].
- [10] Gazzetta ufficiale dell'Unione Europea, «EUR-Lex,» Unione Europea, 19 06 2018. [Online]. Available: <https://eur-lex.europa.eu/legal-content/IT/TXT/PDF/?uri=CELEX:32018L0844&from=EN>. [Consultato il giorno 07 01 2020].
- [11] S. Ciciriello, «Nuova direttiva UE 2018-844,» 11 09 2018. [Online]. Available: <https://www.ediltecnico.it/65215/nuova-direttiva-ue-2018-844-efficienza-energetica-cosa-cambia/>. [Consultato il giorno 14 12 2019].

- [12] S. Laurent, «Il nuovo metodo orario dinamico per il calcolo dei fabbisogni dell'edificio nel nuovo pacchetto EPDB,» Ingenio, 14 11 2019. [Online]. Available: <https://www.ingenio-web.it/23568-il-nuovo-metodo-orario-dinamico-per-il-calcolo-dei-fabbisogni-delledificio-nel-nuovo-pacchetto-epbd>. [Consultato il giorno 09 01 2020].
- [13] L. Sarto e C. Castagna, «Cambia la norma energetica di riferimento,» Logical Soft, 07 09 2017. [Online]. Available: <https://www.logical.it/focus/cambia-la-norma-energetica-di-riferimento>. [Consultato il giorno 02 01 2020].
- [14] R. Nidasio, «Efficienza Energetica degli edifici. Applicabilità delle nuove EN,» Ente Italiano di Normazione, 21 02 2018. [Online]. Available: http://uni.com/index.php?option=com_content&view=article&id=6830%3Aefficienza-energetica-degli-edifici-applicabilita-delle-nuove-en&catid=171&Itemid=2612#. [Consultato il giorno 03 01 2020].
- [15] A. Castagna, «Approfondimenti tecnici per l'edilizia. UNI EN ISO 52016,» Logical Soft, 09 03 2018. [Online]. Available: <https://www.logical.it/approfondimenti-tecnici-per-edilizia/pubblicata-in-italia-la-norma-energetica-uni-en-iso-52016>. [Consultato il giorno 04 01 2020].
- [16] Comitato Termotecnico Italiano Energia e Ambiente, «La UNI TS 11300,» CTI Energia e Ambiente, [Online]. Available: <https://www.cti2000.eu/la-uni-ts-11300/>. [Consultato il giorno 07 01 2020].
- [17] Acca Software, «Decreto requisiti minimi quadro sintesi,» Acca Software, 02 07 2015. [Online]. Available: <http://biblus.acca.it/decreto-requisiti-minimi-quadro-sintesi/>. [Consultato il giorno 03 01 2020].
- [18] Comitato Termotecnico Italiano Energia e Ambiente, «Statuto CTI,» CTI Energia e Ambiente, 15 10 2004. [Online]. Available: https://www.cti2000.it/doc/statuto_CTI.pdf. [Consultato il giorno 06 01 2020].
- [19] Comitato Termotecnico Italiano Energia e Ambiente, «Breve storia del CTI,» CTI Energia e ambiente, [Online]. Available: <https://www.cti2000.it/index.php?controller=sezioni&action=show&subid=1>. [Consultato il giorno 05 01 2020].
- [20] Comitato Termotecnico Italiano Energia e Ambiente, «Validazione Software,» CTI Energia e Ambiente, 02 12 2019. [Online]. Available: <https://www.cti2000.it/index.php?controller=sezioni&action=show&subid=62>. [Consultato il giorno 05 01 2020].
- [21] Comitato Termotecnico Italiano Ambiente e Energia, «Regolamento per la verifica di strumenti di calcolo e software commerciali,» CTI Ambiente e Energia, [Online]. Available: <https://www.cti2000.it/index.php?controller=pubblicazioni&action=show&id=36358>. [Consultato il giorno 06 01 2020].
- [22] gbXML, «gbxml,» gbXML, [Online]. Available: <https://www.gbxml.org/index.html>. [Consultato il giorno 19 01 2020].

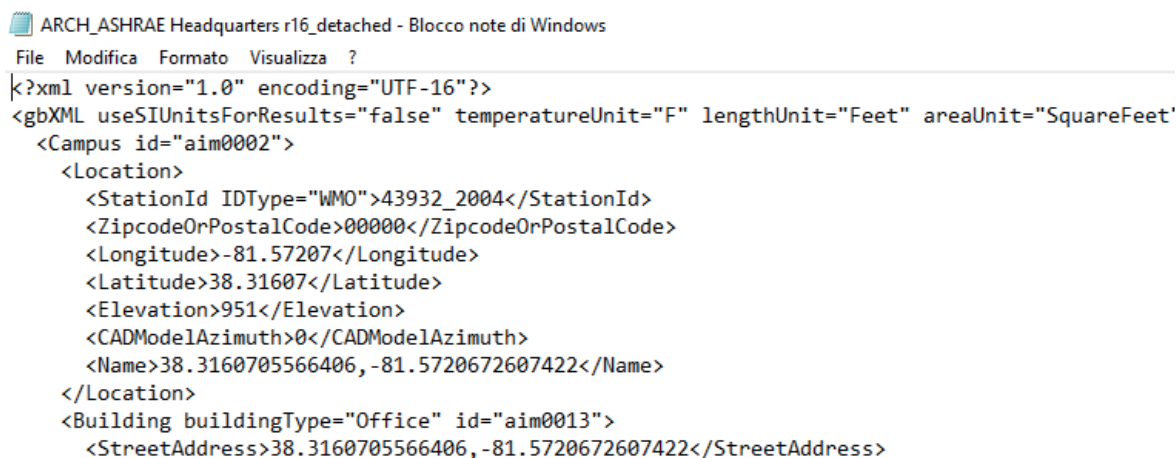
- [23] R. Messier, «Extensible Markup Language,» ScienceDirect, 2016. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/B9780128019498000076>. [Consultato il giorno 03 02 2020].
- [24] BuildingSMART International, «buildingsmart,» BuildingSMART International, [Online]. Available: <https://www.buildingsmart.org/>. [Consultato il giorno 03 01 2020].
- [25] 01Building, «Standard BuildingSMART,» 01building, 12 08 2019. [Online]. Available: <https://www.01building.it/bim/standard-buildingsmart/>. [Consultato il giorno 07 01 2020].
- [26] L. Nissim, «Normazione tecnica BIM a cosa serve,» IBIMI, 11 12 2015. [Online]. Available: <http://www.ibimi.it/normazione-tecnica-bim-a-cosa-serve/>. [Consultato il giorno 11 01 2020].
- [27] Library of Congress, «EXPRESS data modeling language, ISO 10303-11,» Library of Congress, 27 02 2017. [Online]. Available: <https://www.loc.gov/preservation/digital/formats/fdd/fdd000449.shtml>. [Consultato il giorno 11 01 2020].
- [28] Library of Congress, «Step-file, ISO 10303-21,» Library of Congress, 27 02 2017. [Online]. Available: <https://www.loc.gov/preservation/digital/formats/fdd/fdd000448.shtml>. [Consultato il giorno 10 01 2020].
- [29] N. Brugaletta, «Linguaggi programmazione,» EnneBi, [Online]. Available: <https://ennebi.solira.org/linprog/pag03.html>. [Consultato il giorno 20 01 2020].
- [30] M. Kilkelly, “What is Dynamo Revit,” Archsmarter, 08 02 2018. [Online]. Available: <https://archsmarter.com/what-is-dynamo-revit/>. [Accessed 25 01 2020].
- [31] M. Hamedani, «CodeWithMosh,» [Online]. Available: <https://codewithmosh.com/>. [Consultato il giorno 02 02 2020].
- [32] R. Rotti, «BIM e analisi energetica: interoperabilità tra software Il caso studio di FCA Mirafiori,» Politecnico di Torino, Torino, 2018.
- [33] Autodesk, «Revit/en/detail,» Autodesk, 25 06 2018. [Online]. Available: <https://apps.autodesk.com/RVT/en/Detail/Index?id=6193770166503453647&appLang=en&os=Win64>. [Consultato il giorno 18 01 2020].
- [34] Assobim, «Bim Normativa UNI 11337,» Assobim, 30 09 2019. [Online]. Available: <https://www.01building.it/bim/bim-normativa-uni-11337/>. [Consultato il giorno 20 12 2019].
- [35] S. Alessandrini, «Valutazione e certificazione dei professionisti del BIM,» ingenio, 03 12 2019. [Online]. Available: <https://www.ingenio-web.it/25261-valutazione-e-certificazione-dei-professionisti-del-bim-secondo-uni-11337-7-in-consultazione-pubblica-la-unipdr>. [Consultato il giorno 06 01 2020].
- [36] M. Bizzotto, «Normativa BIM UNI 11337,» 19 07 2019. [Online]. Available: <http://www.marcobizzotto.it/news-1/bim/normativa-bim/uni-11337.html>. [Consultato il giorno 09 12 2019].

- [37] H. Bourg, «Quello che hai sempre voluto sapere sulla ISO 19650,» Graphisoft Italia, 02 08 2019. [Online]. Available: <https://blog.archicad.it/tutto-quello-che-hai-sempre-voluto-sapere-sulla-iso-19650>. [Consultato il giorno 28 11 2019].
- [38] M. Carradori, «Le esportazioni dei modelli informativi in formato ifc,» 18 03 2019. [Online]. Available: <https://www.ingenio-web.it/22921-lesportazione-dei-modelli-informativi-in-formato-ifc>. [Consultato il giorno 15 01 2020].
- [39] P. Citelli e A. Meneghelli, «BIM ifc brutto e cattivo,» 05 10 2019. [Online]. Available: <https://www.ingenio-web.it/22944-bim-ifc-brutto-e-cattivo>. [Consultato il giorno 15 01 2020].
- [40] A. Lanari, «Building Information Modelling e illuminazione,» Luce e design, 02 11 2018. [Online]. Available: <https://www.lucenews.it/uni-113372017-bim-building-information-modelling-e-illuminazione/>. [Consultato il giorno 15 12 2019].
- A. Osello, Il futuro del disegno con il BIM per ingegneri e architetti, Palermo: Dario Flaccovio, 2012.
 - C. Eastman, P. Teicholz, R. Sacks e K. Liston, BIM Handbook: A guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors, Hoboken, New Jersey: John Wiley & Sons, 2011.
 - M. Lo Turco, Il Bim e la rappresentazione infografica nel processo edilizio. Dieci anni di ricerche e applicazioni, Marzo 2015: ARACNE editrice S.r.l., 2015.
 - M. Del Giudice, Il Disegno e l'Ingegnere: BIM handbook for Building and Civil Engineering Students, Torino: Levrotto & Bella, 2019.
 - B. Dong, K. Lam, Y. Huang e Dobbs, «A comparative study of the IFC and gbXML informational infrastructures for data,» 2007. [Online]. Available: http://www.ibpsa.org/proceedings/BS2007/p363_final.pdf. [Consultato il giorno 20 09 2019].
 - A. Osello e F. M. Ugliotti, BIM: verso Il catasto del futuro: Conoscere, digitalizzare, condividere. Il caso studio della Città di Torino, Torino: Gangemi Editore, 2017.
 - A. Shadrina, Framework for the transfer of building materials data between the BIM and, Settembre 2015.
 - A. Akbarieh, Systematic Investigation of Interoperability Issues between Building Information Modelling and Building Energy Modelling, Marzo 2018.
 - I. Ivanova, K. Kiesel e A. Mahdavi, «BIM-generated data models for EnergyPlus: A comparison of gbXML,» [Online]. Available: http://www.ibpsa.org/proceedings/BSA2015/9788860460745_51.pdf. [Consultato il giorno 14 11 2019].

- K. Katsigarakis, G. N. Lilis, G. Giannakis e D. Rovas, «An IFC Data Preparation Workflow For Building Energy Performance Simulation,» 10 07 2019. [Online]. Available: https://www.researchgate.net/publication/334448038_An_IFC_data_preparation_workflow_for_building_energy_performance_simulation. [Consultato il giorno 4 12 2019].
- S. Pinheiro, J. O'Donnell, R. Wimmer, V. Bazjanac, S. Muhic, T. Maile, J. Frisch e C. v. Treeck, «Model View Definition For Advanced Building Energy Performance Simulation,» 09 2016. [Online]. Available: https://www.researchgate.net/publication/308614558_Model_View_Definition_for_Advanced_Building_Energy_Performance_Simulation. [Consultato il giorno 15 12 2019].
- G. N. Lilis, G. I. Giannakis, G. Kontes e D. Rovas, «Semi-automatic thermal simulation model generation from IFC data,» 09 2014. [Online]. Available: https://www.researchgate.net/publication/312274723_Semi-Automatic_thermal_simulation_model_generation_from_IFC_data. [Consultato il giorno 12 12 2019].
- C. K. Sandvik e F. Fougner, «BIM as a tool for sustainable design,» 10 2018. [Online]. Available: https://www.researchgate.net/publication/328403782_BIM_as_a_tool_for_sustainable_design. [Consultato il giorno 10 11 2019].
- F. R. Cecconi, L. C. Tagliabue, S. Maltese e M. Zuccaro, «A multi-criteria framework for decision process in retrofit optioneering through interactive data flow,» 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S187770581731754X>. [Consultato il giorno 18 09 2019].

6 Allegati

Allegato A: File di esempio formato gbXML scaricato dal sito <http://www.gbxml.org>



```
ARCH_ASHRAE Headquarters r16_detached - Blocco note di Windows
File Modifica Formato Visualizza ?
<?xml version="1.0" encoding="UTF-16"?>
<gbXML useSIUnitsForResults="false" temperatureUnit="F" lengthUnit="Feet" areaUnit="SquareFeet"
  <Campus id="aim0002">
    <Location>
      <StationId IDType="WMO">43932_2004</StationId>
      <ZipcodeOrPostalCode>00000</ZipcodeOrPostalCode>
      <Longitude>-81.57207</Longitude>
      <Latitude>38.31607</Latitude>
      <Elevation>951</Elevation>
      <CADModelAzimuth>0</CADModelAzimuth>
      <Name>38.3160705566406, -81.5720672607422</Name>
    </Location>
    <Building buildingType="Office" id="aim0013">
      <StreetAddress>38.3160705566406, -81.5720672607422</StreetAddress>
```

Fig. 107: Edificio esempio in formato gbXML

<http://www.gbXML.org>

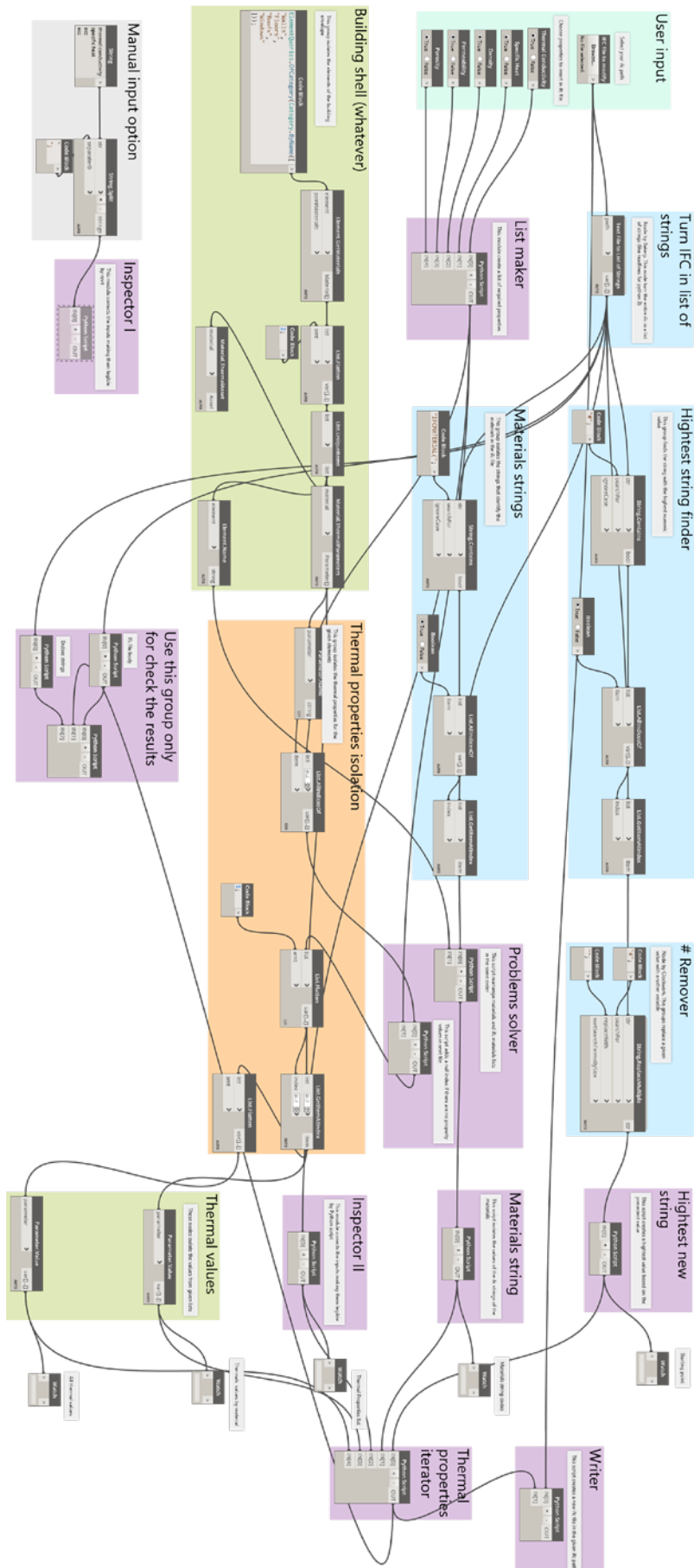
Allegato B: Estratto file IFC modificato del modello FCA



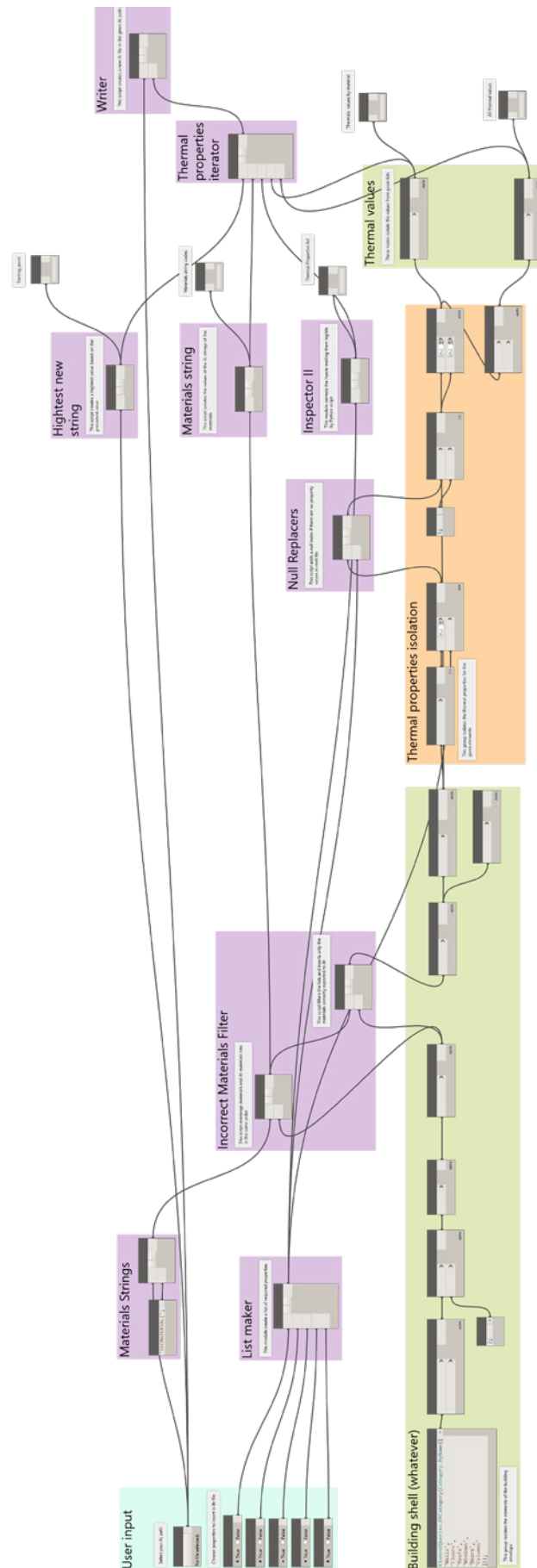
```
190605_MODELLO ARCHITETTONICO_nuova configurazione locali_riferimenti_aggiornati MPI - Blocco note di Windows
File Modifica Formato Visualizza ?
650,#464675,#464745,#465077,#475691,#475702,#483626,#484632,#484702,#485232,#485302,#485313,#485397,#485678
491803,#491814,#491900,#491911,#491997,#492042,#492128,#492158,#492242,#492253,#492337,#492348,#492432,#492
#545901= IFCPRESENTATIONLAYERASSIGNMENT('S-COLS-____-OTLN',$, (#308055,#308088,#318632,#318643,#348650,#3486
#545902= IFCPROPERTYINGLEVALUE('ThermalConductivity',$,IFCTHERMALCONDUCTIVITYMEASURE(0.54),$);
#545903= IFCPROPERTYINGLEVALUE('SpecificHeatCapacity',$,IFCSPECIFICHEATCAPACITYMEASURE(0.84),$);
#545904= IFCPROPERTYINGLEVALUE('MassDensity',$,IFCMASSDENSITYMEASURE(1550.0),$);
#545905= IFCPROPERTYINGLEVALUE('VaporPermeability',$,IFCVAPORPERMEABILITYMEASURE(182.4),$);
#545906= IFCPROPERTYINGLEVALUE('Porosity',$,IFCNORMALISEDRAIOMEASURE(0.001),$);
#545907= IFCPROPERTYINGLEVALUE('ThermalConductivity',$,IFCTHERMALCONDUCTIVITYMEASURE(0.025),$);
#545908= IFCPROPERTYINGLEVALUE('SpecificHeatCapacity',$,IFCSPECIFICHEATCAPACITYMEASURE(1.0035),$);
#545909= IFCPROPERTYINGLEVALUE('MassDensity',$,IFCMASSDENSITYMEASURE(1.2),$);
#545912= IFCPROPERTYINGLEVALUE('ThermalConductivity',$,IFCTHERMALCONDUCTIVITYMEASURE(0.51),$);
#545913= IFCPROPERTYINGLEVALUE('SpecificHeatCapacity',$,IFCSPECIFICHEATCAPACITYMEASURE(0.96),$);
#545914= IFCPROPERTYINGLEVALUE('MassDensity',$,IFCMASSDENSITYMEASURE(1120.0),$);
#545915= IFCPROPERTYINGLEVALUE('VaporPermeability',$,IFCVAPORPERMEABILITYMEASURE(0.0),$);
#545916= IFCPROPERTYINGLEVALUE('Porosity',$,IFCNORMALISEDRAIOMEASURE(0.001),$);
#545917= IFCPROPERTYINGLEVALUE('ThermalConductivity',$,IFCTHERMALCONDUCTIVITYMEASURE(1.19),$);
#545918= IFCPROPERTYINGLEVALUE('SpecificHeatCapacity',$,IFCSPECIFICHEATCAPACITYMEASURE(1.0),$);
#545919= IFCPROPERTYINGLEVALUE('MassDensity',$,IFCMASSDENSITYMEASURE(2300.0),$);
#545920= IFCPROPERTYINGLEVALUE('VaporPermeability',$,IFCVAPORPERMEABILITYMEASURE(182.4),$);
#545921= IFCPROPERTYINGLEVALUE('Porosity',$,IFCNORMALISEDRAIOMEASURE(0.001),$);
#545922= IFCPROPERTYINGLEVALUE('ThermalConductivity',$,IFCTHERMALCONDUCTIVITYMEASURE(0.209),$);
#545923= IFCPROPERTYINGLEVALUE('SpecificHeatCapacity',$,IFCSPECIFICHEATCAPACITYMEASURE(0.657),$);
#545924= IFCPROPERTYINGLEVALUE('MassDensity',$,IFCMASSDENSITYMEASURE(950.0),$);
#545925= IFCPROPERTYINGLEVALUE('VaporPermeability',$,IFCVAPORPERMEABILITYMEASURE(182.4),$);
#545926= IFCPROPERTYINGLEVALUE('Porosity',$,IFCNORMALISEDRAIOMEASURE(0.001),$);
#545927= IFCPROPERTYINGLEVALUE('ThermalConductivity',$,IFCTHERMALCONDUCTIVITYMEASURE(0.333),$);
#545928= IFCPROPERTYINGLEVALUE('SpecificHeatCapacity',$,IFCSPECIFICHEATCAPACITYMEASURE(1.0035),$);
#545929= IFCPROPERTYINGLEVALUE('MassDensity',$,IFCMASSDENSITYMEASURE(1.2),$);
#545932= IFCPROPERTYINGLEVALUE('ThermalConductivity',$,IFCTHERMALCONDUCTIVITYMEASURE(0.65),$);
#545933= IFCPROPERTYINGLEVALUE('SpecificHeatCapacity',$,IFCSPECIFICHEATCAPACITYMEASURE(0.84),$);
#545934= IFCPROPERTYINGLEVALUE('MassDensity',$,IFCMASSDENSITYMEASURE(1100.0),$);
#545935= IFCPROPERTYINGLEVALUE('VaporPermeability',$,IFCVAPORPERMEABILITYMEASURE(0.0),$);
#545936= IFCPROPERTYINGLEVALUE('Porosity',$,IFCNORMALISEDRAIOMEASURE(0.001),$);
#545937= IFCPROPERTYINGLEVALUE('ThermalConductivity',$,IFCTHERMALCONDUCTIVITYMEASURE(17.0),$);
#545938= IFCPROPERTYINGLEVALUE('SpecificHeatCapacity',$,IFCSPECIFICHEATCAPACITYMEASURE(0.5),$);
#545939= IFCPROPERTYINGLEVALUE('MassDensity',$,IFCMASSDENSITYMEASURE(7900.0),$);
#545940= IFCPROPERTYINGLEVALUE('VaporPermeability',$,IFCVAPORPERMEABILITYMEASURE(0.0),$);
#545941= IFCPROPERTYINGLEVALUE('Porosity',$,IFCNORMALISEDRAIOMEASURE(0.001),$);
#545942= IFCPROPERTYINGLEVALUE('ThermalConductivity',$,IFCTHERMALCONDUCTIVITYMEASURE(0.024),$);
```

Fig. 108: Estratto file IFC

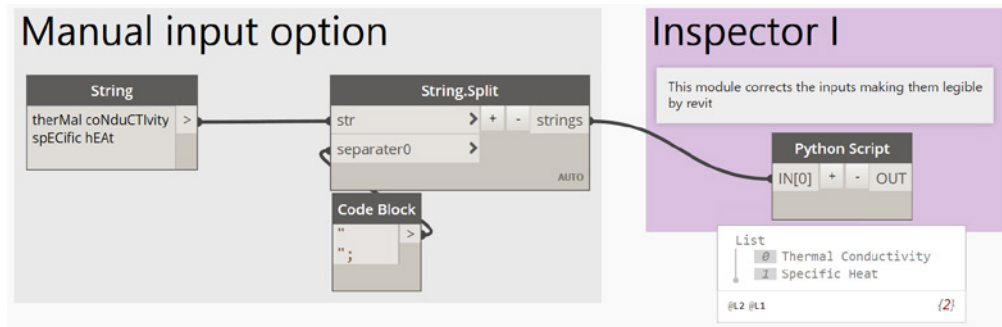
Allegato C: Algoritmo iterazione proprietà termiche, prima versione (meno performante)



Allegato D: Algoritmo iterazione proprietà termiche, seconda versione (più performante)



Allegato E: Script Inspector I



Python script:

```
import re
```

```
def capitalize_string(s):  
    return re.sub('(?<=^)[a-z]|(?<=\\s)[a-z]', '{ }', s).format(*map(str.upper, re.findall('(?<=^)[a-z]|(?<=\\s)[a-z]', s)))
```

```
# The inputs to this node will be stored as a list in the IN variables.
```

```
thermal_properties = IN[0]
```

```
# Place your code below this line
```

```
x = []
```

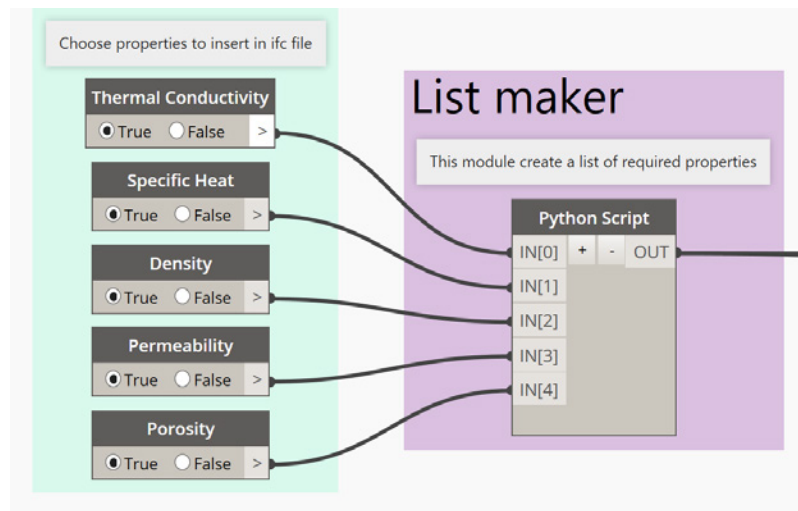
```
for it in thermal_properties:
```

```
    x.append(it.title())
```

```
# Assign your output to the OUT variable.
```

```
OUT = x
```

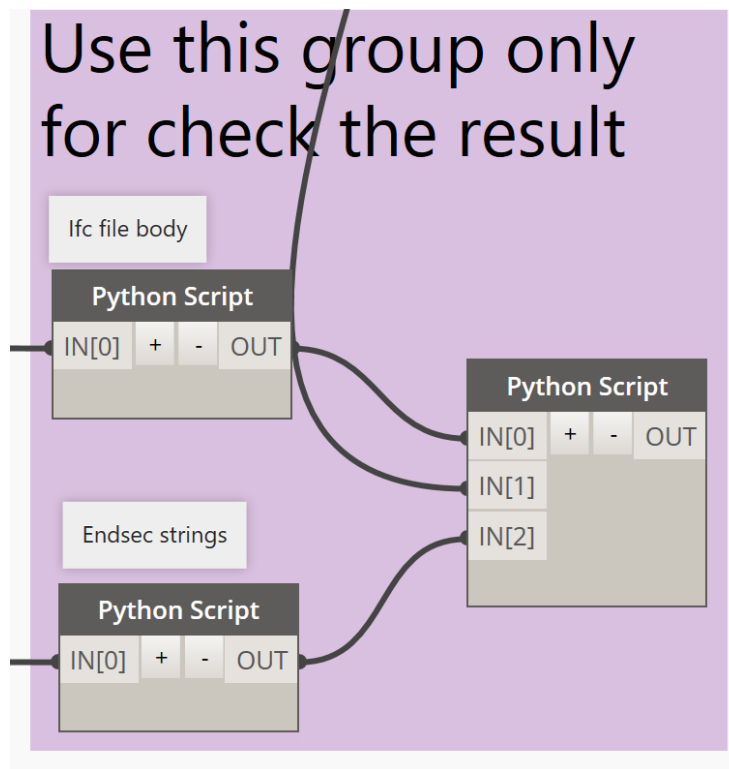
Allegato F: Script liste proprietà termiche



Python script:

```
thermal_conductivity = IN[0]
specific_heat = IN[1]
density = IN[2]
permeability = IN[3]
porosity = IN[4]
# Place your code below this line
properties = []
if thermal_conductivity == True:
    properties.append("Thermal Conductivity")
if specific_heat == True:
    properties.append("Specific Heat")
if density == True:
    properties.append("Density")
if permeability == True:
    properties.append("Permeability")
if porosity == True:
    properties.append("Porosity")
# Assign your output to the OUT variable.
OUT = properties
```

Allegato G: Script gruppo controllo risultati



Python script:

```
original_ifc = IN[0]
```

```
tp_iterator = IN[1]
```

```
end_sec = IN[2]
```

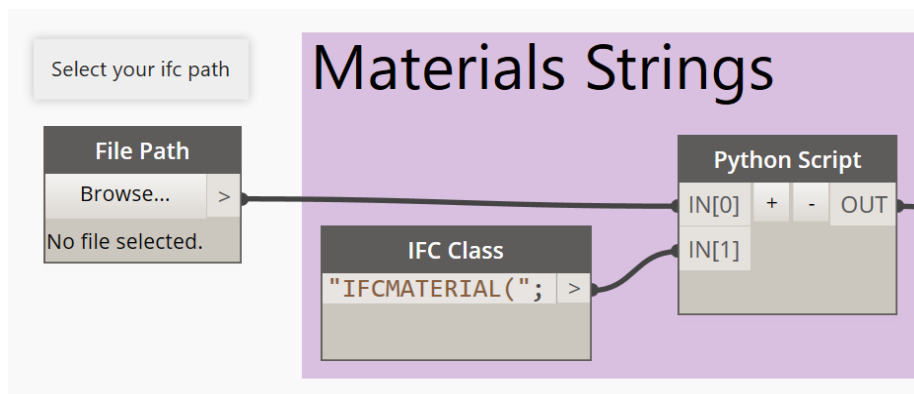
```
# Place your code below this line
```

```
final_ifc = original_ifc + tp_iterator + end_sec
```

```
# Assign your output to the OUT variable.
```

```
OUT = final_ifc
```

Allegato H: Script ricerca classe ifc



Python script:

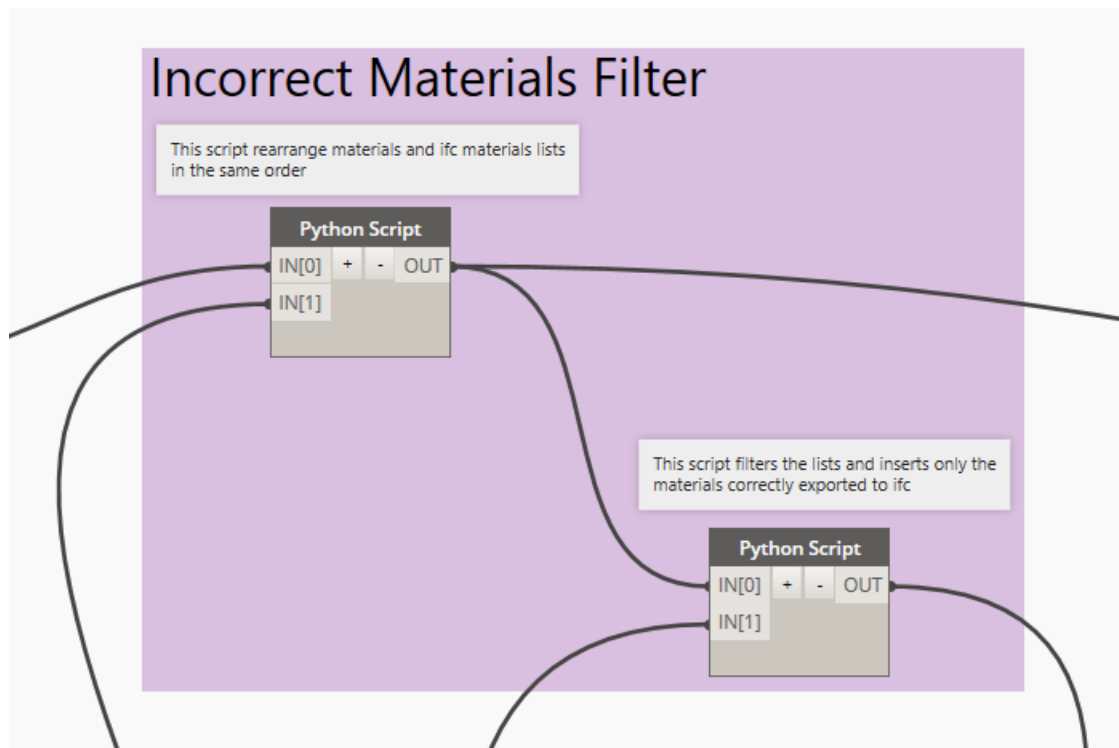
```
import sys
sys.path.append(r'C:\Program Files\IronPython 2.7\Lib')
import os

# The inputs to this node will be stored as a list in the IN variables.
path = IN[0]
request = IN[1]
materials_strings = []
materials_strings_value = []
# Place your code below this line
with open(path, "r") as xfileobj:
    text = xfileobj.readlines()

for echo in text:
    if request in echo:
        materials_strings.append(echo)

# Assign your output to the OUT variable.
OUT = materials_strings
```

Allegato I: Script filtro materiali



Python script 1:

```
ifc_input = IN[0]
revit_input = IN[1]
# Place your code below this line
re_list = [("(" + mike + ")") for mike in revit_input]
```

```
list_sorted = []
for element in re_list:
    for item in ifc_input:
        if element in item:
            list_sorted.append(item)
            ifc_input.remove(item)
        else:
            pass
# Assign your output to the OUT variable.
OUT = list_sorted
```

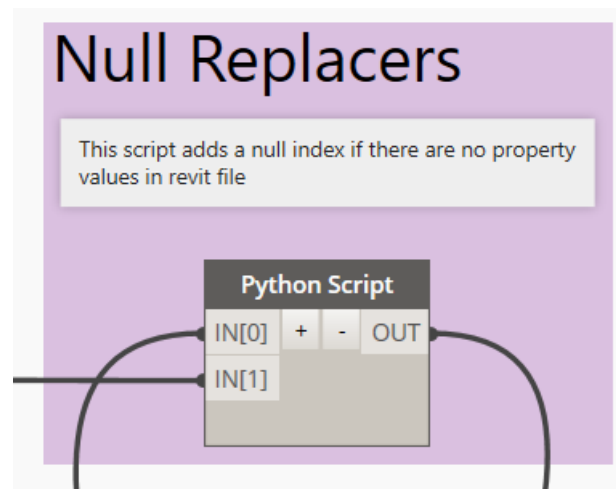
Python script 2:

```
ifc_input = IN[0]
revit_input = IN[1]
# Place your code below this line
re_list = [("(" + mike + ")") for mike in revit_input]

revit_list_sorted = []
for element in re_list:
    for item in ifc_input:
        if element in item:
            revit_list_sorted.append(element[1:-1])
    else:
        pass

# Assign your output to the OUT variable.
OUT = revit_list_sorted
```

Allegato L: Script filtro valore matrice nullo



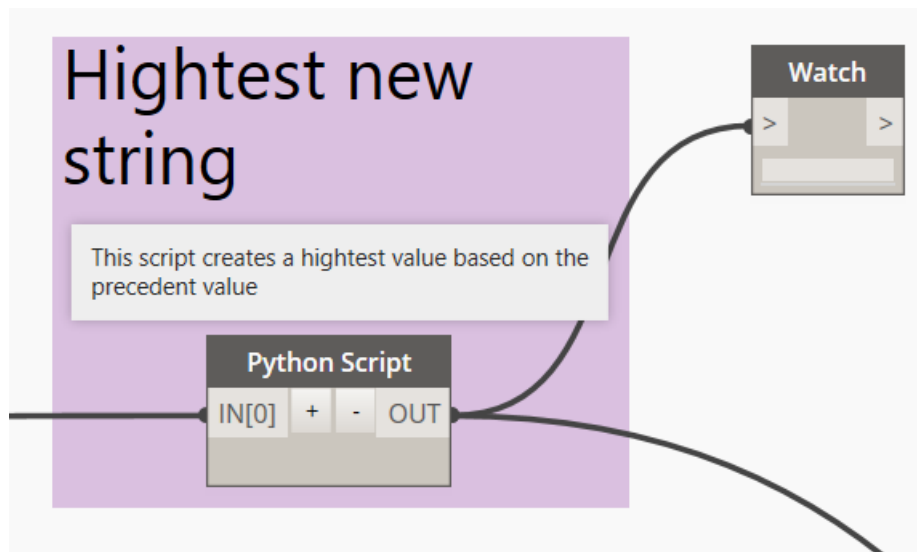
Python script:

```
lists = IN[0]
required_properties = IN[1]
# Place your code below this line
sub_lists_len = len(required_properties)
new_list = []
for omega in lists:
    for alpha in omega:
        b = 'None'
        if alpha == []:
            alpha = b
        else:
            pass
        new_list.append(alpha)

chunks = [new_list[x:x + sub_lists_len] for x in range(0, len(new_list), sub_lists_len)]

# Assign your output to the OUT variable.
OUT = chunks
```

Allegato M: Script nuova stringa IFC



Python script:

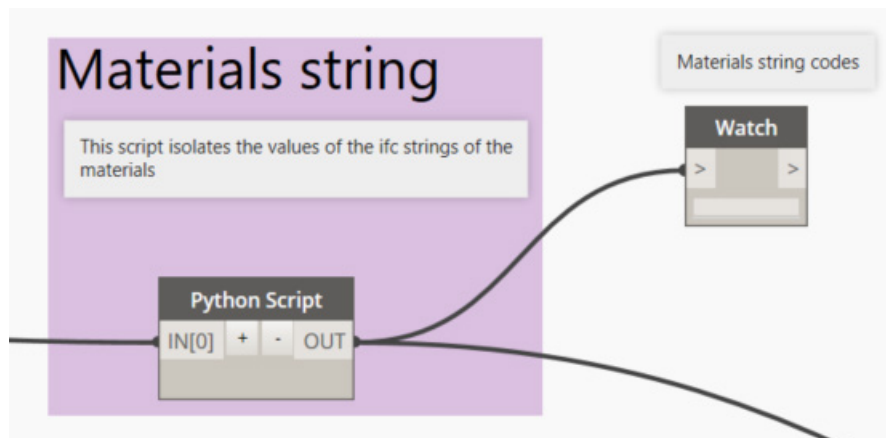
```
import sys
sys.path.append(r'C:\Program Files\IronPython 2.7\Lib')
import os

# The inputs to this node will be stored as a list in the IN variables.
path = IN[0]
ifc_entity_list = []
# Place your code below this line
with open(path, "r") as xfileobj:
    text = xfileobj.readlines()

for charlie in text:
    if "#" in charlie:
        position = charlie.index("#")
        ifc_entity_list.append(charlie[1:position])

yankee = int(ifc_entity_list[-1]) + 1
# Assign your output to the OUT variable.
OUT = yankee
```

Allegato N: Script filtro valore matrice nullo



Python script:

```
stings = IN[0]
```

```
# Place your code below this line
```

```
new_list = [] # This list contain string codes of all materials in this ifc  
for elements in stings:
```

```
    position = elements.index("=")
```

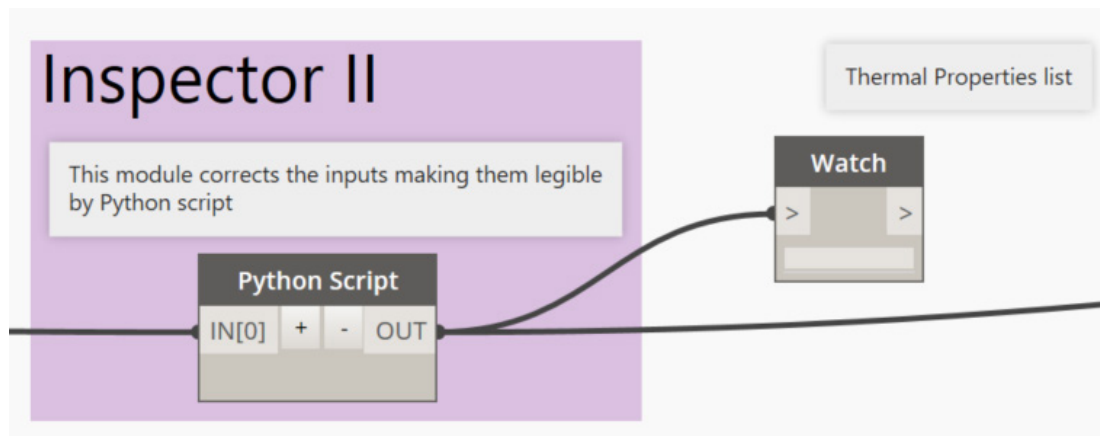
```
    new_list.append(elements[0:position])
```

```
# Assign you output to the OUT variable.
```

```
OUT = new_list
```

Allegato O: Script correttore

(da usare solo se i nomi delle proprietà vengono inseriti manualmente)



Python script:

```
import re
```

```
# The inputs to this node will be stored as a list in the IN variables.
```

```
thermal_properties = IN[0]
```

```
# Place your code below this line
```

```
space_remove = []
```

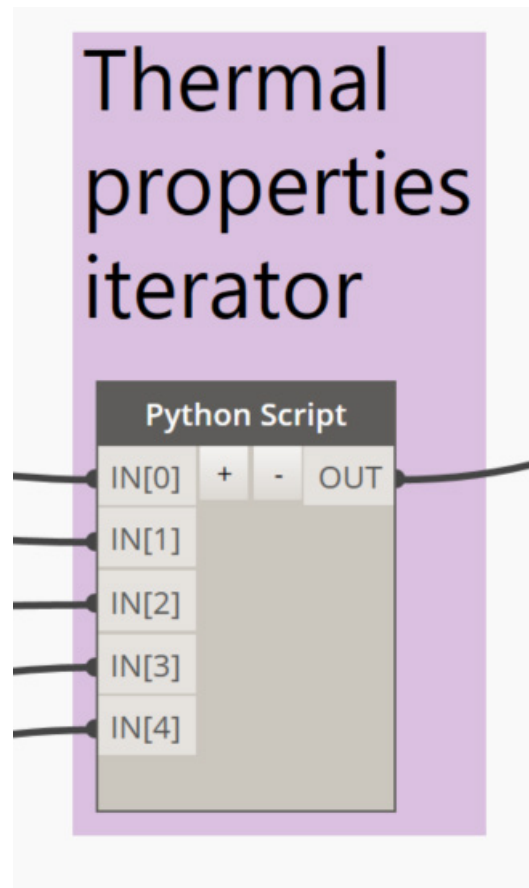
```
for element in thermal_properties:
```

```
    space_remove.append(element.replace(" ", ""))
```

```
# Assign your output to the OUT variable.
```

```
OUT = space_remove
```

Allegato P: Script iteratore



```
def numbers(first_number, last_number, step=1):  
    return range(first_number, last_number, step)
```

```
def replace_item(string, turned_string):  
    if string == "ThermalConductivity":  
        turned_string.append(psv1)  
    elif string == "SpecificHeat":  
        turned_string.append(psv2)  
    elif string == "Density":  
        turned_string.append(psv3)  
    elif string == "Permeability":  
        turned_string.append(psv4)  
    elif string == "Porosity":  
        turned_string.append(psv5)  
    return turned_string
```

```

def chunks(extent, part=2):
    for i in range(0, len(extent), part):
        yield extent[i:i + part]

# The inputs to this node will be stored as a list in the IN variables.
starting_point = IN[0]
material_ref = IN[1]
thermal_prop = IN[2]
values = IN[3]
tv_unit_numbers = IN[4]

# Place your code below this line

raw_result = []
valid_result = []

psv1 = "IFCPROPERTYSINGLEVALUE('ThermalConductivity',$,
IFCTHERMALCONDUCTIVITYMEASURE("
psv2 = "IFCPROPERTYSINGLEVALUE('SpecificHeatCapacity',$,IFCSPECIFICHE
ATCAPACITYMEASURE("
psv3 = "IFCPROPERTYSINGLEVALUE('MassDensity',$,IFCMASSDENSITY
MEASURE("
psv4 = "IFCPROPERTYSINGLEVALUE('VaporPermeability',$
,IFCVAPORPERMEABILITYMEASURE("
psv5 = "IFCPROPERTYSINGLEVALUE('Porosity',$,IFCNORMALISED
RATIOMEASURE("

tv_unit = [(str(t_values)) for t_values in tv_unit_numbers]

ifc_entity_strings = []
for element in thermal_prop:
    ifc_entity_list = replace_item(element, ifc_entity_strings)

```

```
material_matrix = [b for a in material_ref for b in ifc_entity_strings]
matrix_length = len(material_matrix)
end_point = starting_point + matrix_length
string_value = [(("#" + str(ryu) + "= ") for ryu in numbers(starting_point, end_point))]
string_value_2 = [(("#" + str(ryu)) for ryu in numbers(starting_point, end_point))]
zip_all = zip(string_value, material_matrix, tv_unit)
for x, y, z in zip_all:
    raw_result.append(x + y + z + ",$);")
for omega in raw_result:
    if "None" not in omega:
        valid_result.append(omega)

# First part end here

sub_0_len = (len(values[0]))    # Length of the first sublist in list
values_len = (len(values))      # Length of the list
data = range(starting_point, end_point)

chunks = [string_value_2[x:x + sub_0_len] for x in range(0, len(data), sub_0_len)]
reunion = []
for nat in chunks:
    correct_format = ("(" + str(nat) + ")")
    replacement = correct_format.replace("'", "").replace("[", "").replace("]", "").replace("
", "")
    reunion.append(replacement)

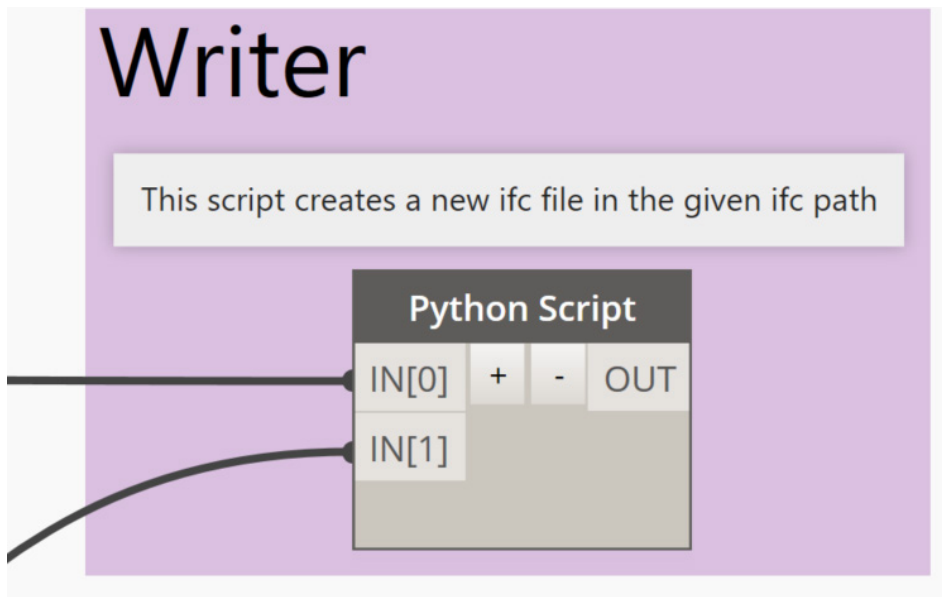
ref_num = len(material_ref)
new_end_point = end_point + ref_num

second_string_value = [(("#" + str(ken) + "= IFCMATERIALPROPERTIES('Pset_
MaterialThermal',$)") for ken in numbers(end_point, new_end_point))]
zip_4_second = zip(second_string_value, reunion, material_ref)
for x, y, z in zip_4_second:
```

```
raw_result.append(x + y + “,” + z + “;”)  
valid_result.append(x + y + “,” + z + “;”)
```

```
# Assign your output to the OUT variable.  
OUT = valid_result
```

Allegato P: Script iteratore



Python script:

```
import sys
sys.path.append(r'C:\Program Files\IronPython 2.7\Lib')
import os

# The inputs to this node will be stored as a list in the IN variables.
path = IN[0]
list_of_custom = IN[1]

# Place your code below this line
with open(path, "r") as xfileobj:
    text = xfileobj.readlines()
    ifc_body = text[:-3]
    end = text[-3:]

path_splitter = os.path.split(path)
dir_path = path_splitter[0]
file_path = path_splitter[1]
add = " MPI"
new_name = file_path[:-4] + add
complete_path = os.path.join(dir_path, new_name+".ifc")
```

```

with open(complete_path, "w") as xcopy:
    for x in ifc_body:
        xcopy.write(x)
    for y in list_of_custom:
        xcopy.write(y)
        xcopy.write("\n")
    for z in end:
        xcopy.write(z)
# Assign your output to the OUT variable.
OUT = "success"

```

Allegato Q: Qrcode tabella valutazioni software

