

# HEAL

**Housing for Emergency and Affordable Living.  
Pensare parametrico**

Rebecca Cagnotto

**Politecnico di Torino**  
Dipartimento di Architettura e Design  
A.A. 2020/2021



Tesi di Laurea Magistrale in Architettura Costruzione e Città

**HEAL**  
Housing for Emergency and Affordable Living  
Pensare parametrico

RELATRICE  
**Prof.ssa Michela Barosio**

CO-RELATORE  
**Arch. Andrea Tomalini**

CANDIDATA  
**Rebecca Cagnotto**  
Matricola 267093

# ABSTRACT

Questo lavoro nasce in un periodo caratterizzato da una pandemia mondiale, che ha dimostrato quanto il nostro mondo si ancora contraddistinto da una forte incertezza e quanto i modelli abitativi flessibili siano sempre più importanti.

Ho, quindi, deciso di lavorare su moduli di alloggi flessibili a prezzi accessibili in grado di aiutare in situazioni di emergenza, da quella sanitaria a quella abitativa.

Per questo ho partecipato a un concorso di idee indetto da Bee Breeders, in modo da avere uno spunto iniziale da cui partire per lo sviluppo del mio progetto.

Per riuscire al meglio nel mio obiettivo ho scelto di utilizzare un approccio parametrico al problema. Questo tipo di metodologia permette di scomporre il problema in elementi più semplici da risolvere. Inoltre, l'approccio parametrico consente di gestire in modo chiaro numerose variazioni all'interno dello stesso elemento.

In questo modo nasce questa tesi, che ha come obiettivo quello di focalizzarsi innanzitutto sulla messa a fuoco di un processo, il cui fine è quello di creare un prototipo di aggregazione di abitazioni, che possano essere progettate, modificate e adattate velocemente e facilmente, per essere flessibili in diverse situazioni o aree geografiche.

A seguito della progettazione iniziale si è anche deciso di introdurre l'utilizzo di software digitali di progettazione parametrica. Questa tipologia di programmi, basati sull'uso di algoritmi, permettono di esplicitare in modo diretto e chiaro

il processo necessario per giungere alle soluzioni cercate e allo stesso tempo, consentono di gestire e controllare quantità enormi di dati e variazioni.

Il campo della progettazione parametrica è talmente vasto e caratterizzato ancora da cambiamenti frequenti, che il progetto di questa tesi potrebbe essere sviluppato ancora su moltissimi fronti, ma l'obiettivo primario è stato quello di individuare un processo coerente, che possa essere applicato automaticamente in altri casi e contesti, fornendo uno strumento di supporto per la progettazione.

In conclusione, questo percorso di ricerca mi ha aiutato a capire che i moderni strumenti di rappresentazione digitale, se ben gestiti, possono essere un supporto indispensabile, ma allo stesso tempo, senza una metodologia chiara e lo sviluppo di una procedura progettuale definita non si riesce a gestire in modo efficace la complessità di un progetto.

# INDICE

	Introduzione	9			
<b>1</b>	Radici e sviluppo dell'approccio parametrico	13	<b>3</b>	Seconda fase. Lo sviluppo digitale	81
	1.1 Precedenti			2.1 Perché il parametrico?	
	1.2 Svolta computazionale			2.2 L'algoritmo	
	1.3 La logica del processo parametrico: il diagramma			2.3 Wasp	
				2.3 Grasshopper	
<b>2</b>	Prima fase. Il progetto di concorso	31		Conclusioni	103
	2.1 Il bando			Bibliografia	107
	2.2 I requisiti			Ringraziamenti	112
	2.3 Casi studio				
	2.3 L'abaco				

# INTRODUZIONE

La progettazione parametrica può essere definita come una serie di domande per stabilire le variabili di un progetto e l'utilizzo di software digitali possono essere utilizzati per facilitare la gestione della varietà di soluzioni.

In questa epoca in cui i processi sono sempre in rapida evoluzione si devono adottare nuovi modi di pensare e progettare. Il pensiero parametrico come metodologia modifica il processo di progettazione convenzionale e permette di trattare più variabili che vanno a definire una serie di regole.

Con l'approccio parametrico si richiede di iniziare a pensare con le regole e i parametri di progettazione e non con soluzioni progettuali predefinite.

Utilizzare una serie di problemi e variabili per risolvere il progetto consente di controllare la progettazione entro un intervallo stabilito, a partire dalla gestione degli elementi singoli si riesce a ottenere risultati chiari a livello globale dell'aggregazione.

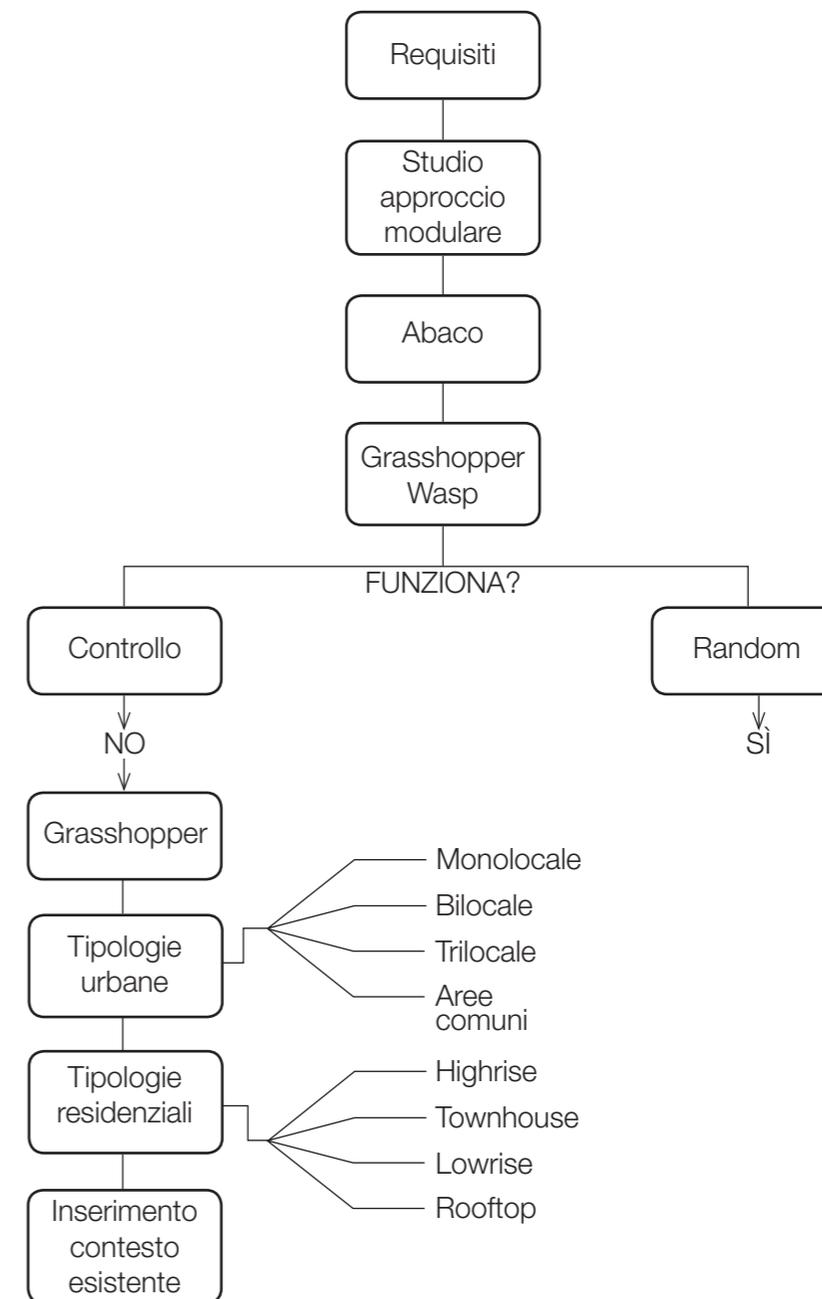
Questa metodologia permette di esplicitare le fasi di analisi e sintesi, consentendo una definizione chiara e precisa del processo di progettazione e facendo capire facilmente i problemi e come poterli risolvere.

Per riuscire a descrivere al meglio il processo seguito durante lo svolgimento di questo lavoro, l'elaborato è stato diviso in tre capitoli.

Il primo capitolo si focalizza sull'evoluzione del processo parametrico, a partire dalla classicità fino ad arrivare ai giorni nostri, con un approfondimento sulla metodologia e logica di questo tipo di approccio.

Successivamente, il secondo capitolo descrive le fasi progettazione seguite per rispondere al concorso di idee a cui ho partecipato come spunto iniziale della mia ricerca. Partendo dalla richiesta di risolvere la crisi abitativa delle città contemporanee si è definito un processo di progettazione che possa essere utilizzato a varie scale e in molteplici situazioni, creando un prototipo di edificio residenziale adattabile e flessibile.

Infine, il terzo capitolo si sofferma sullo sviluppo digitale del progetto con l'utilizzo di software di progettazione parametrica, che permettono una gestione e un controllo delle variabili e delle aggregazioni del prototipo studiato e allo stesso tempo riescono a evidenziare il processo progettuale seguito.



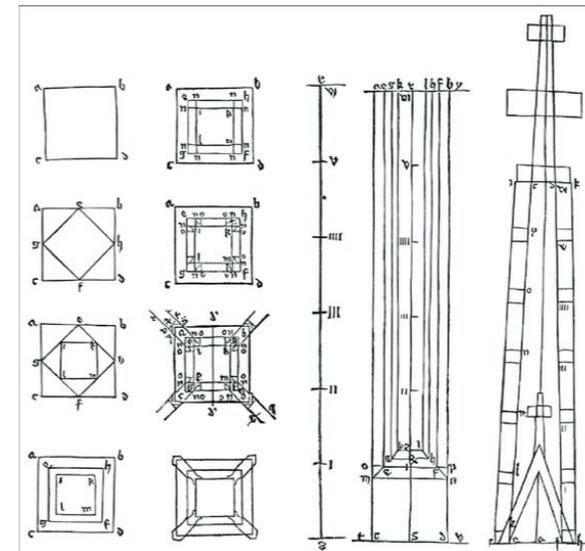
**capitolo 1**

**Radici e sviluppo  
dell'approccio  
parametrico**

# 1.1 Precedenti

## L'ANTICHITÀ E IL MEDIOEVO

L'approccio parametrico in quanto metodo nasce ben prima di quanto ci si possa immaginare. Se le radici del Parametrisimo digitale si possono riscontrare a partire dagli anni '70 e '80 del Novecento, il vero precedente si può trovare già a partire dall'antichità classica e dal Medioevo, quando i trattati di architettura erano trasmessi senza immagini e gli elementi e le loro proporzioni erano trasmessi solo grazie all'uso della spiegazione scritta<sup>[1]</sup>.



*Disegni tratti dal primo libro di architettura illustrato, ma si può ancora notare come i disegni rappresentino dei diagrammi che indicano le regole geometriche di costruzione e non una rappresentazione di un modello ripetibile.*

*Fonte: Roriczer M., "Das Büchlein von der Fialen Gerechtigkeit", 1486.*

Già nel I secolo a.C. **Vitruvio**, con il suo *De Architectura*, faceva uso di questo tipo di approccio, i vari elementi architettonici erano descritti attraverso l'uso di regole e sequenze logiche, per esempio nel formato di una sequenza "if-then", quelle che al giorno d'oggi possono essere chiamate algoritmi procedurali. Seguendo questo tipo di approccio i vari elementi non sono oggetti singoli, ma classi di oggetti, in quando l'applicazione delle regole date creava un vasto numero di elementi simili, con radici comuni, ma allo stesso tempo molto diversi fra loro.

L'architettura gotica funziona ancora come gli scritti di Vitruvio, quindi con la descrizione di tutte le fasi propedeutiche alla costruzione di un elemento, ma senza poter controllare la forma finale di ognuno di essi. Oggetti con la stessa funzione erano sì simili e appartenevano alla stessa classe, ma avevano al loro interno una serie di variazioni tipiche di quel modo di costruire e progettare. Si può dire che l'architettura classica e medievale fosse "generativa e basata su regole", come l'architettura parametrica attuale<sup>[1]</sup>.

## L'EPOCA MODERNA

Con l'avvento dell'epoca moderna si ha anche la fine della variazione, dopo il *De re aedificatoria* di Leon Battista Alberti (1404-1472) i **trattati iniziano a essere accompagnati da illustrazioni**, che facilitano la copia identica degli elementi descritti, così come accade nei trattati del Vignola (1507-1573) e di Palladio (1508-1580). L'approccio parametrico va, quindi, a scemare sempre di più, soprattutto con lo scoppio della Rivoluzione industriale, quando la diffusione della produzione in serie fa sì che si stabilisca la **ricerca dell'identico**. La situazione rimane tale fino almeno al XX secolo, quando la ricerca della variazione riprende piede e si crea una nuova rivoluzione nel modo di progettare, le variazioni tipiche dell'architettura antica sono prodotte in massa grazie alle macchine tipiche dell'epoca moderna<sup>[1]</sup>.



Copertura voltata della Sagrada Família  
Fonte: <https://www.barcelonabusturistic.cat/it/basilica-della-sagrada-familia>

## I PRECURSORI

Antoni Gaudì e Frei Otto possono essere considerati come precursori del Parametriso moderno, poiché hanno influenzato il modo di pensare e progettare di numerosi architetti contemporanei<sup>[2]</sup>.

Grazie ai lavori di analisi di Mark Burry, l'architetto che si sta attualmente occupando della ricostruzione della Sagrada Família, si può dire che **Gaudì** usasse totalmente un approccio parametrico alla progettazione. Grazie all'uso preciso della geometria riesce a fondere insieme forme geometriche complesse, come parabole iperboliche e catenarie, utilizzando regole e parametri, uno su tutti la gravità, dando vita a modelli flessibili e variabili.



Cripta della cappella della colonia Güell. Si notano le colonne e le paraboloidi iperboliche, che sono infinitamente variabili parametricamente.  
Fonte: <http://spagna.ilreporter.com/cripta-Gaudi-colonia-guell/>

Le colonne è dove si può meglio vedere l'uso del parametrico nei lavori di Gaudì, che manipola geometrie semplici, come parallelepipedi, per creare forme complesse. Le colonne variano in dimensione e forma in base alla loro posizione e le loro variazioni seguono regole precise stabilite in partenza (l'altezza e il diametro sono direttamente proporzionali, così come l'angolo di rotazione degli elementi che le compongono e la loro altezza e diametro)<sup>[3]</sup>.

Dalle forme di Gaudì si può risalire alle geometrie iniziali e al processo della loro costruzione, "non inventa forme; le scopre". Questo approccio diventa ancora più esplicito nei lavori dell'architetto tedesco Frei Otto<sup>[4]</sup>.

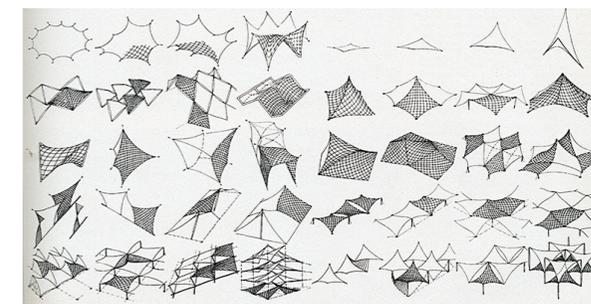
**Frei Otto** condivide con Gaudì l'utilizzo di modelli flessibili per creare le sue geometrie, per esempio modelli sospesi, creando volumi architettonici che seguono la forma che la forza di gravità impone loro<sup>[5]</sup>. I suoi lavori sono il risultato di un'attenta osservazione dei fenomeni naturali e fisici, come si può vedere dai suoi esperimenti con le bolle di sapone o con le strutture ramificate. La natura non viene copiata, ma studiata per capirne i processi di auto-generazione delle forme<sup>[4]</sup>.

Il termine "**architettura parametrica**" risale a ben prima dell'uso sistematico dei computer, quando negli anni '40 Luigi Moretti, con l'aiuto del matematico Bruno De Finetti, conia il termi-

ne, riferendosi alla relazione fra il progetto architettonico e le equazioni parametriche.

Secondo Moretti "*i parametri e le loro interrelazioni diventano il codice del nuovo linguaggio architettonico [...] L'impostazione dei parametri e la loro relazione deve essere supportata dalle tecniche e dagli strumenti offerti dalle scienze più attuali, in particolare dalla logica, dalla matematica e dall'informatica*".<sup>[6]</sup>

L'approccio computazionale al progetto è stato usato anche come processo cognitivo prima dell'uso massiccio dei computer. Come per esempio nei lavori di **Christopher Alexander**, che nel 1977 in "*A pattern language*" cerca e studia un processo analitico preciso e chiaro per descrivere il progetto architettonico, con dati semplici di input e output e in cui il problema progettuale viene scomposto e ogni parte diventa un termine di questo linguaggio, appunto un pattern<sup>[7]</sup>.



Frei Otto. Variazioni e combinazioni di strutture di tende.  
Fonte: Casabella, vol. 301, 1966, p. 35

# 1.2

## Svolta computazionale

### OBJECT E OBJECTILE

Secondo Mario Carpo il Parametrisimo digitale come lo intendiamo oggi è nato nel 1988 in *"The Fold"* di Gilles Deleuze, dove il filosofo cerca di andare a fondo della teoria del calcolo differenziale di Leibniz.

Il matematico usa variabili e parametri (x, y, a, b, c...) per identificare un insieme generico di funzioni, quando questi parametri vengono sostituiti da un numero si rappresenta una funzione specifica. Deleuze chiama questa **classe di oggetti**, che però non sono nessun oggetto in particolare, "objectile", cioè un elemento generico<sup>[1]</sup>.

L'objectile è un concetto creato in architettura da Bernard Cache, come risposta alla moderna standardizzazione capitalista. Consente, infatti, infinite variazioni parametriche, così da permettere la riproduzione in serie di parti non identiche<sup>[8]</sup>.

Questo tipo di logica si prestava particolarmente bene per la progettazione basata su computer e sta alla base della teoria delle serie non-standard.

Per non-standard si intende una serie che è definita dalle sue variazioni e differenze, è un insieme in cui ogni elemento costituente ha qualcosa in comune con gli altri, cioè in cui tut-

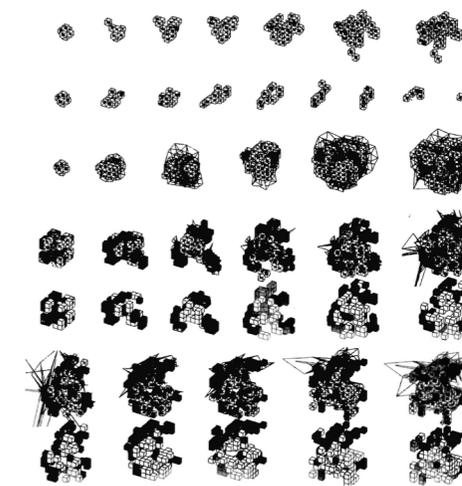
ti gli elementi hanno algoritmi in comune; dal punto di vista visuale una serie non-standard è composta da potenzialmente infiniti elementi tutti diversi, ma simili fra loro<sup>[9]</sup>.

### ARCHITETTURA EVOLUTIVA

In *"An evolutionary architecture"* **John Frazer** espone il percorso nella ricerca di una nuova tecnica di progettazione digitale.

Il focus principale della ricerca era di trovare il modo per gestire le regole che definiscono il progetto architettonico.

Per fare ciò Frazer e la sua squadra hanno creato nuovi tools e software in grado di soddisfare le loro esigenze.



Aggregazioni ottenute al variare di parametri.  
Fonte: Frazer J., *"An Evolutionary Architecture"*, Architecture Association, Londra 1995

I calcoli potevano essere ridotti facilmente a istruzioni logiche semplici (come “and” e “nor”), attraverso cui si poteva creare ogni funzione logica, gerarchicamente a partire dalle funzioni più basilari. Frazer dimostra, quindi, che l'essenziale è il modello concettuale, non tanto il software in sé, e che bisogna iniziare a ripensare al modo in cui si fanno le cose abitualmente <sup>[10]</sup>.

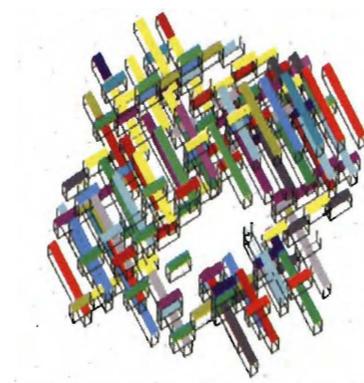
Per esplorare i sistemi rule-based Frazer ha utilizzato la teoria dei polyautomata, cioè un ramo della teoria computazionale, che utilizza più automi interconnessi fra loro, che lavorando in parallelo formano un automa più grande. Una classe specifica di polyautomata sono i “**cellular automata**”, da cui si può facilmente notare come da regole semplici possano crearsi risultati complessi. Sono costituiti da una serie di cellule, che possono avere varie dimensioni, ognuna di loro può avere almeno due stati (per esempio: 0 o 1, vero o falso...) e cellule vicine con cui hanno delle relazioni spaziali precise. Il comportamento di un “cellular automaton” (chiamato comportamento globale) è determinato da regole di transizione, cioè regole che definiscono lo stato di una cella singola (chiamate regole locali) in ogni momento in relazione allo stato della stessa cella e dei suoi vicini nel momento precedente.

Tutto ciò porta a creare comportamenti anche

molto complessi, che in alcuni casi possono anche sfociare in situazioni di errori o caos <sup>[10]</sup>.

Negli stessi anni si stavano evolvendo anche altre numerose tecniche digitali utili a risolvere problemi per miglioramenti e sviluppi consecutivi, dando la possibilità di selezionare le soluzioni che danno il risultato migliore e scartando le altre.

Fra questi metodi c'è anche il modello adattivo di John Holland, che richiede la progressiva modifica di una struttura grazie all'azione ripetuta di alcuni operatori (prende come esempio i cromosomi che rappresentano la struttura per la genetica e le mutazioni come gli operatori). Questo concetto forma le basi per il modello adattivo, usato anche da Frazer, in cui avvengono successive modifiche alla struttura in risposta all'ambiente circostante, che quindi influenza le scelte che vengono fatte <sup>[10]</sup>.



*Esempio di cellular automaton. Le regole stabilite servono a creare una serie di trasformazioni proporzionali.  
Fonte: Frazer J., “An Evolutionary Architecture”, Architecture Association, Londra 1995*

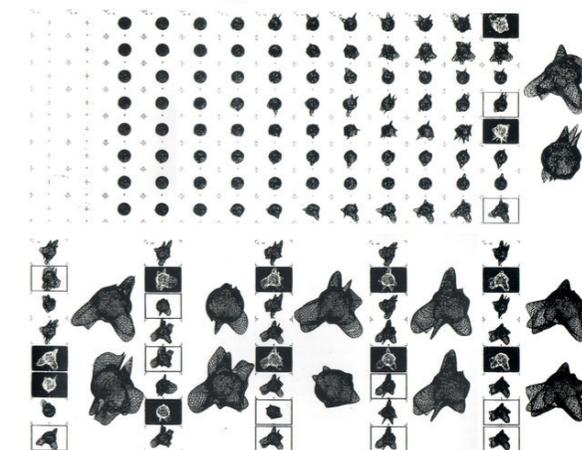
Da questo tipo di approccio derivano gli “**algoritmi genetici**”, cioè parametri in forma codificata, che possono controllare il problema posto. Essi vengono per lo più utilizzati nelle situazioni in cui è relativamente semplice definire i problemi posti e i criteri scelti da rispettare per arrivare alla soluzione <sup>[10]</sup>.

Dagli algoritmi genetici derivano anche i “biomorfi”, sviluppati da Richard Dawkins, una classe di strutture ad albero, bidimensionali e ricorsivi. La loro crescita è controllata da una rappresentazione genetica, che determina un particolare aspetto della struttura (per esempio il numero di bracci). Si tratta, però, di una metodologia limitata, poiché tutte le informazioni contenute nelle scelte scartate vengono perse <sup>[10]</sup>.

Un passo avanti è stato fatto con i “sistemi di classificazione”, che non si concentrano più sulla scelta della migliore soluzione, ma sulla popolazione che ha imparato a rispondere in maniera più soddisfacente agli input dati. Il sistema riceve le informazioni dall'ambiente, vengono poi controllate con le regole imposte, che vengono quindi applicate <sup>[10]</sup>.

Infine, per riuscire ad ottenere i risultati sperati,

in modo da riuscire a controllare problemi più complessi con un grande numero di iterazioni, Frazer introdusse anche gli “algoritmi gerarchici genetici”, dove un algoritmo riesce ad imparare i criteri per selezionare il risultato di quello successivo <sup>[10]</sup>.



*Esempio di algoritmo genetico. Il sistema risponde ad un insieme di input ambientali e valuta il successo della risposta. La natura della risposta si basa sul feedback dell'ambiente e le risposte più riuscite vengono gradualmente sviluppate.  
Fonte: Frazer J., “An Evolutionary Architecture”, Architecture Association, Londra 1995*

## ARCHITETTURA PARAMETRICA OGGI

L'architettura parametrica oggi non è estremamente differente da quella descritta da Moretti negli anni '40, ma è cambiata la terminologia ad essa legata.

Wassim Jabi nel 2013 definisce il progetto parametrico come **“un processo basato sul pensiero algoritmico che permette l'espressione di parametri e regole che insieme definiscono, codificano e chiarificano la relazione tra gli intenti della progettazione e le risposte della progettazione”**<sup>[11]</sup>.

Il progetto è basato su un modello parametrico, cioè **“un algoritmo che genera modelli costituiti da una geometria e da caratteristiche (ad esempio il materiale). Questo algoritmo utilizza funzioni e variabili, comprendendo sia variabili dipendenti che indipendenti. Alcune delle variabili indipendenti possono avere uno status più di primo piano, come interfaccia al modello parametrico - questi sono indicati come parametri del modello”** (Patrick Janssen)<sup>[11]</sup>.

Da questo si riesce a capire come diversi sistemi parametrici possano portare a stili ed estetiche diversi e come l'uso del parametrico serva prima di tutto a descrivere una geometria

flessibile, più che a definire uno stile preciso<sup>[11]</sup>.

Nella Biennale di Architettura di Venezia del 2008 nasce, per mano di **Patrick Schumacher**, il termine e il concetto di “Parametrismo”, definito nel **manifesto** come il nuovo grande stile dopo il Modernismo e in cui una delle principali tecniche è l'utilizzo di elementi modulari e adattabili. **“L'architettura d'avanguardia contemporanea sta rispondendo alla domanda di un maggiore livello di complessità articolata attraverso il riattrezzamento dei suoi metodi sulla base di sistemi di progettazione parametrica.**



Zaha Hadid Architects, Kartal-Pendik Masterplan, Istanbul, Turkey, 2006. Studio del tessuto urbano. Variazione morfologica dei blocchi.

Fonte: Schumacher P., “Parametricism - A New Global Style for Architecture and Urban Design”, in “AD Architectural Design - Digital Cities”, Vol. 79, Vol. 4, Luglio/Agosto 2009

*Lo stile architettonico contemporaneo che ha raggiunto una pervasiva egemonia all'interno delle avanguardie architettoniche contemporanee può essere inteso al meglio come un programma di ricerca basato sul paradigma parametrico. Proponiamo di chiamare questo stile: Parametrismo.”*<sup>[12]</sup>

Più recentemente, nel 2012, Schumacher ha coniato il termine “Parametricism 2.0”, per rilanciare l'idea di Parametrismo, non solo più incentrata su un'intenzionalità stilistica, ma anche su problemi sociali e ambientali<sup>[13]</sup>.

Il Parametrismo è l'unico stile architettonico in grado di sfruttare a pieno la rivoluzione digitale degli ultimi anni e che riesce a rispondere alle nuove esigenze strutturali e ambientali. Grazie al suo valore fondamentale di adattabilità e flessibilità, implica anche un certo grado di differenziazione regionale, permettendo di adeguarsi al clima, al sito di progetto, ai materiali locali...<sup>[2]</sup>.

Perché il Parametrismo abbia successo è necessario sfruttare le nuove e sempre più complesse tecnologie digitali, che devono, però, al contempo essere accompagnate dai valori e tecniche già emersi nel processo progettuale prima dell'avvento del digitale, in modo da evitare di rimanere nello sterile estetismo<sup>[10]</sup>.

Nelle parole di Luigi Moretti, **“l'architettura parametrica è in grado di aprire le porte per forme nuove e rivoluzionarie e di dare vita a nuovi comportamenti e approcci umani”**<sup>[11]</sup>.

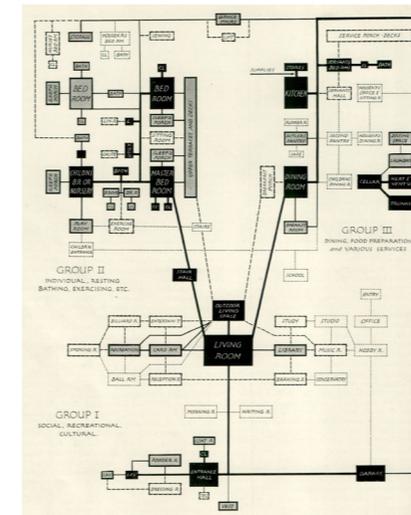
# 1.3

## La logica del processo parametrico: il diagramma

Nella progettazione architettonica il disegno è sicuramente uno degli strumenti fondamentali per descrivere e immaginare gli spazi, ma per gestire e capire i processi che generano il progetto possono essere utili strumenti, digitali o meno, di sintesi dei dati [14].

Il diagramma è uno di questi strumenti, in grado di **sintetizzare e analizzare** dati complessi, rappresentando al contempo le dinamiche e i cambiamenti del progetto, è la spazializzazione di un'astrazione o riduzione di un concetto o fenomeno.

Il diagramma si trova tra la forma e la parola, per questo è spesso considerato un dispositivo performativo più che rappresentativo [15].



The country house chart. Room by room. Diagramma funzionale. Fonte: Garcia M., "The diagrams of architecture", John Wiley & Sons, Hoboken 2010

Essendo stati utilizzati in diverse forme dall'inizio delle pratiche architettoniche, hanno avuto anche diverse definizioni e funzioni, arrivando solo nel Ventesimo secolo ad essere l'argomento principale di testi specialistici [16].

Come mezzo è anche in grado di organizzare le informazioni e le regole del progetto, diventando utile non solo come strumento di sintesi, ma anche come strumento conoscitivo e decisionale per trovare nuove possibili soluzioni. Il diagramma può descrivere sia il contesto sia la forma, offrendo un mezzo per verificare il primo e trovare la seconda. Per questo è uno strumento essenziale nella progettazione, poiché è in grado di creare un ponte tra i requisiti e la forma [17].

I diagrammi riescono a gestire un gran numero di parametri e variabili, aiutando nella generazione di nuove forme spaziali [14].

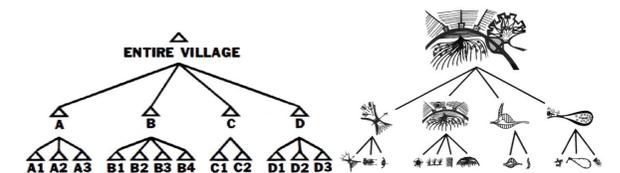
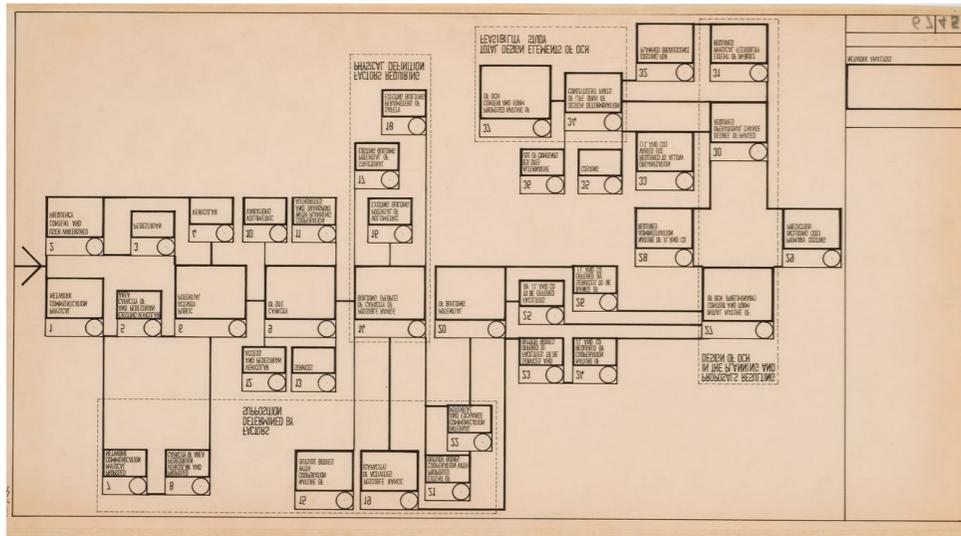


Diagramma per studiare la riorganizzazione di un villaggio indiano. Fonte: Alexander C., "Note sulla sintesi della forma", Saggiatore, Milano 1973

Ciò è alla base di numerose teorie nate dagli anni '60 in poi, come il concetto di "objectile", dove l'oggetto diventa classe di oggetti, ognuno differente dall'altro, ma con una matrice comune, introdotto in campo architettonico da Gilles Deleuze, che definisce il diagramma come "una macchina astratta... una mappa di relazioni fra le forze".

Da questo anche numerosi altri ricercatori danno la loro spiegazione, come Anthony Vidler che, insoddisfatto della definizione enciclopedica, è convinto che "il diagramma illustra una definizione, aiuta nella dimostrazione di una proposizione, rappresenta il corso dei risultati di qualsiasi azione o processo" [16].

Nella **progettazione contemporanea** l'uso di diagrammi è ancora ampiamente sfruttato, basti pensare ai progetti per il Parc de la Villette a Parigi nel 1983. Il vincitore del concorso, Bernard Tschumi, considera il diagramma come "una rappresentazione grafica di un concept", viene utilizzato per descrivere l'interazione tra lo spazio e il movimento dei corpi che lo occupano. Il diagramma, quindi, non è solamente sfruttato per l'analisi dello spazio di progetto, ma anche come strumento in grado di aiutare nello sviluppo futuro dell'architettura [16].



Cedric Price. Oxford Corner House, London. Diagramma di connessioni.  
Fonte: <https://www.cca.qc.ca/en/articles/vol.s/4/origins-of-the-digital/74854/information-archaeologies>

Nella pagina seguente: UNStudio. Diagramma dell'afflusso di persone nella stazione centrale di New York.  
Fonte: <https://www.unstudio.com/en/page/12062/ifcca>

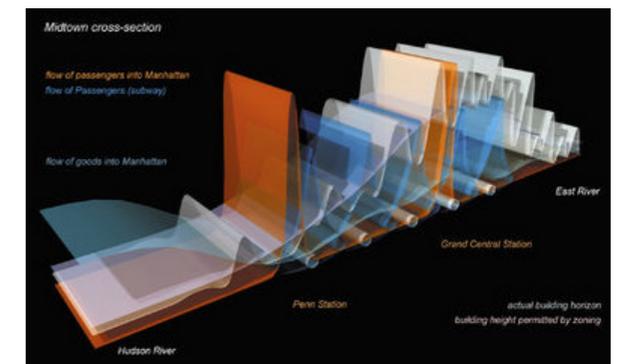
Patrick Schumacher, padre del Parametrisimo, distingue fra **diagrammi ordinari ed extra-ordinari**. Si parla del primo quando la relazione fra il diagramma astratto e l'entità concreta rappresentata è chiara, mentre il secondo è inteso in senso deleuziano, cioè una macchina astratta che lascia risultati aperti, si possono definire quasi come delle proto-rappresentazioni; inoltre, distingue anche fra diagrammi metrici e parametrici, i primi hanno caratteristiche fisse e determinate, i secondi variabili e limitati all'interno di un intervallo stabilito [16].

Al giorno d'oggi la maggior parte dei diagrammi utilizzati sono categorizzabili come ordinari e parametrici, utilizzati anche in tecniche digitali di scripting e programmi di progettazione parametrica. Infatti, l'utilizzo dei diagrammi è facilmente visibile nell'uso di software di progettazione, mostrando un discostamento dal mero esito formale, ma ponendo maggiore risalto al processo applicato per giungere alla soluzione o soluzioni finali [16].

Uno degli studi che si occupa approfonditamente della ricerca sui diagrammi è UNStudio. Il diagramma è considerato come uno strumento per creare nuove soluzioni innovative, attraverso la compressione di informazioni e

viene utilizzato soprattutto per spingere l'architettura verso forme sempre più astratte. Secondo i fondatori, Ben van Berkel e Caroline Bos, il diagramma riesce a unire il tempo e l'azione nel processo progettuale, in un modo che i modelli tridimensionali non sono in grado di fare. Non è un modello per il progetto e non rappresenta un oggetto esistente, ma è essenziale per la sua produzione, introducendo anche caratteristiche intuitive e sconnesse da una logica lineare [16].

Globalmente, gli studi di architettura si affidano ancora ai diagrammi per riuscire a collegare differenti aspetti della progettazione, poichè sono fra i pochi strumenti in grado di avere un alto grado di efficienza e versatilità. Spesso vengono anche utilizzati come mezzo per arrivare a risultati temporanei o dare una forma a un'idea iniziale, diventando estremamente utili per valutare le opzioni possibili e giungere più facilmente alla soluzione finale [17].



## NOTE

1 Carpo M., *Parametric Notations*, in “Architectural Design”, vol. 86, Marzo 2016, pp. 24-29

2 Schumacher P., *Gearing up to impact the global built environment*, in “AD. Parametricism 2.0. Rethinking architecture’s agenda for the 21st century”, vol. 86, vol. 2, Marzo/Aprile 2016, pp. 8-17

3 Hernandez C. R. B., *Thinking parametric design: introducing parametric Gaudí*, in “Design Studies”, vol. 27, vol. 3, Maggio 2006, pp. 309-324

4 Songel J. M., *Frei Otto and the debate about the genesis of architectural form*, in “EGA. Revista de Expresión Gráfica Arquitectónica”, vol. 15, Ottobre 2010

5 Burry M., *Antoni Gaudí and Frei Otto: Essential Precursors to the Parametricism Manifesto*, in “AD. Parametricism 2.0. Rethinking architecture’s agenda for the 21st century”, vol. 86, vol. 2, Marzo/Aprile 2016, pp. 30-35

6 Tedeschi A., *AAD Algorithms-Aided Design. Parametric strategies using grasshopper*, Le Penseur, Brienza 2014

7 Pota G., *Parametrizzare la mente dell’architetto. Il caso di Tour Bois le Prêtre*, in “Ardeth”, Vol. 5, Novembre 2019, pp. 78-95

8 Carpo M., *The Digital Turn in Architecture 1992-2012*, John Wiley & Sons, Chichester 2012

9 Carpo M., *The alphabet and the algorithm*, MIT Press, Cambridge 2011

10 Frazer J., *An Evolutionary Architecture*, Architecture Association, Londra 1995

11 Frazer J., *Parametric Computation: History and Future*, in “AD. Parametricism 2.0. Rethinking architecture’s agenda for the 21st century”, vol. 86, vol. 2, Marzo/Aprile 2016, pp. 18-23

12 Schumacher P., *Parametricism as Style. Parametricism Manifesto*, Londra 2008

13 Schumacher P., *The Autopoiesis of Architecture*, Vol II: “A New Agenda for Architecture”, John Wiley & Sons, Chichester 2012

14 Barosio M., Gugliotta R., *Dal tipo al diagramma: la rappresentazione del dato come strumento euristico per il progetto di architettura*, in “Dienne. Building Information Modeling, Data & Semantics”, vol. 7, Dicembre

2020, pp. 28-39

15 Somol R. E., *Dummy Text, or The Diagrammatic Basis of Contemporary Architecture*, Universe, New York 1999

16 Garcia M., *The diagrams of architecture*, John Wiley & Sons, Hoboken 2010

17 Alexander C., *Note sulla sintesi della forma*, Saggiatore, Milano 1973

18 Fedorchenko M., *The tools of mediation: Extending the Diagrammatic Project*, in “Philosophy Kitchen - Rivista di filosofia contemporanea”, Vol. 3, 2019

**capitolo 2**

**Prima fase**

**Il progetto di  
concorso**

# 2.1

## Il bando

Come spunto iniziale per la mia tesi ho scelto di partecipare ad un concorso di architettura in cui potessi concentrarmi sull'aspetto processuale del progetto e applicare le teorie precedentemente apprese e illustrate.

Il concorso scelto fa parte di una serie di challenge indette da Bee Breeders volti a trovare **soluzioni per il problema abitativo** in varie parti del mondo.

In questo caso si tratta del Melbourne Affordable Housing Challenge, in cui l'area di progetto è, appunto, l'area metropolitana di Melbourne, nello stato del Victoria in Australia sud-orientale.

Melbourne è la città con il più alto tasso di crescita della popolazione nel paese. In un anno, dal 2016 al 2017, ha avuto un tasso di crescita del 2.7%, con un incremento di cittadini di 125'000 unità. Le previsioni affermano che entro il 2021 la popolazione arriverà a superare i 5 milioni e gli 8 milioni entro il 2050. Questo richiederà la costruzione di più di 1.6 milioni di nuove case nei prossimi 35 anni.

L'aumento esponenziale della domanda di alloggi, non accompagnato dallo stesso livello di crescita dell'offerta, ha causato un aumento massiccio nei costi delle case.

Il prezzo medio attuale di una casa nell'area metropolitana di Melbourne è di 836'000 \$,

mettendole fuori portata per la maggior parte delle persone, soprattutto le famiglie.

Il concorso richiede di trovare una possibile soluzione a questa crisi abitativa della città.

L'area di progetto non è delineata nello specifico, quindi, si richiede che il progetto sia abbastanza flessibile da potersi adattare a diverse zone di Greater Melbourne.

Non sono definite caratteristiche particolari, ma che si tenga in considerazione le varie possibili dimensioni abitative in grado di adattarsi alle diverse tipologie di utenze (dal singolo alla famiglia).

Si richiede di progettare un concept di alloggio a **prezzi accessibili**, che possa essere implementato facilmente per aumentare la capacità residenziale all'interno della città.

Deve essere minimo nell'uso di terreno e materiali, ma non è definita nessuna dimensione minima per l'alloggio.

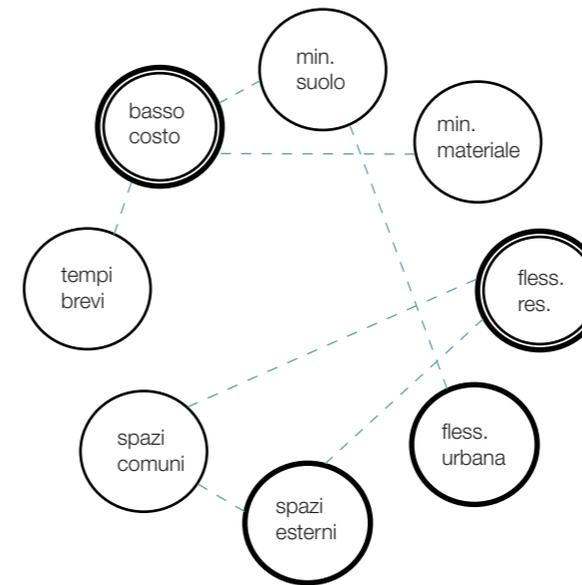
Le soluzioni dovrebbero, inoltre, essere abbastanza **flessibili** da potersi adattare a diverse capacità residenziali e diverse località in tutta la città.

## 2.2

# I requisiti

### FASE INIZIALE

Lo step iniziale del processo è stato quello di definire in modo puntuale i **requisiti** del progetto, unendo le richieste fatte dal concorso e le caratteristiche che ho riscontrato potessero essere utile per giungere ad una soluzione soddisfacente.



Dati anche gli ultimi avvenimenti dell'anno passato e quello in corso ho ritenuto necessario aggiungere delle caratteristiche che si sono riscontrate indispensabili per numerose persone.

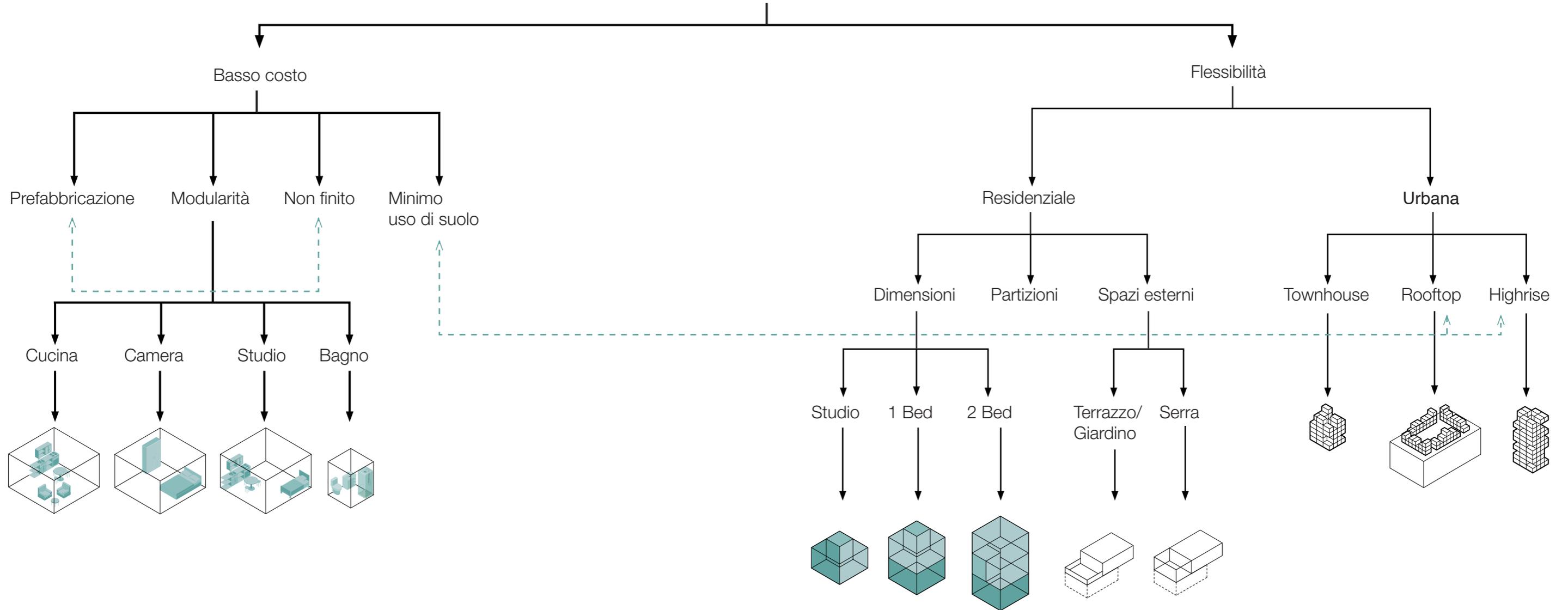
Quindi, oltre ad essere flessibile, avere prezzi accessibili e utilizzare meno materiale e suolo possibile, l'alloggio tipo del progetto deve ave-

re un accesso a uno spazio aperto, in modo da poter avere uno sbocco sull'esterno anche quando non può essere garantito nella vita di tutti i giorni; inoltre il blocco di alloggi che si va a formare presenta una serie di spazi comuni, che possono essere modificati in base alle esigenze e infine, ho ritenuto necessario che la costruzione possa essere completata in tempi rapidi, in modo da riuscire a fornire nuove abitazioni velocemente, soprattutto in tempi di possibili crisi emergenziali.

Successivamente attraverso un **diagramma** i requisiti più importanti sono stati scomposti in sottocategorie, in modo da poter giungere alla definizione delle caratteristiche essenziali del progetto.

Il diagramma, inoltre, aiuta anche a comprendere le relazioni tra i parametri scelti, favorendo la selezione delle soluzioni più efficaci. Da questo si è giunti ad alcune decisioni fondamentali per il progetto, come il processo di produzione e le tipologie di combinazione degli alloggi.

# AFFORDABLE HOUSING



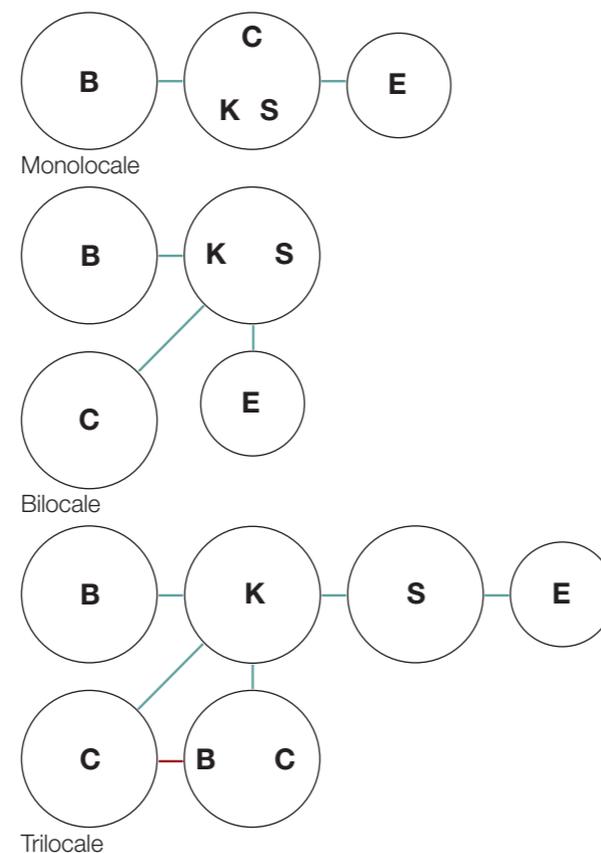
I requisiti principali sono, come si è detto, il basso costo degli alloggi e la flessibilità del progetto. Scomponendo il problema diventa evidente che il basso costo può essere raggiunto lavorando su almeno due fronti: la produzione di elementi prefabbricati e l'uso minimo di suolo. La prima strategia porta a pensare di poter realizzare strutture modulari, che, allo stesso tempo, facilitano la diminuzione dei tempi di costruzione, mentre la seconda si ricollega alla flessibilità urbana del progetto, riuscendo ad adattare l'edificio in contesti in cui si ha una scarsità di spazi e quindi il costo del terreno aumenta esponenzialmente.

Al fine di risolvere la crisi abitativa, come richiesto dal concorso, la soluzione più interessante è abbinare la **costruzione modulare** con l'**abitazione minima**, in questo modo si riesce a combinare l'esigenza di utilizzo di minor spazio possibile e mantenere i costi accessibili. Grazie alla progettazione di spazi minimi si riescono a sfruttare tutti quegli spazi lasciati liberi dal tessuto urbano ormai estremamente denso, come i vuoti urbani di dimensioni ridotte o, addirittura, coperture piane inutilizzate.

## L'ABITARE MINIMO

L'idea dell'alloggio minimo prende piede a partire dagli inizi del 1900, in risposta al problema abitativo per le masse popolari nelle città industrializzate, soprattutto in Germania. A svilup-

pare a fondo queste teorie furono in particolare i maestri del razionalismo, fra cui **Alexander Klein**, che pubblica numerosi scritti focalizzati su metodi oggettivi e scientifici per la progettazione di unità immobiliari minime, tenendo sempre in primo piano il rapporto tra la forma e le esigenze fisiche e psicologiche dell'utente<sup>[1]</sup>.



Diagrammi studio iniziale delle piante del monocale, bilocale e trilocale

## LA PROGETTAZIONE MODULARE

La progettazione modulare dello spazio abitativo rende certamente più facile creare luoghi flessibili e adattabili. Inoltre, le case modulari possono essere prodotte molto più velocemente, essendo costruite in fabbrica e poi trasportate nell'area del progetto.

L'idea è stata sviluppata nel 1930 in seguito alle prime Conferenze Internazionali di Architettura Moderna. **Le Corbusier** iniziò a formulare l'idea di costruire case in serie, create in fabbrica come se fossero aerei o automobili, create rapidamente e a basso costo. In questo modo, la casa sarebbe diventata uno strumento, accessibile a tutti, sano e sicuro<sup>[2]</sup>. La progettazione modulare unita all'utilizzo degli spazi minimi previsti dal CIAM ha permesso di avere meno sprechi possibili e una migliore efficienza dello spazio utilizzato.

Negli ultimi anni sta tornando in scena il design modulare e prefabbricato, poiché questo tipo di progettazione consente di creare nuovi spazi abitativi per le emergenze in breve tempo, con costi contenuti e a basso impatto sull'ambiente.

Gli alloggi modulari permettono di utilizzare materiali più eco-sostenibili e riciclabili e consentono di produrre meno rifiuti, poiché è possibile controllare la produzione in modo molto preci-

so in base alle esigenze dell'utenza<sup>[3]</sup>. Pertanto, è possibile creare moduli di costruzione con un alto livello tecnico, ma riuscendo a raggiungere costi più accessibili e mantenendo la flessibilità nella progettazione.

Grazie a questo approccio, ogni modulo può essere scambiato, modificato o eliminato nell'aggregazione finale, senza grossi problemi. Si tratta di un tipo di design molto versatile, che, a seconda delle esigenze, può essere utilizzato per installazioni permanenti o temporanee e consente anche un alto grado di personalizzazione, sia a livello globale dell'edificio che a livello del singolo modulo.

Grazie all'utilizzo della progettazione modulare i diversi blocchi possono essere assemblati in modalità differenti. In questo modo si creano svariate soluzioni, che si riescono ad adattare a numerose situazioni.

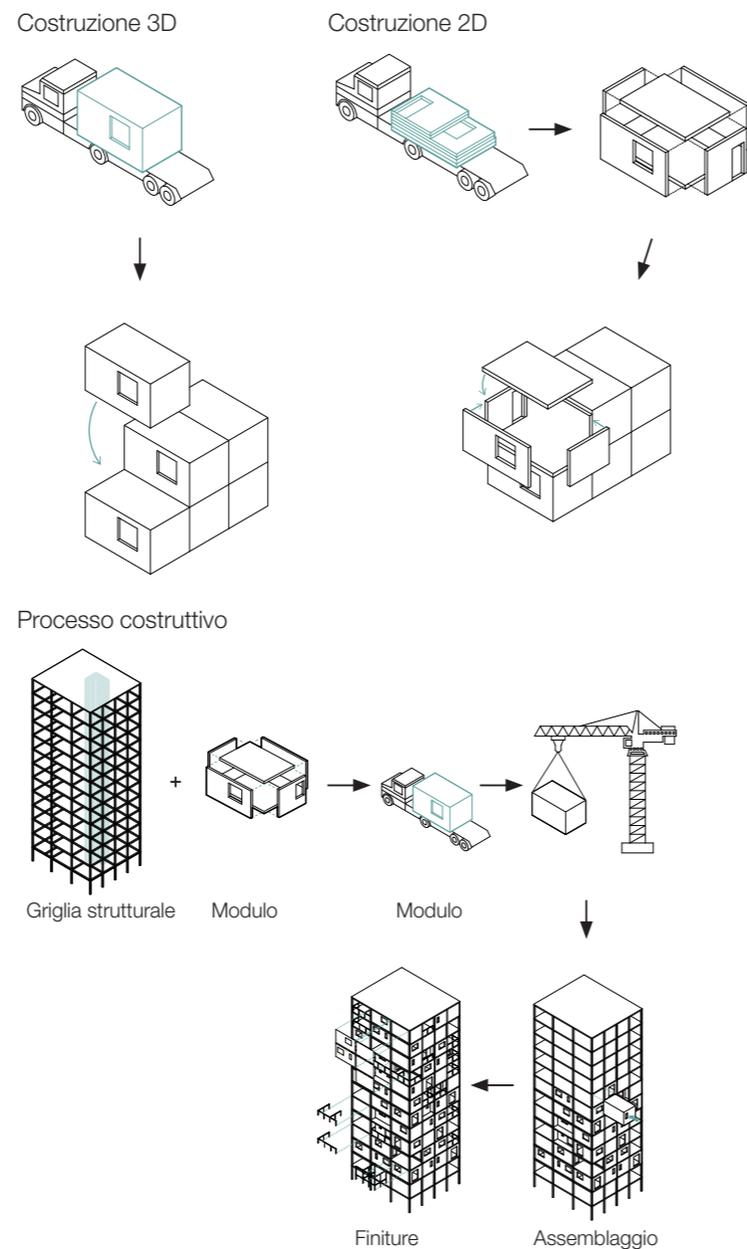
Nel caso specifico del concorso si richiedeva di avere un progetto che fosse flessibile per le varie zone di Melbourne, quindi si è pensato a **tre soluzioni possibili** per l'aggregazione delle abitazioni: la townhouse, blocchi residenziali più bassi, per adattarsi alle zone più gentrificate della città, caratterizzate da case a pochi piani, l'highrise, una soluzione adatta alle zone con un tessuto urbano più denso e il rooftop, la sopraelevazione in cui si riescono a sfruttare i tetti piani di numerosi edifici, altrimenti inutilizzati.

Il **processo costruttivo** legato alla progettazione modulare aiuta a diminuire i tempi di costruzione dal 20 al 50%<sup>[4]</sup>, permettendo tempistiche più rapide anche in situazioni emergenziali.

La parte strutturale del blocco residenziale, in acciaio, viene costruita in cantiere, mentre, nello stesso momento, i vari moduli di appartamenti vengono creati in fabbrica.

I moduli possono essere trasportati direttamente in 3D, già montati, oppure scomposti nei loro elementi, in caso le dimensioni scelte fossero troppo grandi per il trasporto su ruota. Una volta arrivati in cantiere, possono essere inseriti nella griglia strutturale, in breve tempo e senza la necessità di manodopera specializzata.

Inoltre, i moduli possono, potenzialmente, essere spostati, permettendo nuove configurazioni dell'edificio o agli utenti di trasferire i propri alloggi in altre zone o in altre città.



A lato: diagrammi del processo di costruzione modulare

## 2.3

# Casi studio

Prima di definire il progetto vero e proprio ho eseguito una ricerca su vari casi studio di affordable housing, in modo da riuscire in seguito a definire con maggior precisione e coscienza le caratteristiche e requisiti necessari.

I casi studio sono stati divisi in base alle caratteristiche prese maggiormente in considerazione ai fini di poter essere applicate anche nel progetto che si sta sviluppando.

LEGENDA:

-  MODULARITÀ
-  STRUTTURA
-  SPAZI ESTERNI

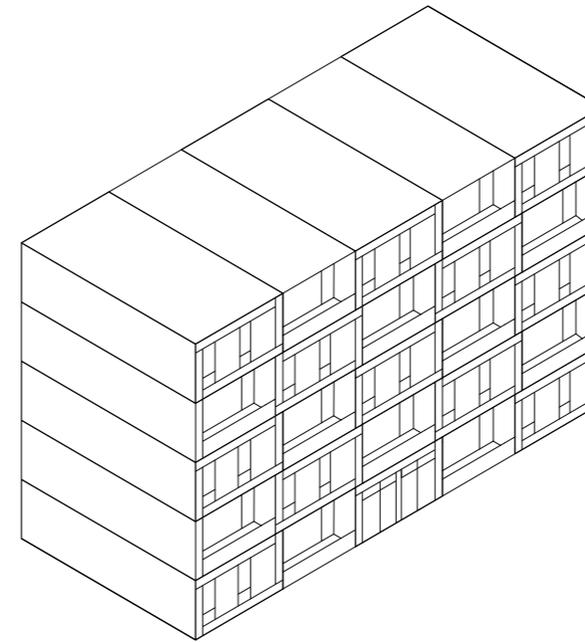
**Progetto:** Dortheavej Residence  
**Localizzazione:** Copenhagen, Danimarca  
**Anno di realizzazione:** 2018  
**Architetti:** BIG

L'edificio offre 66 nuove case per cittadini con basso reddito nella zona nord-ovest di Copenhagen, rispondendo alla richiesta dell'associazione non-profit Lejerbo.

Il progetto è creato da 66 moduli impilati per un totale di 5 piani. L'edificio leggermente curvo delimita una corte pubblica.

Gli elementi prefabbricati sono impilati in modo da dare a ogni appartamento un piccolo terrazzo e ai blocchi cucina-soggiorno 1 m in più di altezza. Sul lato rivolto verso la corte interna si hanno grandi finestre a tutt'altezza, per dare più luce naturale possibile.

Le dimensioni degli appartamenti spaziano da 60 a 115 m<sup>2</sup> e sono stati utilizzati materiali semplici, come legno e cemento, per poter creare spazi abitativi generosi.



A sinistra: Assonometria di parte dell'edificio  
 In basso: Diagramma distribuzioni e funzioni di due alloggi tipo

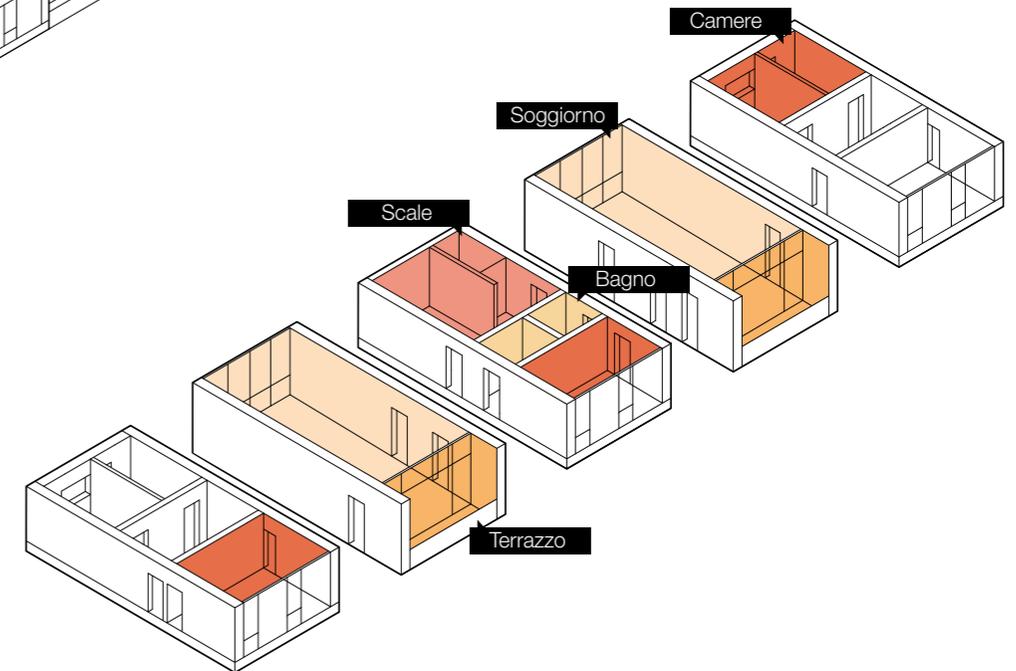


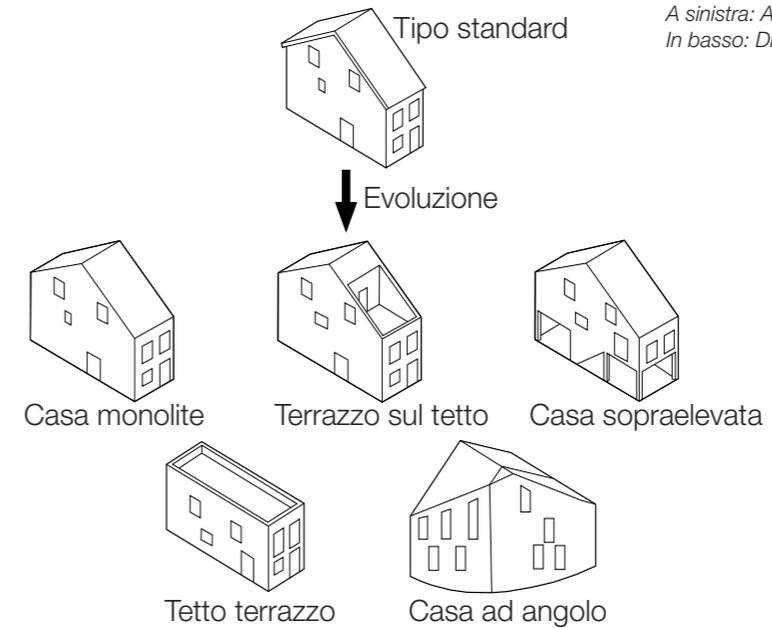
Foto in alto: Vista dalla corte interna  
 Foto in basso: Vista interna dalla cucina-soggiorno  
 Fonte: <https://www.archdaily.com/903495/homes-for-all-dortheavej-residence-bjarke-ingels-group>

**Progetto:** Traumhaus Funari  
**Localizzazione:** Mannheim, Germania  
**Anno di realizzazione:** 2015  
**Architetti:** MVRDV

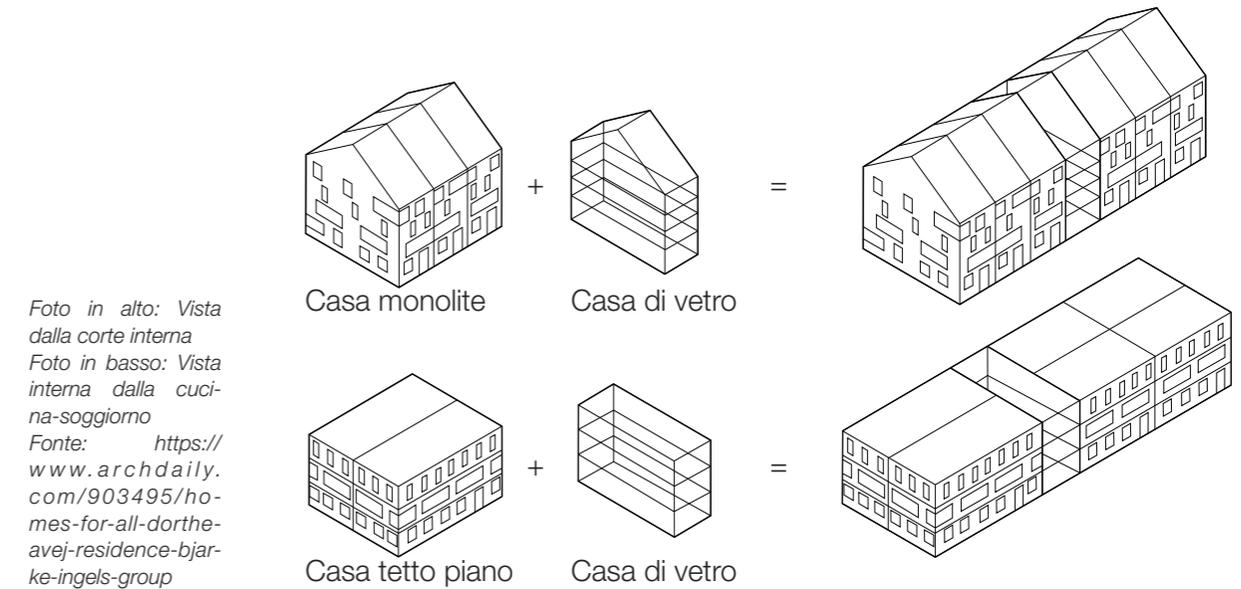
Gruppo di edifici a Mannheim in Germania. Creati in collaborazione con Tramhaus, produttore di alloggi di alta qualità a basso costo, grazie all'uso di elementi standardizzati.

Sono state pensate abitazioni di varie tipologie per adattarsi a diversi tipi di famiglie e demografia. Ogni casa ha un giardino privato personalizzabile.

I vari modelli di abitazione vengono anche combinati in due blocchi di appartamenti, che saranno quindi caratterizzati da mixité sociale.



A sinistra: Assonometria di parte dell'edificio  
 In basso: Diagramma distribuzioni e funzioni di due alloggi tipo

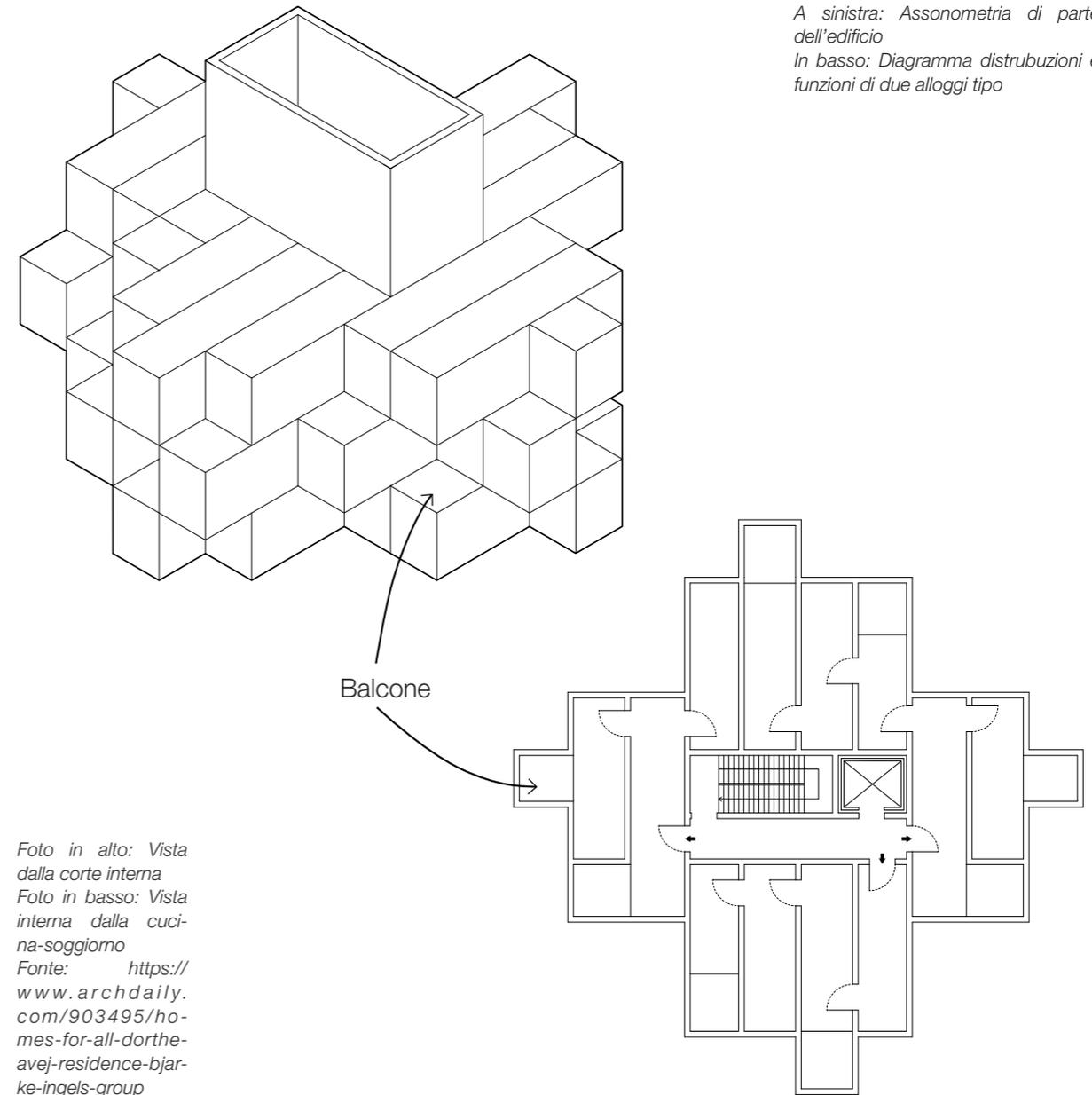


Grattacielo di 18 piani con struttura in legno. Oltre alle unità residenziali ospita anche aree pubbliche, come bar, asilo, laboratori.

Unici elementi fissi sono le fondazioni e il nucleo di base, le unità abitative vengono pre-assemblate e sono costituiti da pannelli CLT prefabbricati.

Ogni appartamento è caratterizzato da grandi terrazze per supportare grandi sistemi di vegetazione

**Progetto:** Toronto Tree House  
**Localizzazione:** Toronto, Canada  
**Anno di realizzazione:** 2017  
**Architetti:** Precht



A sinistra: Assonometria di parte dell'edificio  
 In basso: Diagramma distribuzioni e funzioni di due alloggi tipo

Foto in alto: Vista dalla corte interna  
 Foto in basso: Vista interna dalla cucina-soggiorno  
 Fonte: <https://www.archdaily.com/903495/homes-for-all-dorthe-avej-residence-bjarke-ingels-group>

**Progetto:** Alvenaria Social Housing  
**Localizzazione:** Lisbona, Portogallo  
**Anno di realizzazione:** 2013  
**Architetti:** Fala Atelier



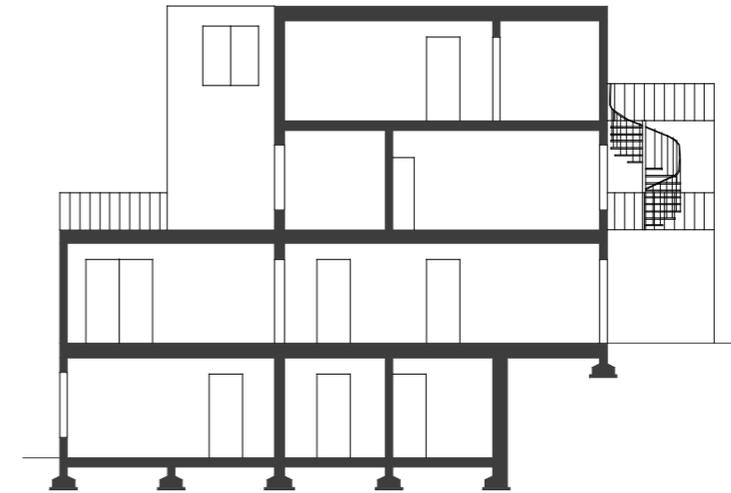
Progetto di concorso per creare una nuova identità del quartiere Alvenaria di Lisbona. Si compone di un modulo base di 2.55 m che subisce variazioni e moltiplicazioni.

L'uso del modulo rende possibile una grande flessibilità per adattarsi alle esigenze dell'utente e fa diventare il progetto interattivo. Gli interni sono arricchiti dalla presenza di spazi esterni, come balconi o giardini.

Il modulo è caratterizzato da una struttura in metallo prefabbricata, sia l'interno che l'esterno, lasciati piuttosto spogli, possono essere modificati per adattarsi alla vita del quartiere.



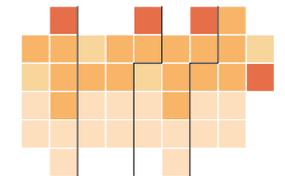
Foto in alto: vista globale estrena  
 Foto in basso: vista da una corte interna  
 Fonte: <https://www.archdaily.com/411410/alvenaria-social-housing-competition-entry-fala-atelier>



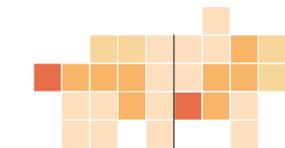
A sinistra: sezione di alloggio tipo  
 In basso: moduli base delle unità abitative e aggregazione dei moduli



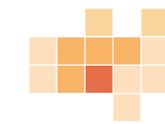
Piano 0



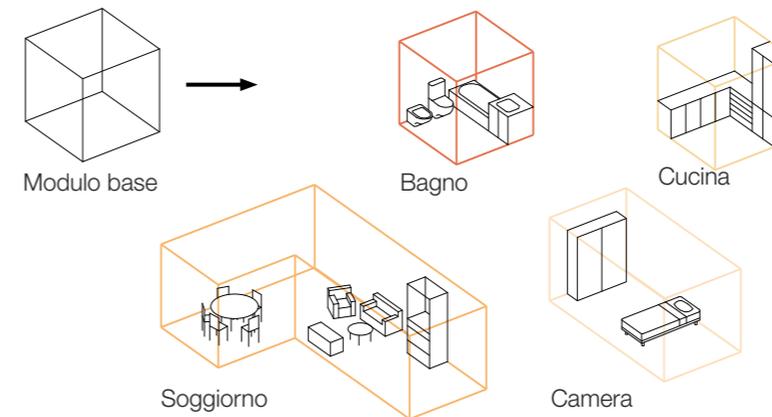
Piano 1



Piano 2



Piano 3



**Progetto:** Nest Toolkit  
**Localizzazione:** Los Angeles, USA  
**Anno di realizzazione:** 2019  
**Architetti:** Brooks + Scarpa

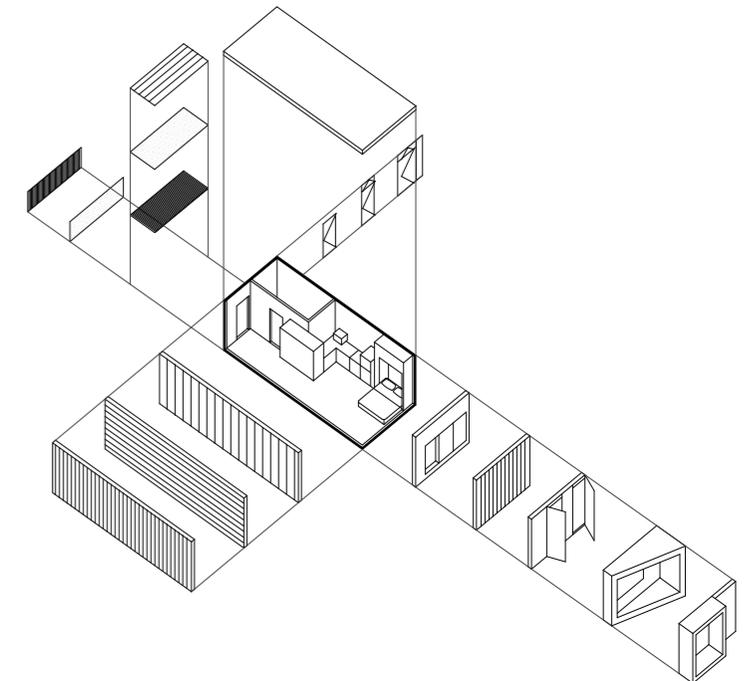
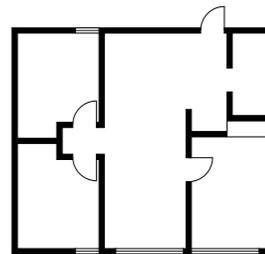
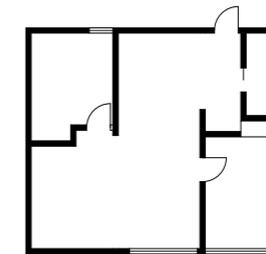
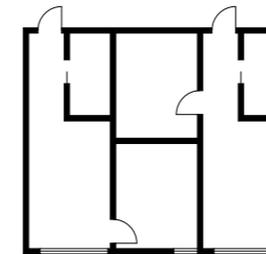
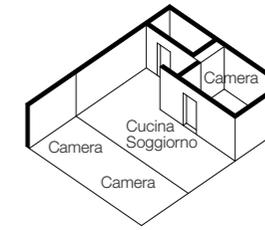
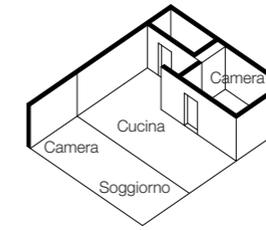
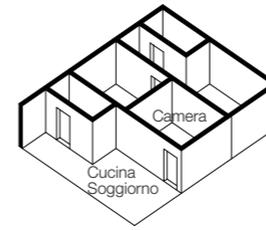
Creato per risolvere la carenza di alloggi a Los Angeles per i senzatetto. Il progetto può essere scalato per adattarsi a diverse aree della città, andando anche a riempire i vuoti urbani più piccoli.

Sono stati progettati diverse tipologie di unità abitative, a partire da situazioni temporanee fino ad alloggi familiari.

I moduli sono costituiti da elementi prefabbricati per ridurre i costi e le tempistiche di costruzione.



Viste esterne di possibili soluzioni dell'edificio  
 Fonte: <https://www.archdaily.com/927411/brooks-plus-scarpa-design-a-toolkit-for-affordable-housing>



In alto: possibili tipologie di unità abitative: 1 camera, 2 camere, 3 camere  
 A destra: esploso con varie possibilità di elementi che compongono un modulo

Il progetto è studiato per essere flessibile e adattarsi a diverse aree geografiche, situazioni climatiche e terreni.

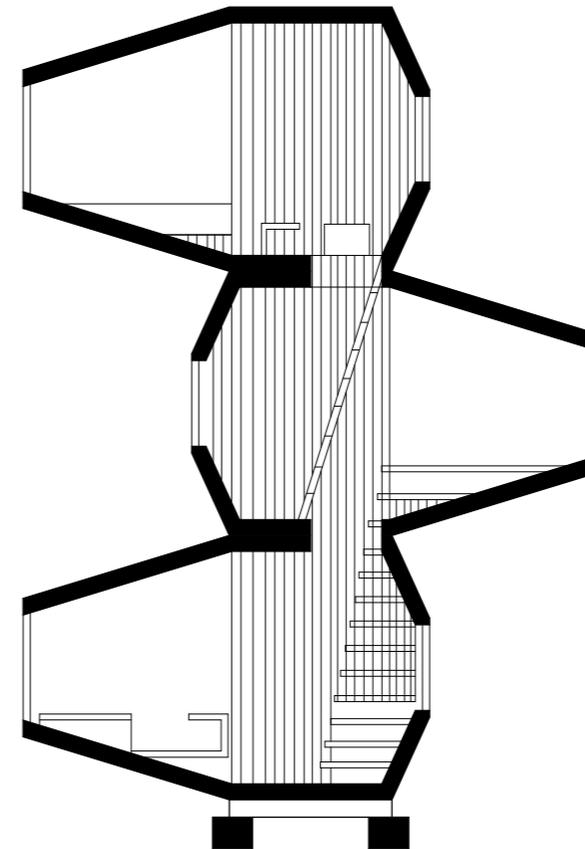
La dimensione minima dell'alloggio lo rende ideale per vacanze, abitazioni per ricerche o rifugi, rendendo possibile anche un trasporto molto semplice.

I moduli, creati con una struttura in legno, possono essere aggregati per formare abitazioni singole o fino a un massimo di 4-6 persone e possono essere combinati verticalmente o orizzontalmente.

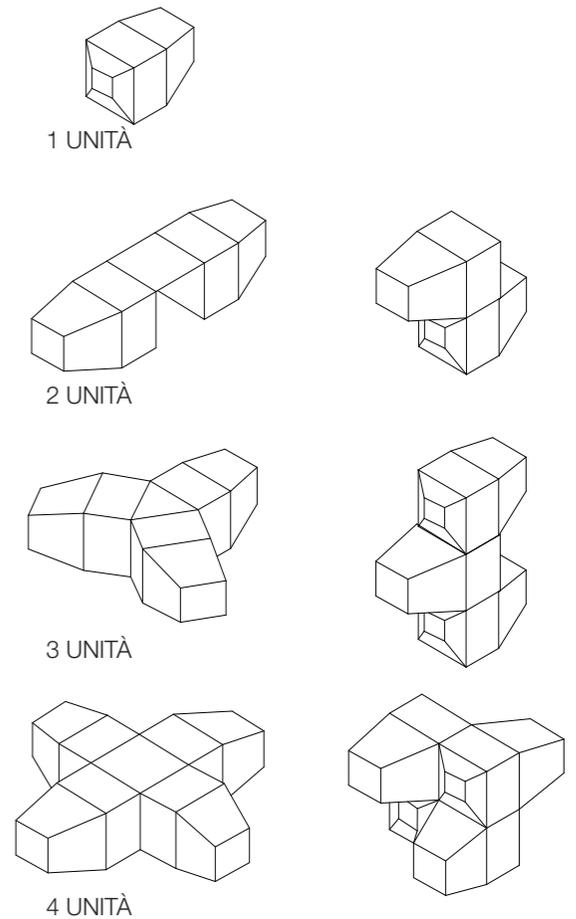
**Progetto:** Living Unit  
**Localizzazione:** Milano, Italia  
**Anno di realizzazione:** 2017  
**Architetti:** Ofis Architects



*In alto: vista esterna complessiva  
 A sinistra: vista interna  
 Fonte: <https://www.archdaily.com/874241/living-unit-ofis-architects>*



*In alto: sezione di un'unità abitativa  
 A destra: possibili aggregazioni dei moduli*

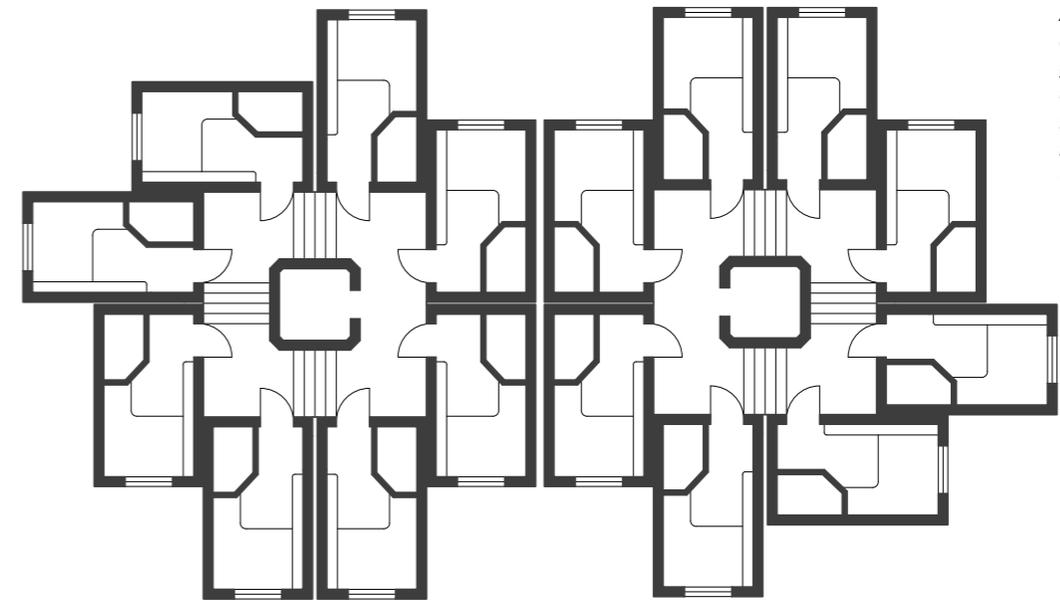


Il modulo che costituisce gli alloggi è stato pensato per ospitare persone che lavorano in centro Tokyo durante la settimana.

Ogni alloggio può essere inserito o eliminato dalla torre in qualsiasi momento, diventando uno dei primi prototipi di architettura sostenibile e attenta al riciclo.

La torre di 14 piani è costituita da 140 moduli di 4 x 2.5 m, che sono stati prodotti in fabbrica e poi trasportati nell'area di progetto.

**Progetto:** Nakagin Capsule Tower  
**Localizzazione:** Tokyo, Giappone  
**Anno di realizzazione:** 1972  
**Architetti:** Kisho Kurokawa



A sinistra: pianta delle torri con aggregazione di moduli  
 In basso: assonometria di un modulo tipo

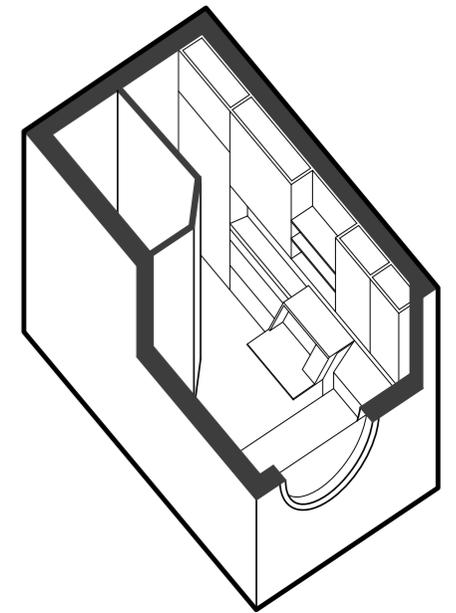


Foto in alto: vista esterna complessiva  
 Foto in basso: Vista interna di un modulo  
 Fonte: <https://www.archdaily.com/110745/ad-classics-nakagin-capsule-tower-kisho-kurokawa>

7

**Progetto:** The Stack Modular Housing  
**Localizzazione:** New York, USA  
**Anno di realizzazione:** 2014  
**Architetti:** GLUCK+

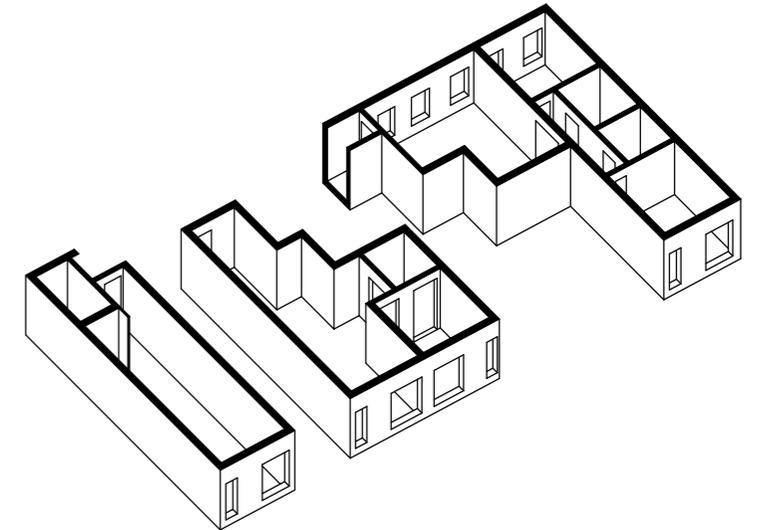
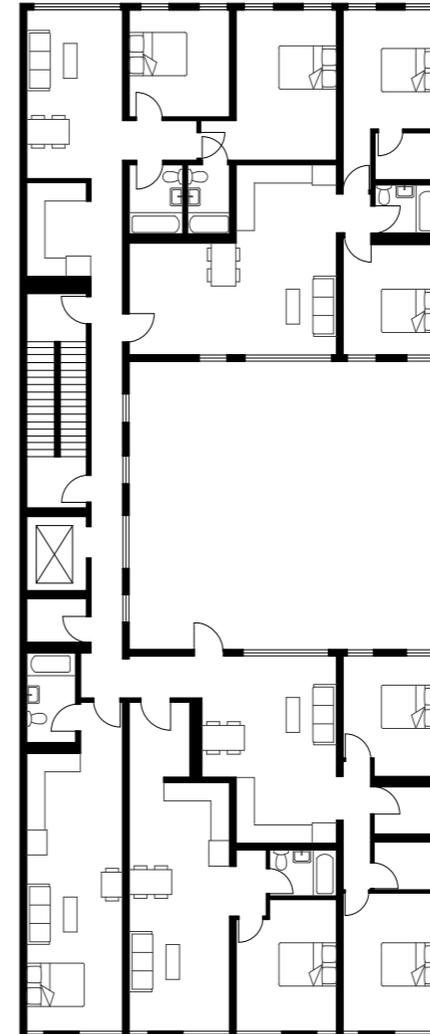
Il progetto cerca di risolvere la necessità di alloggi a prezzi moderati a New York.

Per far ciò si sfrutta un piccolo sito inutilizzato e la costruzione modulare off-site, permettendo la realizzazione di alloggi in tempi più rapidi.

Si tratta del primo edificio prefabbricato in acciaio e cemento di New York, è costituito da 28 unità abitative. La struttura e i moduli vengono costruiti in un periodo di massimo 10 mesi, riducendo le tempistiche classiche di almeno 6 mesi.



In alto: vista esterna  
In basso: vista dal soggiorno  
Fonte: <https://www.archdaily.com/943491/the-stack-modular-housing-in-manhattan-gluck-plus>



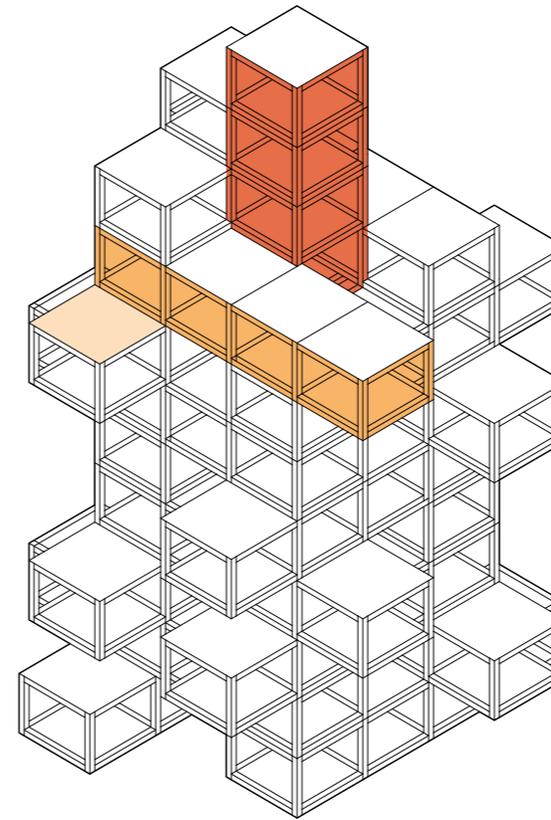
In alto: assonometria dei tre alloggi tipo  
A sinistra: pianta di un piano dell'edificio

La torre è costruita con legname locale ed è caratterizzata dalla presenza di giardini sui tetti delle abitazioni, privati o condominiali.

Gli appartamenti possono essere impilati per un totale di 10 piani e sono costituiti da 4 scatole ciascuno per un totale di circa 75 mq.

Gli alloggi sono pensati per essere a basso costo e velocemente assemblati.

**Progetto:** Tree House  
**Localizzazione:** Various  
**Anno di realizzazione:** 2016  
**Architetti:** Rogers Stirk Harbour + Partners



A sinistra: Assonometria. Evidenziati la distribuzione verticale, un alloggio tipo e lo spazio esterno.  
 In basso: Pianta di un piano tipo

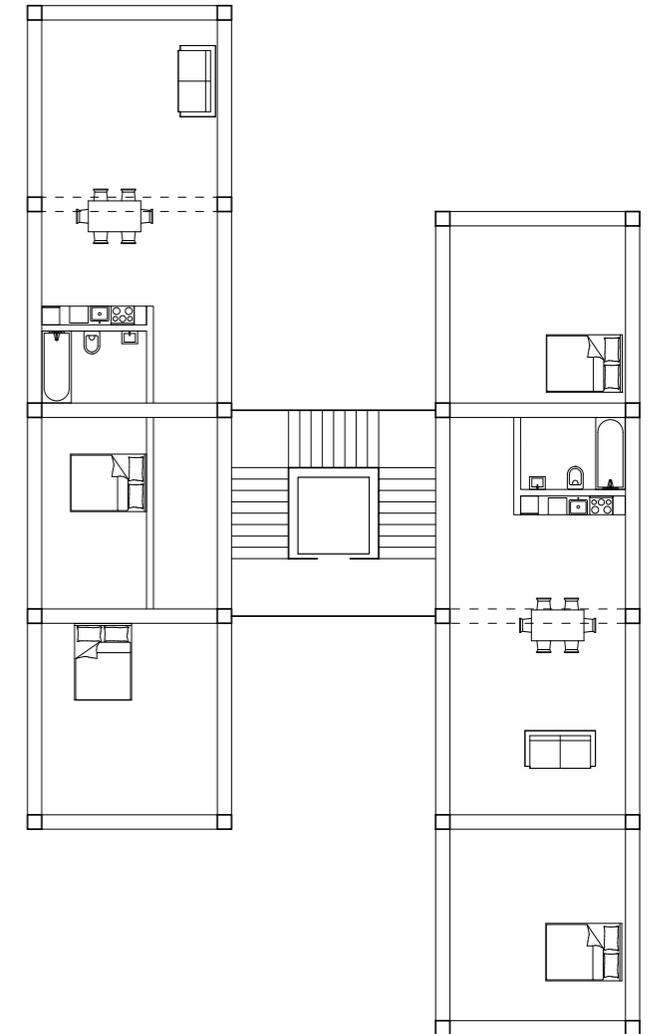
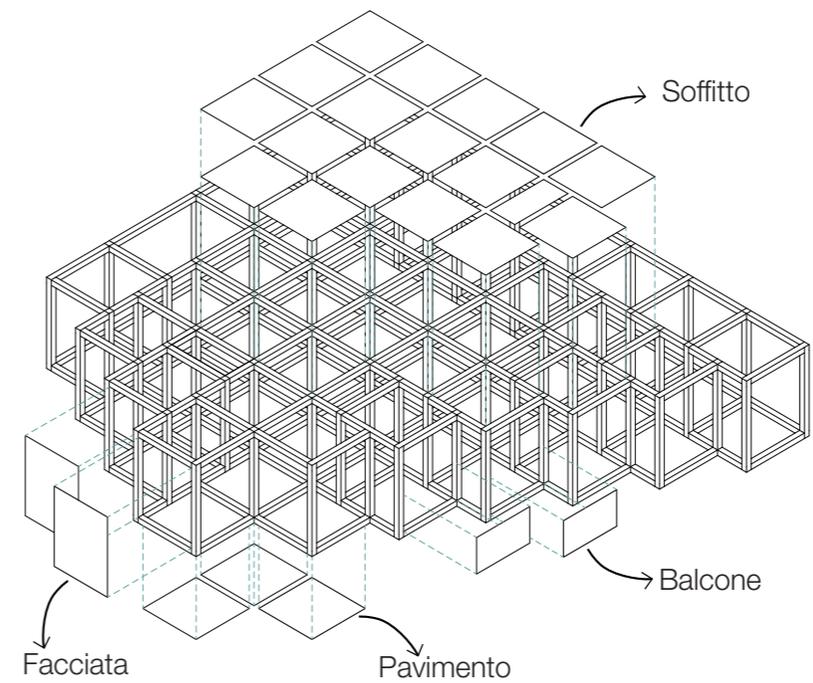


Foto in alto: vista esterna  
 Foto in basso: vista di una parte dell'edificio  
 Fonte: <https://www.rsh-p.com/projects/tree-house/>

Costruzione modulare, per dare la possibilità di progettare il proprio appartamento scegliendo da un catalogo di moduli prefabbricati, inseribili nel telaio che costituisce la torre.

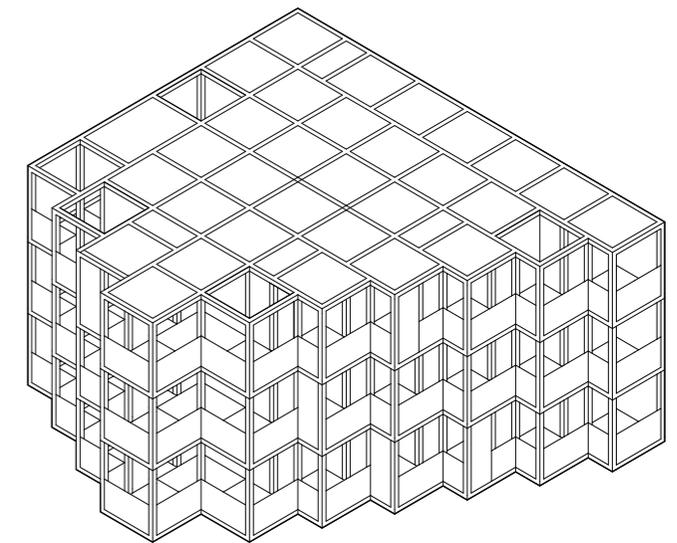
La costruzione è divisa in 8 elementi separati: struttura, pareti, facciata, soffitto, pavimenti, infrastrutture, balconi, impianti. La struttura e l'infrastruttura sono i soli elementi fissi.

**Progetto:** Vijayawada Tower  
**Localizzazione:** Vijayawada, India  
**Anno di realizzazione:** 2015  
**Architetti:** Penda



A sinistra: Assonometria di parte dell'edificio  
 In basso: Diagramma distribuzioni e funzioni di due alloggi tipo

Foto in alto: Vista dalla corte interna  
 Foto in basso: Vista interna dalla cucina-soggiorno  
 Fonte: <https://www.archdaily.com/903495/homes-for-all-dorthe-avej-residence-bjarke-ingels-group>

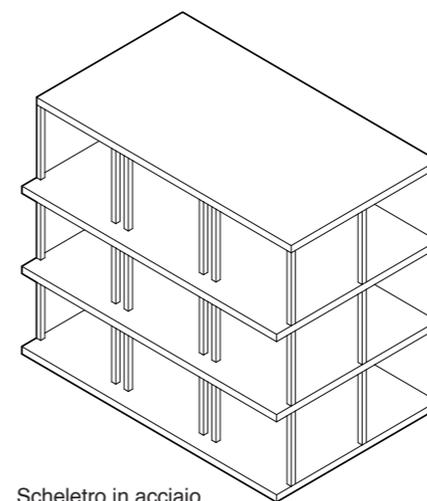


**Progetto:** Kasita  
**Localizzazione:** Various  
**Anno di realizzazione:** 2017  
**Architetti:** Jeff Wilson

Micro abitazioni prefabbricate di 30 mq, ogni appartamento è caratterizzato da un box vetro a sbalzo.

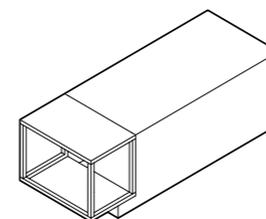
Ogni blocco può essere inserito in una struttura a telaio in acciaio, per poter creare un insieme di unità impilate in verticale e in orizzontale.

Assemblabile in una settimana e i blocchi possono essere spostati se necessario in qualunque momento.

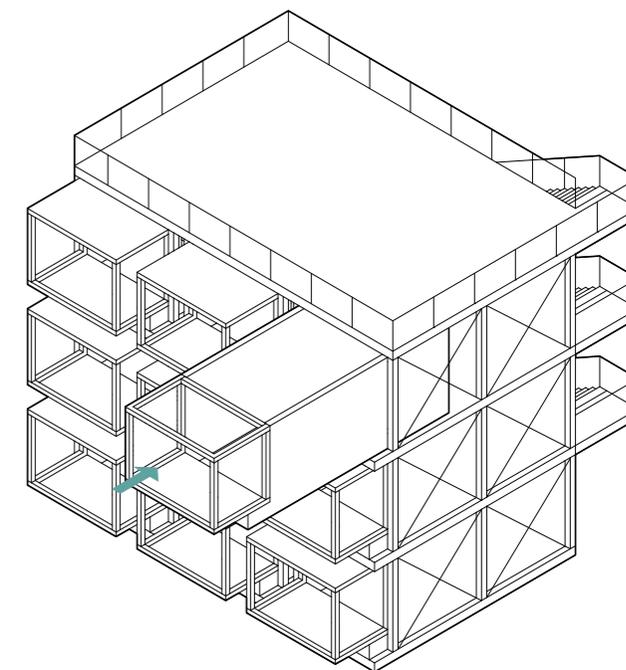


Scheletro in acciaio

+



Abitazione 30 mq



A sinistra: Assonometria di parte dell'edificio

In basso: Diagramma distribuzioni e funzioni di due alloggi tipo

Foto in alto: Vista dalla corte interna  
 Foto in basso: Vista interna dalla cucina-soggiorno  
 Fonte: <https://www.archdaily.com/903495/homes-for-all-dortheavej-residence-bjarke-ingers-group>

Trasformazione di edifici anni '60, con minori interventi possibili.

Vengono aggiunti giardini d'inverno e balconi, per ampliare lo spazio utilizzabile degli appartamenti. I giardini d'inverno, di 3.80 o 2 m di profondità, sono posti sulle facciate sud degli edifici.

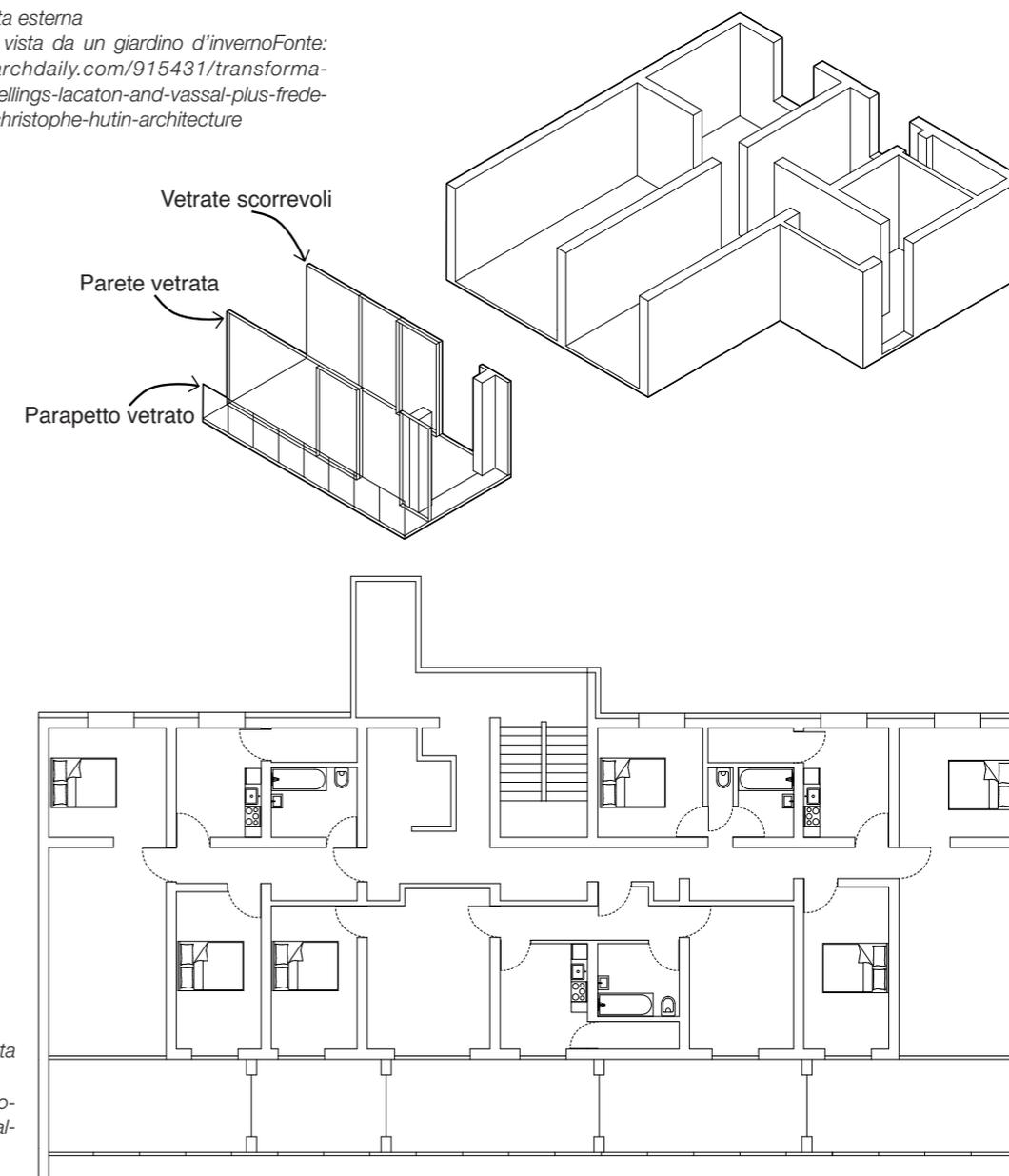
I giardini d'inverno e i balconi sono abbinati a tende termiche, che assicurano l'ombreggiamento, il comfort termico d'estate e d'inverno e un risparmio energetico.

**Progetto:** Transformation de 530 logements  
**Localizzazione:** Bordeaux, Francia  
**Anno di realizzazione:** 2017  
**Architetti:** Lacaton & Vassal



Foto in alto: vista esterna

Foto in basso: vista da un giardino d'inverno  
 Fonte: <https://www.archdaily.com/915431/transformation-of-530-dwellings-lacaton-and-vassal-plus-frederic-druot-plus-christophe-hutin-architecture>

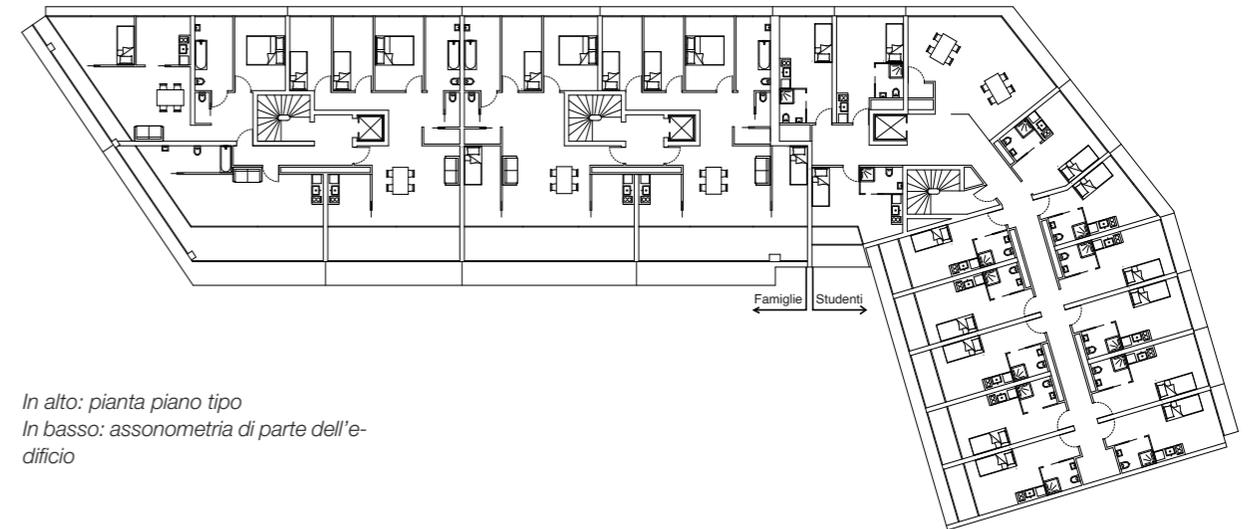


**Progetto:** Logements Étudiants & Sociaux  
**Localizzazione:** Parigi, Francia  
**Anno di realizzazione:** 2014  
**Architetti:** Lacaton & Vassal

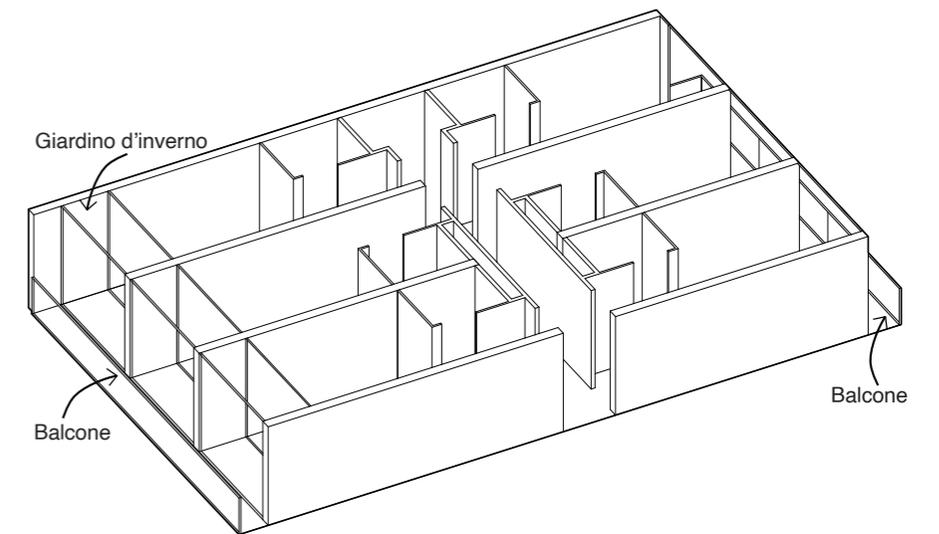
Edificio composto da 98 alloggi per studenti e 30 di edilizia residenziale pubblica.

Nella maggioranza dei casi gli appartamenti possono godere delle viste sui due fronti. I soggiorni e le cucine sono poste a sud o sud-est e si aprono su un giardino d'inverno profondo 2.10 m, che si estende in un balcone di 1 m. Le camere si trovano sul lato nord e si aprono su un balcone.

Gli alloggi per studenti, da 19 a 23 mq, si affacciano sul balcone dal lato nord e sul giardino d'inverno, di 3.70 m, dal lato sud



*In alto: pianta piano tipo  
 In basso: assonometria di parte dell'edificio*



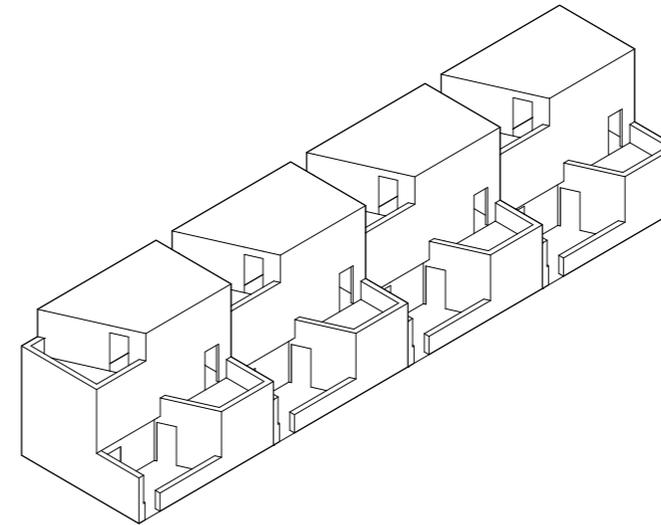
*Foto in alto: Vista dalla corte interna  
 Foto in basso: Vista dal giardino d'inverno  
 Fonte: <https://www.lacatonvassal.com/index.php?idp=75>*

**Progetto:** Hannibal Road Gardens  
**Localizzazione:** Londra, Regno Unito  
**Anno di realizzazione:** 2012  
**Architetti:** Peter Barber Architects

Stecca di 8 villette a schiera, costruite sul retro di un sito di case popolari degli anni '60, rivestite di scandole di legno per riprendere l'estetica del quartiere.

Ogni casa ha, sul tetto e al piano terra, almeno 2 piccole terrazze, che andranno personalizzate.

La residenza più grande ha 7 camere, una 6 e le altre 3 o 4.



A sinistra: Assonometria dell'edificio  
 basso: Assonometria di un alloggio  
 tipo

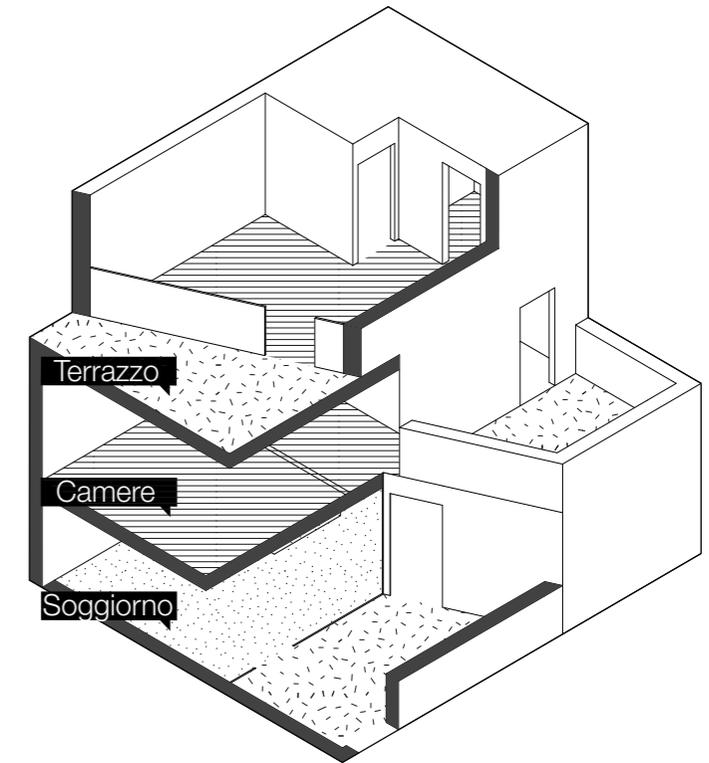


Foto in alto: Vista complessiva  
 Foto in basso: Vista interna di un alloggio  
 Fonte: <https://www.dezeen.com/2012/10/22/hannibal-road-gardens-housing-by-peter-barber/>

# CAP 2.4

## L'abaco

Per definire e riassumere in modo chiaro le variabili di progetto viene costruito un abaco dei componenti, cioè una sorta di matrice o catalogo da cui l'utenza può scegliere le configurazioni che meglio si adattano alle sue esigenze.

Verrà scelta la tipologia di volumetria dell'edificio, a seconda della conformazione del lotto, e gli spazi comuni da destinare all'edificio. In seguito si potrà scegliere lo sviluppo del proprio alloggio e i moduli che lo compongono e, infine, si selezioneranno tutte le tipologie di finiture, in modo da rendere gli appartamenti il più personalizzabili possibile.

Gli appartamenti sono composti da **moduli base di 3x5 m**, costruiti con un sistema modulare composito leggero, in modo da riuscire a diminuire i costi di trasporto e montaggio e avere la possibilità di imbullonare i moduli alla struttura invece di usare una saldatura, permettendo un montaggio e smontaggio più facile e rapido, senza necessità di manodopera specializzata.

I moduli possono essere aggregati in orizzontale o verticale, a formare tre tipologie di appartamenti diversi, da **30, 45 e 75 m<sup>2</sup>** oppure essere lasciati singoli o uniti per creare gli spazi condivisi dell'edificio, come aree di lavoro, palestre, lavanderie, terrazzi...

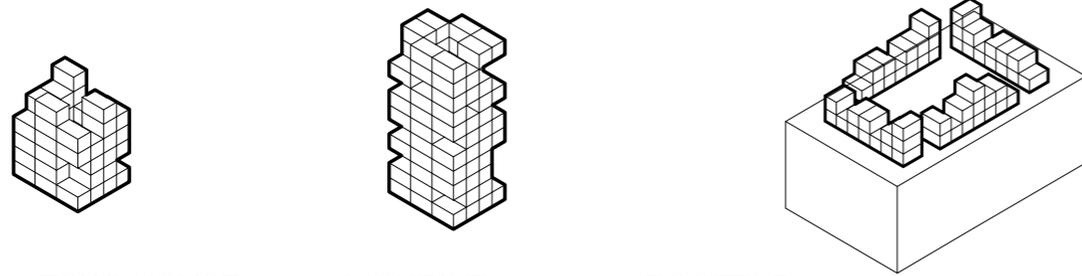
Per poter rendere gli alloggi personalizzabili si

possono scegliere le aperture e gli spazi esterni più indicati al proprio stile di vita e, infine, anche le finiture esterne, che possono essere realizzate con materiali riciclati e riciclabili, come legno, plastica o alluminio, adattandosi anche ai materiali più caratteristici dell'area in cui viene sviluppato il progetto.

Nel caso di aggregazione dei moduli in edifici con più piani, la distribuzione verticale è posta al centro, questo per massimizzare le opportunità di luce diurna e per assicurare sempre uno spazio esterno per gli spazi abitativi, che si troveranno lungo il perimetro della costruzione.

ABACO DELLE VARIAZIONI

TIPOLOGIE URBANE



TOWNHOUSE

HIGHRISE

ROOFTOP

SPAZI COMUNI



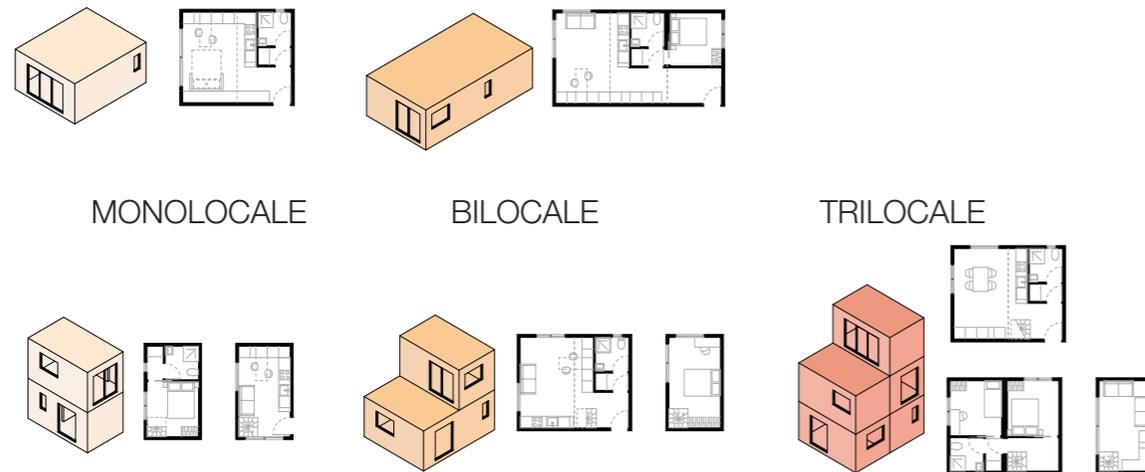
LAVANDERIA

PALESTRA

CO-WORKING

TERRAZZO/  
GIARDINO

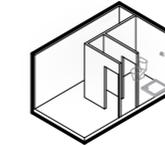
TIPOLOGIE RESIDENZIALI



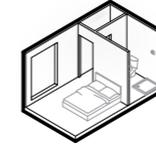
MONOLOCALE

BILOCALE

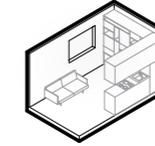
TRILOCALE



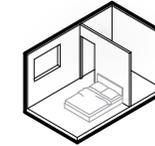
BAGNO



CAMERA +  
BAGNO

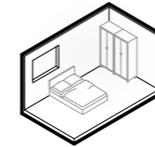


CUCINA +  
SOGGIORNO

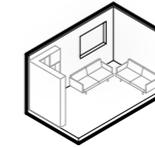


CAMERA  
PICCOLA

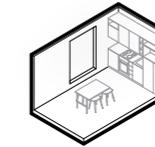
MODULI



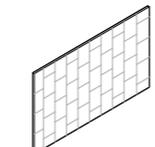
CAMERA



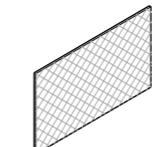
SOGGIORNO



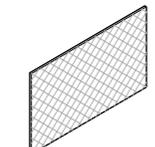
CUCINA



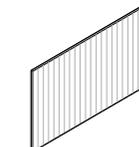
ALLUMINIO



PLASTICA

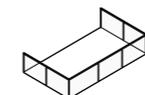


SUGHERO



LEGNO

FINITURE



BALCONE



GIARDINO  
D'INVERNO



OMBREGGIAMENTO/  
PARTIZIONE



FINESTRA  
2x2.1 m



FINESTRA  
0.5x1.1 m



FINESTRA  
1.5x1.1 m



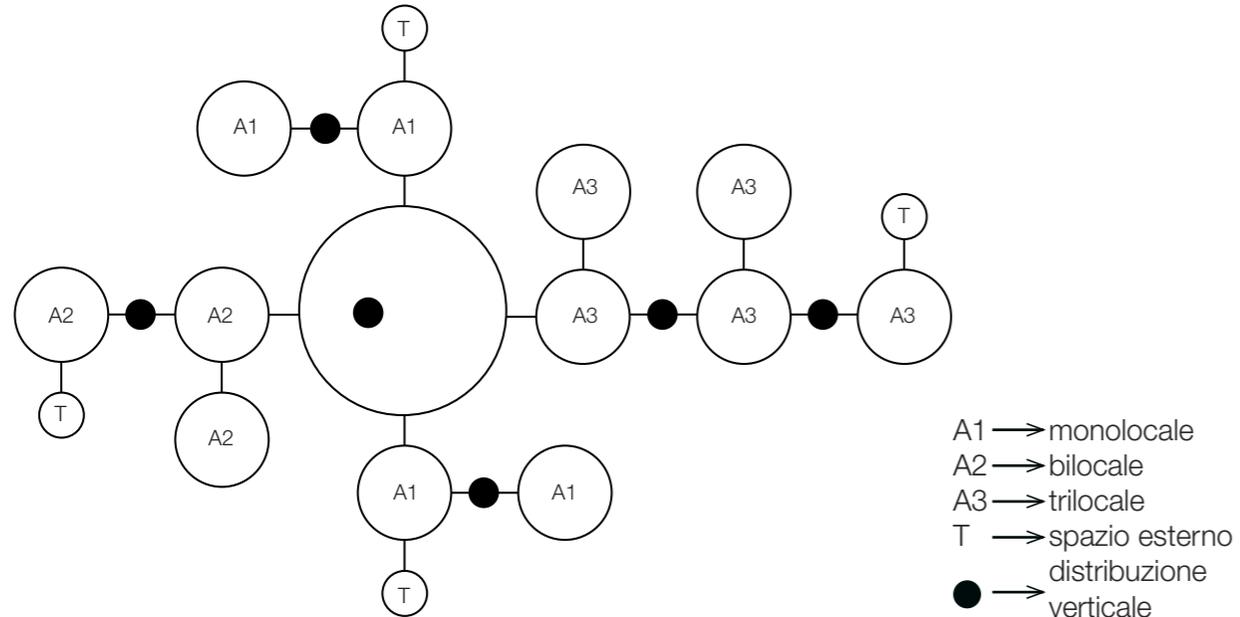
FINESTRA  
1.5x2.1 m



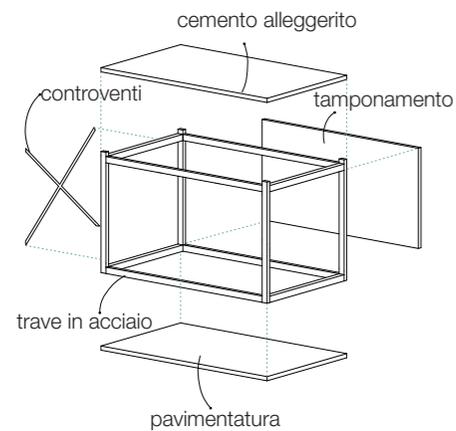
FINESTRA  
3x2.1 m

AFFACCI

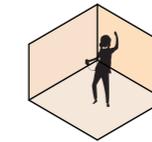
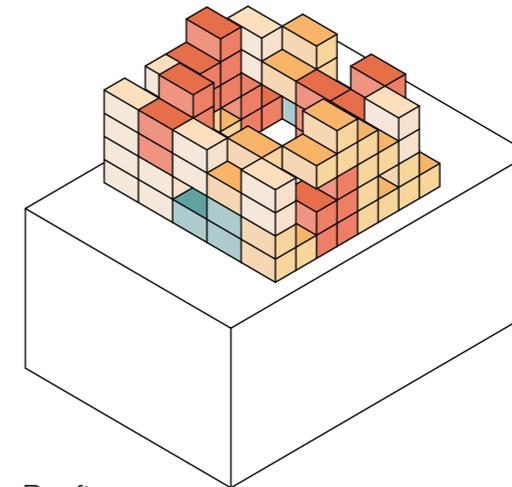
## DISTRIBUZIONE PIANO TIPO



## STRUTTURA MODULARE



## DISTRIBUZIONE TIPOLOGICA DEI MODULI NEL BLOCCO RESIDENZIALE



### MONOLOCALE

30 m<sup>2</sup> - Ideale per ospitare una persona. Lo spazio è essenziale, ma è garantita un'area esterna.



### BILOCALE

45 m<sup>2</sup> - Ideale per una coppia o per un singolo che voglia più spazio.



### TRILOCALE

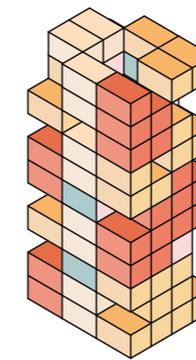
75 m<sup>2</sup> - Ideale per le famiglie con uno o due figli. L'alloggio può ospitare due o tre camere.



### SPAZI COMUNI

Possono avere diverse dimensioni e ospitare diverse funzioni in base alle esigenze degli utenti.

Rooftop



Highrise

## NOTE

1 Klein A., Baffa Rivolta M. e Rossari A. (a cura di), *Lo studio delle piante e la progettazione degli spazi negli alloggi minimi : scritti e progetti dal 1906 al 1957*, Mazzotta, Milano 1975

2 Le Corbusier, Cerri P. e Nicolini P. (a cura di), *Verso una architettura*, Longanesi, Milano 2015

3 Ahn Y. H., Kim K., *Sustainability in modular design and construction: a case study of 'The Stack'*, in "International Journal of Sustainable Building Technology and Urban Development", vol. 5, vol. 4, Ottobre 2014, pp. 250-259

4 Bertram N., Fuchs S., Mischke J., Palter R., Strube G., Woetzel J., *Modular construction: from projects to products*, McKinsey & Company, Giugno 2019

## **capitolo 3**

### **Seconda fase**

### **Lo sviluppo digitale**

# 3.1

## Perchè il parametrico?

“Il progetto non è un risultato, ma un **processo**. Non è una descrizione, ma una prescrizione e una performance.”<sup>[1]</sup>

Per questo per lo studio e l'analisi di questo caso è stato scelto un approccio parametrico. Pensare parametrico significa trovare il problema da risolvere e riformularlo in modo da renderlo meno complesso, attraverso la **sin-tetizzazione** e la **scomposizione**. Si devono trovare gli elementi più semplici che compongono il problema, al fine di capire quali siano gli elementi più rilevanti e riuscire a trattarli con più facilità.

L'uso di **software** non è necessario per uno sviluppo parametrico del progetto, ma aiutano nel lavoro di astrazione e concettualizzazione del problema. Attraverso l'utilizzo di programmi si riesce a dedurre modelli e schemi, in modo da generalizzare e parametrizzare un concetto. I software di progettazione parametrica al giorno d'oggi vengono utilizzati principalmente per lo sviluppo di progetti costituiti da forme geometriche complesse, andando a riprendere la definizione di parametrismo di Schumacher, che lo intende, ancora prima di tutto, come uno stile architettonico. In questo caso, invece, si è scelto di utilizzare questa tipologia di programmi per provare a indagare il progetto dal punto di vista organizzativo, più che estetico. I dati inseriti come input dell'algoritmo servono in primo luogo a studiare le varie soluzioni possibili in risposta al problema posto.

Questo tipo di approccio può aiutare la progettazione su vari fronti. Innanzitutto, si riesce a definire più chiaramente il problema progettuale da risolvere e allo stesso tempo, si riesce a rendere più leggibile e fruibile il processo utilizzato per giungere alle soluzioni finali, evidenziando le scelte effettuate.

Infine, dà la possibilità di controllare e gestire numerosi parametri e dati, permettendo di allargare il campo di applicazione del progetto, di svilupparlo in modo più approfondito e coordinare più discipline allo stesso momento, giungendo a numerose potenziali soluzioni in breve tempo.

Inoltre, in questo caso, si è scelto l'utilizzo di software parametrici, perchè sono in grado di aiutare nella progettazione mantenendola decontestualizzata. Un requisito primario del progetto è, infatti, la sua flessibilità urbana. Il concorso di idee iniziali richiedeva che fosse possibile sfruttarlo per le diverse aree della città di Melbourne, ma si è scelto di allargare ulteriormente il campo, cercando di creare un prototipo di edificio residenziale che sia in grado di aumentare la capacità abitativa di una città, adattandosi alle varie esigenze che si potrebbero riscontrare e alle caratteristiche della zona presa in considerazione.

# 3.2

## L'algoritmo

Il passaggio successivo è stato trasportare il concept di progetto su strumenti software di tipo parametrico, attraverso la creazione di un **algoritmo**, che potesse in poco tempo realizzare e modificare l'aggregazione dei moduli.

L'algoritmo utilizza linguaggi di programmazione in grado di esprimere le istruzioni eseguibili da un computer attraverso una procedura passo-passo. L'algoritmo è una procedura utilizzata per **dare una soluzione a una domanda o eseguire un compito** attraverso un elenco finito di istruzioni di base ben definite ed è in grado di suddividere il problema in passi più semplici, che possono, quindi, essere calcolati facilmente.

Ha tre proprietà essenziali: deve avere un set di istruzioni chiare, che non possono essere fraintese, deve avere un set di input ben definito e deve generare un output preciso<sup>[2]</sup>.

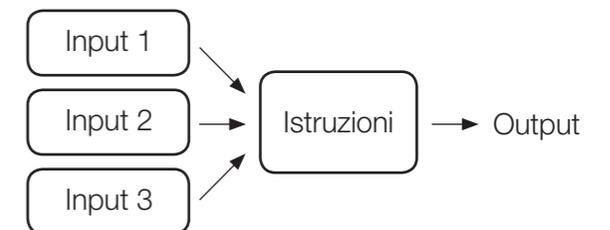
Gli algoritmi possono produrre anche geometrie, che non sono più disegnate a mano o con il mouse, ma definite da istruzioni espresse in un linguaggio specifico proprio del programma che si sta utilizzando, cioè uno scripting, che consiste in due ambienti di lavoro, l'editor e lo spazio di modellazione 3D, e produce due tipologie di output: l'algoritmo e la geometria a lui associata<sup>[2]</sup>.

Per modellare parametricamente il progetto è stato scelto come software Grasshopper. Grasshopper, un plug-in per Rhinoceros, è un editor basato su nodi sviluppato da David Rutten per Robert McNeel & Associates.

Si tratta di un linguaggio di programmazione VPL, Visual Programming Language, cioè una tipologia di visualizzazione degli script e delle funzionalità dei software. I componenti vengono trascinati nell'area di lavoro e concatenati tra loro, si crea, quindi, una connessione tra i parametri e la geometria<sup>[3]</sup>.

In questo modo si permette di velocizzare notevolmente il calcolo dei comandi, infatti cambiando gli input si creano output diversi senza la necessità di inserire nuovamente tutte le istruzioni, cambiando un parametro si modifica automaticamente la geometria generata.

Inoltre, questo tipo di linguaggio consente di generare algoritmi senza l'obbligo di conoscere a fondo i linguaggi di programmazione, permettendo anche ad architetti con conoscenze di scripting basilari di gestire un numero sempre maggiore di informazioni.



Rappresentazione schematica di un algoritmo

# 3.3

## Wasp

Per procedere all'assemblaggio dei moduli in ambiente Grasshopper si è deciso di utilizzare il plug-in Wasp, creato appositamente per generare aggregazioni da Andrea Rossi.

**Wasp** è un insieme di componenti Grasshopper, sviluppato in Python, diretto a rappresentare e progettare con elementi discreti. Ogni componente include le informazioni di base necessarie per il processo di aggregazione, la geometria, la posizione delle aggregazioni, l'orientamento, ecc... Le aggregazioni vengono generate in funzione a una serie di regole stabilite dall'utente, descritte come istruzioni per definire i vincoli e i limiti dell'aggregazione stessa<sup>[4]</sup>.

Per generare l'aggregazione è stato innanzitutto necessario creare **sei tipologie di modulo**, cioè le sei differenti soluzioni di appartamento, presenti nel progetto (BOX1, BOX2, BOX3, BOX4, BOX5, BOX6), che si differenziano solamente per dimensione. (Figura 1)

Ogni modulo è stato modellato parametricamente inserendo per ogni sua dimensione uno slider, che permette di modificare ogni grandezza molto rapidamente e in qualsiasi momento del processo.

Volendo rendere i moduli parametrici, per semplicità, contengono già nel parallelepipedo che li costituisce anche gli spazi esterni, in modo da garantire agli alloggi come asset di base lo spazio aperto.

Una volta creati i diversi moduli si è passati alla generazione delle **regole di connessione**, che stabiliscono quali moduli si possono connettere fra loro e quali invece no.

Nella figura 2 è possibile vedere il pannello di Grasshopper che definisce le regole di connessione stabilite, le stringhe sono costruite nel linguaggio BOXx|a\_BOXy|b, che indica che la connessione a del modulo x deve innestarsi sulla connessione b del modulo y (per esempio: BOX1|2\_BOX3|0 indica che la connessione 2 del modulo 1 si aggrega con la connessione 0 del modulo 3).

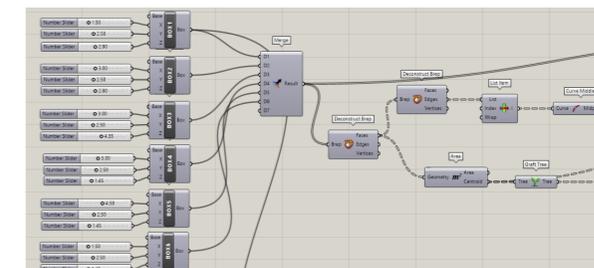


Figura 1. Creazione dei moduli

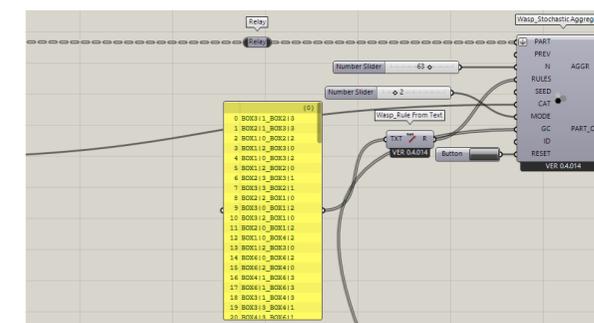


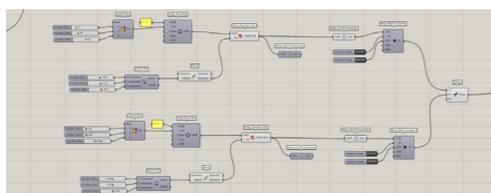
Figura 2. Regole di aggregazione

Una volta definite le regole si può scegliere fra due tipologie di **aggregazione: stocastica o field-driven**. La prima è un tipo di aggregazione casuale, mentre la seconda segue un campo prestabilito, solitamente una geometria personalizzata e funziona in modo aggiuntivo, quindi aumentando il numero di parti in un'aggregazione aggiunge solo nuove parti su quelle già presenti, senza ricalcolare l'aggregazione precedente.

In questo caso è stata scelta l'aggregazione stocastica, controllata con l'aiuto di vincoli spaziali (mesh constraints). Mi sono soffermata sulla progettazione della torre, in cui le connessioni sono maggiori e più complesse rispetto alle altre casistiche pensate nel concept di progetto, per poi adattare la modellazione anche alle altre soluzioni.

Quindi sono state create le mesh al cui interno avverrà l'aggregazione. Queste rappresentano il volume della torre stessa e al suo interno quella dello spazio occupato dalla distribuzione verticale, che dovrà essere lasciato libero dai moduli. Anche le volumetrie della torre, come i moduli, sono totalmente parametricizzate, in modo da avere la possibilità di modificare le loro dimensioni in qualsiasi momento. A questo punto si può procedere con l'aggregazione. (Figura 3)

Figura 3. Creazione delle mesh della torre



In questo momento si iniziano a riscontrare i primi problemi. L'aggregazione che viene effettuata non rispetta tutte le regole inserite, poiché alcuni moduli che, secondo le istruzioni date, non dovrebbero connettersi, si uniscono tra loro. Di conseguenza, vengono aggiunte delle **costrizioni**, che sottolineano i vincoli di progetto che non vengono rispettati e allo stesso tempo viene anche inserito un componente (parts catalog), che permette di scegliere il numero o la percentuale per ogni categoria di elemento, in modo da poter controllare la quantità di ogni tipologia di alloggio presente. (Figura 4+5)

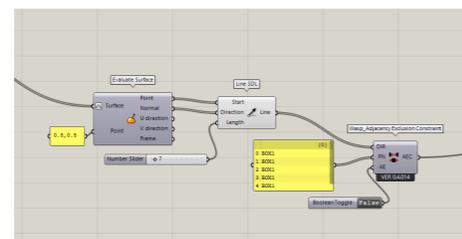


Figura 4. Costrizioni aggiuntive

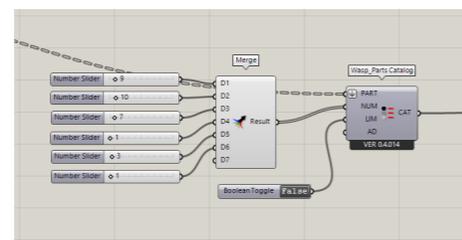


Figura 5. Parts Catalog. Usato per scegliere le percentuali di moduli per ogni tipologia

Anche avendo implementato le regole, Wasp non riesce a posizionare in modo corretto i moduli all'interno delle mesh, mentre nel momento in cui quest'ultime si eliminano e si esegue un'aggregazione totalmente randomica, il programma è in grado di rispettare le istruzioni date. (Figura 6+7)

Dopo aver contattato la numerosa community di supporto e soprattutto il creatore di Wasp, Andrea Rossi, si è capito che il problema risiede nella mesh interna della distribuzione verticale; infatti, se si diminuiscono drasticamente le sue dimensioni il problema si risolve. Questo è dovuto molto probabilmente al fatto che si tratta di programma ancora relativamente giovane e sempre in fase di miglioramento, non è escluso, quindi, che in futuro i problemi riscontrati possano essere risolti molto facilmente.

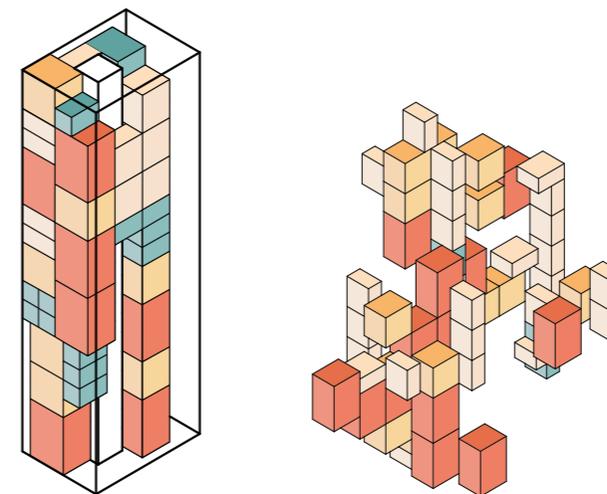


Figura 6 e 7. Più a sinistra: aggregazione non corretta dei moduli. A destra: aggregazione randomica corretta



A questo punto, con l'edificio completo e diviso in tanti moduli base, si vogliono separare i blocchi delle file centrali occupati dalla **distribuzione verticale** e dagli appartamenti più piccoli da quelli occupati dal resto degli alloggi e dalle aree comuni.

Per riuscire in ciò, si è creato uno script di Python. In questo viene inserito come input un'espressione per ogni grandezza dell'edificio prevista che identifica l'area di un piano, per far sì che ogni possibilità sia valutata correttamente. Se la condizione dettata dall'espressione viene considerata vera darà come esito un output specifico.

L'output è costituito da un codice binario, in cui gli 0 rappresentano i moduli che si vogliono scartare e gli 1 i moduli centrali da selezionare; per ogni grandezza di torre prevista viene creato un codice binario abbinato, in modo da selezionare i moduli in maniera corretta. (Figura 4)

Una volta ottenuta la divisione, si prosegue separando ulteriormente i moduli che compongono la distribuzione verticale centrale da quelli dei monocali, utilizzando nuovamente un codice binario, in cui lo 0 rappresenta gli appartamenti e l'1 il blocco scale. Infine, si è aggiunta un'espressione per evitare che si crei quest'ultimo quando l'edificio è più basso di tre piani, cioè quando non si ha più di un appartamento sopra l'altro. (Figura 5)

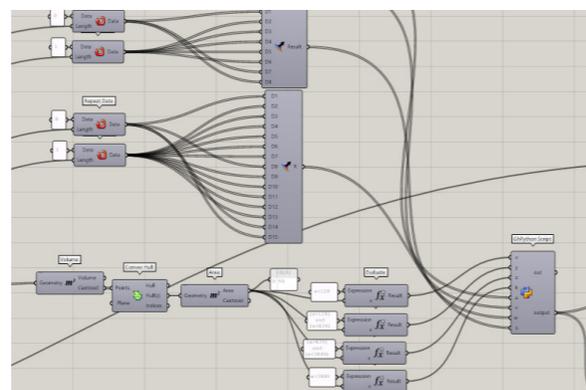
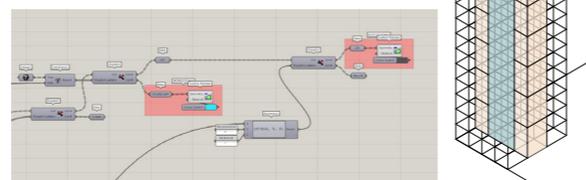


Figura 4. In alto, script Python con input collegati. In basso, dettaglio dello script Python



Figura 5. Divisione dei moduli dei monocali e della distribuzione verticale. In basso, script di Grasshopper.



L'ultimo passaggio è riuscire a selezionare in modo randomico dai moduli rimanenti le altre **tipologie di appartamento** e le aree comuni.

Il primo metodo provato è stato l'utilizzo di un componente Random. I moduli sono stati, quindi, divisi in due gruppi, attraverso l'utilizzo di due componenti Cull Pattern, che sono in grado di eliminare elementi di una lista usando una ripetizione di valori associati al componente. In questo caso sono stati collegati con un algoritmo randomico, in grado di selezionare alcuni moduli in modo totalmente casuale. Con questa metodologia si creano delle aggregazioni di moduli, ma deve essere poi il professionista a scegliere la destinazione dei singoli, poiché non si riescono a definire delle regole specifiche che discretizzino la quantità di moduli che si aggregano fra loro. (Figura 6)

Il secondo metodo, che si è rivelato essere quello funzionante, invece, prevede di separare le facciate restanti in due gruppi (uno per ogni facciata), attraverso l'utilizzo del componente Dispatch, che similmente a Cull pattern, riesce a dividere un insieme di dati in due insiemi separati, ma diversamente da quest'ultimo fornisce entrambe le liste come outputs.

A questo punto si tratta di separare in modo randomico le due facciate per poter ottenere il mix di appartamenti che si preferisce. Per far ciò mi sono avvalsa di uno script trovato sul forum di Grasshopper, sviluppato da Laurent Delrieu nel 2017 e modificato per essere applicato al caso specifico<sup>[5]</sup>.

Il processo in breve richiede di generare una griglia di punti e creare rettangoli con dimensioni diverse. In seguito, si prende un punto casuale contenuto nella griglia e si controlla se i rettangoli contengono i punti della griglia. Se la risposta è affermativa si inserisce la forma, si eliminano i punti utilizzati e si reitera il processo fino a ottenere una griglia completamente occupata.

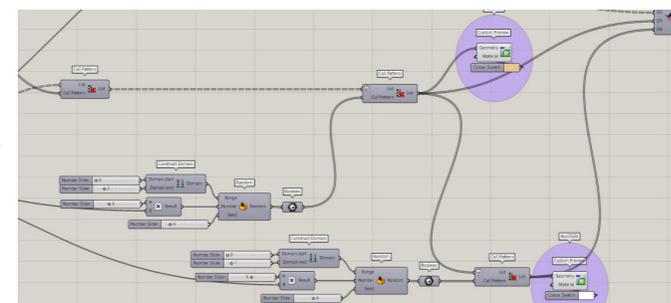
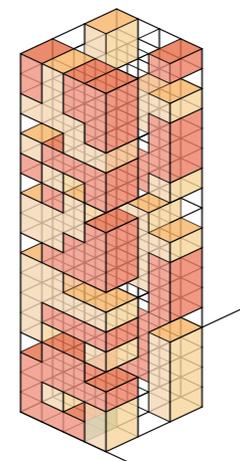


Figura 6. Divisione randomica dei moduli degli appartamenti rimanenti e degli spazi comuni.

Viene, quindi, come detto precedentemente, creata una griglia di partenza, che ha come modulo base un quadrato di 1x1 unità e come dimensioni le stesse della facciata della torre, che possono essere modificate in qualsiasi momento con l'uso di due slider. (Figura 7)

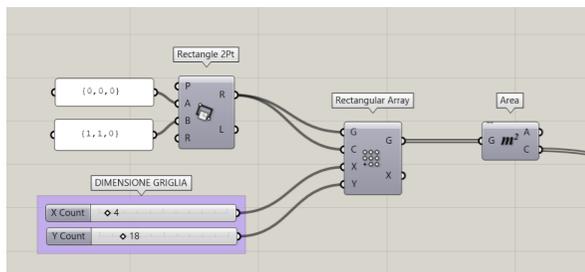


Figura 7. Creazione della griglia iniziale.

In un secondo tempo, vengono creati le varie tipologie di appartamenti, utilizzando due Gene Pool, cioè un componente di Grasshopper in grado di contenere un insieme di variabili, uno per la larghezza e l'altro per l'altezza dei moduli. In questo caso si creano quattro soluzioni per gli alloggi (2x3, 4x1, 3x1, 2x2), che prevedono al loro interno anche lo spazio utilizzato per balconi o terrazzi e la soluzione del modulo singolo 1x1 per gli spazi comuni. Un terzo Gene Pool viene utilizzato per poter decidere e gestire la quantità di moduli per ogni tipologia che si vogliono poter posizionare, si inseriscono per primi i moduli più grandi e via via fino a raggiungere quello più piccolo, una volta posizionati verranno eliminati dalla lista, altrimenti il programma riprova fino al raggiun-

gimento del numero di tentativi massimo. I tre Gene Pool vengono collegati a due componenti Duplicate Data, che permettono di duplicare i moduli fino al raggiungimento della quantità richiesta. (Figura 8)

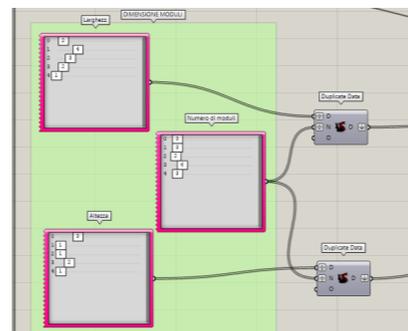
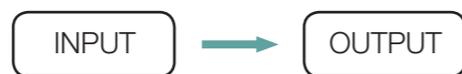


Figura 8. Creazione dei moduli con l'utilizzo di Gene Pool.

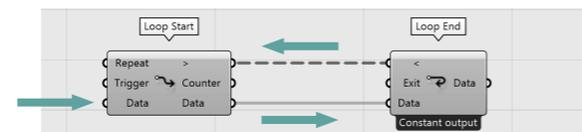
Successivamente si deve far partire una serie di iterazioni, in modo che vengano soddisfatte le richieste, per far ciò si è scelto di utilizzare Anemone, un plug-in di Grasshopper, che permette di creare loop all'interno del programma.



Logica computazionale. Mono-direzionale, da input a output.



Logica ricorsiva. Si crea un ciclo di dati, manda indietro un nuovo input.



Anemone. Segue una logica ricorsiva.

Per prima cosa si inserisce un componente di inizio, Loop Start, a cui vanno collegati come input, il numero di ripetizioni da eseguire, un button per innescare l'inizio di un nuovo computo e i vari dati da mandare in loop. (Figura 9) Fra gli output, oltre ai dati generati, sono presenti il numero di iterazioni fatte e la connessione per il componente di fine, Loop End.

Per riuscire a inserire i moduli gli output vengono collegati a una serie di componenti, che attivano la ricerca delle unità della griglia che possono essere occupate dai rettangoli. (Figura 10)

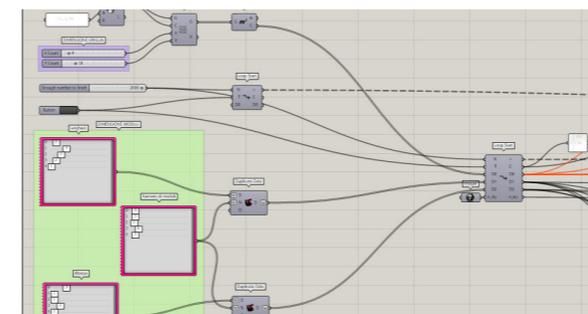


Figura 9. Inizio del processo. Loop Start e input ad esso collegati.

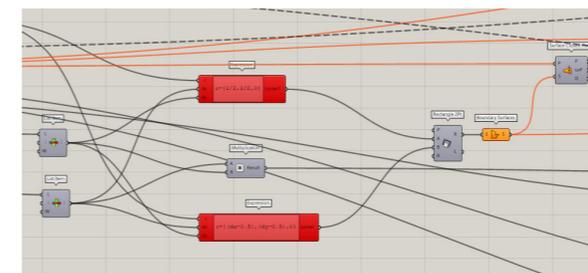


Figura 10. Ricerca delle unità della griglia da occupare con i moduli.

Infine, si collega all'input Exit del Loop End i parametri che permettono di definire la fine del processo. Le iterazioni si fermeranno quando si concludono i dati iniziali del ciclo, cioè quando il numero di dati della griglia diventano 0, poiché tutti sono stati utilizzati nel processo. (Figura 11)

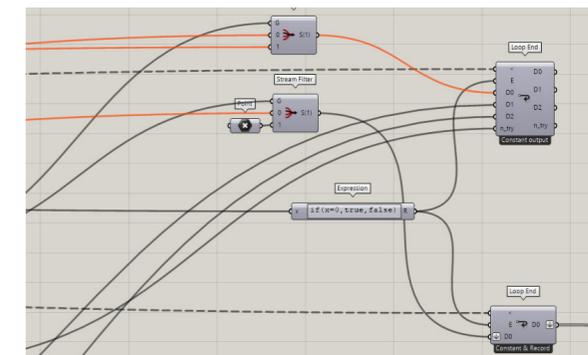


Figura 11. Fine del processo. Loop End e input ad esso collegati.

La parte finale dello script è composta dalla visualizzazione grafica della suddivisione della facciata in moduli. In questo momento le superfici e la griglia, che sono state realizzate sul piano xy, vengono ruotate sul piano verticale e scalate per adattarsi alla facciata. Inoltre, vengono anche applicati i colori ai rettangoli differenziando le varie tipologie di alloggi presenti. (Figura 12)

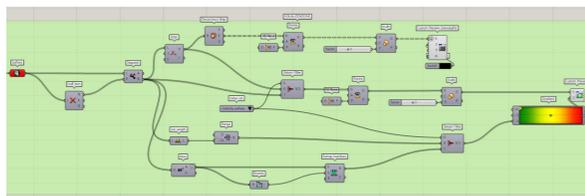


Figura 12. Script della visualizzazione grafica dei moduli in facciata

Tutto il processo spiegato viene ripetuto per la seconda facciata, con l'aggiunta di un componente di movimento collegato a uno slider per controllare lo spostamento delle superfici e della griglia, in modo che si allineino al prospetto. (Figura 13)

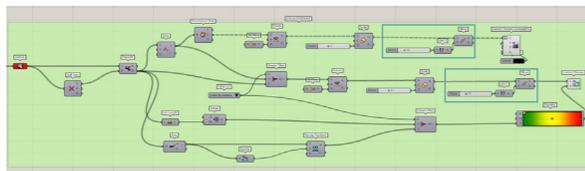
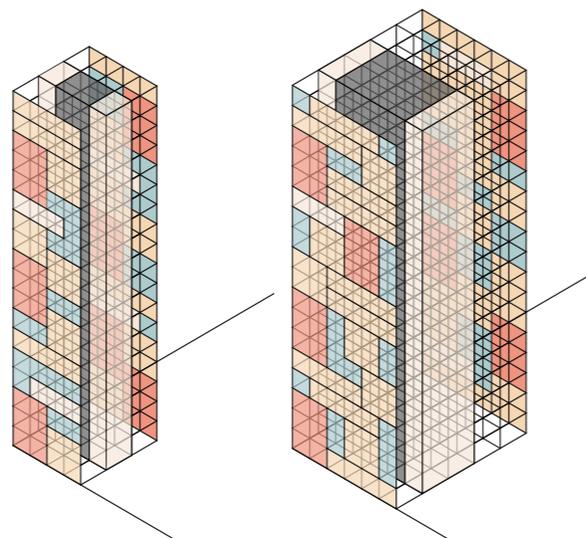
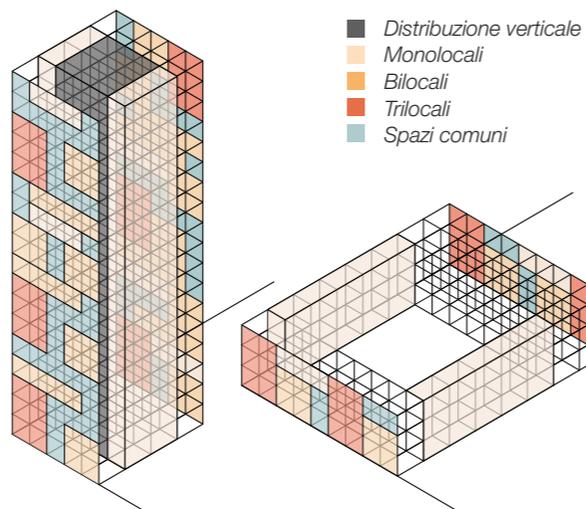


Figura 13. Script della visualizzazione grafica dei moduli della facciata retrostante. In evidenza i componenti di movimento per allineare le superfici al prospetto.

## LE DIVERSE TIPOLOGIE DI EDIFICIO



In senso antiorario: edificio da 4x3 moduli di base, 5x4, 9x8, 6x5.



- Distribuzione verticale
- Monolocali
- Bilocali
- Trilocali
- Spazi comuni

## L'INSERIMENTO DEL CONTESTO

Una volta modellato l'edificio singolo si tratta di creare la possibilità di produrre delle sopraelevazioni, cioè posizionare l'aggregazione sopra le coperture piane inutilizzate degli **edifici esistenti**.

Per fare ciò si è scelto di utilizzare un plug-in di Grasshopper: Gismo, che, attraverso la connessione con il sito OpenStreetMap, consente di generare automaticamente l'ambiente urbano e la geometria del terreno in base alle coordinate latitudine-longitudine e al raggio della posizione<sup>[6]</sup>.

In questo caso è stata scelta una parte di downtown di Melbourne, zona estremamente densa, ma con numerosi edifici che si prestano a ospitare una sopraelevazione. Si è, quindi, collegato l'indirizzo, che può essere modificato in qualsiasi momento, e il raggio d'azione attorno al punto scelto. (Figura 14)

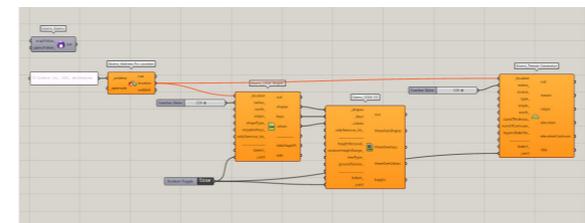


Figura 14. Impostazione del contesto esistente con l'utilizzo del plug-in Gismo

Dopo aver ottenuto gli edifici esistenti si è voluto separarli in quattro gruppi, il cui elemento di discretizzazione è l'area della copertura. In questo modo i blocchi residenziali si andranno a posizionare sulle superfici in maniera proporzionale alla loro grandezza.

Per prima cosa si è costruito un riquadro di delimitazione, Bounding Box, in modo da riuscire a selezionare per tutti gli edifici la faccia superiore, cioè quella occupata dalla copertura e si sono dovuti ordinare i dati raccolti per ottenere il dominio in cui si andrà a operare. (Figura 15)

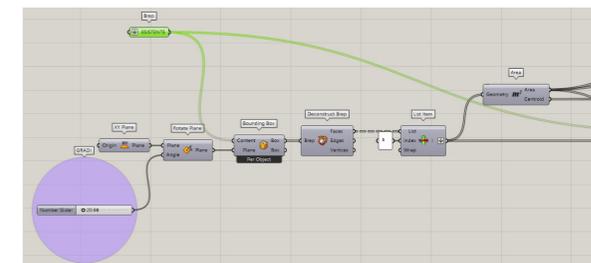


Figura 15. Creazione delle Bounding Box e selezione delle facce superiori occupate dalle coperture.

In seguito, si è scelto di restringere il campo di dominio attraverso l'utilizzo di due slider, in modo da poterlo gestire in qualsiasi momento della progettazione, ciò perché in questo modo si riescono a eliminare tutti quegli edifici che sarebbero troppo piccoli per ospitare una nuova costruzione. (Figura 16)

Il passo successivo è stato quello di creare



## NOTE

1 Maurizio Ferraris, *Il progetto dettato*, in “Ar-deth”, vol. 1, Ottobre 2017, pp. 31-47

2 Tedeschi A., *AAD Algorithms-Aided Design. Parametric strategies using grasshopper*, Leenseur, Brienza 2014

3 Tedeschi A., *Architettura parametrica : introduzione a Grasshopper*, Leenseur, Brienza 2010

4 Wasp - *Discrete Design for Grasshopper*, <https://github.com/ar0551/Wasp>, consultato il 18/06/2021

5 <http://www.grasshopper3d.com/forum/topics/rectangle-false-packing-sample-and-example?commentId=2985220%3AComment%3A1678379>, consultato il 2/06/2021

6 *Gismo*, <https://github.com/stgeorges/gismo>, consultato il 18/06/2021

# CONCLUSIONI

Questa ricerca cerca di costituire un supporto efficace nella gestione della progettazione architettonica di spazi abitabili flessibili, sfruttando la modellazione e le tecniche parametriche.

Un'opportunità certamente sottovalutata data dall'approccio parametrico come strumento di ideazione, è l'opportunità di esplicitare i requisiti definiti dai progettisti e dagli utenti, consentendo anche ai non esperti di partecipare alla fase decisionale, dando vita a un progetto interattivo e a un processo partecipativo.

Inoltre, gli strumenti digitali utilizzati consentono di avere un visione più ampia e organizzata dell'intero progetto, mantenendo comunque un focus sui singoli elementi e permettono di generare un insieme di possibilità e variabili che non potrebbero essere altrimenti facilmente controllate.

Il vincolo principale dell'uso di questa tipologia di software è, sicuramente, che per ottenere risultati sempre più soddisfacenti il progettista dovrebbe possedere una serie di nuove competenze, che sfociano anche nel campo della programmazione, ma che permettono di collegare tra loro aree e discipline diversi, favorendo un ambiente condiviso.

Il pensiero parametrico aiuta ad allontanarsi dall'idea di impostare elementi definiti e statici, ma favorisce lo sviluppo di processi partecipa-

tivi e flessibili, che possono essere adattati e variati seguendo le esigenze che si riscontrano durante il processo.

In questo modo è nata questa ricerca, che vuole avere come obiettivo la definizione di un'ipotesi di flusso operativo, che aiuti nella progettazione di architetture sempre più flessibili e sostenibili.

Durante lo svolgimento della tesi ho preso atto della complessità nella progettazione di un modello simile, con l'introduzione anche di strumenti digitali che richiedono specifiche competenze, ma allo stesso tempo, cercando di risolvere i requisiti del concorso a cui ho partecipato, ho potuto verificare i vantaggi di questo tipo di approccio alla progettazione, che non è tanto un risultato definito quanto un processo.

# **BIBLIOGRAFIA**

## APPROCCIO PARAMETRICO. DIAGRAMMI

Il percorso di tesi è iniziato con lo studio dell'approccio parametrico alla progettazione e, quindi, con la lettura di alcuni testi che aiutano a capire ed esplicitare l'importanza dell'uso del diagramma in architettura.

Alexander C., *Note sulla sintesi della forma*, Saggiatore, Milano 1973

Barosio M., Gugliotta R., *Dal tipo al diagramma: la rappresentazione del dato come strumento euristico per il progetto di architettura*, in "Dienne. Building Information Modeling, Data & Semantics", vol. 7, Dicembre 2020, pp. 28-39

Cavallari Murat A., *Forma urbana ed architettura nella Torino barocca: dalle premesse classiche alle conclusioni neoclassiche*, UTET, Torino 1968

Fedorchenko M., *The tools of mediation: Extending the Diagrammatic Project*, in "Philosophy Kitchen - Rivista di filosofia contemporanea", Vol. 3, 2019

Garcia M., *The diagrams of architecture*, John Wiley & Sons, Hoboken 2010

Klein A., Baffa Rivolta M. e Rossari A. (a cura

di), *Lo studio delle piante e la progettazione degli spazi negli alloggi minimi: scritti e progetti dal 1906 al 1957*, Mazzotta, Milano 1975

Somol R. E., *Dummy Text, or The Diagrammatic Basis of Contemporary Architecture*, Universe, New York 1999

## APPROCCIO PARAMETRICO. SVILUPPO

In seguito si è proseguito con la lettura di testi in riferimento alla storia dell'approccio parametrico, per capirne le origini e le metodologie che ne hanno influenzato l'uso attuale.

Burry M., *Antoni Gaudí and Frei Otto: Essential Precursors to the Parametricism Manifesto*, in "AD. Parametricism 2.0. Rethinking architecture's agenda for the 21st century", vol. 86, vol. 2, Marzo/Aprile 2016, pp. 30-35

Carpo M., *Parametric Notations*, in "Architectural Design", vol. 86, Marzo 2016, pp. 24-29

Carpo M., *The alphabet and the algorithm*, MIT Press, Cambridge 2011

Carpo M., *The Digital Turn in Architecture 1992-2012*, John Wiley & Sons, Chichester 2012

Frazer J., *An Evolutionary Architecture*, Architecture Association, Londra 1995

Frazer J., *Parametric Computation: History and Future*, in "AD. Parametricism 2.0. Rethinking architecture's agenda for the 21st century", vol. 86, vol. 2, Marzo/Aprile 2016, pp. 18-23

Hernandez C. R. B., *Thinking parametric design: introducing parametric Gaudí*, in "Design Studies", vol. 27, vol. 3, Maggio 2006, pp. 309-324

Schumacher P., *Gearing up to impact the global built environment*, in "AD. Parametricism 2.0. Rethinking architecture's agenda for the 21st century", vol. 86, vol. 2, Marzo/Aprile 2016, pp. 8-17

Schumacher P., *Parametricism as Style. Parametricism Manifesto*, Londra 2008

Schumacher P., *The Autopoiesis of Architecture, Vol II: "A New Agenda for Architecture"*, John Wiley & Sons, Chichester 2012

Songel J. M., *Frei Otto and the debate about the genesis of architectural form*, in "EGA. Revista de Expresión Gráfica Arquitectónica", vol. 15, Ottobre 2010

## PROGETTAZIONE MODULARE

Per la progettazione dei moduli di abitazione sono stati letti alcuni testi in riferimento alla progettazione di edifici modulari.

Andrew William Lacey, Wensu Chen, Hong Hao, Kaiming Bi, *Structural response of modular buildings – An overview*, in "Journal of Building Engineering", vol. 16, Dicembre 2017, pp. 45-56

J.Y.R. Liew, Y.S.Chua, Z.Dai, *Steel concrete composite systems for modular construction of high-rise buildings*, in "Structures", vol. 21, Ottobre 2019, pp. 135-149

Huu-Tai Thai, Tuan Ngo, Brian Uy, *A review on modular construction for high-rise buildings*, in "Structures", vol. 28, Dicembre 2020, pp. 1265-1290

## **APPROCCIO PARAMETRICO. DIGITALE**

Infine le letture si sono concluse con alcuni testi sulla progettazione parametrica digitale, che hanno permesso una comprensione più facile e dettagliata dell'uso degli strumenti digitali attuali.

M. Calvano, *Disegno digitale esplicito: rappresentazioni responsive dell'architettura e della città*, Aracne, Canterano 2019

C. Griffa, *Smart Creatures. Progettazione parametrica per architetture sostenibili*, Edil-stampa, Roma 2012

Tedeschi A., *AAD Algorithms-Aided Design. Parametric strategies using grasshopper*, Le Penseur, Brienza 2014

# RINGRAZIAMENTI

Vorrei ringraziare, in primo luogo, la mia relattrice la Prof.ssa Michela Barosio, che mi ha sempre seguito con entusiasmo e disponibilità, spingendomi a dare il massimo in questa fase finale del mio percorso di studi.

Un grazie in particolare va anche all'Arch. Andrea Tomalini, che si è rivelato fondamentale nella sua estrema disponibilità nell'aiutarmi sempre molto velocemente nello sviluppo del progetto con l'utilizzo di software digitali.

Voglio ringraziare anche tutti i miei amici e colleghi che ho incontrato in questi anni nel mio percorso universitario, che mi hanno sempre aiutato e con cui ho condiviso numerosi momenti di ansia e di felicità.

Ringrazio i miei amici di sempre, che mi hanno aiutato a "staccare la spina" dallo studio e con cui ho passato anni di passeggiate e vacanze.

Infine, ringrazio la mia famiglia, che non mi ha mai fatto macare nulla e mi ha sempre dato infinito amore e un solido punto di riferimento.



