

Syracuse University

SURFACE

Architecture Thesis Prep

School of Architecture Dissertations and
Theses

12-2015

Algorithmic Settlements

Benjamin Anderson-Nelson

Follow this and additional works at: https://surface.syr.edu/architecture_tpreps



Part of the [Architecture Commons](#)

Recommended Citation

Anderson-Nelson, Benjamin, "Algorithmic Settlements" (2015). *Architecture Thesis Prep*. 283.
https://surface.syr.edu/architecture_tpreps/283

This Thesis Prep is brought to you for free and open access by the School of Architecture Dissertations and Theses at SURFACE. It has been accepted for inclusion in Architecture Thesis Prep by an authorized administrator of SURFACE. For more information, please contact surface@syr.edu.



Algorithmic Settlements

modeling informal settlements through the use of generative algorithmic design

Ben Anderson-Nelson
Advisor: Brian Lonsway

Background |

Christopher Alexander explains in his "Notes on the Synthesis of Form" that in the act of designing, the designer must choose which information is relevant and which is not. They can draw on specialist experience but fundamentally, "design problems are reaching insoluble levels of complexity." This critical view still admires the designers broad scope when looking into the problem but recognizes the need for a new approach. In a broader sense, what is required is a tool, which assists the operation of an architect. This thesis contends that through developing a tool for simulating the spacemaking of informal settlements, architects could better understand and act upon settlements by hybridizing existing individualism with data-based systemic improvement. Creating a generative tool which can be used to model the informal adaptations occupants might make on formalized settlements will improve the agency with which architects can act in creating and improving those settlements.

Informal settlements are about individual actions, there is a bottom up mentality which pervades their construction and habitation. Materials and spaces are adapted constantly for new purposes and by different independent operators. But out of this construction, which takes place in residual spaces around the city, comes both positive and negative social implications. The designer struggles to understand these implications in their totality, because of the sheer volume of relevant parts. The nature of this creation goes against many of the methods deployed to design cities in the past century. We favor the top down approach, the masterplan. A method that can never encompass all of the parameters which effect and subsequently are affected by a project, and has instead chosen to selectively consider specific attributes. And yet this is by and large the chosen method for governments to address informal settlements. The alternative is localized, sensitive and contextual architectural applications, which are generally more successful, and naturally more time intensive. The impact of these interventions have far reaching effects within the communities that surround them.

Utilizing a typological understanding of the spatial conditions of informal settlements to help suggest how interventions will affect conditions is the goal. Understanding the beneficial aspects of communal space created as well as the nature of adaptability within the settlement provides architects with a better understanding of where and how they might intervene. Simulating the conditions of an informal settlement digitally would help to better understand the individualistic agency involved in their creation. Modeling the inter-relational actions between neighboring entities within the system allows for the system to be understood more complexly than before. Designers would be able to add this tool to their repertoire in order to broaden the scope with which they operate on informal settlements.

Contents |

Research Questions

Key Terms

Case Studies

Informal Settlements

Cidade de Deus

Kowloon Walled City

Torre David

Game Environments

Rust

Minecraft

No Mans Sky

Algorithmic Models

HAS

Kokkugia

Vertical Village

Hypothesis

Inventory

Case Studies

Components

Strategy

Simulation Methods

Customization

Assembly

Questions |

Primary Question:

Through developing a tool for simulating aspects informal settlements, how could architects better understand and act upon informal settlements growth by the adaptability and ownership of existing individualism with data-based systemic improvements?

Secondary Questions:

Can a generative design tool be created which helps retain the adaptability and individuality of informal settlement while operating on a large scale to improve systemic issues?

How can informal settlements be simulated through the use of generative design tools to create large scale models which maintain adaptability and individuality?

How can vernacular and specific components be understood within a generative system without limiting the system's ability to work within different contexts?

Does a generative design tool based on growth have more application to accounting for informality, or acting upon it and in its midst?

Key Terms |

Informal settlement - areas where people have constructed housing and facilities for themselves on land which they do not have a legal claim to; settlements which lack formal planning efforts or zoning codes and are instead a product of individual actions by occupants

Mass customization - using computer aided manufacturing procedures to allow for the objects or components produced to be unique or variable but still economically efficient; typically related to retail or online shopping; the ability to deploy the efficiency of mass production to deliver many variations of a similar idea, for housing this could provide the individuality of personalized spaces (creating more meaning of the individual?)

Individualism - a culture or practice of being self reliant (specifically in respect to constructing or designing one's own architecture); in informal settlements this is the practice of developing one's own space, typically because it is cheaper than paying someone to construct it, and made using found or cheap materials

Adaptation - the ability to change or react to a specific environment; the practice of changing over time to accommodate for new conditions; in biology, evolution is a process which takes place over millions of years, however with more iterations the possibilities are broader, so living things like bacterial evolve more rapidly, this mass-iteration could be applied to housing using algorithmic approaches

Algorithm - a formula or set of operations which define a process by which a problem can be solved and a solution produced; it is a set of parameters set up so that an outcome can be reached through computational methods, in architecture it is usually intended to expedite the process of producing iterations

Ownership - in relation to settlement, it is the idea that the individual living or working in a specific place has added value to that place because of the effort they have put into creating it; when someone has rights to something, they inherently place more value on it and frequently feel incentive to protect/develop it which is a very positive thing in a community

Self-organizing - the ability for a community (in this case) to construct their own spatial rules, these rules can be based on the individual goals of occupants, but lead to larger implications on the form of the community, and as such can be positive or negative (ex: increased density near circulation routes → safety and utility, less density near waste and trash areas → less safe, increase in violence towards women)

Procedural Generation - a process of creating data (visual, numeric, etc) which relies on algorithmic input and output rather than manual creation (commonly used in video games for textures, various content elements, and physical geometries); could be used to model informal settlement in the fact that it

Dependent system - a nested set of algorithms, where each algorithm is dependent on the output of the previous, these algorithms build off each other and have a highly structured method for creating the final outcome of the system; with added complexity, more variety can be created but the system can behave increasingly unpredictable; modeling the informal settlement with this method would allow for different layers of complexity to be developed based on the what the site constraints caused the program to return

Emergence - the idea that when observing processes, larger entities and patterns occur within those systems naturally as a result of the interaction between smaller entities; idea that smaller actions within a system can produce larger observable patterns → this produces natural organization from what would otherwise be chaos → elements have much more meaning as a collective than as an individual due to the larger order produced

Swarm intelligence - collective behavior of organisms where multiple agents interact locally and produce results which indicate emergent behavior patterns (ants, bees, flocking birds, schools of fish etc) → essentially emergence in relation to groups of organisms behaving together

Cellular Automata - grid with each square having specific states it can be in, these change based on stimuli but remain in place in the grid; could be used to abstractly model the informal settlement's density over time or perhaps perimeter growth

Case Studies |

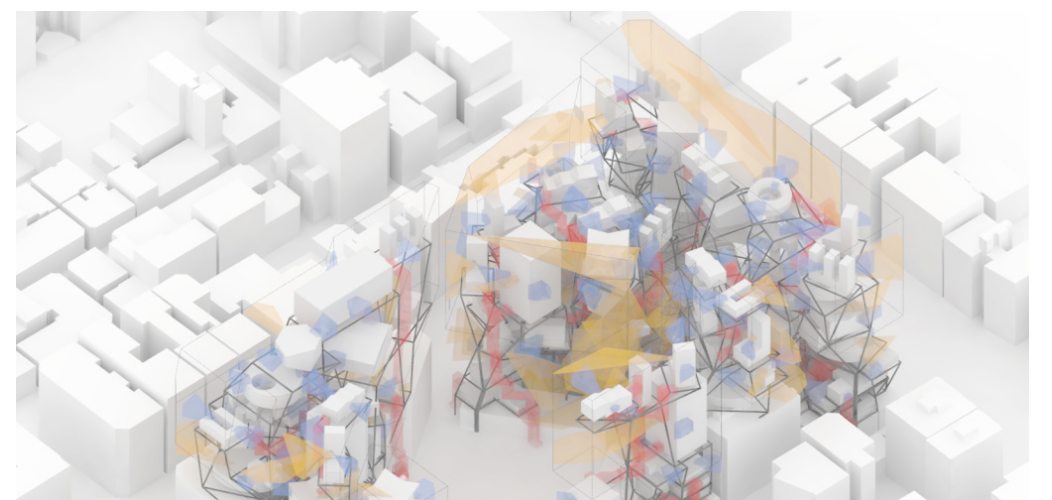
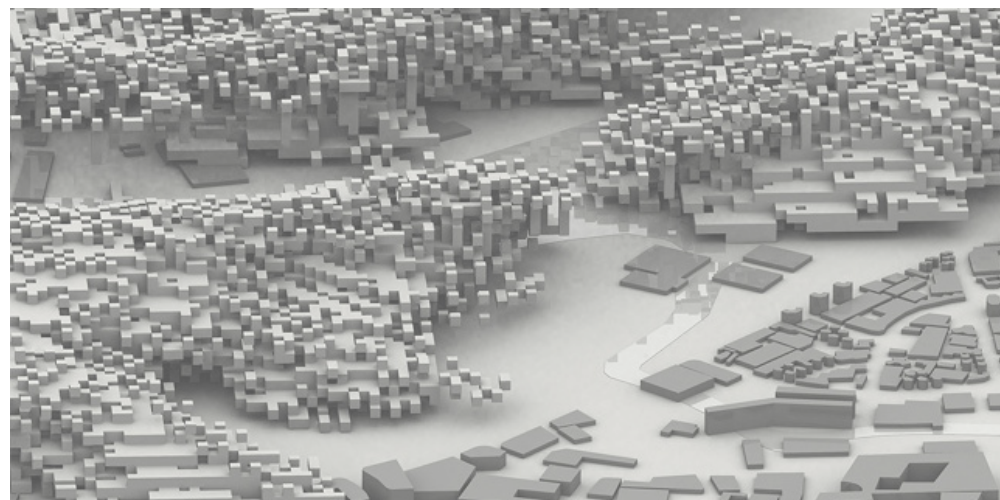
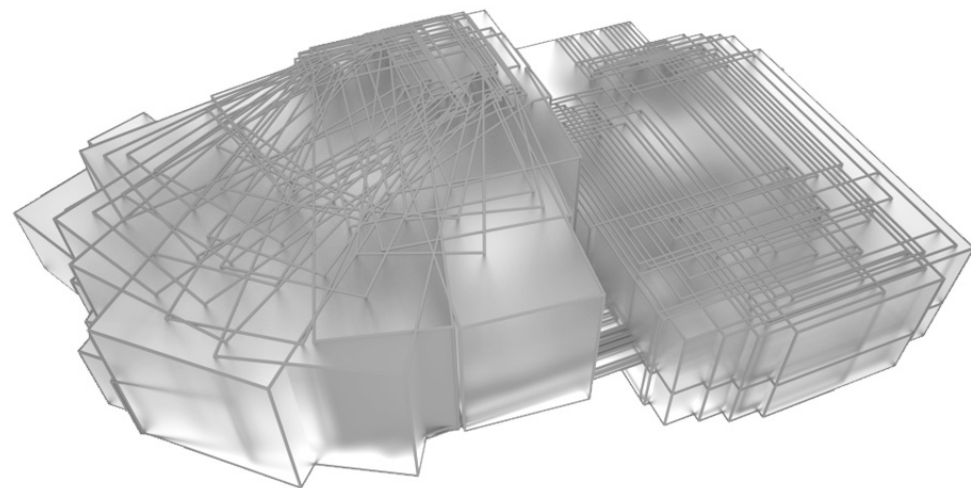
Settlement



Game Environment



Algorithmic Modeling

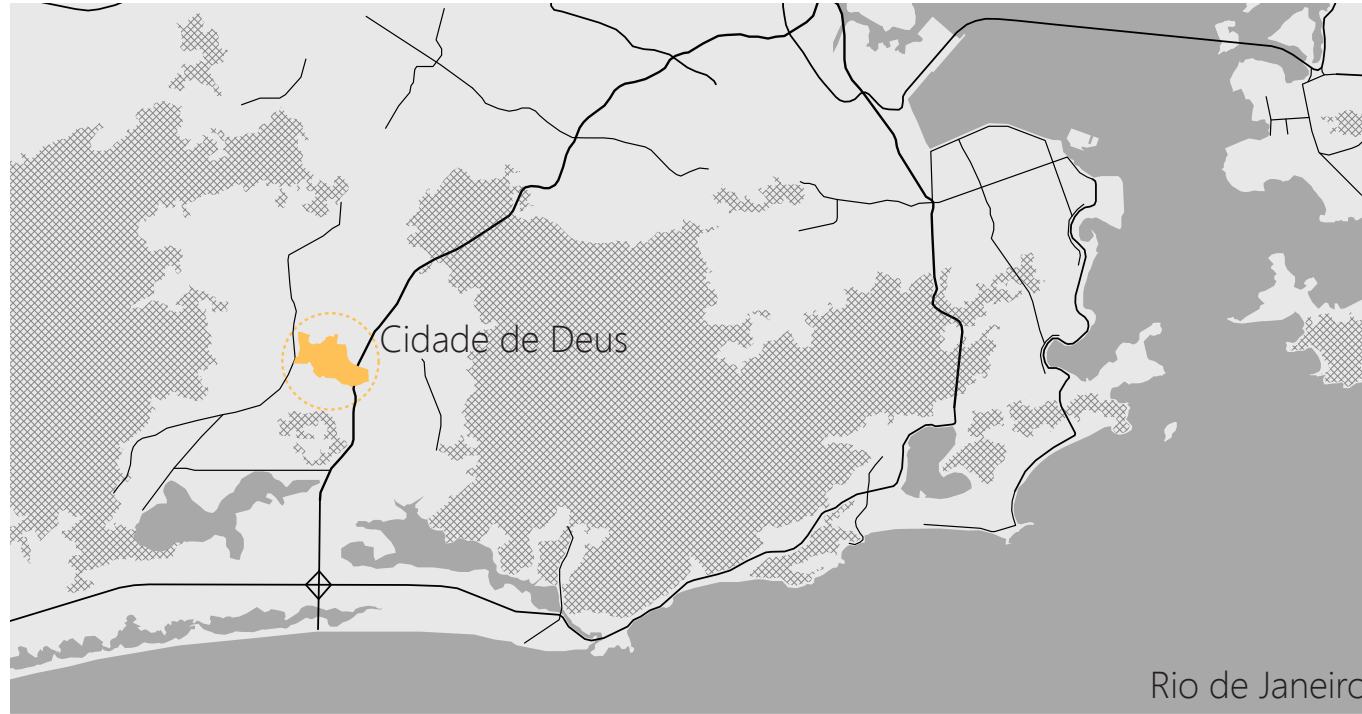


Settlement: Cidade de Deus

Origin

Cidade de Deus was created in an effort to move people from urban favelas around Rio de Janeiro into a more suburban/rural setting outside the city. Its intention was to provide adequate housing, but at the same time it moved urban poor out of the city's more valuable real estate. This suburbanization distanced the occupants from jobs in the city and the new and isolated community was not dense. The suburban planning methods allowed for residents to slowly infill the blocks until it achieved a density similar to that of urban Rio de Janeiro.

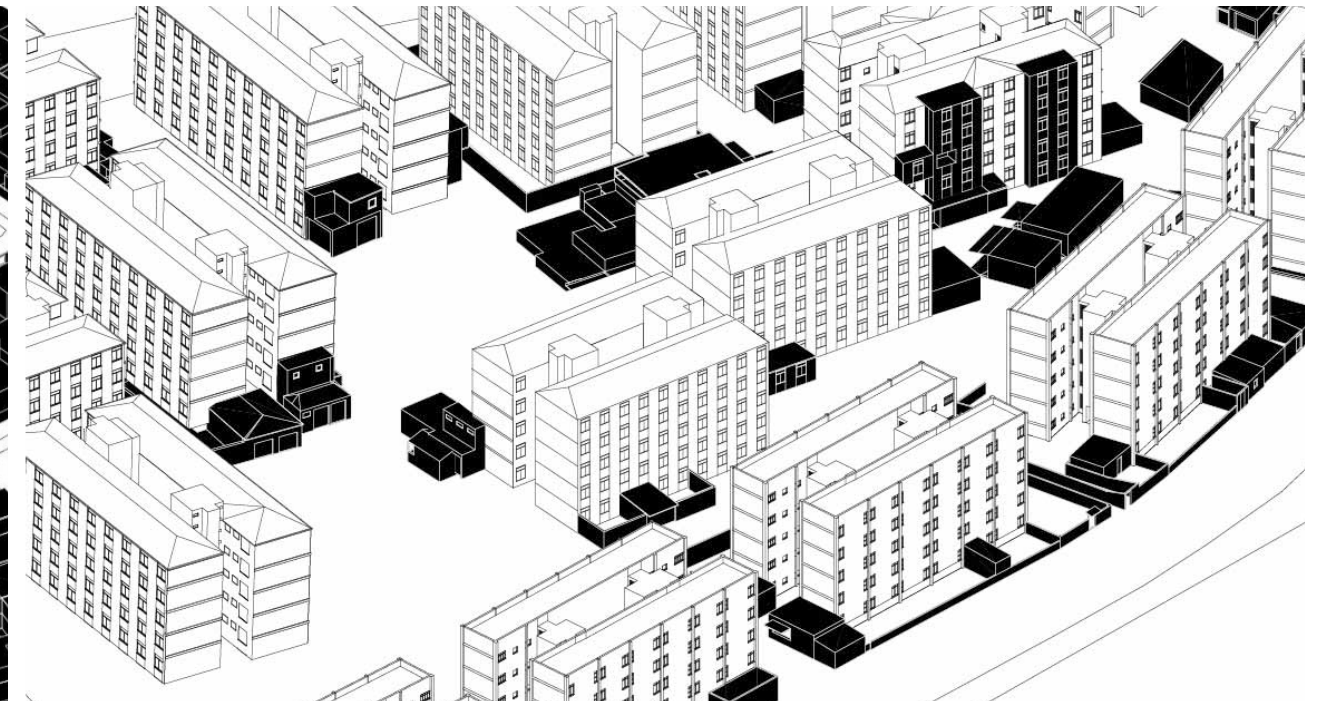
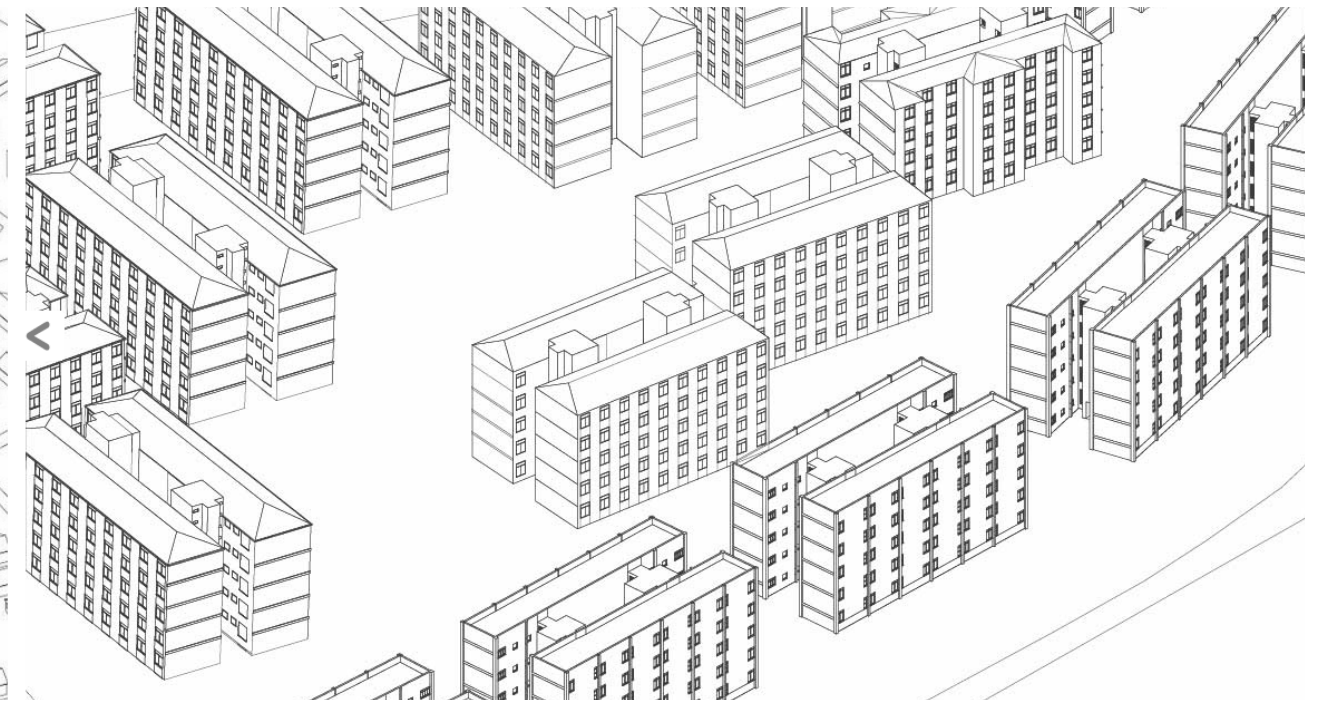
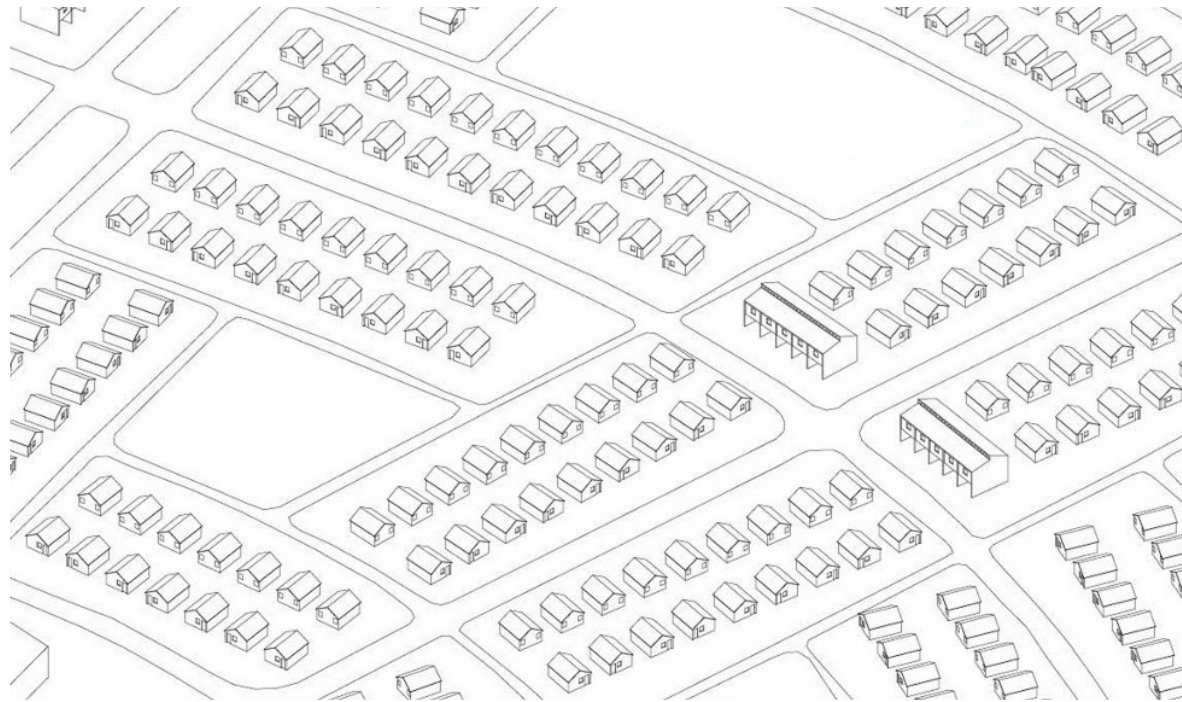
The city today has filled in. This diagram shows the full extent of the Cidade de Deus today with the solid yellow line. The suburban housing units which consisted of two bedrooms, a kitchen/dining room and bathroom have been deconstructed and enveloped by the expansion. The city today has turned from a poor and dangerous settlement to a new place for the growing Brazilian middle class. The city was the target of a strong police crackdown on crime in the early 2000s and has since bounced back. A local currency called CDD has been created which promotes local business and is currently valued at 20% higher than the national Brazilian Real. In 55 years of growth a what was essentially a government initiative to displace poorer people has become a thriving community.



Settlement: Cidade de Deus

Development

The city is composed to two original planning schemes, suburban and vertical mass housing. Over time, both housing typologies were informally expanded on to accommodate the residents needs. The suburban typology grew denser with added structures to buildings and eventually the removal of the single family houses almost entirely, in favor of multiple story residences. Other spaces became stores and other programs. The high rise housing was acted upon in similar ways to create space for stores, garages and other programs. These additions helped to fill in the open spaces between high rises. Open blocks planned into the suburban side of the settlement remained free of structures and are still main gathering spaces. The suburban style houses were much easier to informally build on than the tower housing. The construction and planning of the towers was such that it was often too difficult to add on

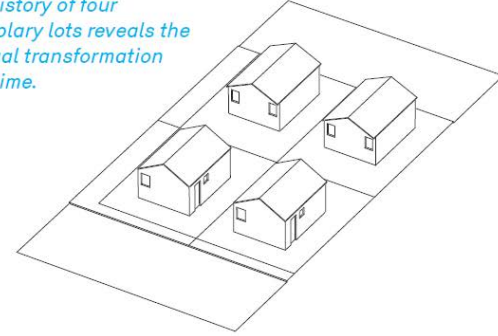


Settlement: Cidade de Deus

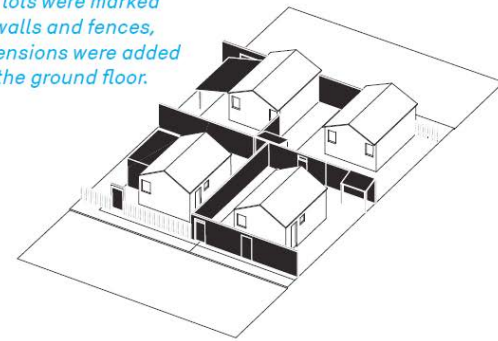
Means and Methods

In the book *Cidade de Deus*; working with informalized mass housing in Brazil, the authors diagram the expansion of the both suburb and high rise housing over time. The addition of ad-hock pieces were added over many years and predominantly at the discretion of the individuals constructing them. The informality stems from the poverty of the community, the nature of construction is a product of the resources available. Each addition has different implications for the community, walls serve to protect but separate individual parcels of land, while porches and awnings can provide inviting spaces. The city is a product of formal master planing as well as informal development. The informal construction is essentially occupants of the city investing in their community in a way because it was one of the only ways real estate was added. This city is a case where a fairly depressed government development was converted into much more valuable space as a result of informal construction and community building.

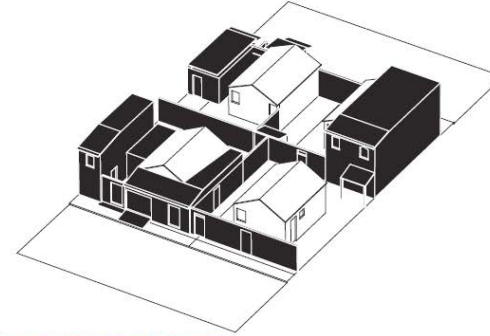
The history of four exemplary lots reveals the gradual transformation over time.



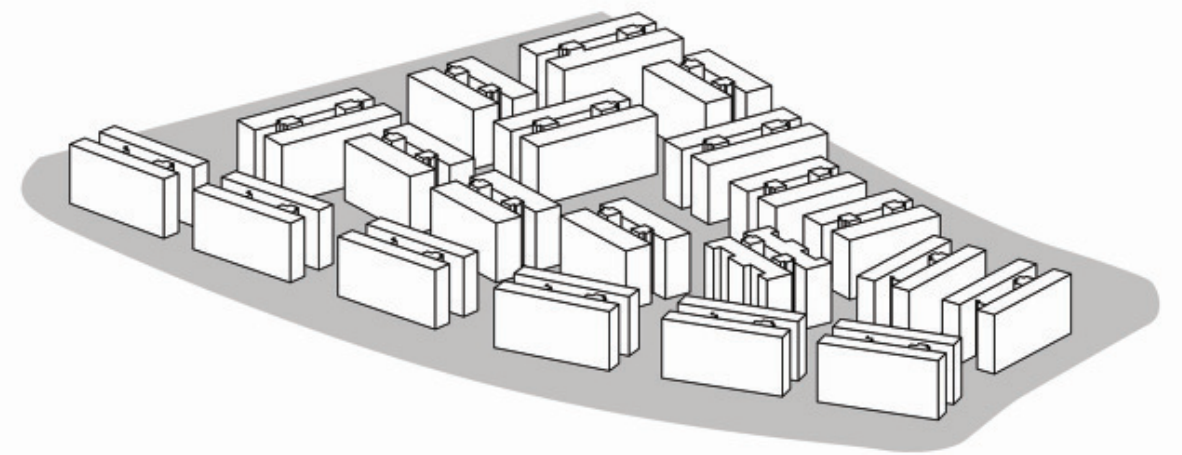
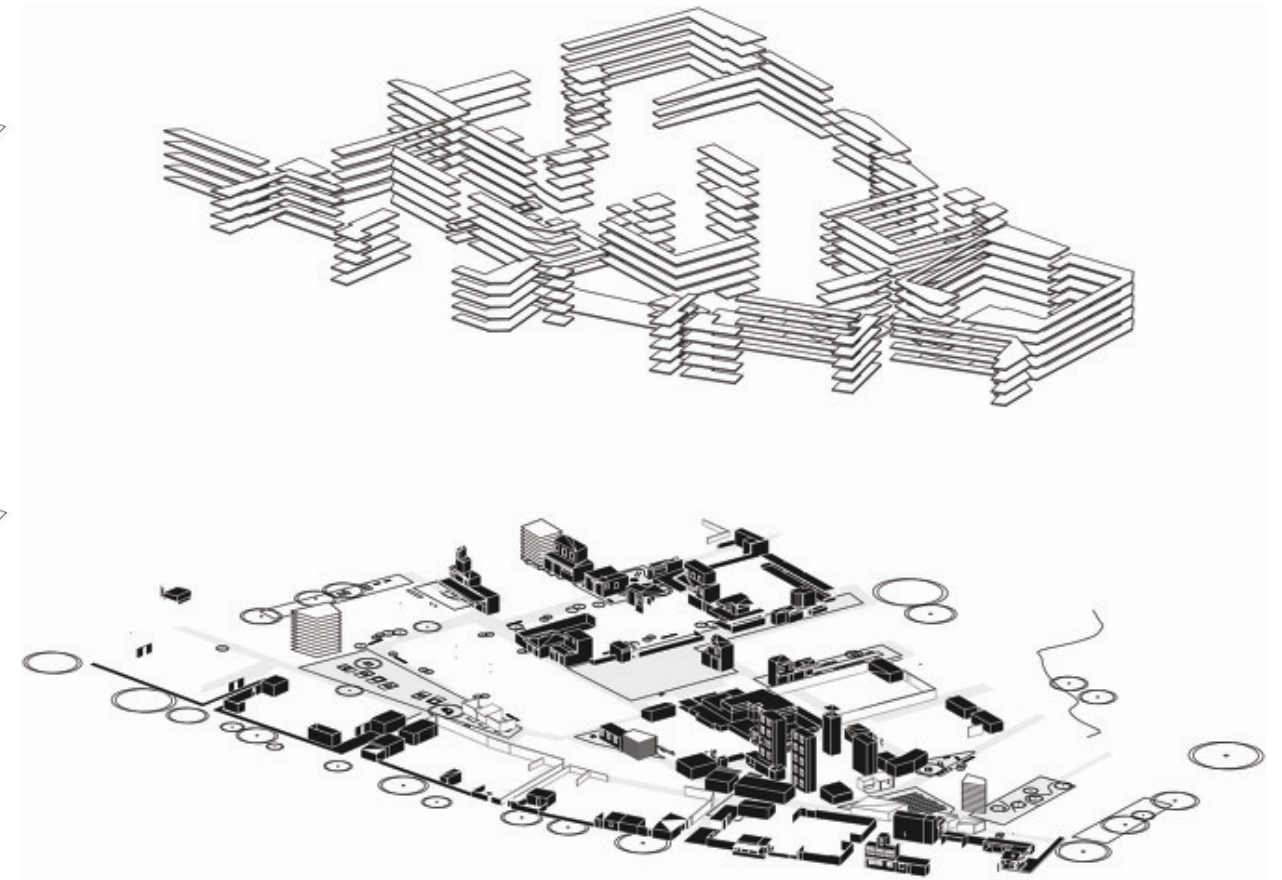
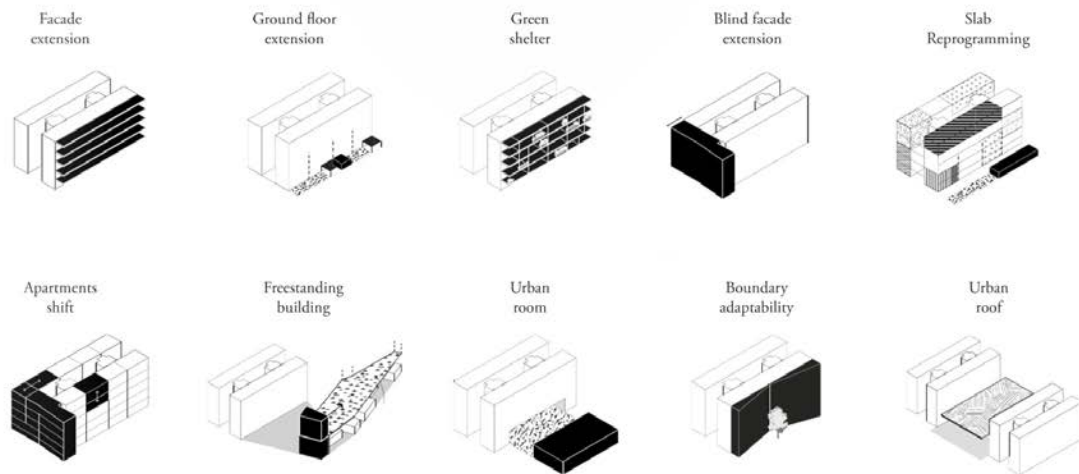
First the borders of the lots were marked by walls and fences, extensions were added on the ground floor.



In further steps additional floors were constructed -in most of the cases the original house was replaced.



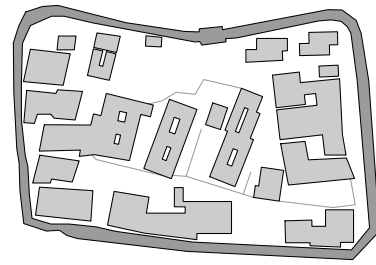
EXTENSIONS CATALOGUE



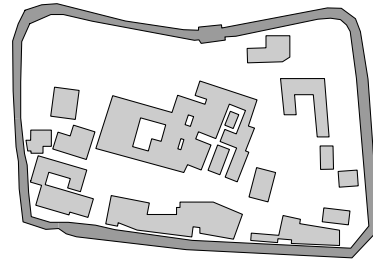
Settlement: Kowloon Walled City

Origins

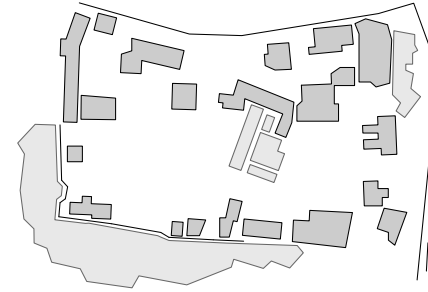
Kowloon Walled City began as a Chinese fort which was constructed between 960 and 1297 BCE. In 1841 the British occupy Hong Kong and by 1898, jurisdiction over the fort and the small "city" it now surrounds is being debated between the Chinese and British. The British finally seize the city, but due to the fact its surrounded by slums, no effort is made to maintain the city. The walls surrounding the city are demolished in 1940 by the Japanese occupying Hong Kong, and from then on, the city becomes a haven for people who are escaping rural life for more successful prospects in the city. Due to its location and confusion surrounding government control, it becomes a haven for poorer people and even lower middle class who constructed business operations within the city. It also became home to drug users, criminals and other more darker aspects of society. On a basic level it grew from a need based desire for more housing space, coupled with a complete lack of governmental oversight or regulation. Developers saw an opportunity for profits unsupervised construction, and built high rises from 1973 onward, increasing the density of the city until it was the highest in the world.



1847



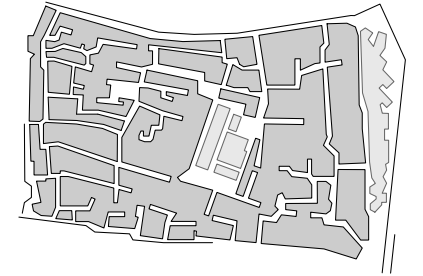
1940



1965



1973



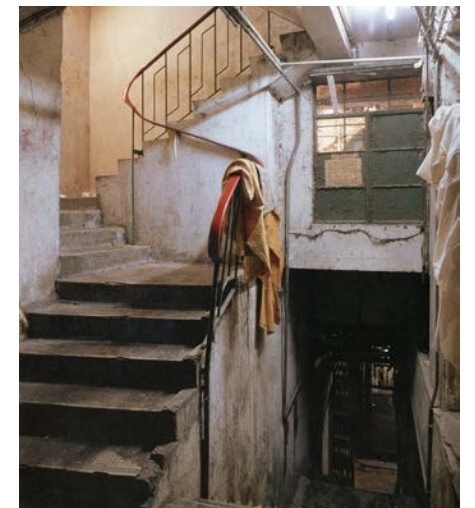
1989



Settlement: Kowloon Walled City

Density

The city started with around 2000 occupants in 1940, grew to 10,000 by 1973 and topped out at around 50,000 by 1990 when it was torn down. The majority of the city was made up of housing for the residents of the city, but it also contained numerous businesses, factories, food processing facilities and even a temple. The city was also home to many more unlawful programs such as gambling facilities, prostitution houses, and drug dens. The illustrated section below shows the density of the city as well as some of the activities going on inside the block. The city also became a place for people in Hong Kong to buy goods at cheaper rates than they could downtown. By the end of its existence, it was providing a valuable market for goods within the city as well as homes for 50,000 people. However the sanitation within the settlement was very poor, over the years the Chinese government had made efforts to bring in clean water and electricity, but largely unsuccessfully.

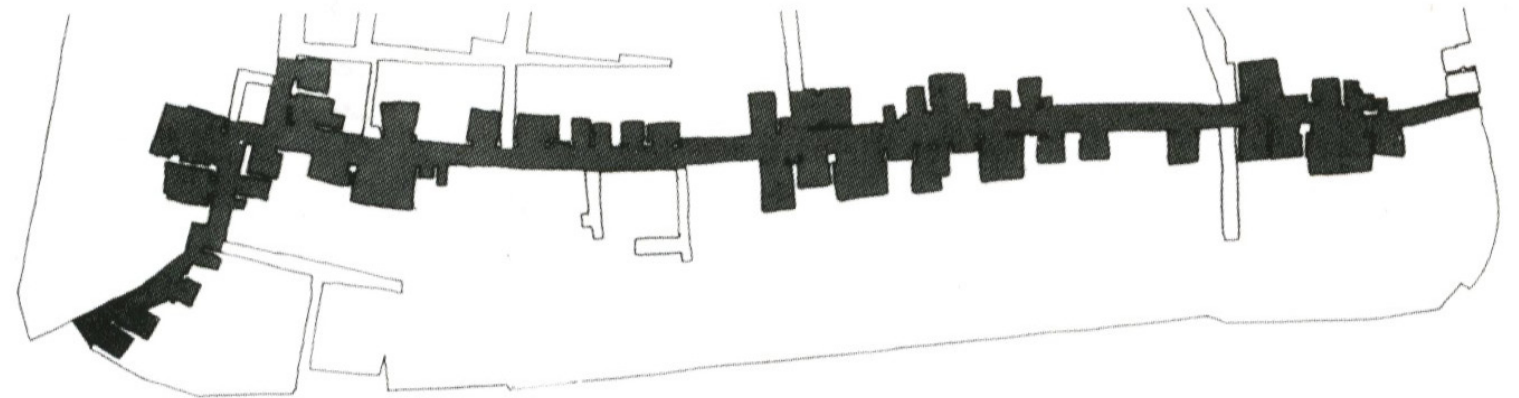
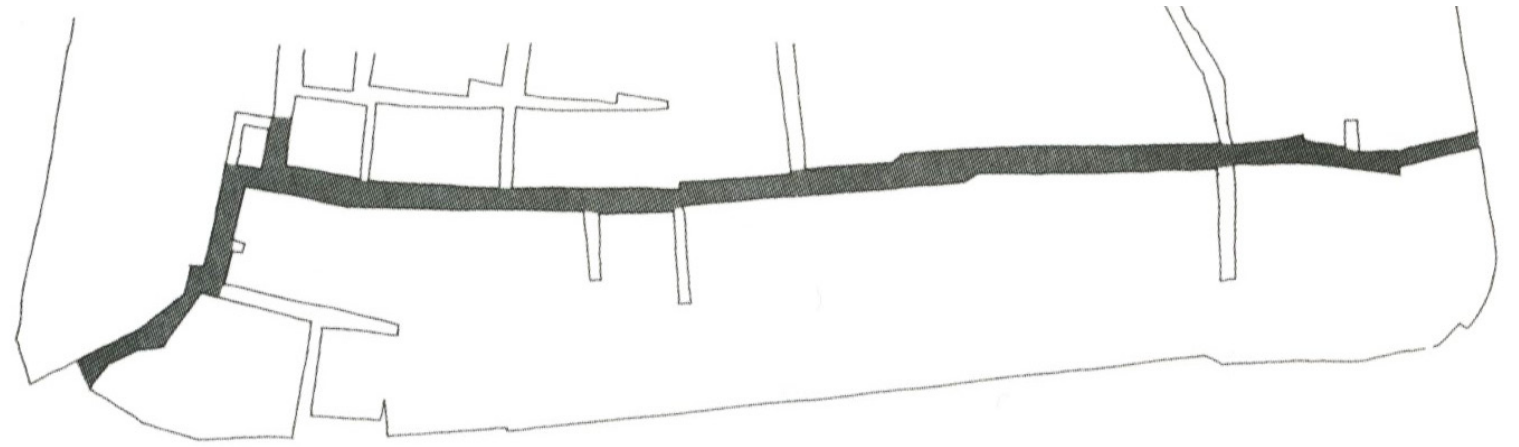
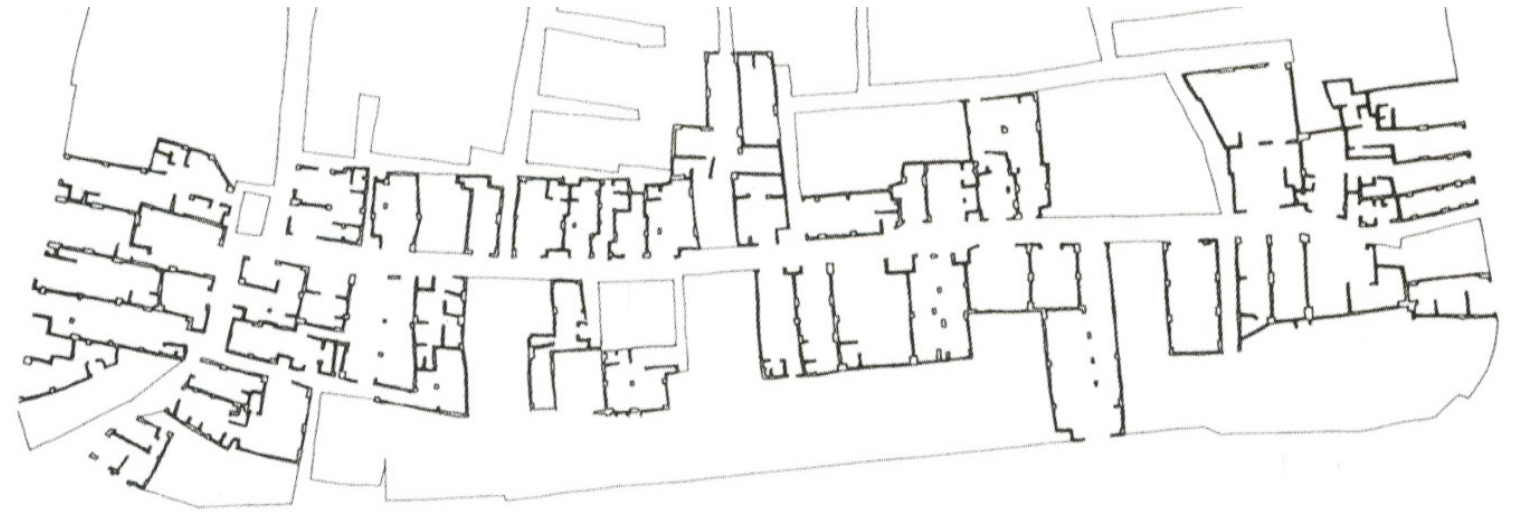


Settlement: Kowloon Walled City

Circulation and Aggregation

These are diagrams from the book *City of Darkness* and they help to illustrate the spatial aspects of Kowloon walled city. The diagrams on the left show how the scale of the buildings changed over the course of the city's development. The city began as smaller single or double story homes. Over time as developers from Hong Kong saw the city as a building opportunity, taller tower housing of up to 14 stories was constructed. The initial small parcels of land were consolidated into larger and taller buildings.

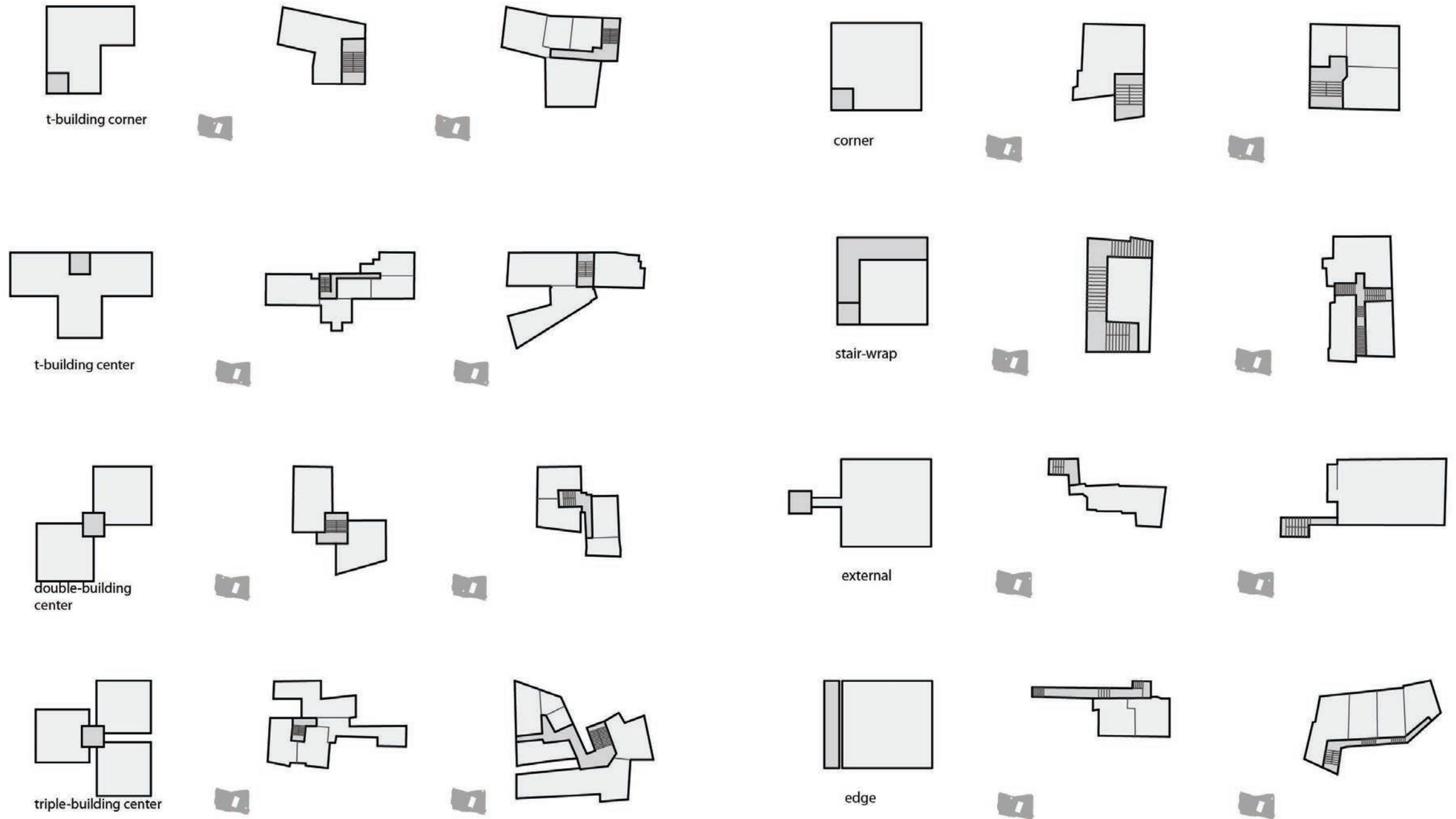
As the city filled in and became a dense block, streets became increasingly compact. They carved through the block and had numerous alcoves for businesses opening up onto the corridors and small public spaces. There is a chaotic, maze-like quality to the circulation, but due to the consolidation of parcels the circulation became somewhat more straight forward.



Settlement: Kowloon Walled City

Apartment Typologies

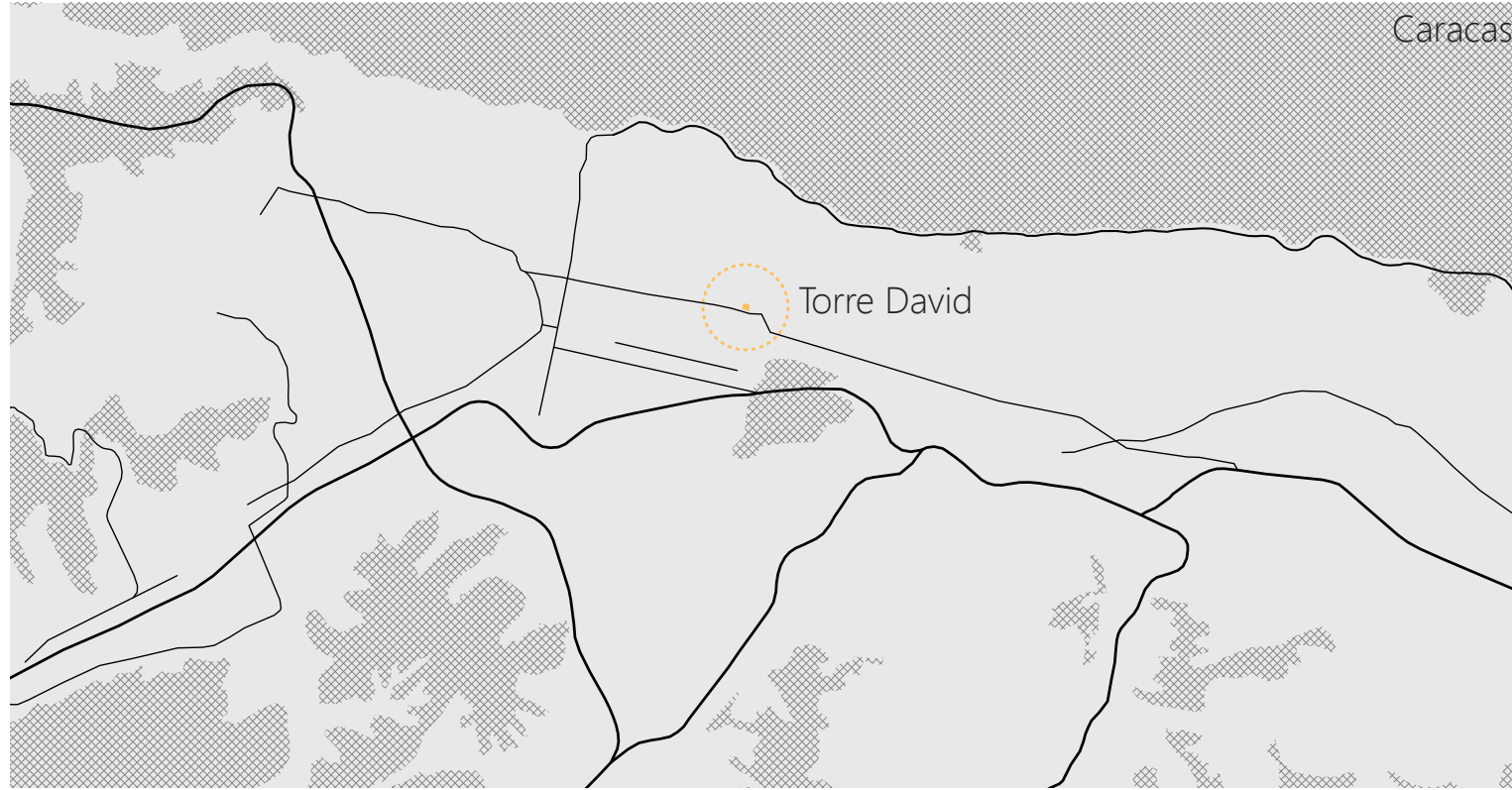
In the book *City of Darkness*, Ian Lambot categorizes apartments relationships with circulation into various typologies. These diagrams are self made reproductions of his typological analysis, and serve to show the different formal interpretations for each typology. Understanding the ways in which the high-rise towers shift form around staircases and access points is crucial to understanding how the city was spatially conceived. Buildings sometimes had public staircases which went through private apartments. And frequently buildings would borrow staircases from the neighboring building and simply punch through the wall of the stairwell in places to provide access to their own corridors. This was highly unsafe and challenging to navigate, but residents frequently took staircases directly to the roof and used that as a means of lateral circulation to get to their own apartment.



Settlement: Torre David

Origins

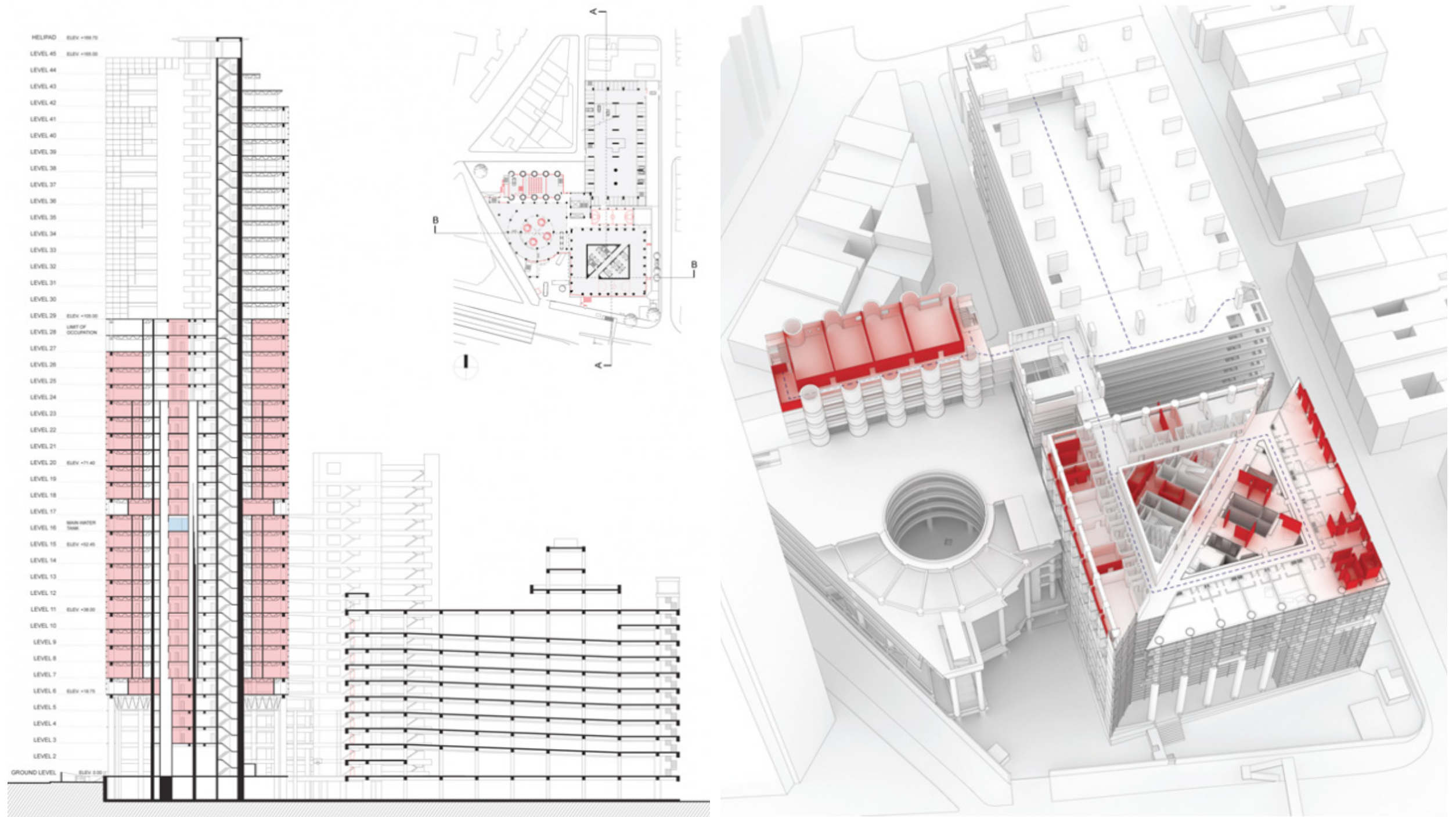
The tower began as a development project that ran into financing issues when the owner died and the recession hit, and so was abandoned in 1994. The government absorbed the bankrupt company which had begun the project. The property was guarded for years, but many of the valuable materials on the site were taken by thieves and scavengers over the years. In 2007, there was a flood in Caracas, and some of the people in affected areas sought shelter in the tower. The guards let the people shelter in the tower, after which the new occupants decided to see how long they could make it last. After months and then years of being left alone in the building, residents began to make their homes more permanent. New residents converted the office tower floors into various configurations of apartment and retail spaces. The tower provided a relatively safe and environmentally protected area for people to live. As people moved into the building and constructed residences, others moved in and constructed stores. The tower functioned as a thriving community for a number of years beginning in 2007 when it was first occupied until 2014 when police evicted the residents. As of now, the usage of the building is currently up for debate.



Settlement: Torre David

Occupancy

The diagram to the left shows the location of various non residential programs within the tower. The majority of stores which opened were grocery and convenience style stores. Other programs which were added were workshops and manufacturing spaces. Due to the unfinished nature of the building, there were no elevators for residents to use, making it increasingly difficult to live higher up. The residents were able to use motorcycles to get up to the first 10 floors due to an adjoining parking garage, but the remaining floors were only accessible by stair. Less able residents like the elderly and handicapped usually took up residences near the bottom, with able people living higher up. The total population living informally in the building was 2,500 at its peak occupation. Its occupation began with the housing crisis in Caracas, where there is a shortage of an estimated 400,000 units. The building was only occupied up to the 28th floor due to issues with getting water high enough up and general accessibility. As can be seen in the section to the right, there was a water tank on the 15th floor which provided residents with a small, but weekly resupplied reservoir of water.

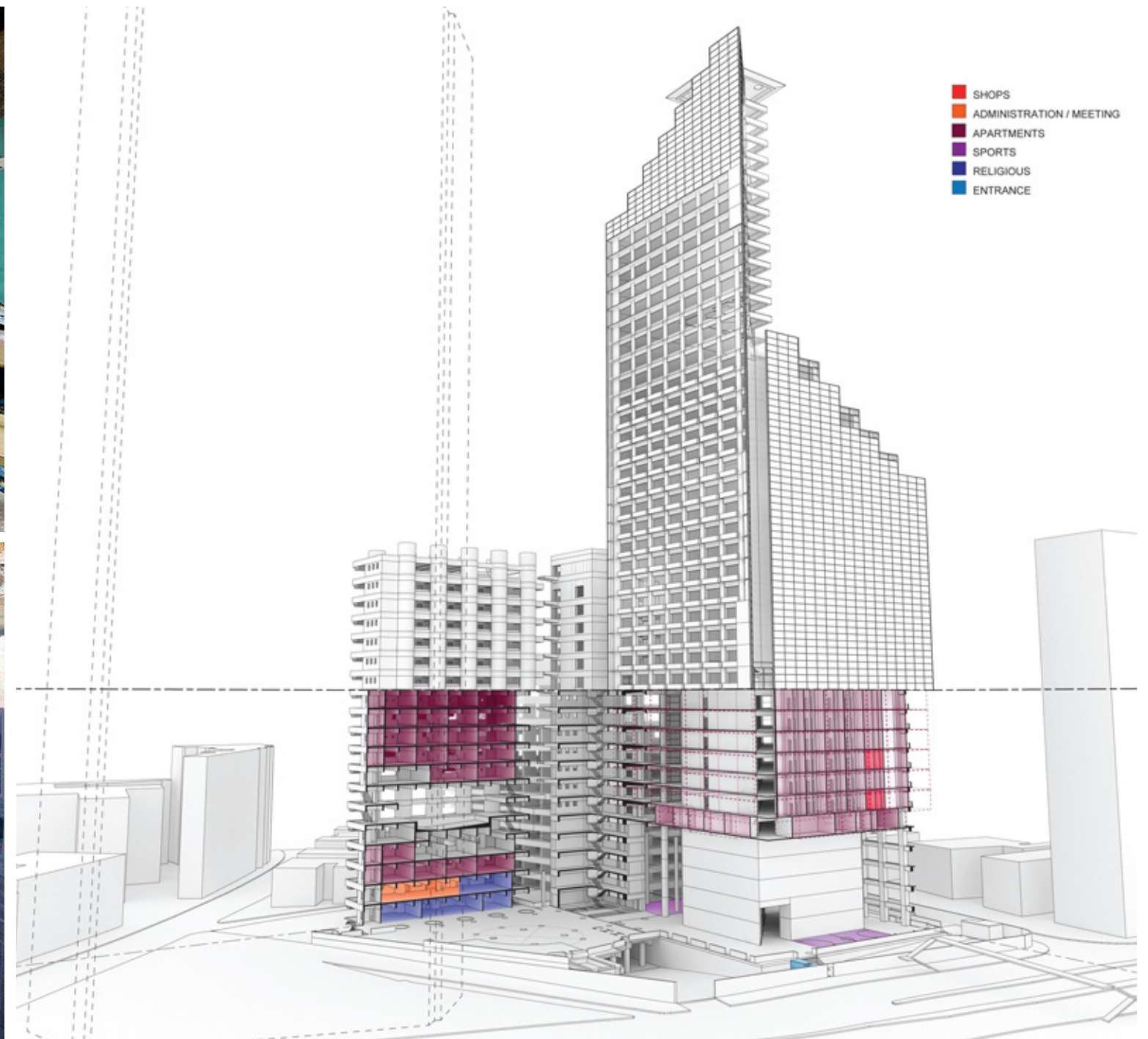


Settlement: Torre David

Programming

This is a drawing made by Urban-Think Tank which looks three dimensionally at programming within the tower. Their goal was to understand the vertical settlement better think about ways in which the settlement could be a location for design experimentation. They developed a comprehensive set of diagrams which explain the technical side of the project, and accompany it with photos and discussion about the buildings occupants. This firm has done extensive projects in the favela's surrounding Caracas and their research into how informal settlements could be acted upon in more sensitive and intelligent ways.

Much of the building was converted into apartment style living, usually with short walls (due to the trussed deck system). On the top left is pictured a woman working in a tailoring shop set up within Torre David. The adjoining parking garage made it possible to get industrial machinery and other commercial equipment up to any of the first 10 floors.



Settlement: Comparison

	Cidade de Deus	Kowloon Walled City	Torre David
Development	Built up overtime as people informally added on to properties	Developers from Hong Kong moved in and consolidated small apartments and houses into high rises	After not being evicted when expected, individuals settled in and walled off a eas for themselves
Origin	Began as government settlement, suburbanising favela dwellers	Began as a fortress, became vacant due to territorial questions and then occupied by squatters	Residents began moving in after a flood, eventually evicted in 2014
Density	Density grew over time, currently its at 36,000 people, which is 4 times the original density	The densest settlement in history, 50,000 people living within the perimeter wall	At its peak it housed 2,400 individuals on 28 of the buildings floo s
Community	Over time the environment has changed, very dangerous for a period, now transitioning into middle class	Large commerce and factory sector, initially very group minded, but became less so	Insulative community, kept outsiders out, less physically able residents were priviledge to ground floor

Case Studies |

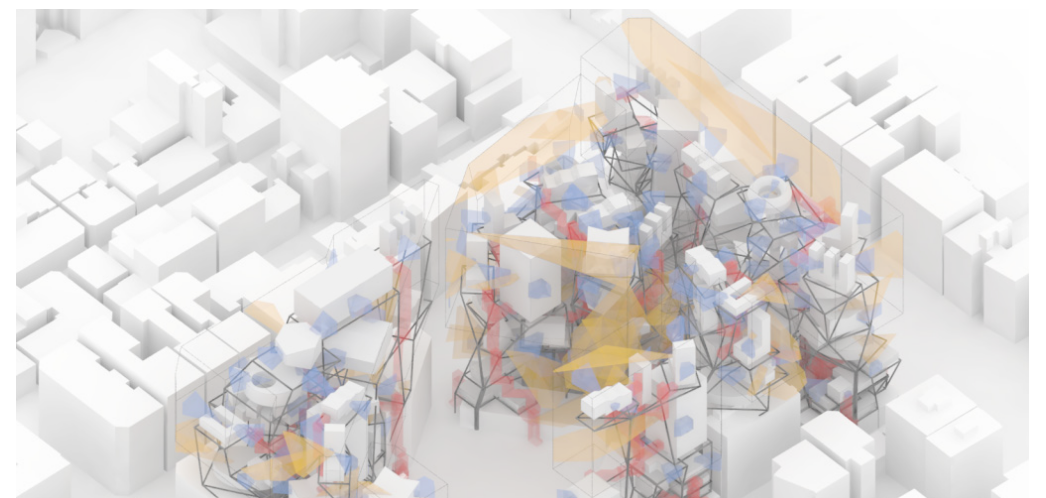
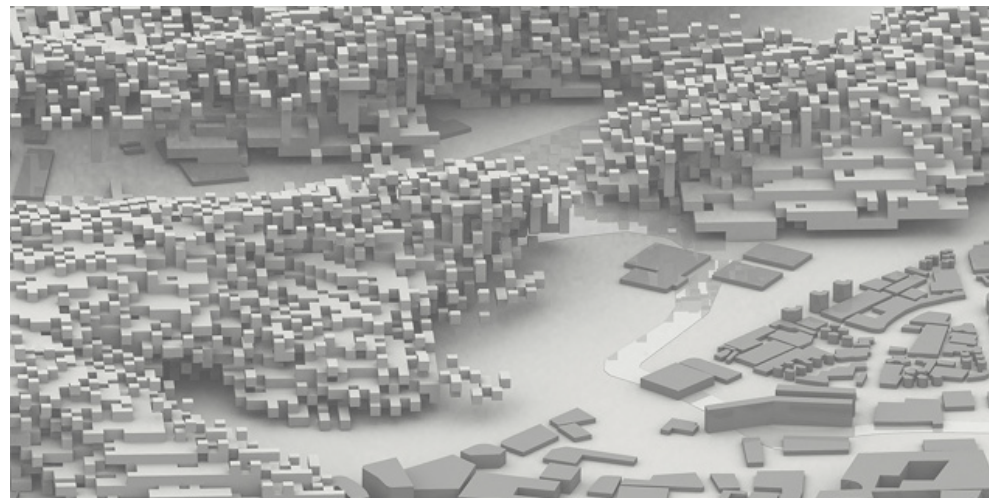
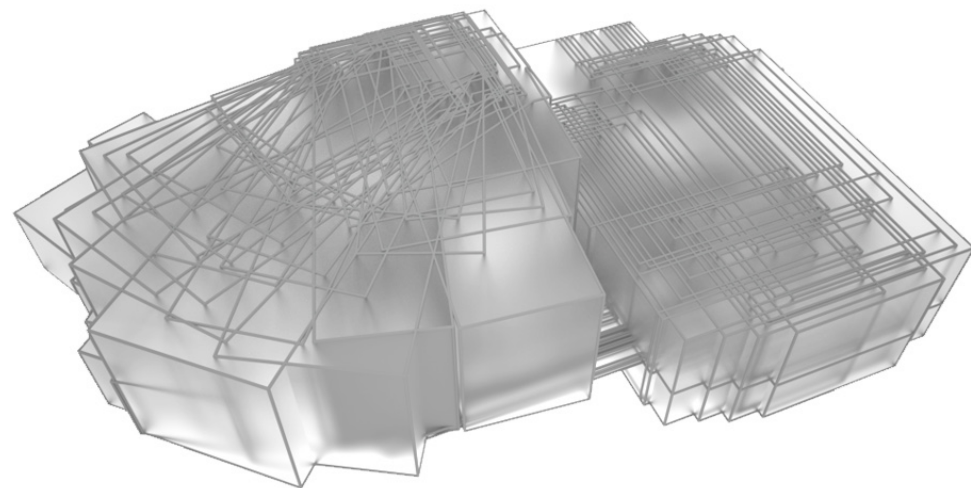
Settlement



Game Environment



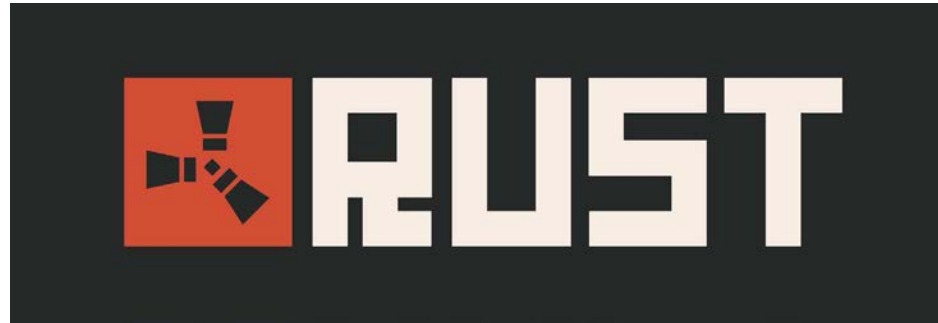
Algorithmic Modeling



Game Environment: RUST

Background

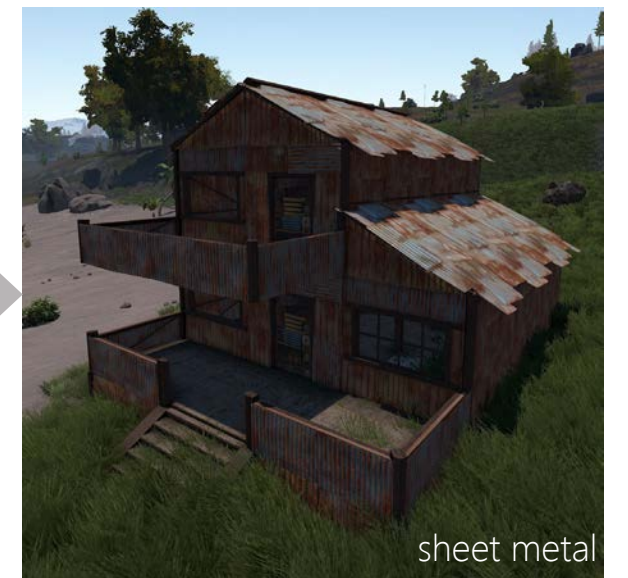
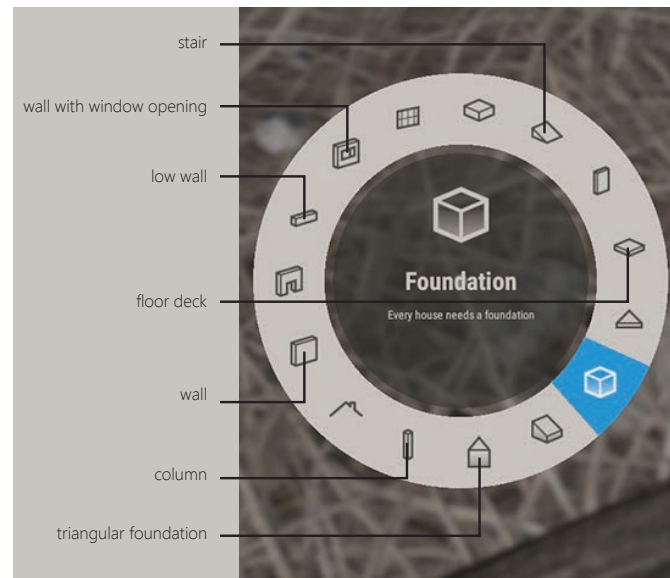
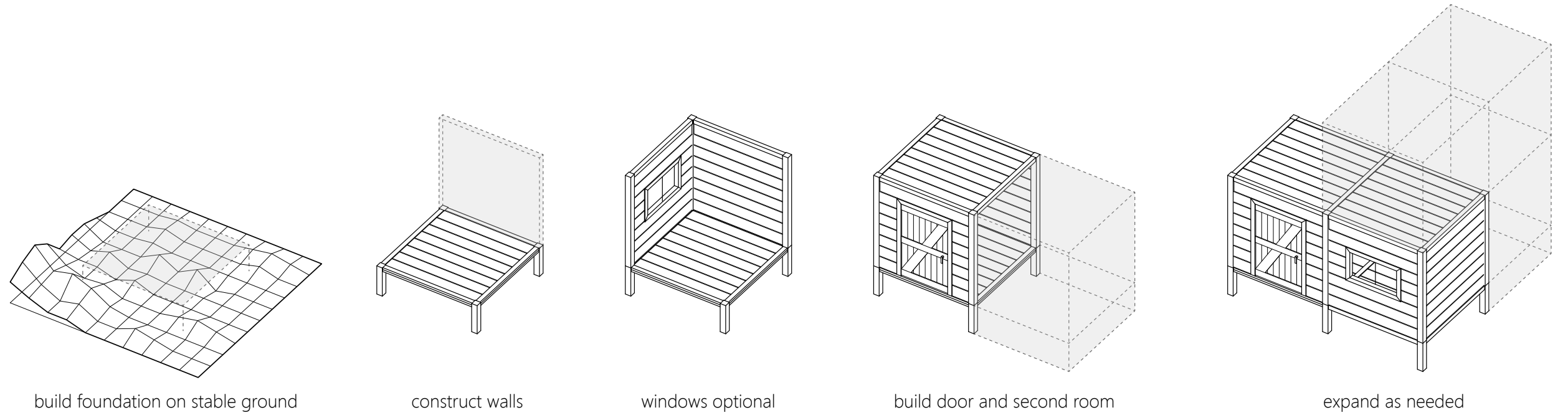
Rust is a game in Alpha release created by Facepunch Studios in the UK. It is a first person, multiplayer, survival game in which players collect and scavenge materials from their environment in order to create various structures and items. Players can attack one another and take each others possessions, making it crucial to build a structure which can protect what you've made and collected. When you log out, your body remains "sleeping" on the server, making it vulnerable to other players. Due to this hostile and resource scarce environment, players creations are interesting to study due to their ingenuity and informality. Often players work together to protect their common interests and form factions. These groups create large structures to protect their valuables and share them with each other. Other groups act as traders, pacifists, raiders, pretend religious orders and a variety of other creative roles. While the context is unlike any real-life scenario, the structures produced often resemble aspects of informal settlements.



Game Environment: RUST

Building Mechanic

Once players have collected resources, they can craft a building plan. Once created, building is done through an interface which appears on the screen when a plan is accessed. It provides a series of selectable options of modular pieces for players to build with. The player decides where to begin building, and the ground must be sufficiently flat to place a foundation. There is no global grid system or module in the game world which constructions have to adhere to. So once a player has set down a piece, every subsequent piece is referring back to the module of that original piece. The main goal when constructing is to make sure that resources are protected by whatever is built. To this end, any initially built material can be upgraded four times, eventually reaching the limit of its defensive qualities at reinforced steel paneling. Upgrading requires incrementally more resources with each tier, and thus requires a great deal of effort by the player, making resources even more valuable.

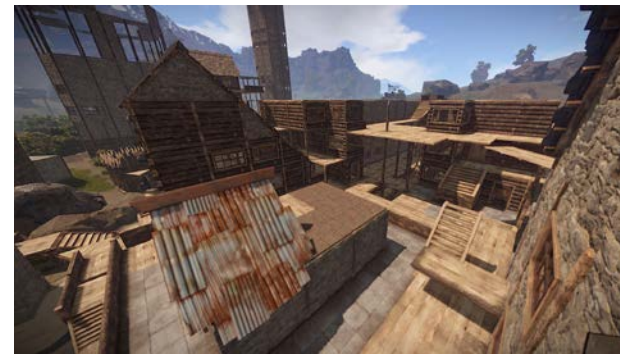


Game Environment: RUST

Community

These are examples of player built towns/villages. These creations are the product of random individual players getting together into larger groups and building together. There are no larger governing rules by which these players are operating, the ability to steal or murder is the same. There is an interesting social aspect to this construction, in that the players benefit more from being packed closely together because of the ability to share materials and protect each other. It is the virtual game equivalent to early villages throughout the world.

Interestingly, due to the textures of the materials in the game, there is a similar aesthetic to that of many informal settlements. More interestingly though is the spaces both intentional and residual that are a result of the construction. Many independent operators come together with different goals, but similar access to materials and building methods and they have created a cohesive community. One large structure which housed all of the players involved might be less safe. This is due to the fact that none of the players living there would have anything of their own invested into the creation, it would simply be living inside someone's created environment. In a sense players have something at stake and so they treat their neighbors as they want to be treated.

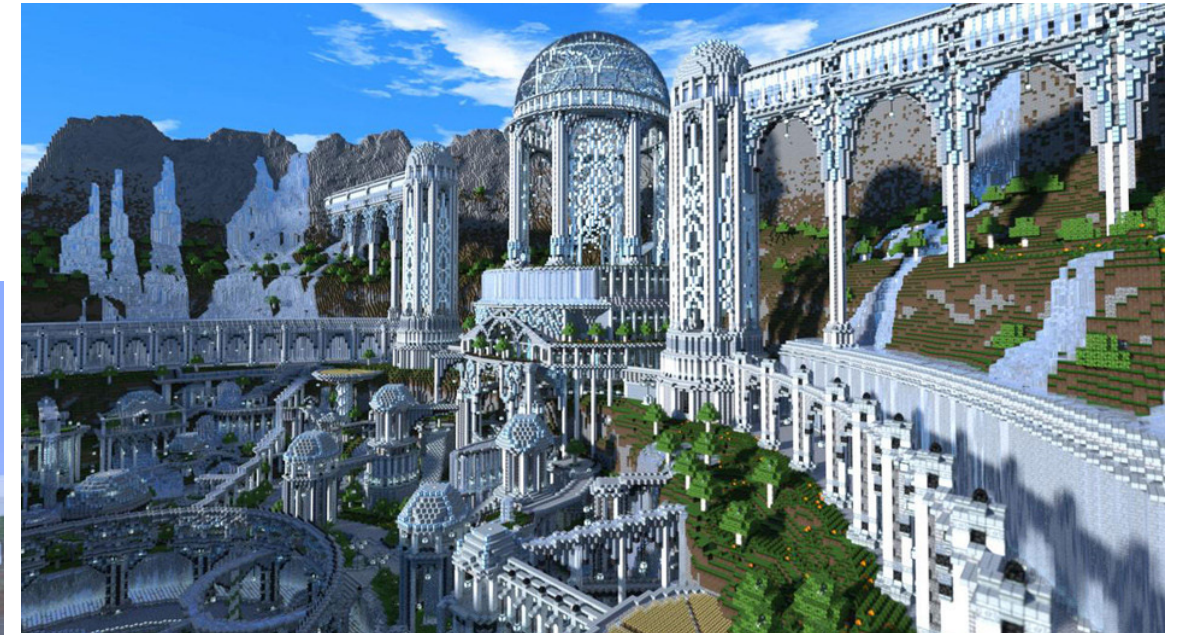


Game Environment: Minecraft

Background

Minecraft is an independently developed game that has acquired a lot of attention in the past 5 years. Due to its game-play concept, it has become an influential example within game design for survival and sandbox games. It abstracts the realism of so many other games in favor of 8-bit textures and simple character models. The biggest element to the game that sets it apart, is the fact that it breaks down the game world into modular voxel blocks which can be altered at will by the player. The game can be played in two different ways, one way is creative, where supplies are limitless and great creations can be built and the other is survival where supplies are collected, and there are hostile creatures. In both, players have managed to create impressive structures which take hundreds of hours to build. Due to the fact that the game was initially created by a single person, Notch, the generation of different worlds players build in, is computational. Every world is generated through a seed value, which is a numeric value that guides the output of the generation. Each value produces the same result every time that it is used to generate a world.

MINECRAFT

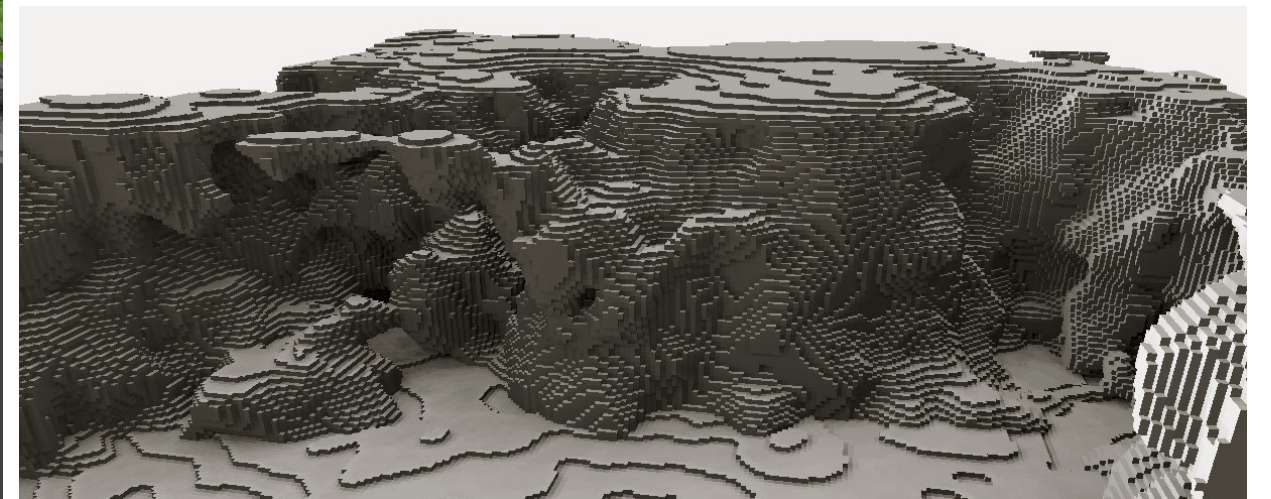
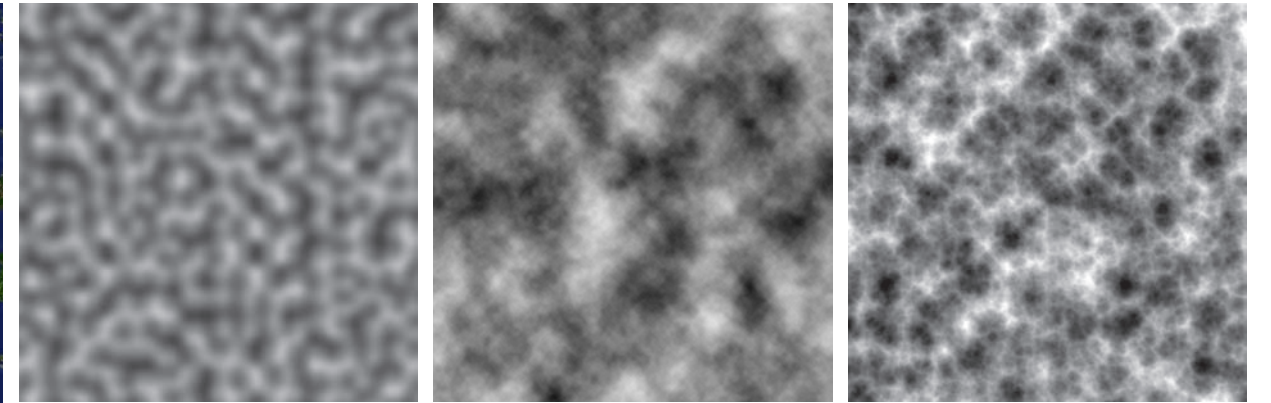
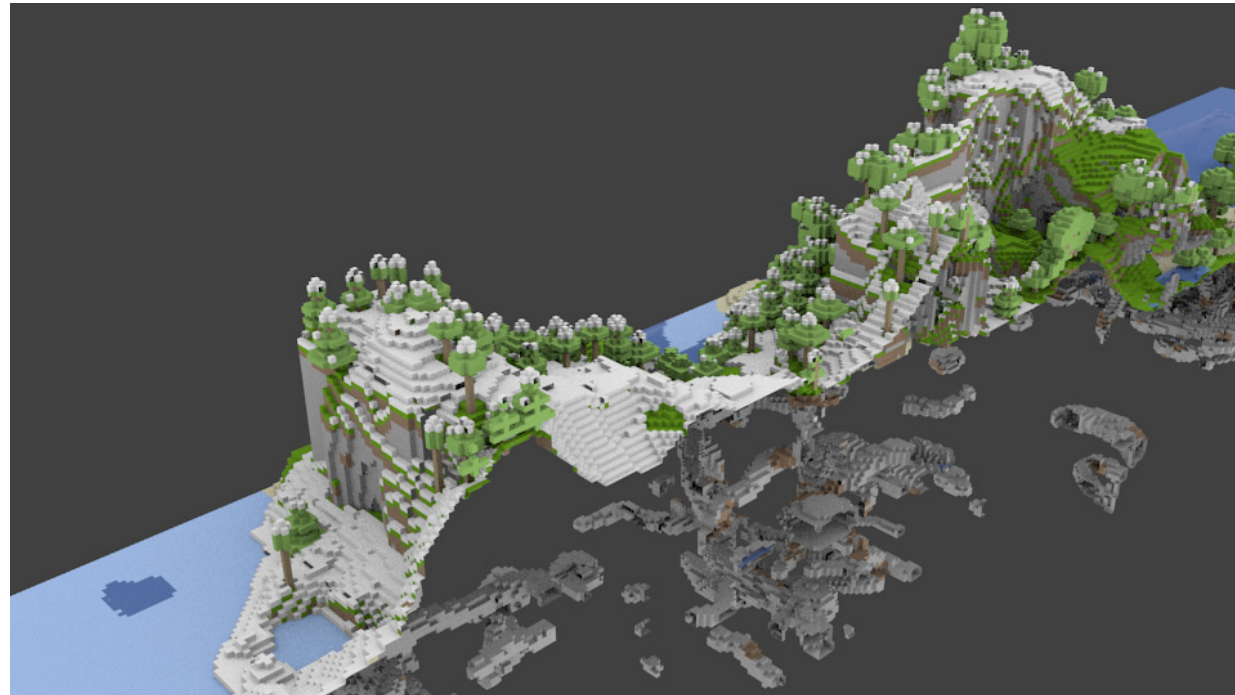


Game Environment: Minecraft

Generation

The generation of the world is done primarily through the use of Perlin noise and Perlin worms. Perlin noise, pictured across the top right, is a gradient noise which is often used in procedural design. It was developed in 1983 by Ken Perlin in order to improve aspects of repetitive computer graphics. Perlin noise appears to be somewhat random, but at the same time, all of the data (gradient values) are at the same unit scale, and thus can be applied evenly. In Minecraft, this method is used as a terrain generator in conjunction with Perlin worms and a variety of other generation methods focused at specific elements. Perlin worms are used to create cave systems in the worlds and can interconnect providing players with dense cave networks.

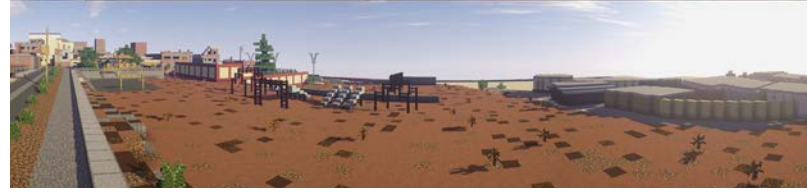
Minecraft also generates cliffs and overhangs, which do not rely on Perlin noise. Additionally, villages and other constructions meant for the player to explore are placed in the map based on a series of location based parameters. These elements will generate only if these parameters are met, and will vary in organization between cases.



Game Environment: Minecraft

Block by Block

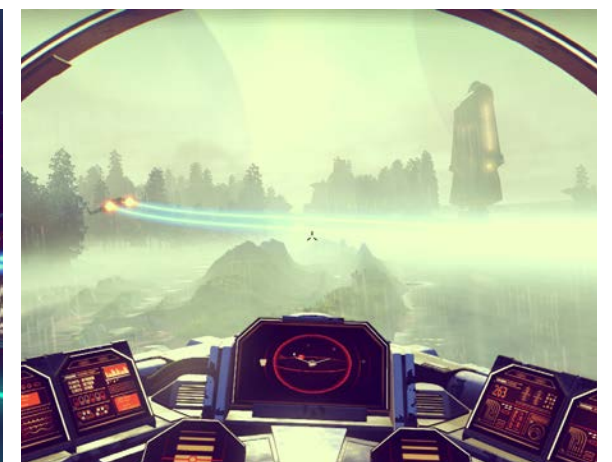
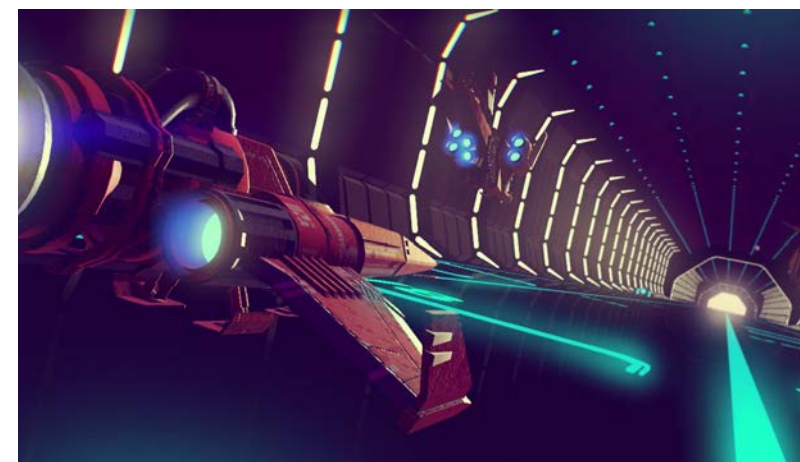
This project is a collaboration between Mojang, the parent company of Minecraft, and UN Habitat. The goal of this project is to get communities around the world to be more involved in local planning and development projects. In this project, Minecraft is being used as a interface between the reality of the site and the communities desires. Mojang builds each site in Minecraft on a 1-1 scale, modeling in similar textures and color palates. Then they work with UN Habitat to help residents realize their goals for the community projects. As a tool for expression, Minecraft's easy interface and concept allow for residents to shows more specifically what they envision. In the left column are real world images compared to their Minecraft equivalents. The realism is surprising based on the simplified graphics in the game. The center column shows members of the communities, both young and old, participating in the design project. And finall , the right column shows the redesigned park based on the residents input.



Game Environment: No Man's Sky

Background

This much anticipated game is currently in development by Hello Games and is set to be released in June 2016. What makes this an interesting model to study is that the entire environment is procedurally generated, meaning that it is created from a series of algorithms. Each planet, spaceship, and animal is created through a series of dependent systems of code, each one building off the previous and depending on variables produced. For example, if a planet is the right distance from a star it might have liquid water, if it has liquid water it might have rivers, and so on, down to the level of aesthetics of the animals and plants on the surface. What is interesting about these systems, is that for the mass customization of planetary, animal and spaceship forms, mathematical algorithms are used. Using the algorithmic method allows for all of the data for each planet to be held within the equation rather than a specific and data intensive 3D model. The game engine can simply build the model of the planet, animals and ships based on the x,y,z location within the universe. Additionally, this allows the universe to be made up of 16 quintillion accessible planets, which makes the game environment infinite to a human player.

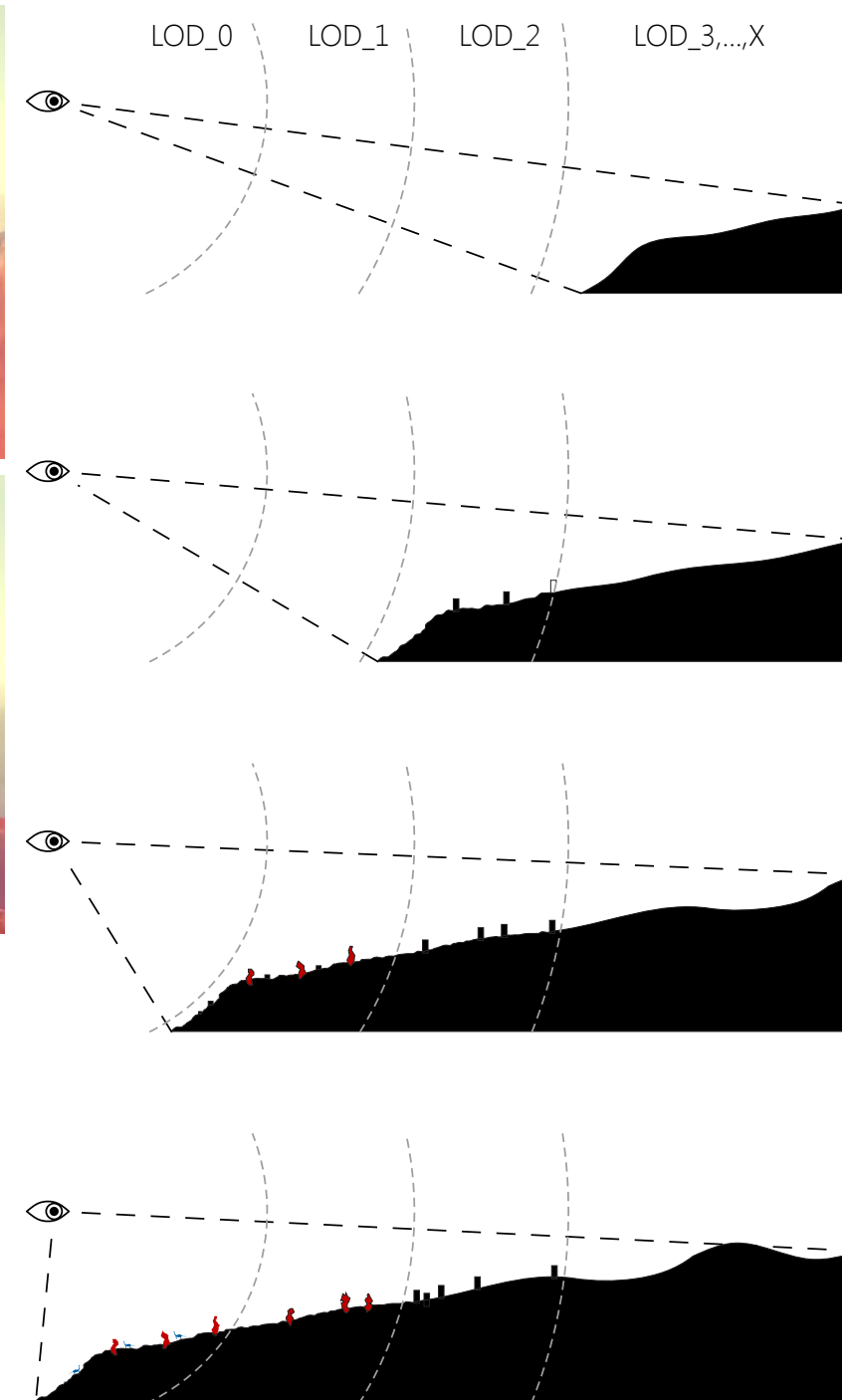
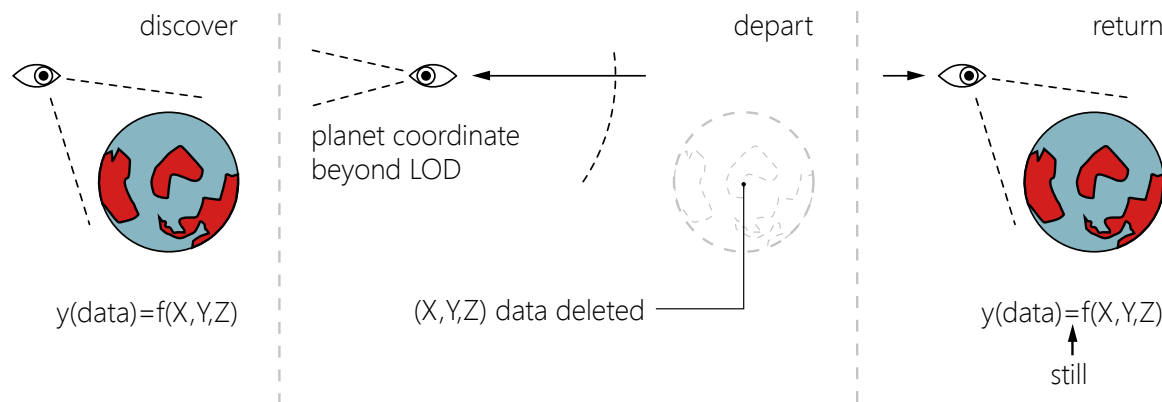
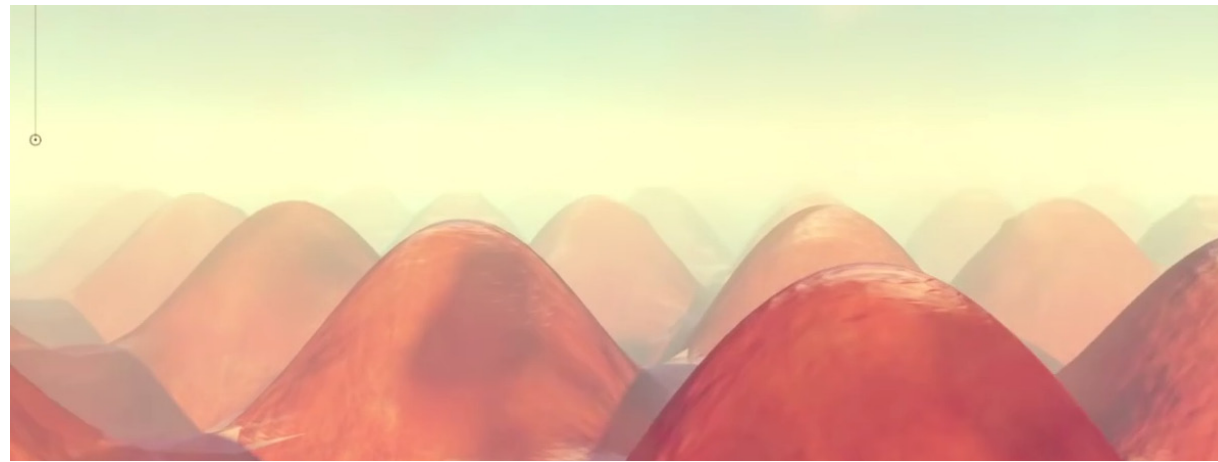


Game Environment: No Man's Sky

Terrain Generation

The terrain of planets is essentially a series of plotted and densely overlapped equations. These equations are designed by the creators of the game to begin with a single numeric input which will produce a unique but specific outcome. This way, at any given moment you can compile the data for a specific x, z coordinate and generate that specific portion of the universe. The static equations allow for the world to be generated exactly the same way each time, for each planet to be unique in the universe. This method is also data efficient, so it loads rapidly and doesn't require much disc space.

Developer Sean Murray explained that in order to debug and look for graphical errors, he has a series of simple equations that he plugs in for form planets. These equations generate very simple geography and make it easy to look for graphical errors. Pictured (top right) is a planet generated using the equation for a sine wave, $y = \sin(x)$. This planet has sine wave form mountains extending over the entirety of its surface. So essentially, the way that the geography is generated for any more complex planet is by using more complex and chaotic equations, as well as overlapping them to create more densely articulated geometry. Then additional detail elements (which are also procedurally generated) are added to flush out the believability of the planet.

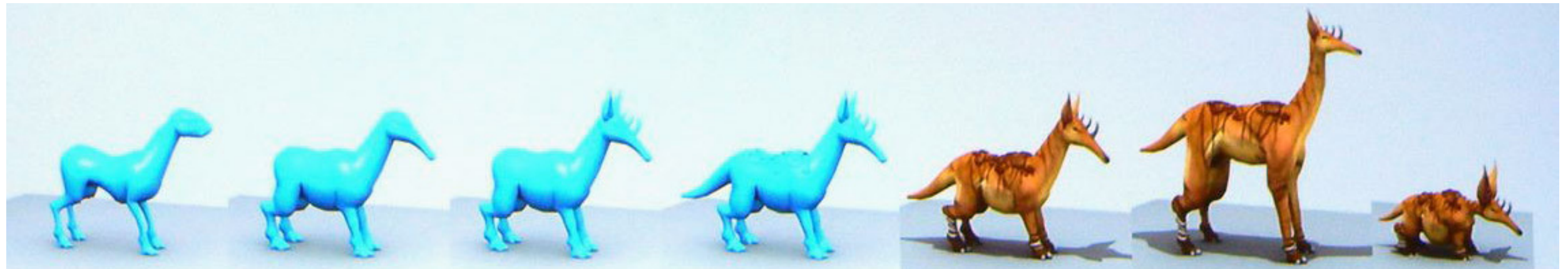
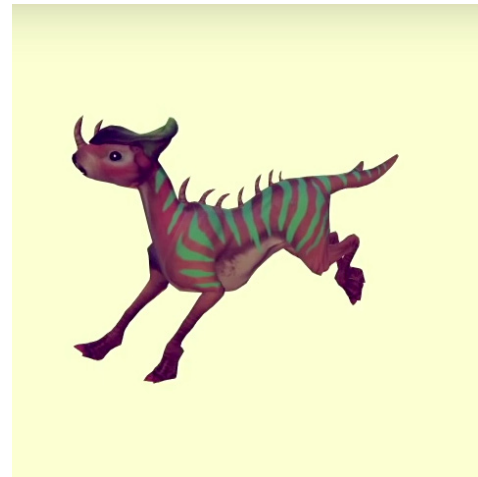
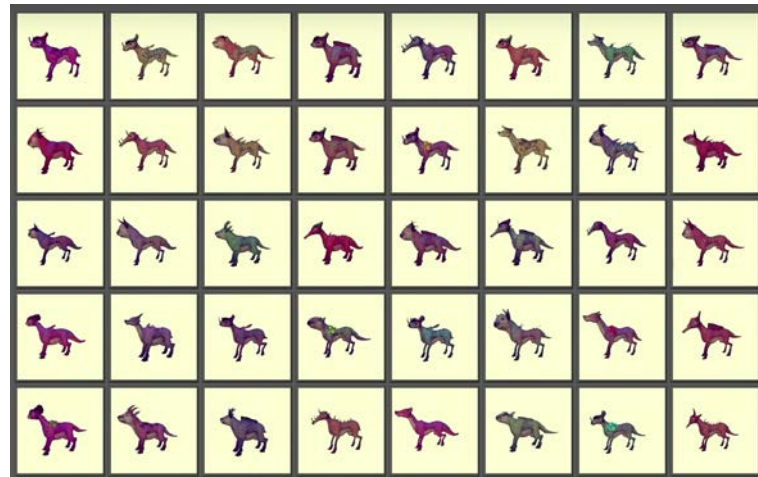
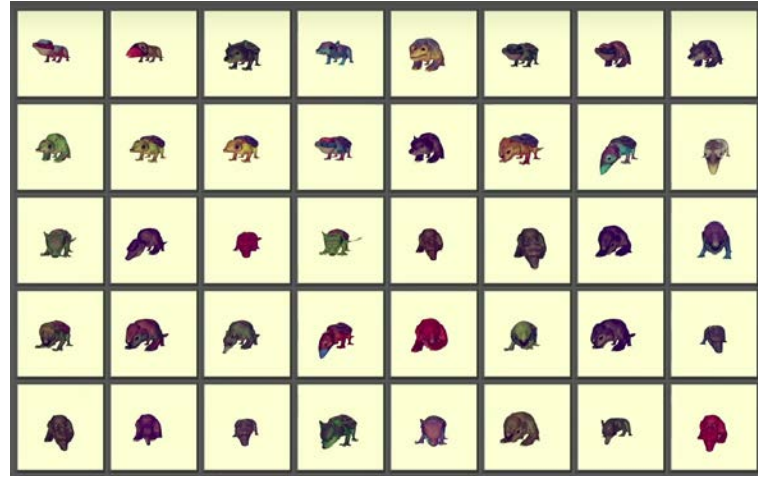


LOD0	Visible	Num Terrain Verts: 0 (0 / 32768)
		Num Water Verts: 0 (0 / 10000)
		Num Quad Verts: 0 (0 / 8000)
		Num Line Verts: 0 (0 / 64000)
		Num Line Indices: 0 (0 / 72000)
LOD1	Visible	Num Terrain Verts: 328476
		Num Water Verts: 0
LOD2	Visible	Num Terrain Verts: 468738
		Num Water Verts: 0

Game Environment: No Man's Sky

Life Generation

The method of generation for animals and plants is different than that of planets, but uses a similar methodology. The proportions and variability of form and color are changeable on sliding numeric scales. By changing the numbers for each animal or plant, the aesthetics change. Utilizing similar equation-based dependent systems allows the developers to input a single number and output the same animal form every time. This can be done with a near infinite amount of numbers, creating a huge amount of variation within the environment. The output from various equations within the generation system is dependent on the generated conditions of the planet the animal or plant is on, such as atmosphere, temperature, terrain etc. Space ships are generated in a similar way, (but without the dependence on the planet conditions) providing each player with their own unique spacecraft.



Game Environment: Comparison

	RUST	No Man's Sky	Minecraft
User Generation	All structures are generated by the user, and are made up of a limited set of modular components	No structures generated by user	All structures generated by user, large number of modular blocks to utilize in creation
Procedural Generation	Not used	Everything, planets, mountains, trees, rocks, animals, plants and spaceships	Major terrain and tunnels are generated
Goal	Survive and build	Unknown, explore?	Survive and build or just build
Community	Have to play together, groups/factions have formed in many servers	Interacting is statistically unlikely	Can play alone or on a server online, online is mostly creative

Case Studies |

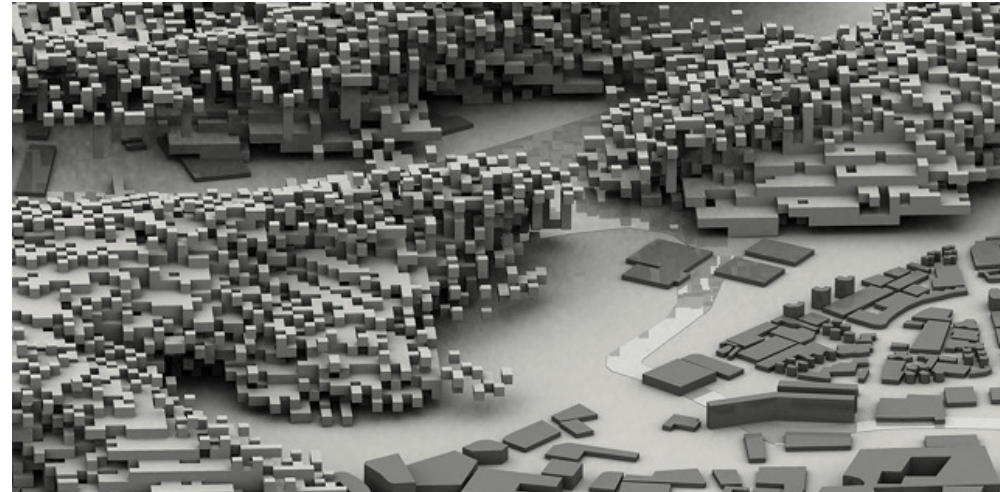
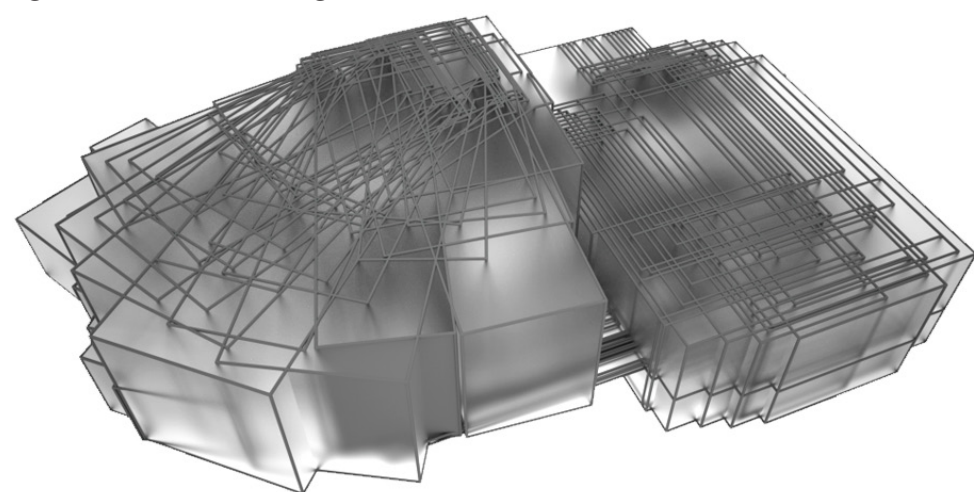
Settlement



Game Environment



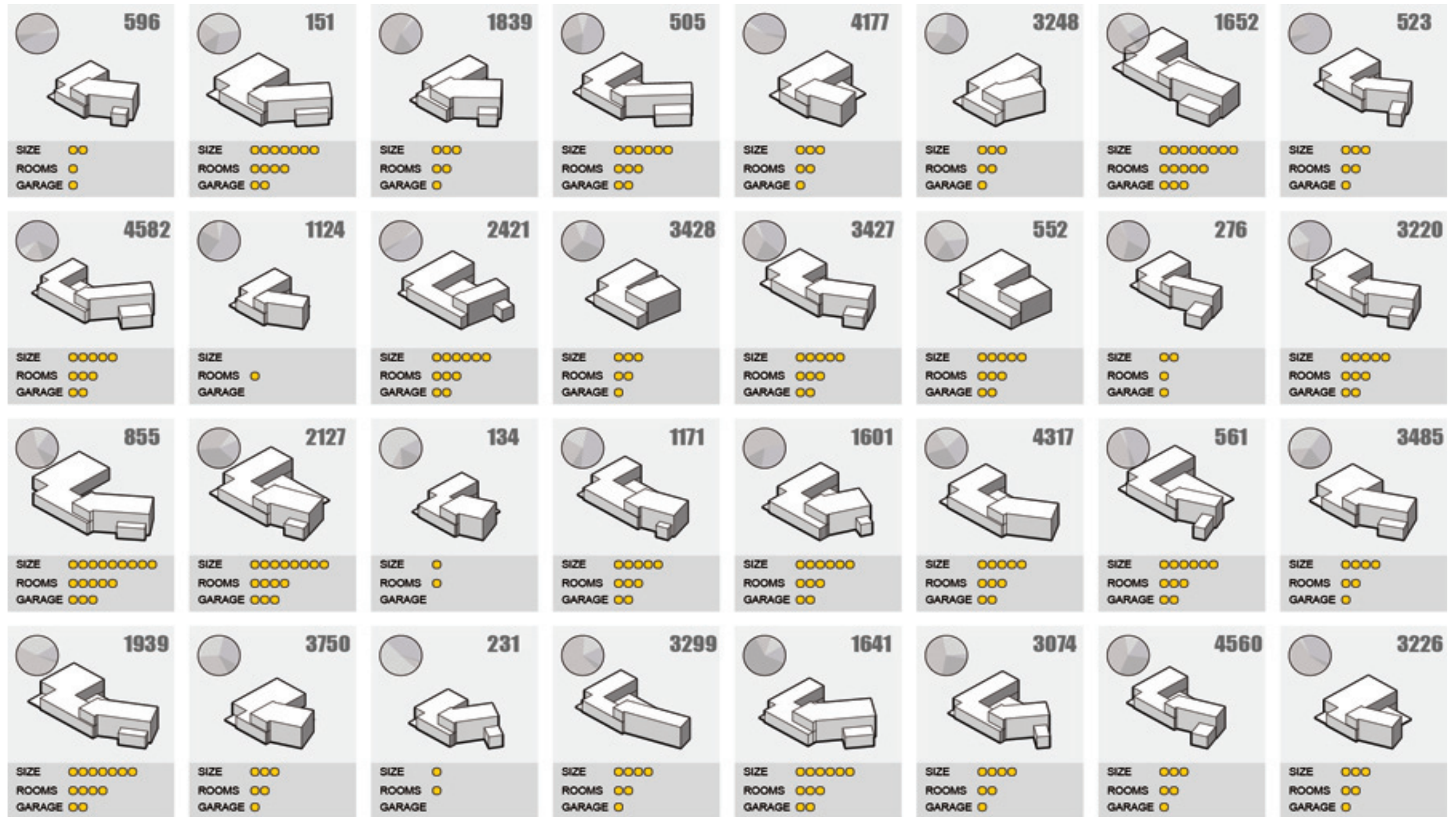
Algorithmic Modeling



Algorithmic Modeling: HAS

Mass Customization

This is a generative design tool developed by the Architecture Research Lab (Alexandra and Michael Bergin) to produce mass customization of single family homes. They argue that according to the AIA when it comes to housing design only 28% have a licensed design professional involved. That leaves 72% to be designed without any major changes to the original drawings by the developer. The mass production of these forms leads to standardized communities without thinking about the implications on the client and above all eliminates the design process. The ARL is building off of Autodesk research into what they call a Housing Agency System. The Housing Agency System creates iterations from a specific input form, using a series of algorithms tied into site, planning and formal strategies, construction systems, detail systems, and building components. These algorithms generate custom designs based off of user input on what specifically they want. They can view a large number of output solutions (right) and pick one which they like.

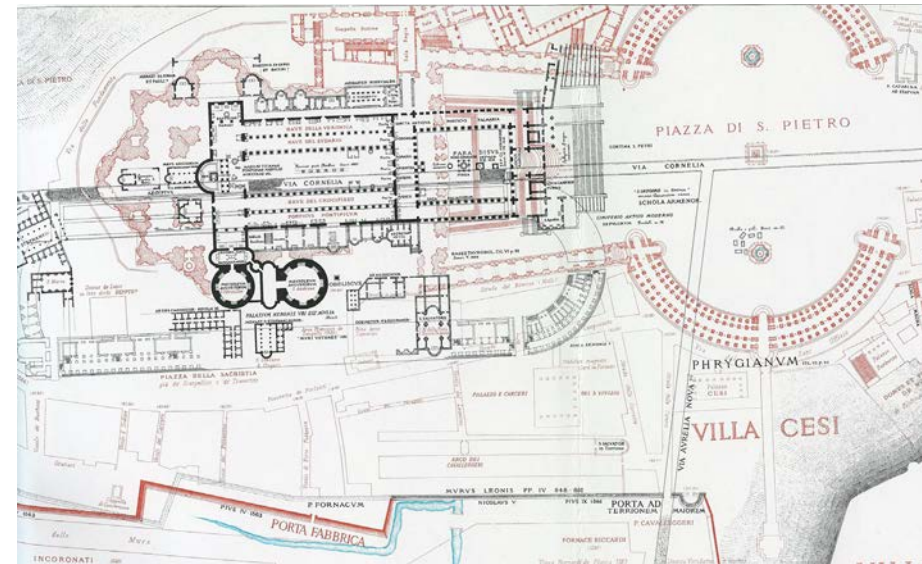
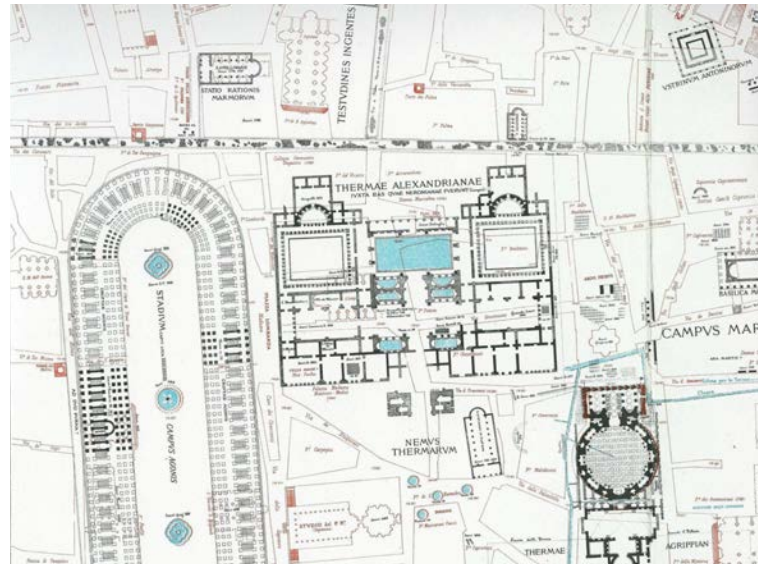
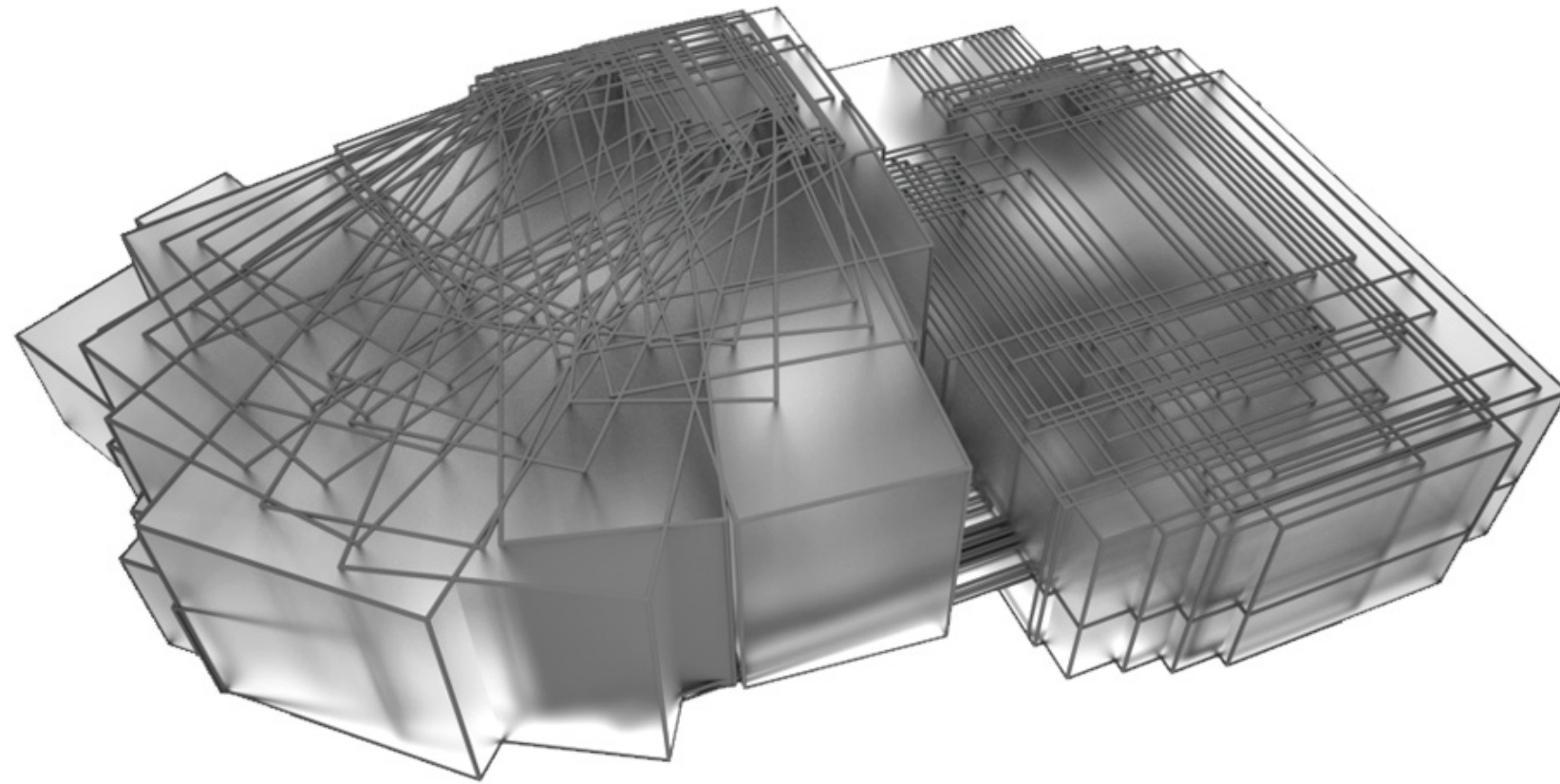


Algorithmic Modeling: HAS

Superimposing Form

ARL created a model which superimposes various configurations of the basic housing unit. What is interesting about this is that it is simultaneously showing 72 different possibilities for the basic model. It provides an overview of all of the spatial arrangements which have been analyzed for this model. The overall form does not change much in either volume or form, but provides numerous options for a person considering the model.

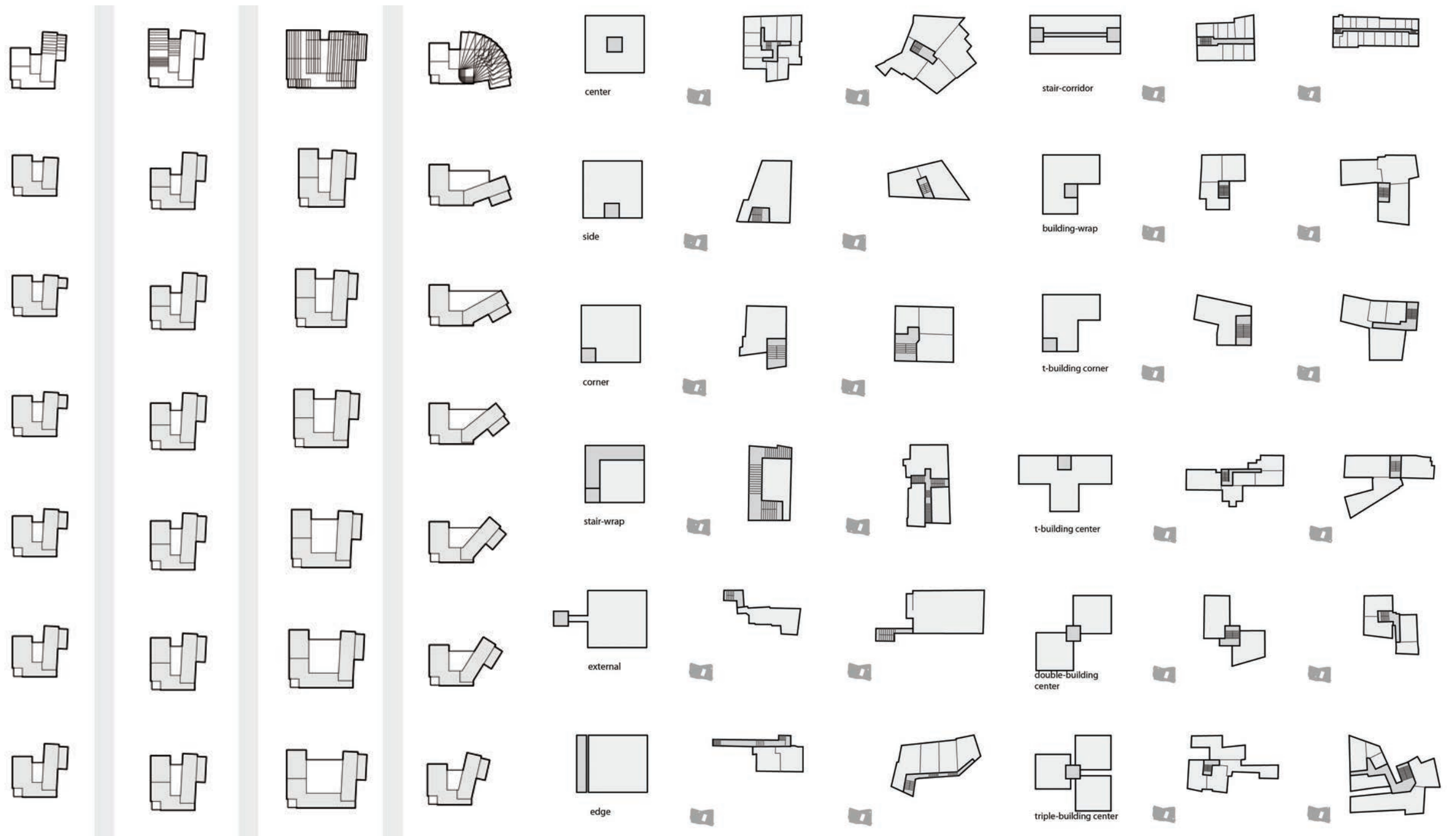
The spatial effect of this overlap below is similar aesthetically to the forms of informal additions in settlements. The image to the right shows the Mississippi river over a period of 300 years. The superimposition of the path of the river allows the viewer to better understand the nature and impact of the river on the surrounding land. Mapping the changes to an informal settlement over time, using a method which looks graphically similar to ARL's could provide insight into the nature of settlement growth. Additionally, it could be used in a method similar to ARL's to provide possible iterations of settlement growth based off of a hypothetical formal origin.



Algorithmic Modeling: HAS

Iteration

The variations produced from the Housing Agency System start with a single base and operate on one aspect of the form at a time. When looking at Kowloon walled city, which was a high density informal settlement built outside Hong Kong, there are identifiable typologies of housing. These typologies are variable due to the constraints of the site, but have relatable origins. In the book *City of Darkness*, the author, Ian Lambot, categorized different apartments based on the relation of the stairwell to the building. These stair/apartment typologies differ based on the construction methods, and space available and even shift from floor to floor. When compared to the Housing Agency System variations produced, there is a clear relation between the formal operations. The possible versions of informal operations in Kowloon walled city could be modeled through a generative design program like the Housing Agency System. This could provide models of possible forms on a site or model potential for growth on an existing building. As a tool for development within existing informal settlements, it could be an efficient way to look at specific sites and see what possible informal additions might be made in the future.

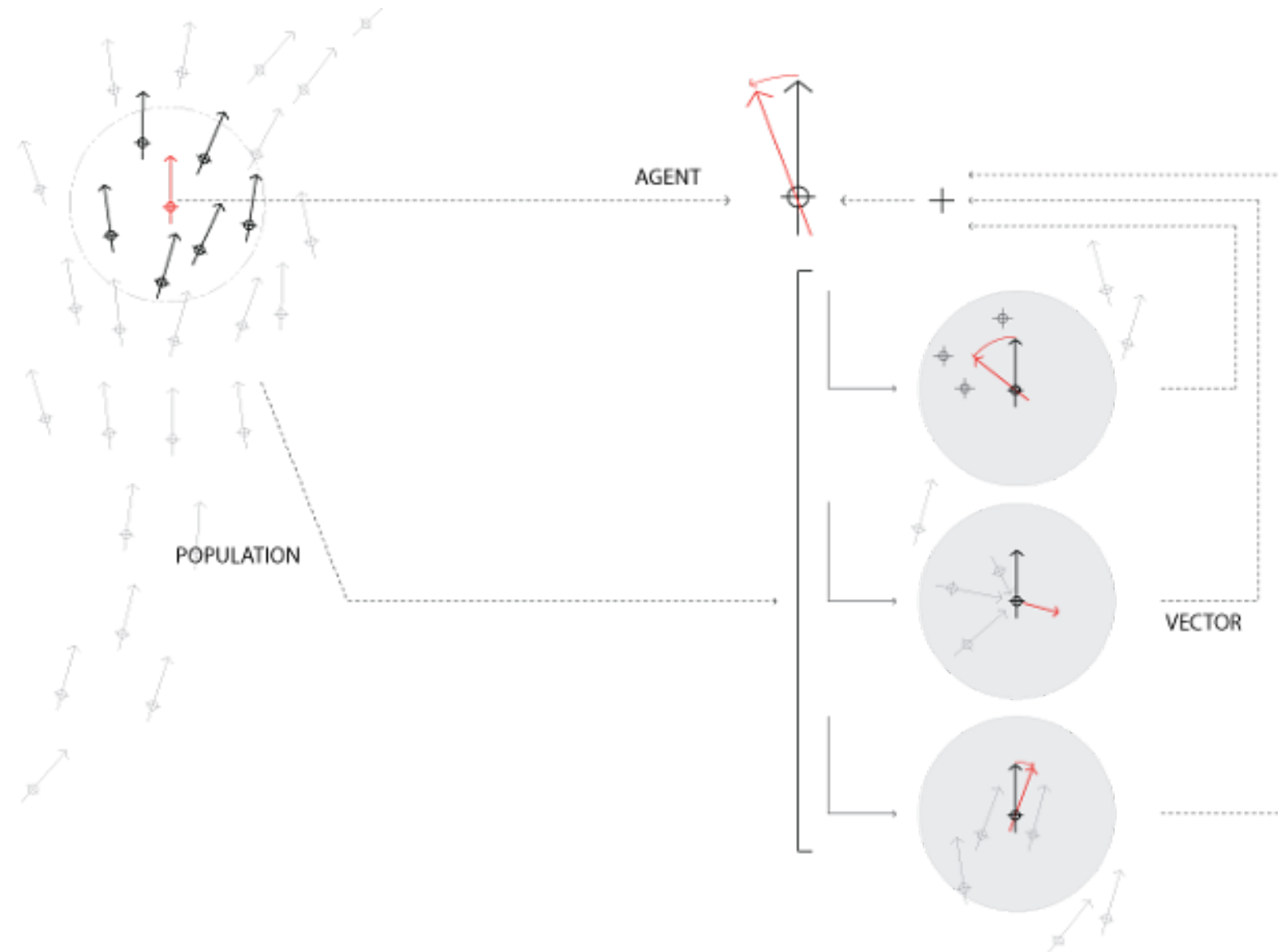


Algorithmic Modeling: Kokkugia

Background

The method deployed by Kokkugia in the majority of their projects is to generate form and organization through the use of autonomous agents. There is focus on decentralizing the decision making process allowing for an emergence of form through small actions. Their agents have specific and limited interactions with one another, but have similar goals. There is an insect-like quality to their methods and even the digital images created. Two projects of theirs that are relevant to this thesis are, the Swarm Urbanism, and Behavioral Urbanism projects.

In Swarm Urbanism, there are two autonomous agent systems working together. One agent set is deployed first and Kokkugia describes it as operating similarly to a termite or ant. It creates pathways and self organizes through stigmergic growth. The second set builds off this scaffold created by the first, but builds into more minimal and optimized structures. The overall goal of the program is to develop ideas about how to create 'self-organizing urban structures.'

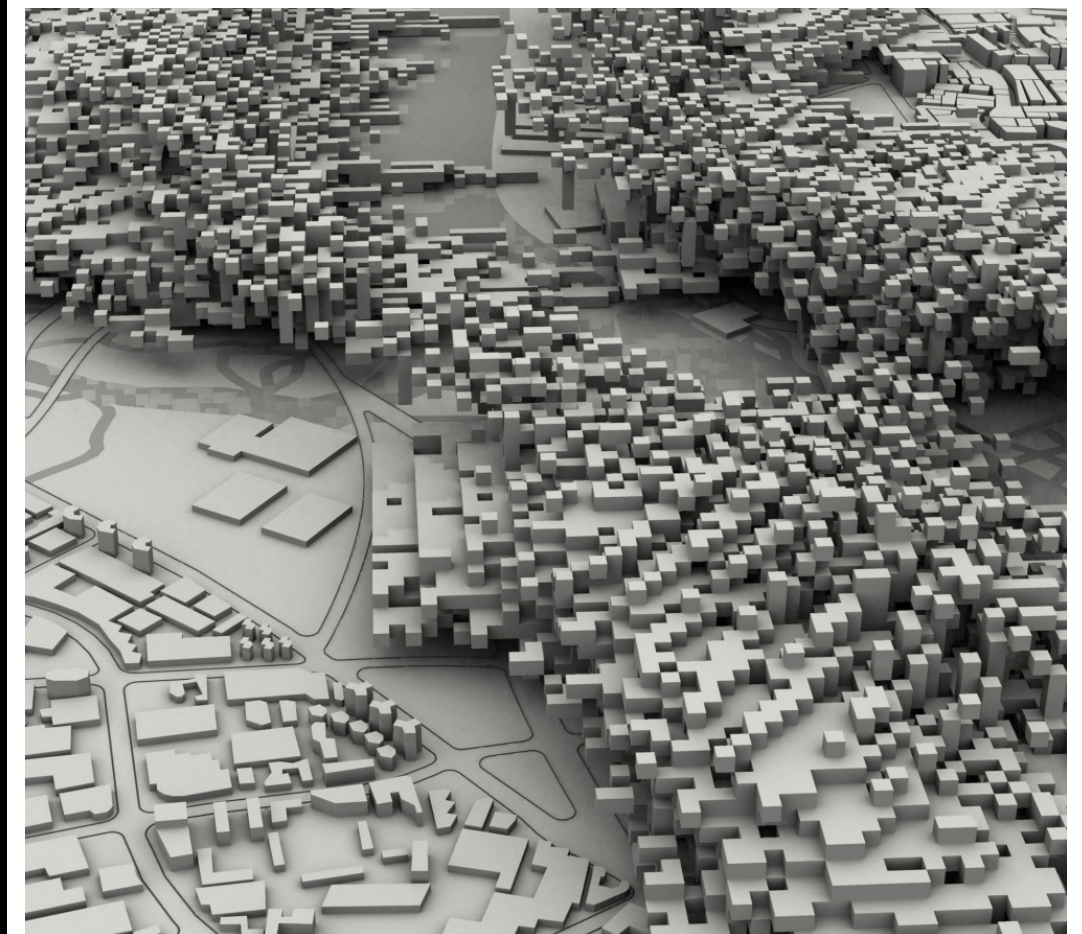
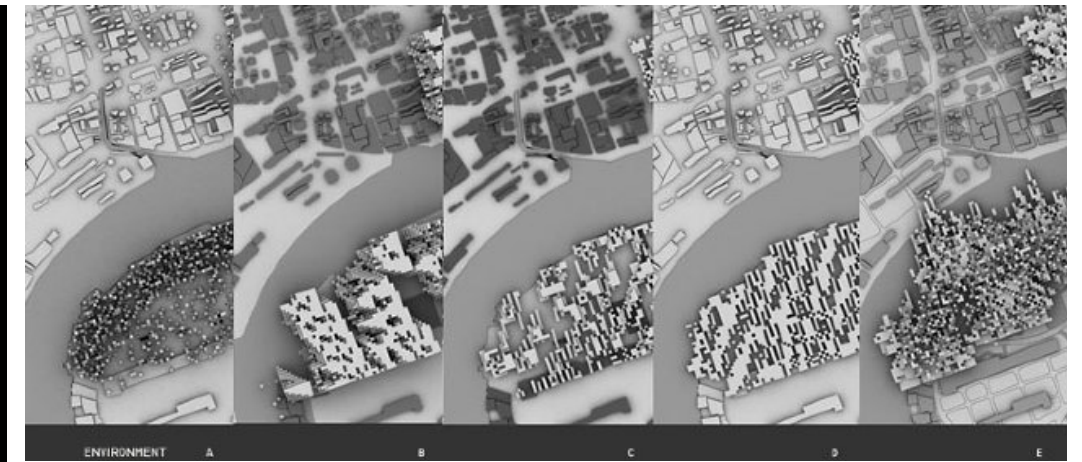
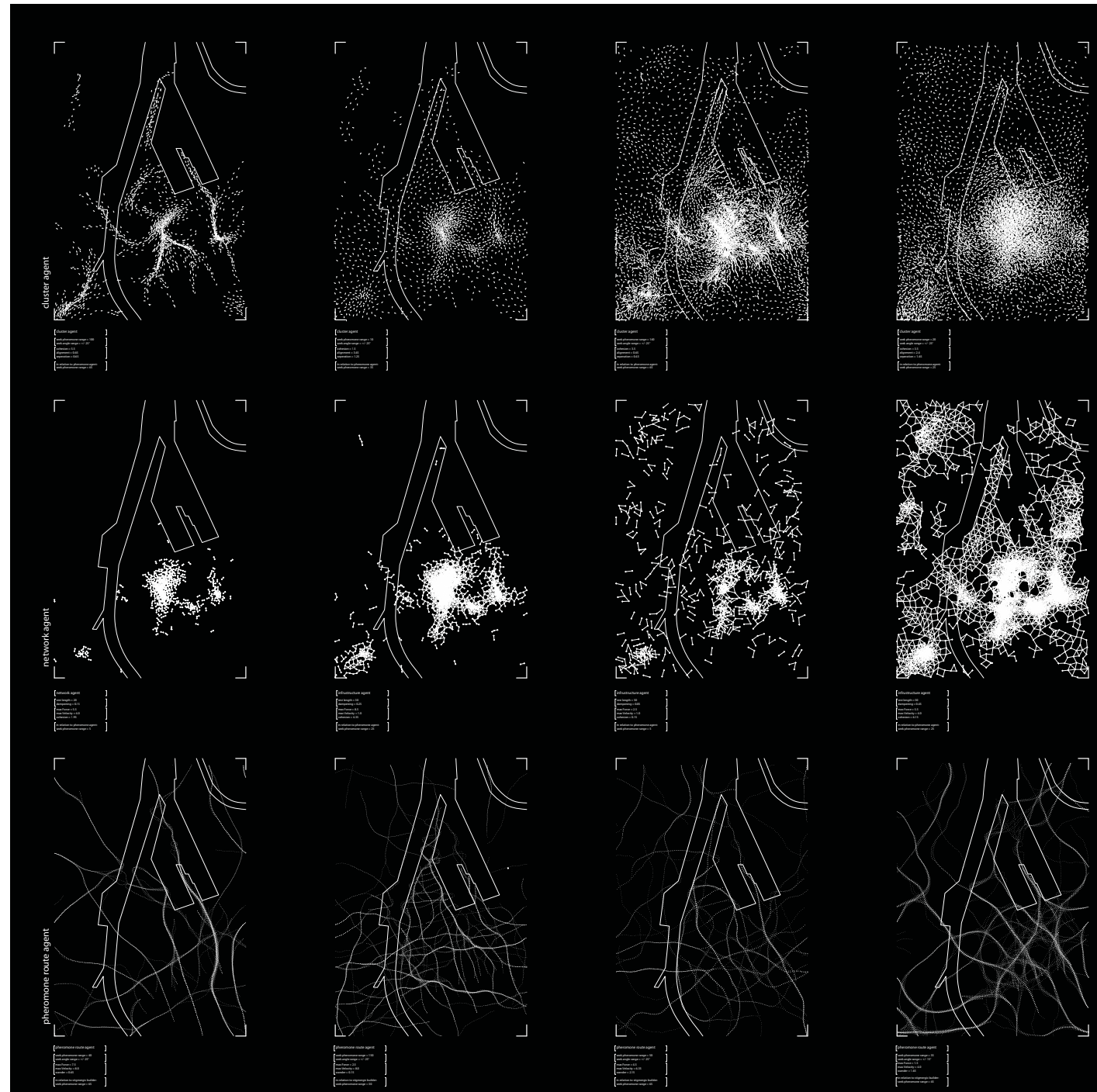


Algorithmic Modeling: Kokkulugia

Swarm and Behavioral

The image on the left is of the Swarm Urbanism project, where it has been superimposed on an urban site in order to attempt to generate infrastructure pathways. The product of this generation seems more chaotic, but produces a compelling set of paths and aggregations. The swarms however seem largely ignorant of the context, and seem to start from a planned point and spill out haphazardly.

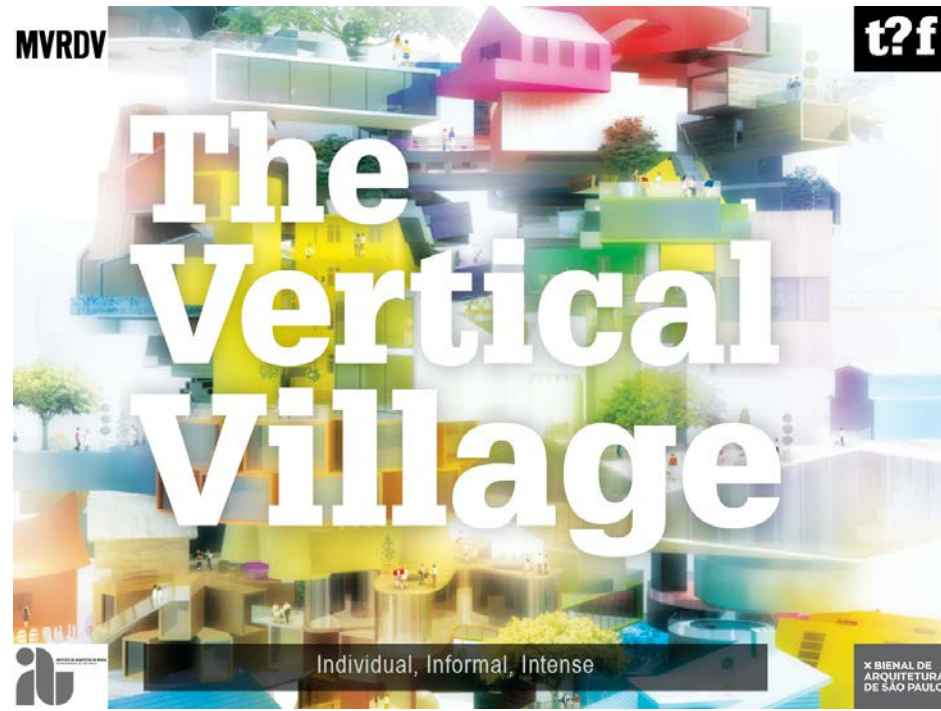
The second image is of the Behavioral Urbanism project. In this project the goal was to use ideas about cellular automata in order to produce aggregated mega structures within the city. There is no program associated with these large scale interventions, but housing seems like a likely scenario. There are different iterations of aggregation based on the various parameters which determine the agent's response. These mega forms also seem to have little to do with their context apart from avoiding the water and site edges.



Algorithmic Modeling: Vertical Village

Background

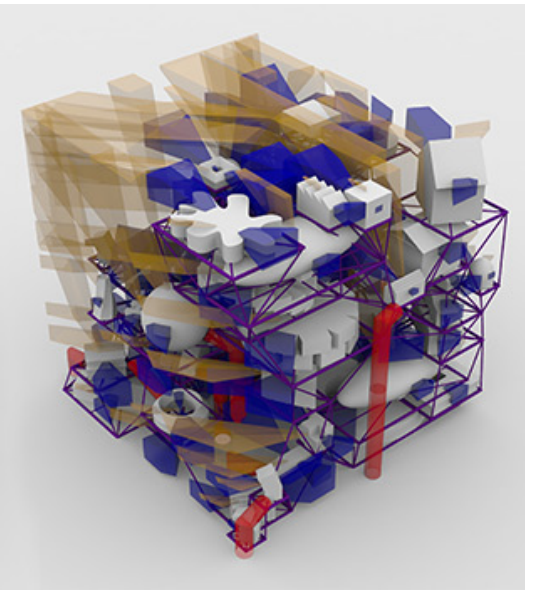
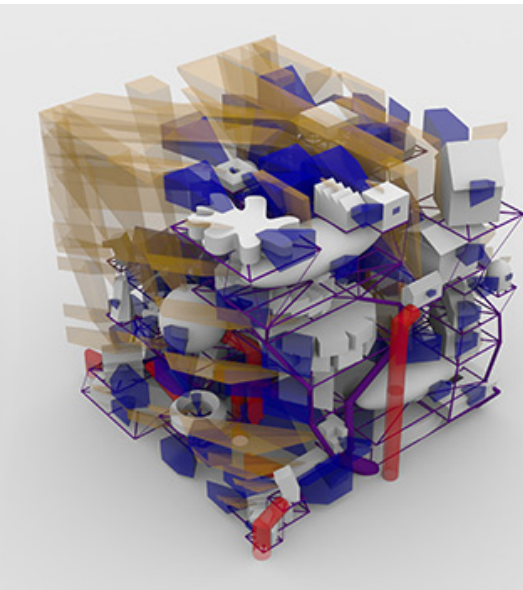
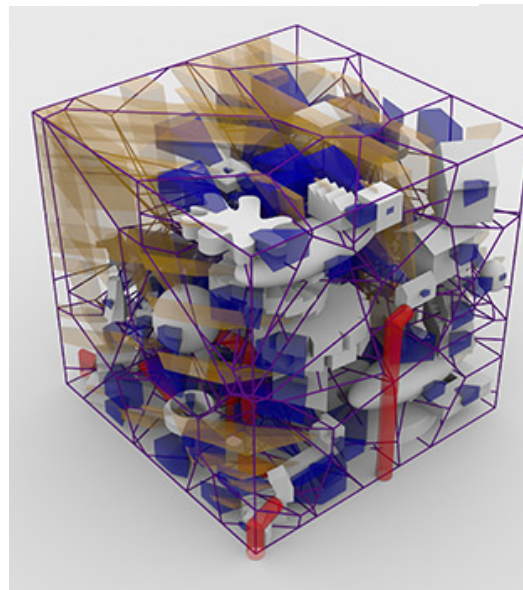
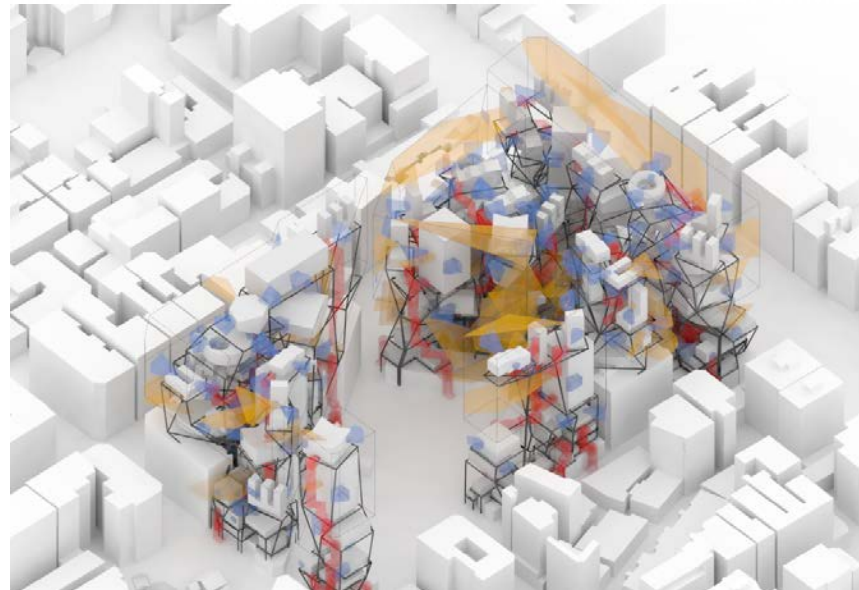
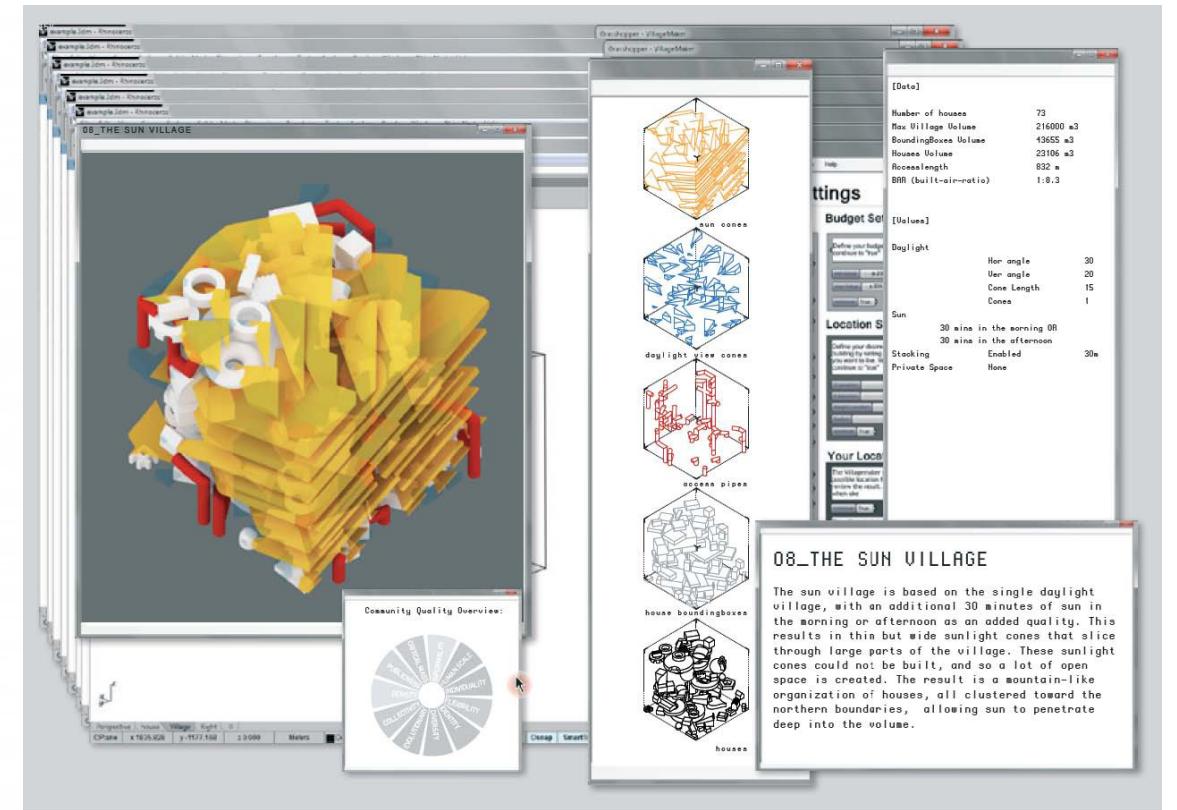
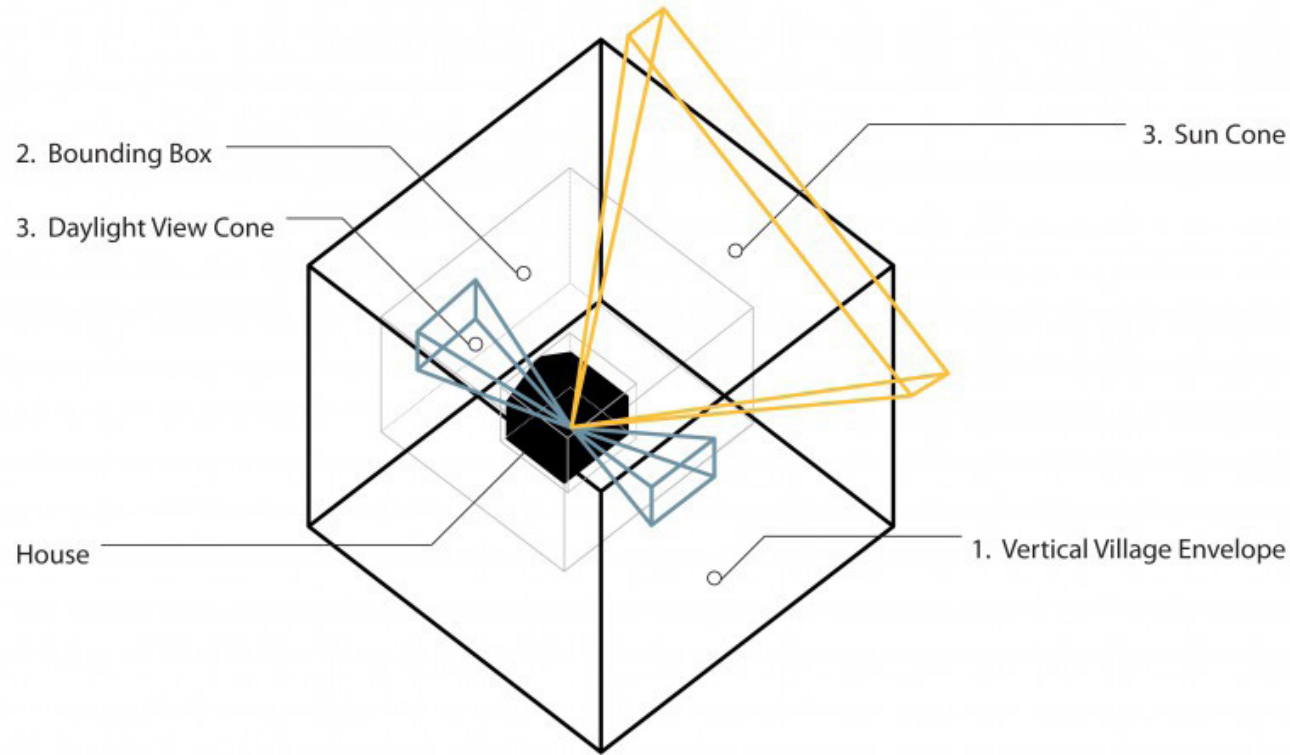
This project grew out of a semester long design studio led by MVRDV about the urbanization of different cities in Asia. What the studio noticed, was that there was a severe lack of affordable housing, with many people waiting years to buy a house. They analyzed the development of smaller scale pockets within the city and analyzed the qualities they maintained. The main argument being that the village mentality of interaction, smaller scale, local interest, community among other things, created an environment that would be positive to emulate or retain when building denser communities in emerging cities. The models produced are their first analogue attempts to produce a vertical community with village like elements. Teams traded models and built in different colors with different goals, all relating back to the qualities of the urban village. Later, they produced models like the one on the bottom left, which consider the possibility of each individual's home having a unique form. The idea stems from mass customizing of housing, and to understand it they built a Grasshopper script called 'House Maker' which allowed users to customize their own house models.



Algorithmic Modeling: Vertical Village

House Parameters

Building off the ideas that we e started in 'House Maker' the studio created another script called 'Village Maker' for aggregating the customized houses. This program took into account specific user defined conditions, such fo m, windows, space between houses and amount of sun. Using these parameters, the 'Village Maker' would generate an aggregated form based on the houses input. It considered a variety of conditions and modeled them in various colors to visualize them. Vertical circulation was a critical issue, it can be seen highlighted in red in the images. The sun cone is in yellow, and the daylight cones are in blue. The houses themselves are in white and are wrapped in purple structure. Each of the parameters could be adjusted and could be modeled with diffe ent intents. Structure for instance could be modeled to be like a scaffold, a tree, and a box truss system. The ability to customize the model to the site as well as iterate through design schemes quickly makes this an interesting case study when considering informal settlements.

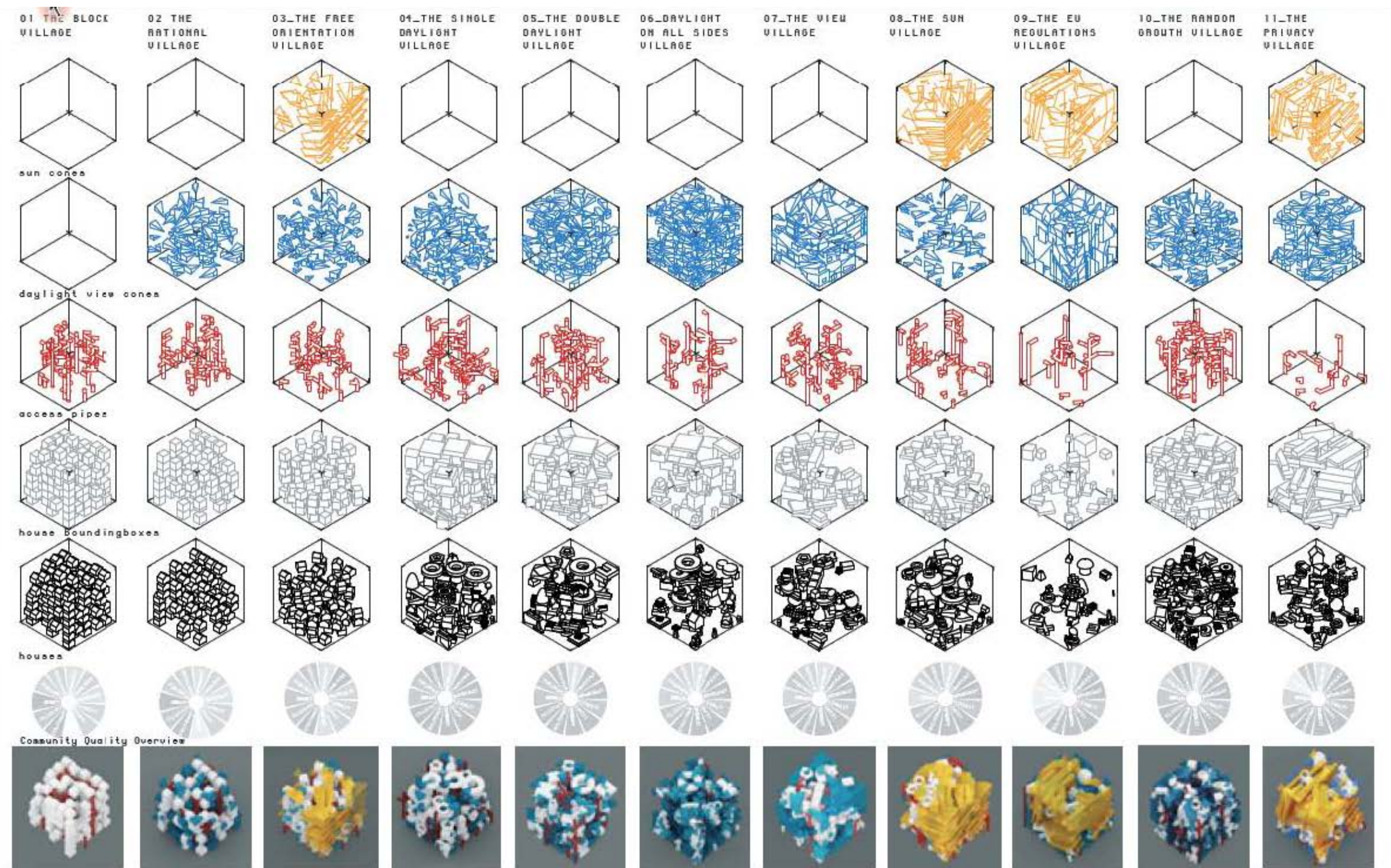


Algorithmic Modeling: Vertical Village

Iteration

This is an array of outputs from the 'Village Maker' and their different compositional elements extracted. In comparing these schemes it becomes apparent that certain input models of aggregation cause more complex systems of structure, access, view or housing organization. Unlike the Kokkugia project this is a tool which is taking a set of parameters (house designs) and iterating through them to produce an optimized result form. Within this system however, the structure and circulation elements act somewhat like autonomous agents in order to create pathways to iterate and check.

This project is designed to provide architects or consumers with a rapid way to iterate through project possibilities. As a concept for a tool it is very successful despite the fact these typologies of construction have inherent flaws. The spatial analysis capabilities within this program allow it to provide relatively specific architectural insight into specific form created by architects.



Algorithmic Modeling: Comparison

	HAS	Kokkugia	Vertical Village
Mass Customisation	Main purpose for the program is to provide models for mass customization of housing	Each iteration produced is dependent on the agents and has high chance of being different	Provides ability to mass customize home as well as automatically configure apartments block
Concept	Customization, providing people with unique homes	Utilizing agent systems to develop structural or architectural networks	Create new village/housing typology that competes with city density
Input	Volumetric building or house model	Site and initial control parameters for each agent	Specific numbers of and models for housing configuration
Output	Multiple variations of form with iterations	Sinuous web-like pathways and aggregated forms	Tested and configured housing forms based on specific and manipulatable parameters

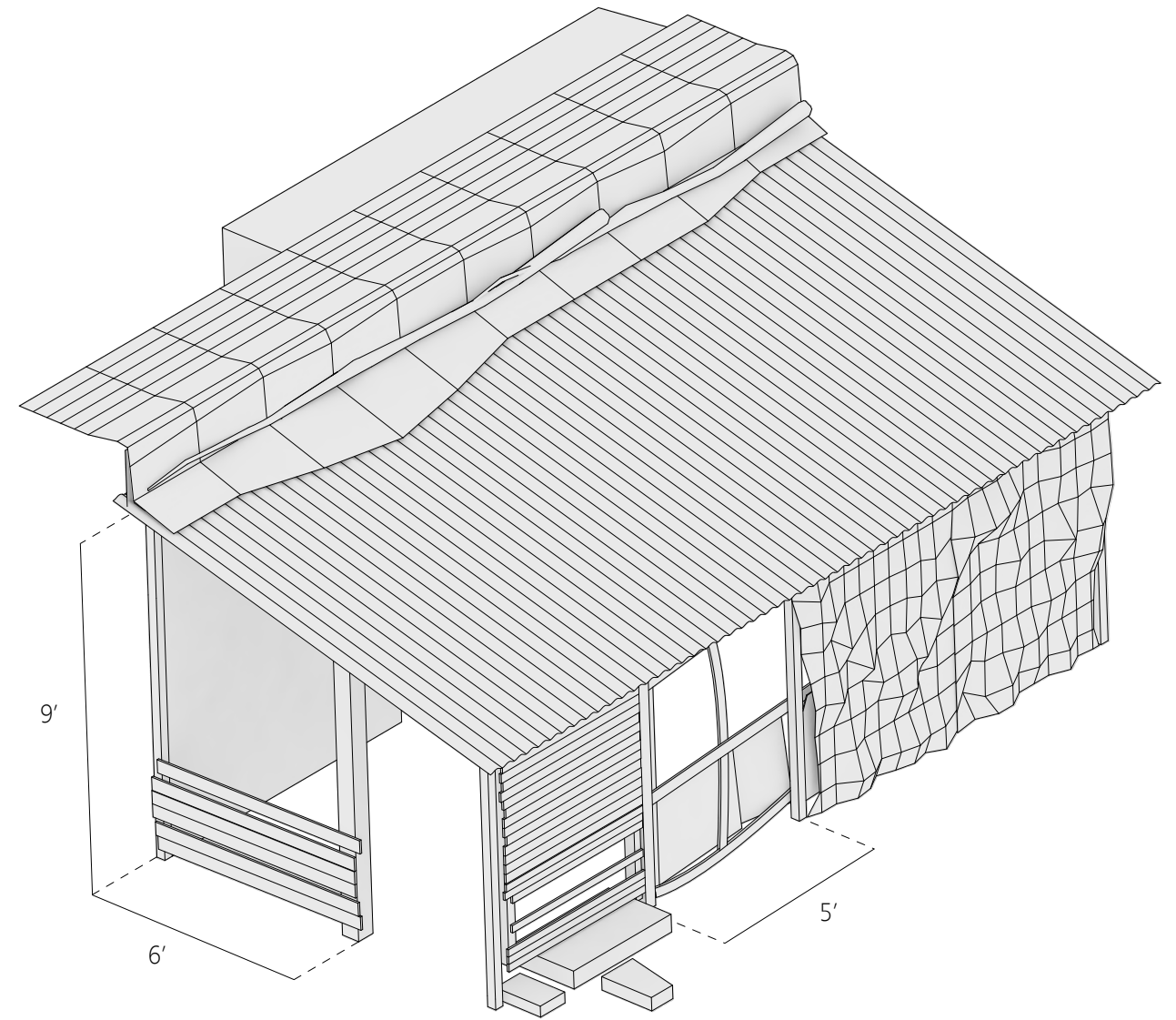
Hypothesis |

The result of a tool run over many iterations, modeling the aggregation of materials and volumes in an informal settlement as accurately as possible, will provide architects with a means of understanding potential growth patterns in informal settlements.

Inventory |

Catalogue a set of components with which to simulate settlements by modeling a series of existing informal buildings from various settlements. Understand methods of mass customization for each component and implementation.

Case Studies | 1 : Exterior Shed

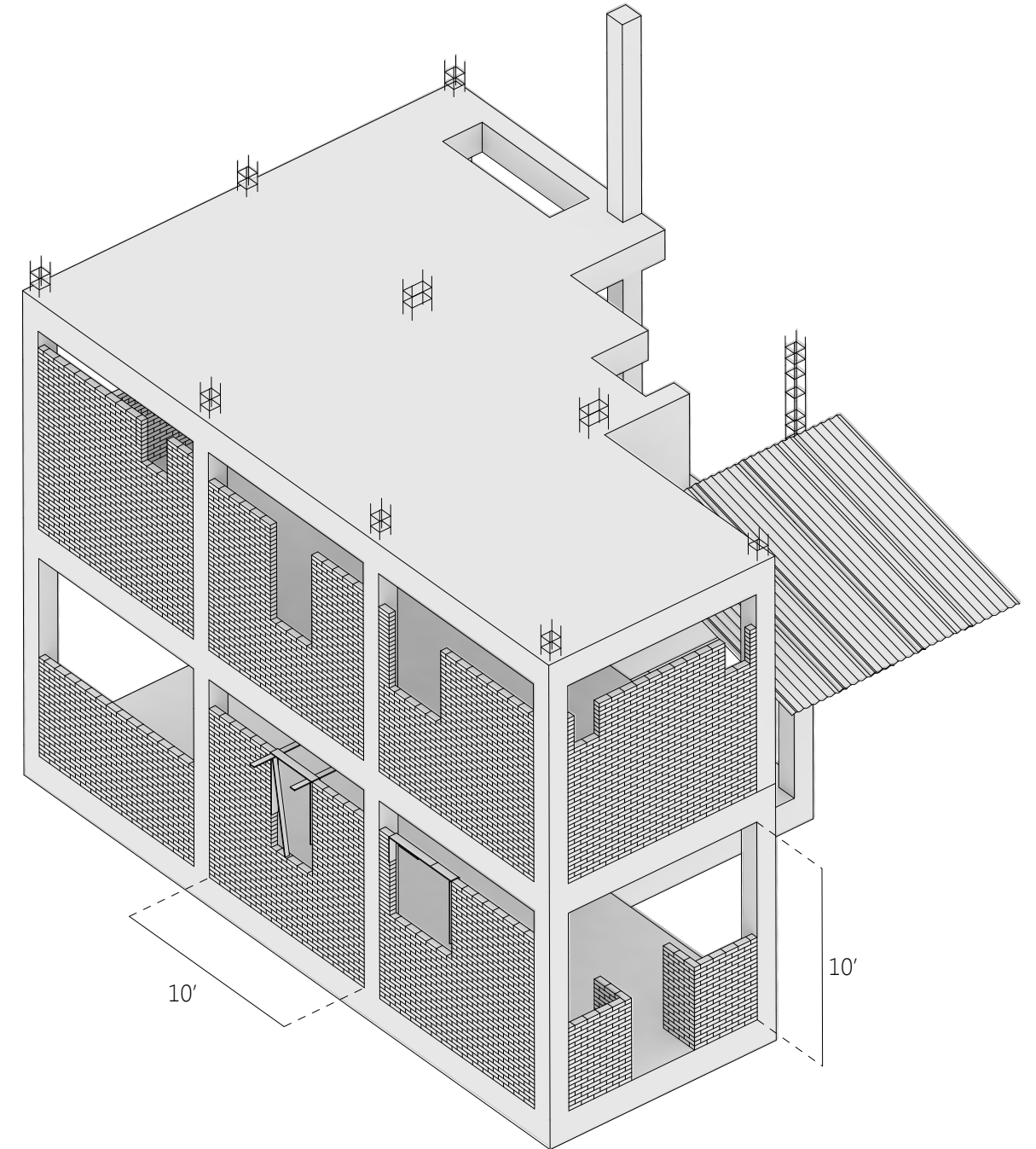


<http://mitei.mit.edu/news/mit-architects-tackle-indias-slum-problem>

Case Studies | 2 : Unfinished Housin

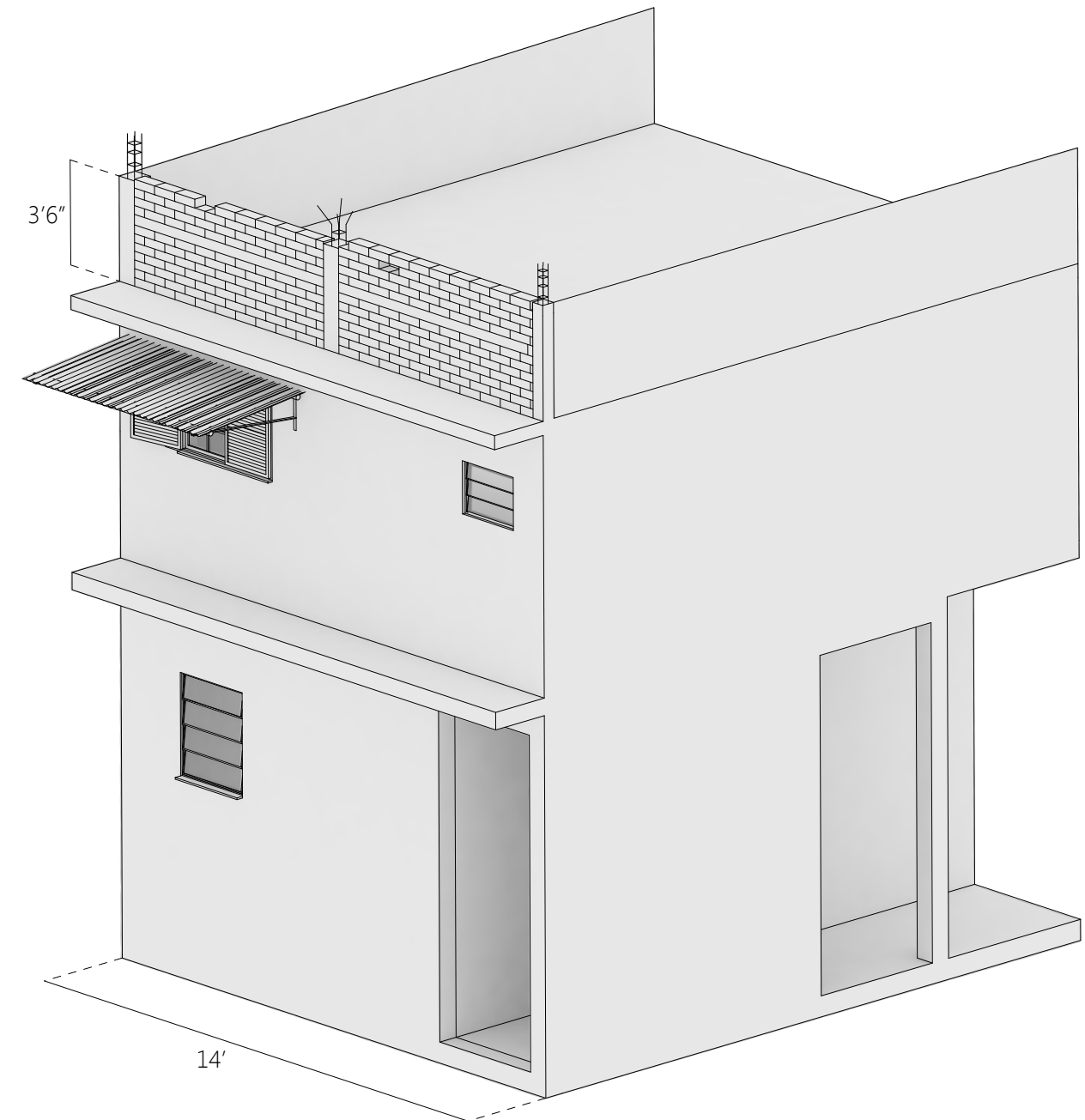


<http://www.designother90.org/solution/yerwada-slum-upgrade/>





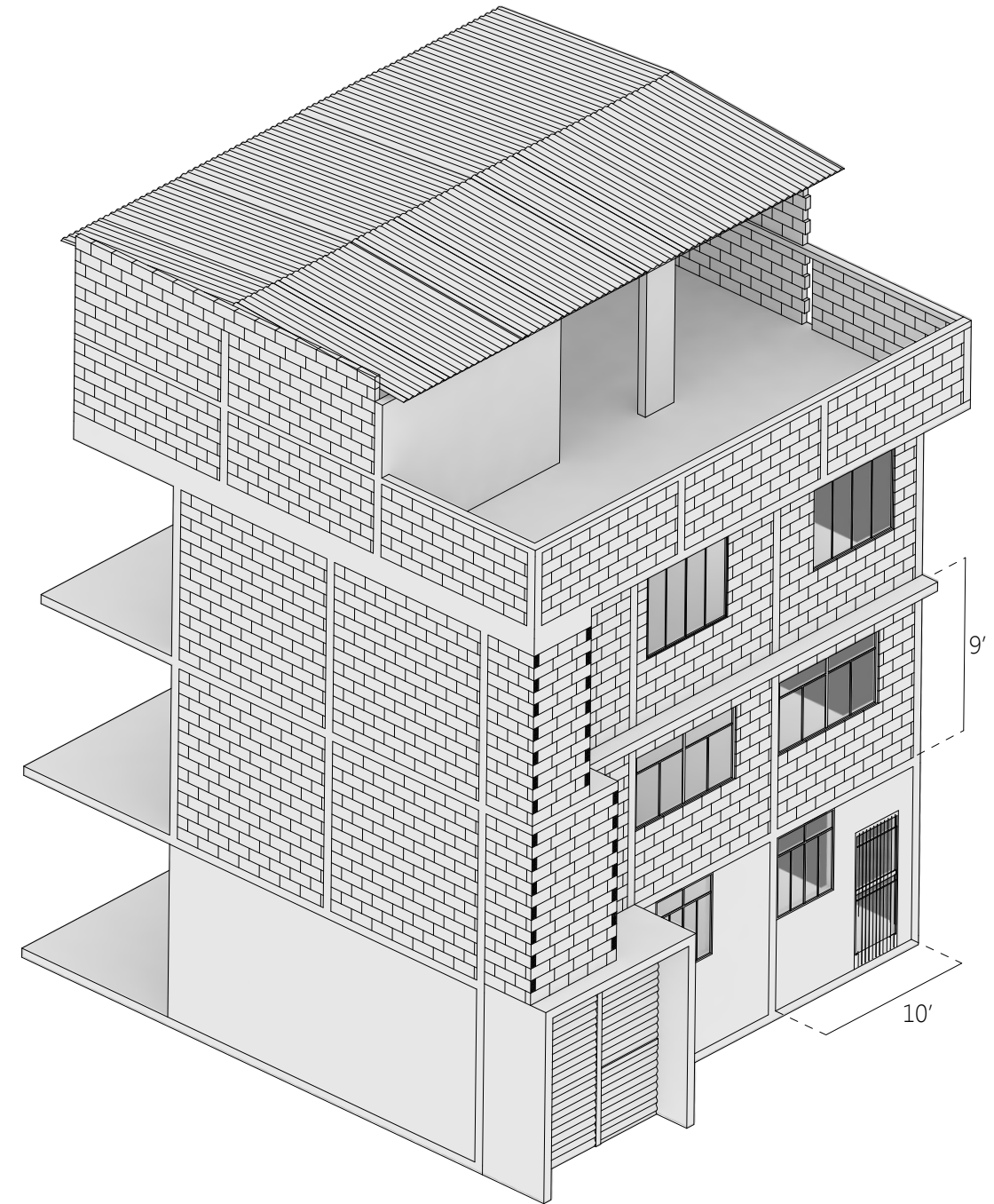
<http://www.archdaily.com/531253/case-study-the-unspoken-rules-of-favela-construction>



Case Studies | 4 : Workspace Housing



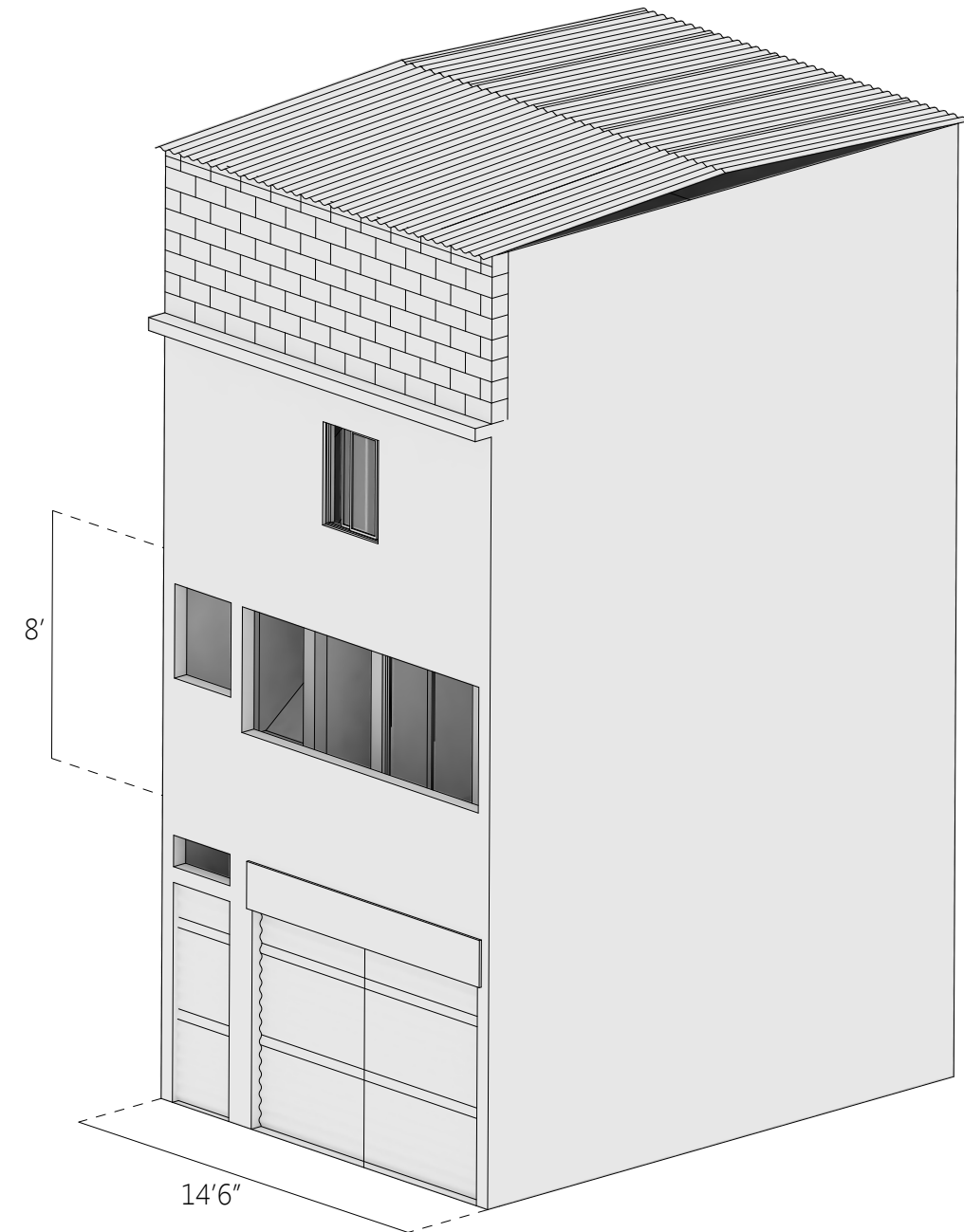
<http://www.archdaily.com/531253/case-study-the-unspoken-rules-of-favela-construction>



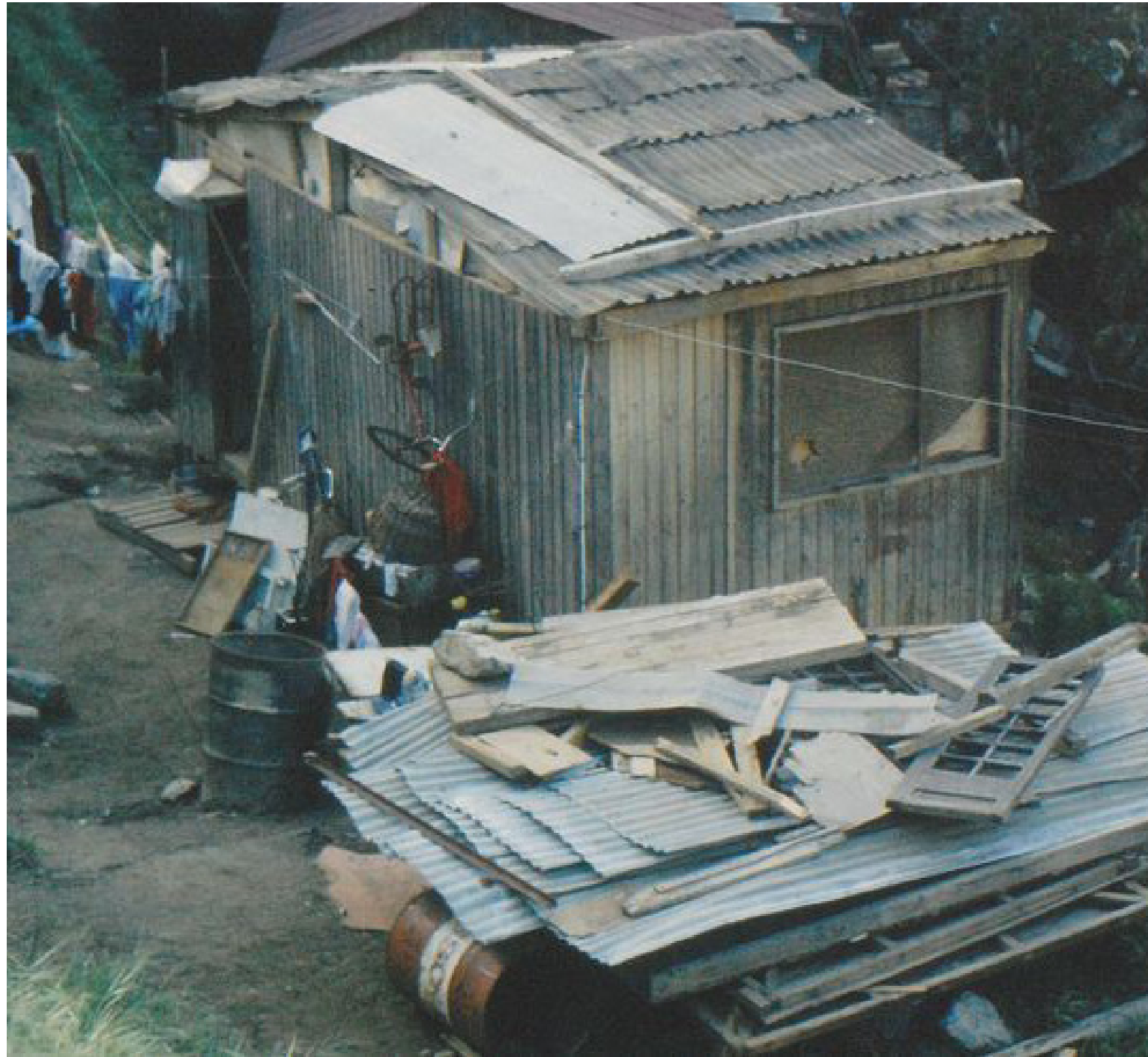
Case Studies | 5 : Storefront Housing



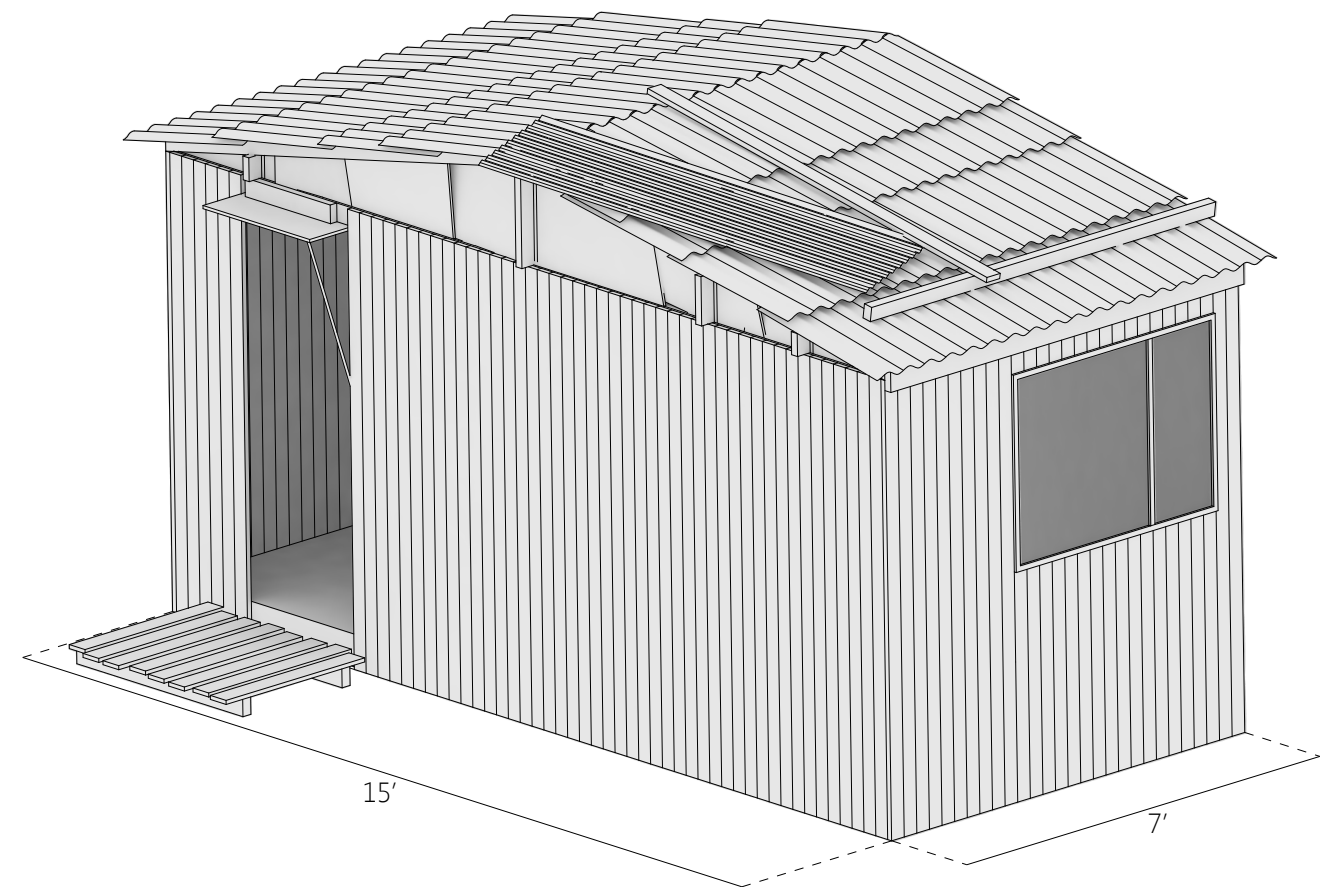
<http://www.archdaily.com/531253/case-study-the-unspoken-rules-of-favela-construction>



Case Studies | 6 : Wooden Shed



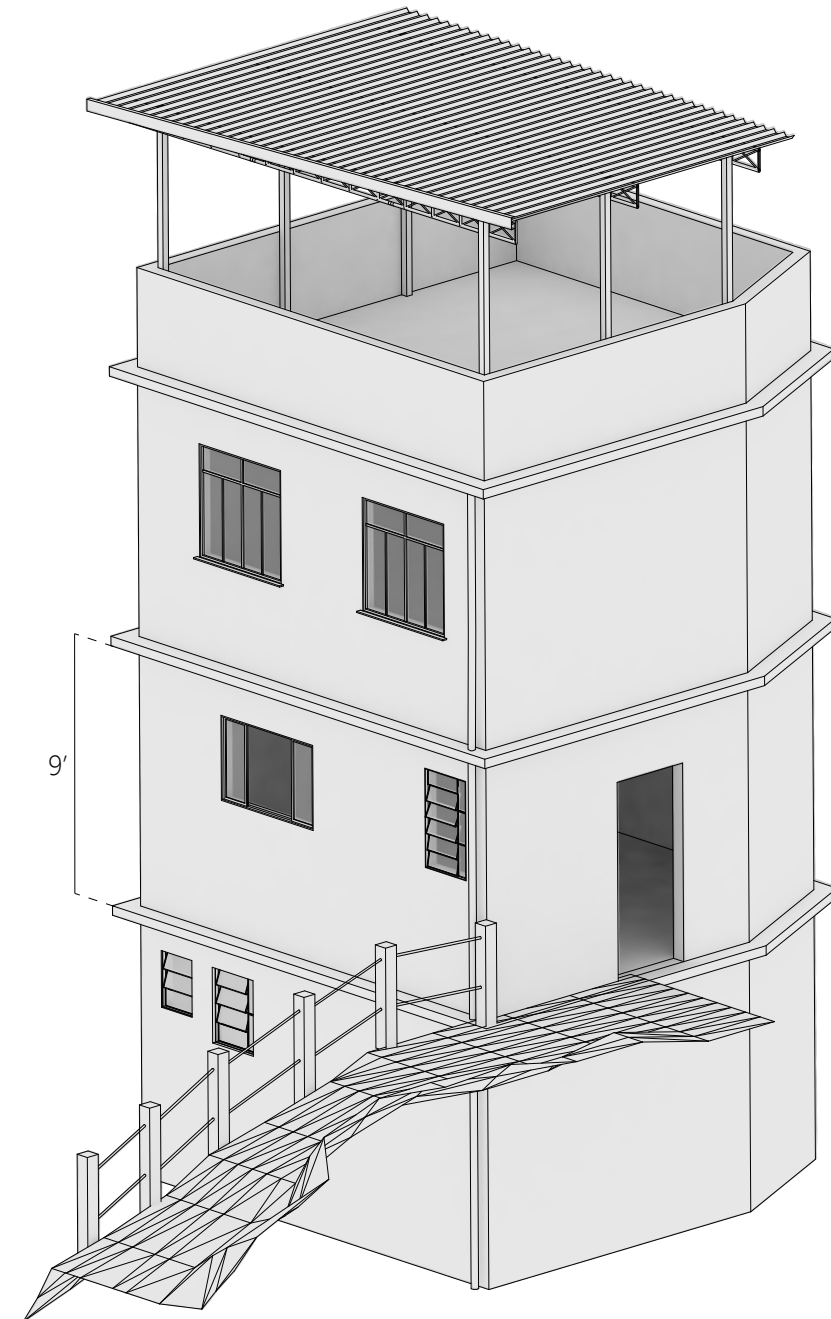
http://insitebuilders.com/blogimages/Blog120513/Insitebuilders-02-DSC_0064_1.jpg

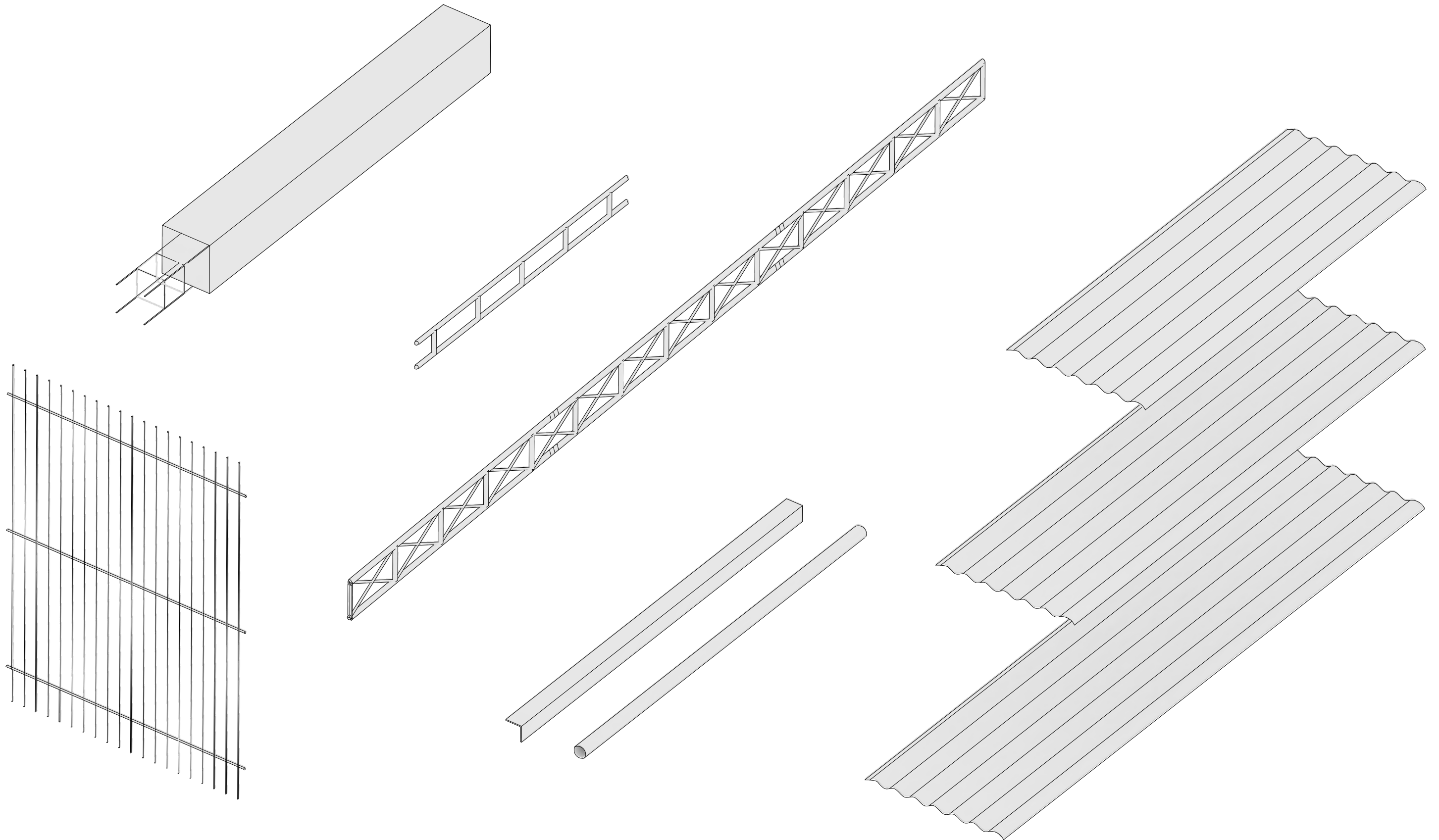


Case Studies | 8 : Hillside Tower

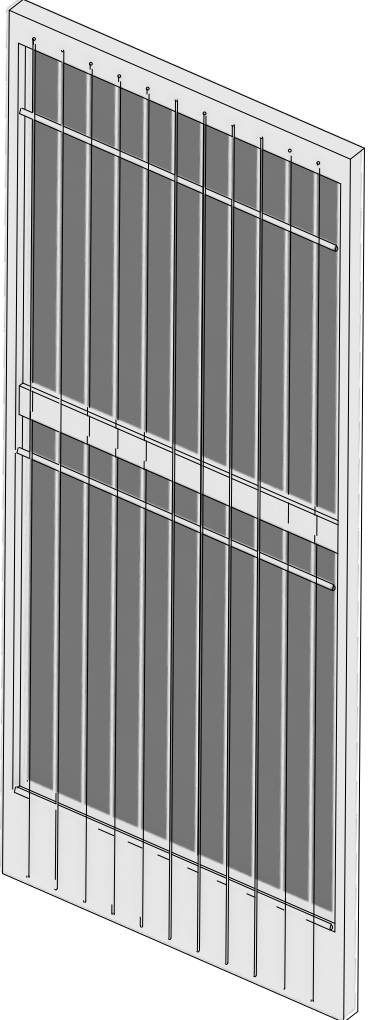
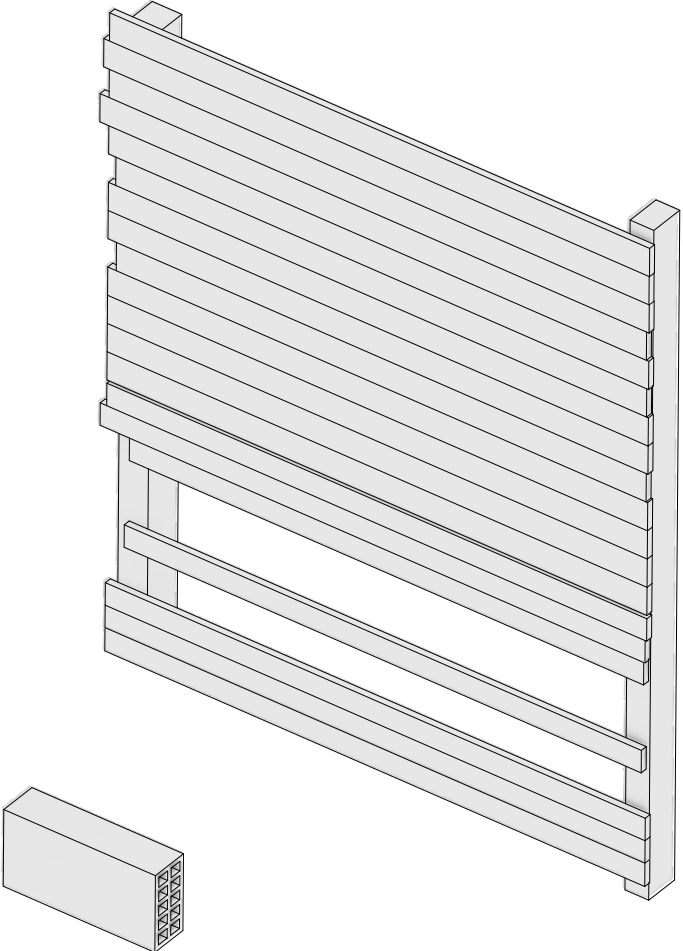
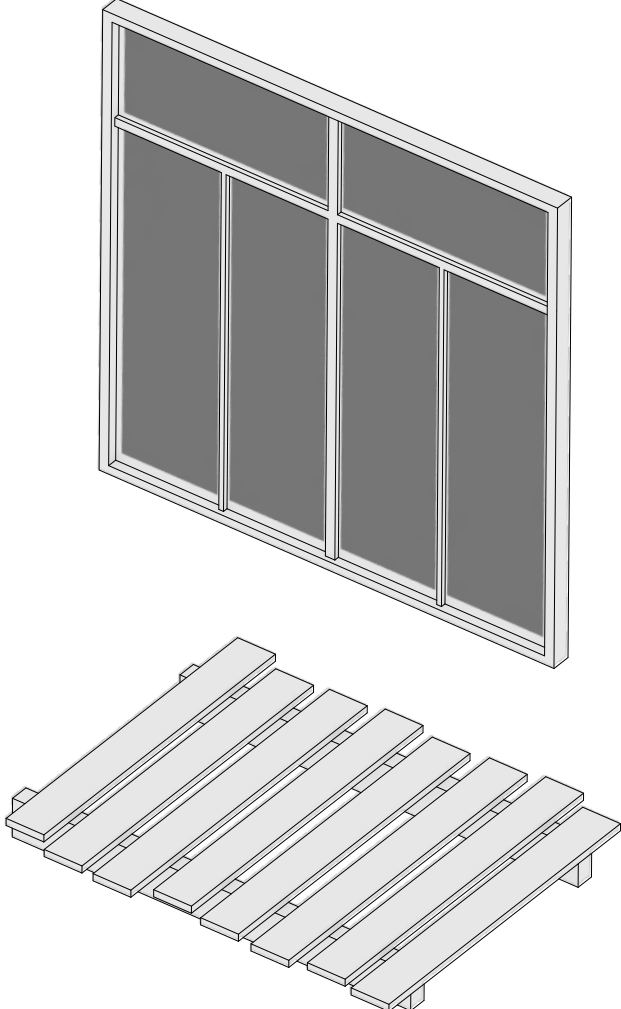
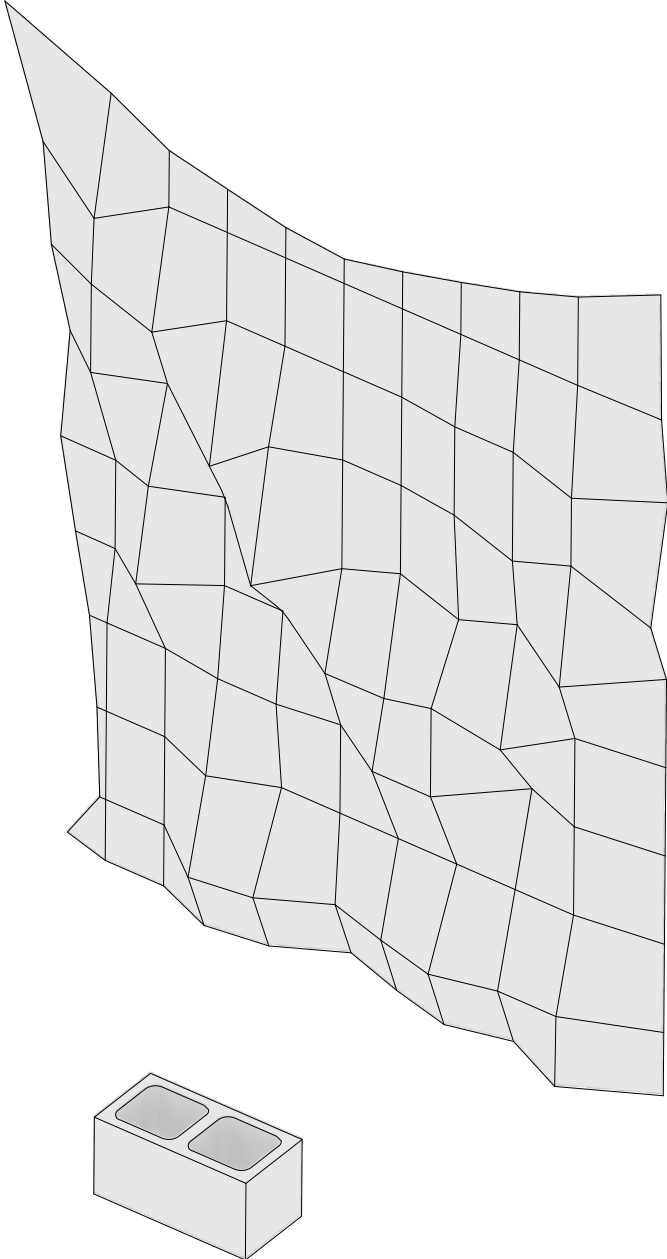


<http://traveldance.net/tag/corcovado/>





Components | Plastic, CMU, Wood and Glass Examples



Strategy |

Use catalogue of parts in conjunction with data from case study settlements to deploy those parts in to as high a degree of accuracy as possible.

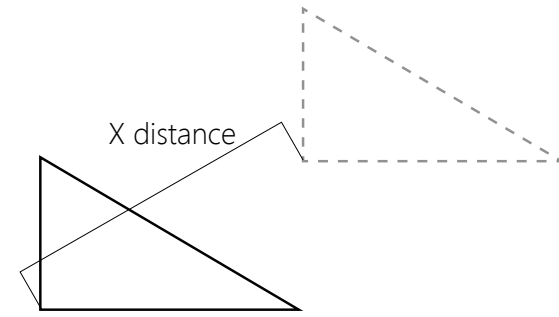
Data |

- Area Occupied
- Number of People
- Average Number of Floors
- Circulation Corridor Widths
- Number of Structures
- Households
- Men and Women
- FAR
- Material Usage by Percentage
- Program (not fully represented)
- Building Legality
- Construction Methods
- Existing Infrastructure
- ...

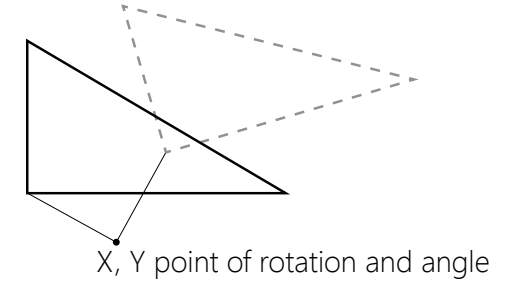
A type of generative system which create 2D and 3D geometric shapes by utilizing a series of specific "shape rules" in combination with a "generation engine."

The engine processes the rules which are transformative operations such as boolean unions, intersection, difference, as well as transformations such as reflection, rotation, scale, and translation this.

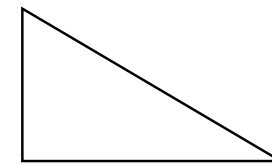
The initial input is modified through these transformations in order to produce a new form.



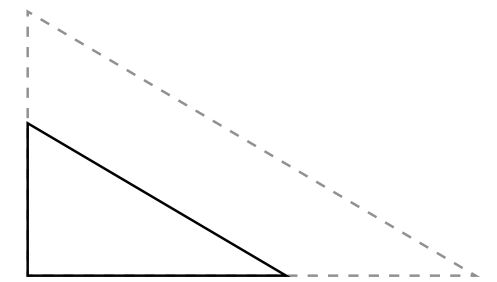
Translation



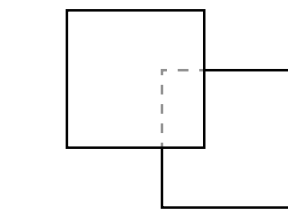
Rotation



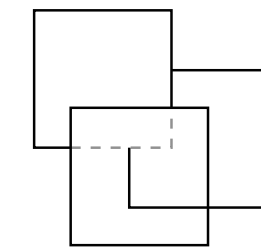
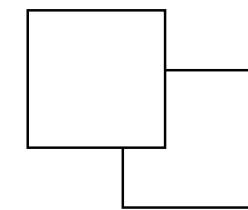
Reflection



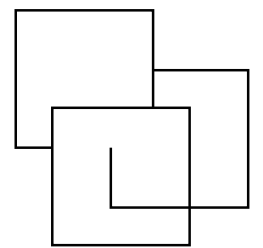
Scaling



new square trims existing



heirarchy of boolean



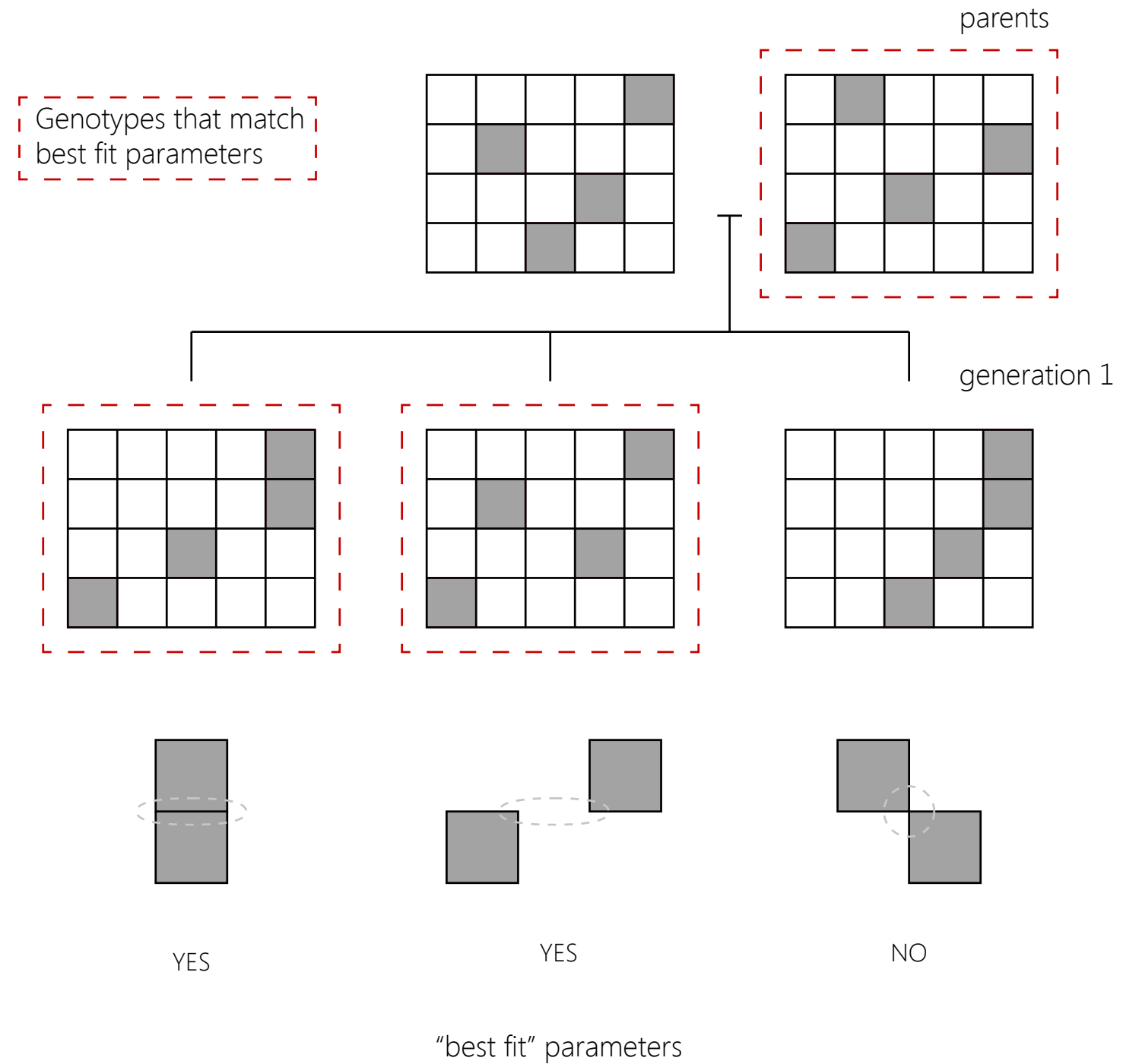
Boolean

This is a method of programming which mimics the recombinant nature of genetic mutation in order to produce new combinations and variety from specific input models

Newly produced combinations are then assessed based on a best fit model (manually or parametric) and then recombined if they meet the requirements, similarly to the way in which mimics the process of natural selection.

They can be assigned a "fitness value" based on their resemblance of the best fit model and ranked accordingly.

The final output will be a set of "offspring" which are algorithmically generated, and sorted according to their correlation with the best fit model.

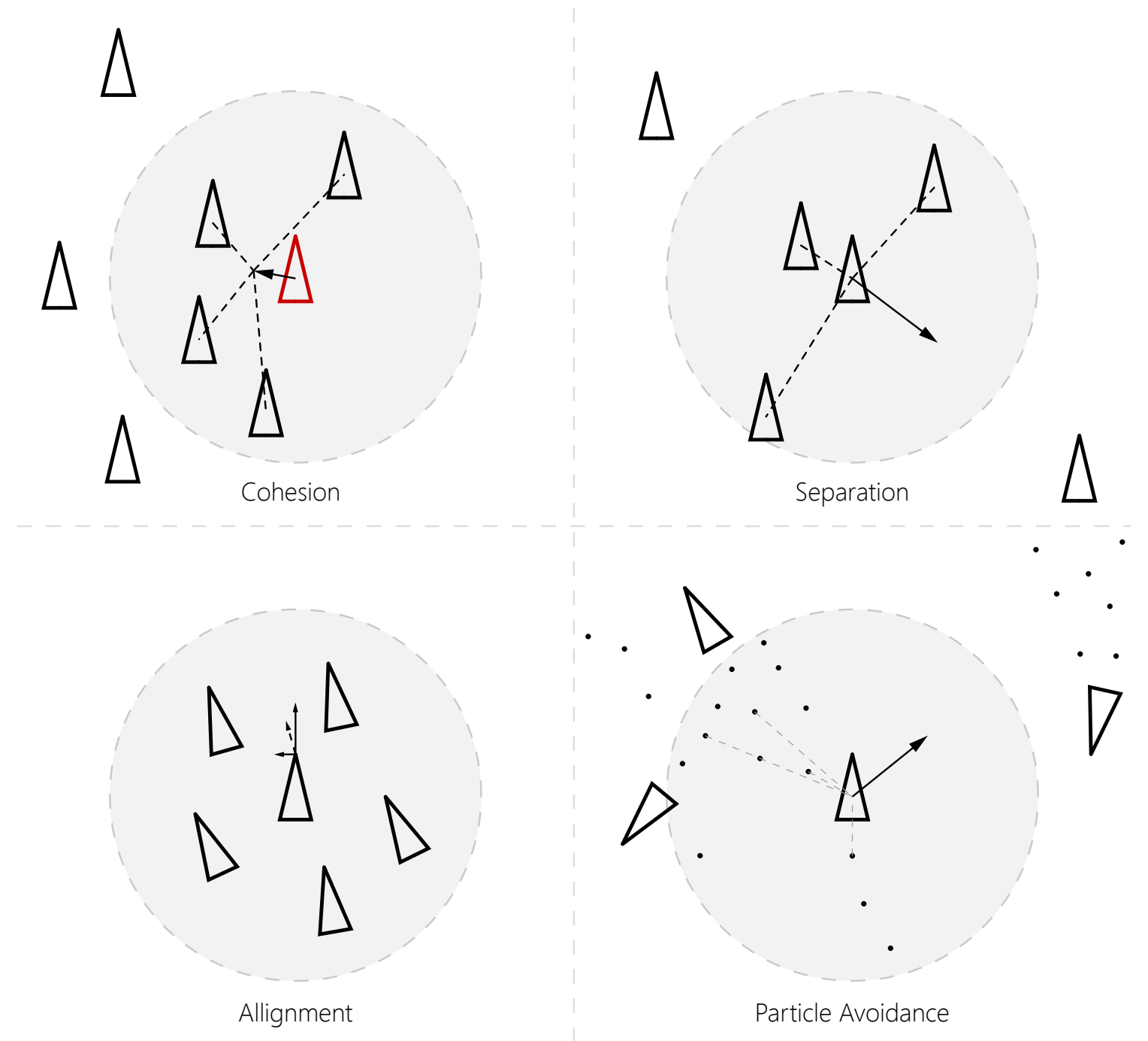


Modeling method based on the actions of individual objects or “agents” within the system which act independently based on a set of criteria.

Usually when making agent based models, the criteria which determine behavior are simple in order to avoid conflict and chaotic results.

The agents start from an origin and are allowed to make a specific number decisions, after which the model produced is assessed based on movement from their origins toward an emergent form.

Typical modeling rules include cohesion and separation where agents steer towards or away from each other or another specific element, and alignment where agents derive their trajectory from the average of their neighbors.



As a type of unsupervised learning, SOMs are classified as artificial neural networks in programming, they are typically used to take multi-dimensional data (x, y, z... R, G, B...) and represent it in lower dimensions.

Usually colors are used for data because of the fact that they carry 3 values if assessed on an RGB scale. And the location is addressed by the x, y coordinate.

The individual neuron's within the system randomly select a sample, then move to the best matching coordinate, scale the neighboring neuron locations and increase the weight of the neuron (resistance to movement/scaling).

This is repeated often more than 1000 times.

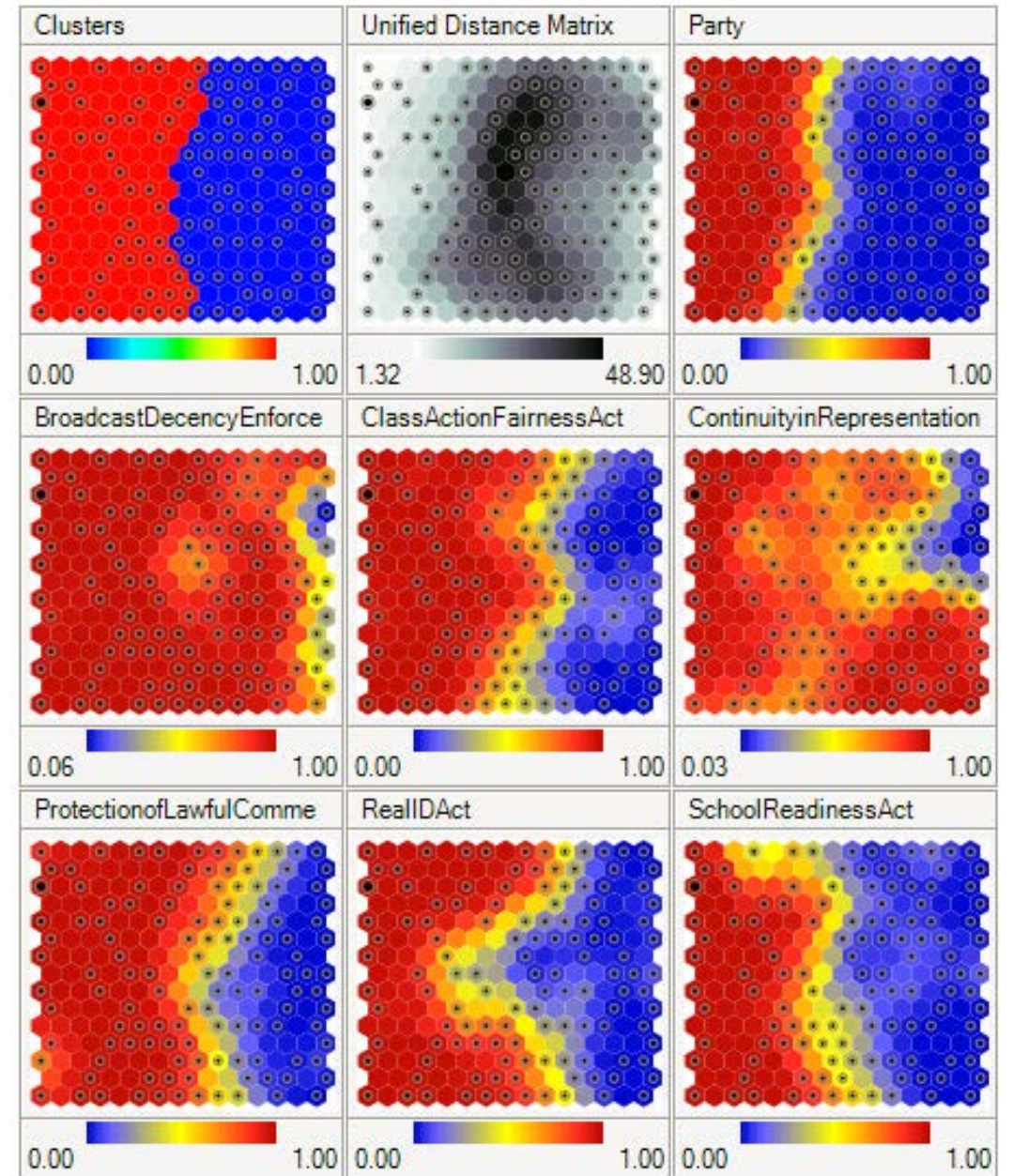
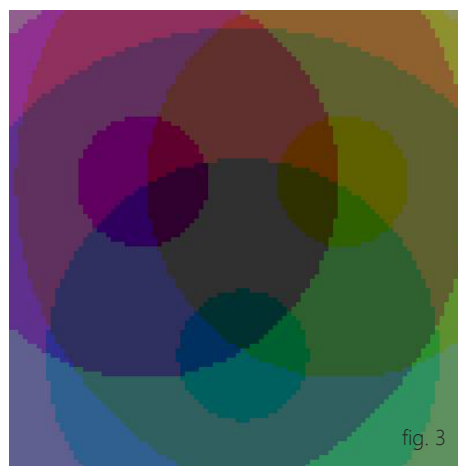
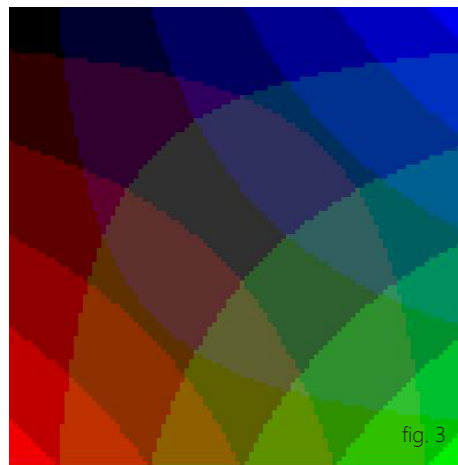


Fig 33, 34, 35 : Germano, Tom. "Self Organizing Maps." Worcester Polytechnic Institute. Last modified March 23, 1999. Accessed December 16, 2015. <http://davis.wpi.edu/~matt/courses/soms/>.

Fig. 36 : "Self-organizing map." Wikipedia. Last modified June 2011. Accessed December 16, 2015. https://en.wikipedia.org/wiki/Self-organizing_map.

Using a regular grid of cells (various shapes will work) with each typically having an "on" and "off" state, an initial cell is used to create a new generation based on an algorithm.

Each generation, in turn creates the next, until the program reaches the maximum iterations prescribed.

Each cell only effects other cells within its "neighborhood" which a e typically adjacent cells, and in turn each new generation is effected by the states of multiple cells.

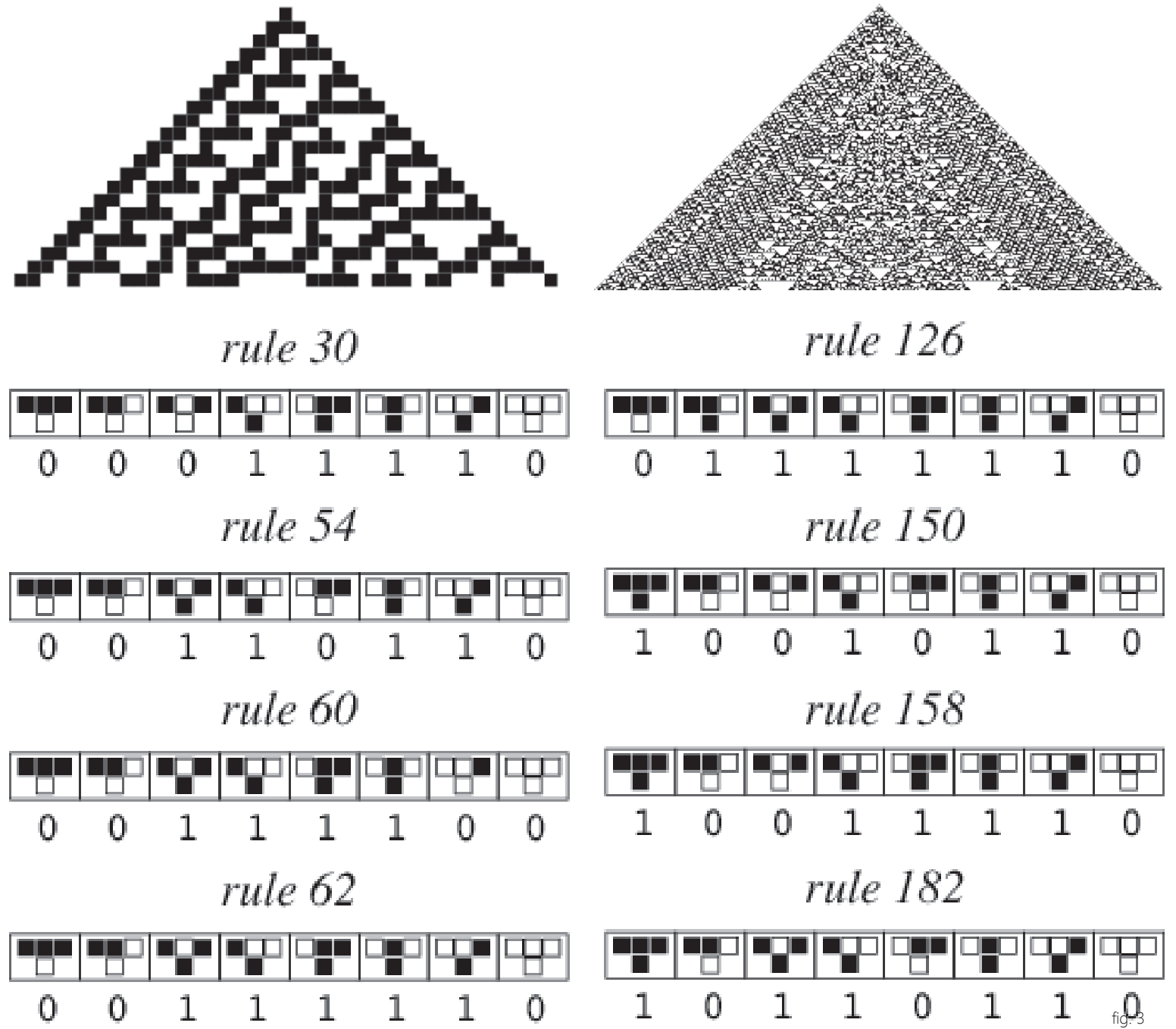
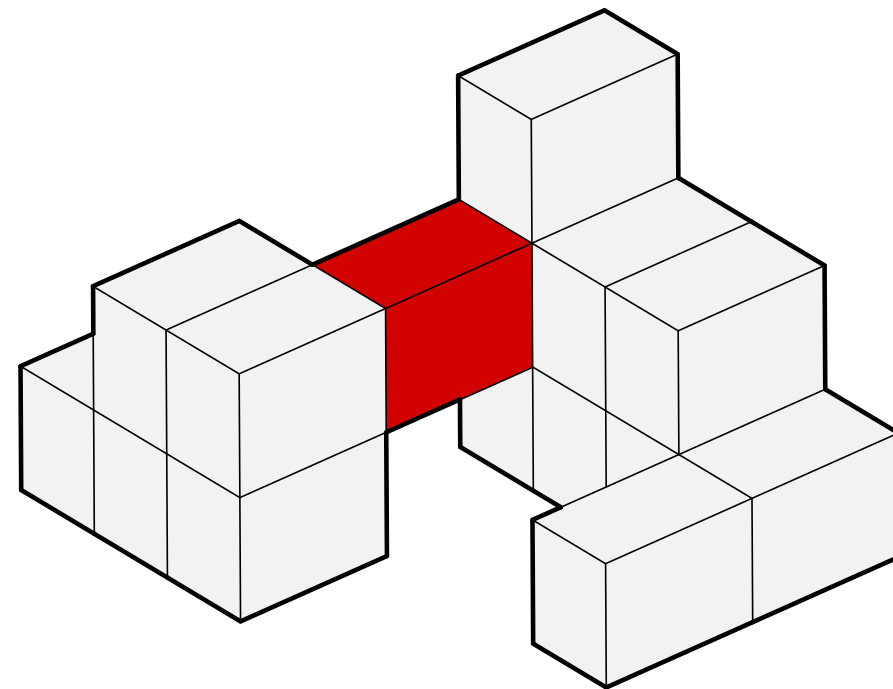
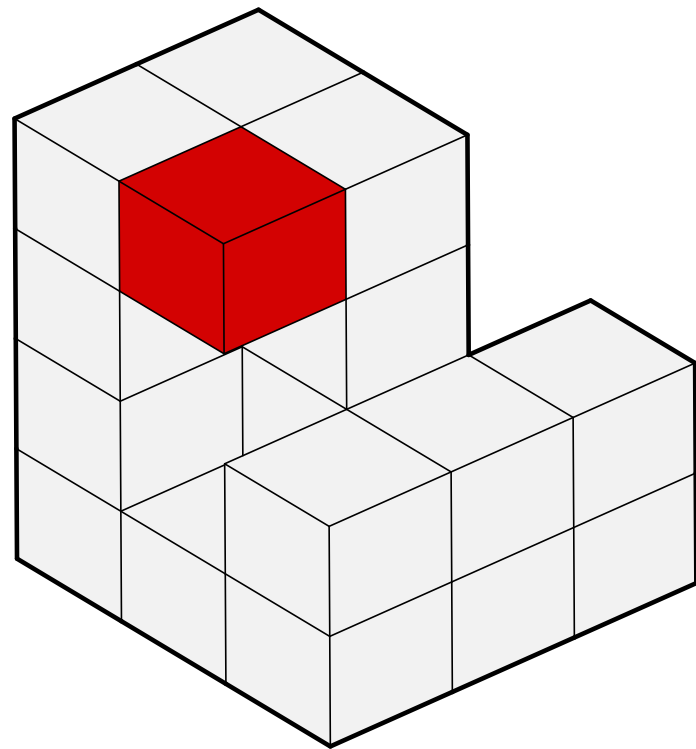
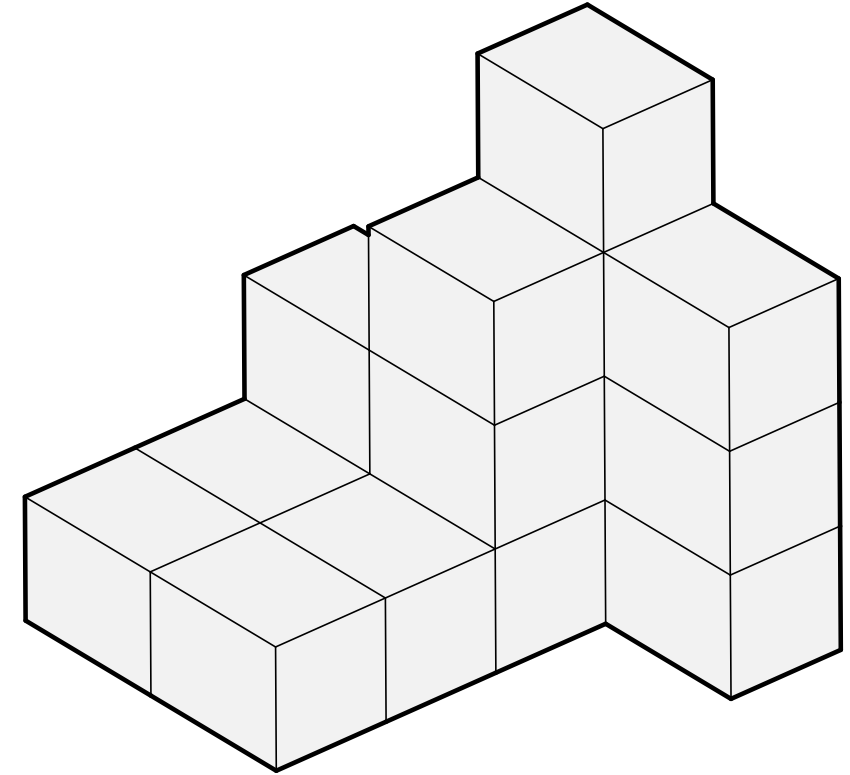
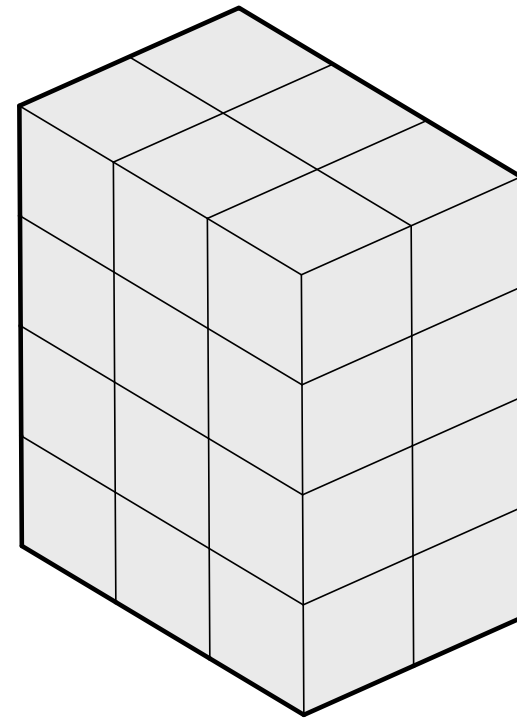
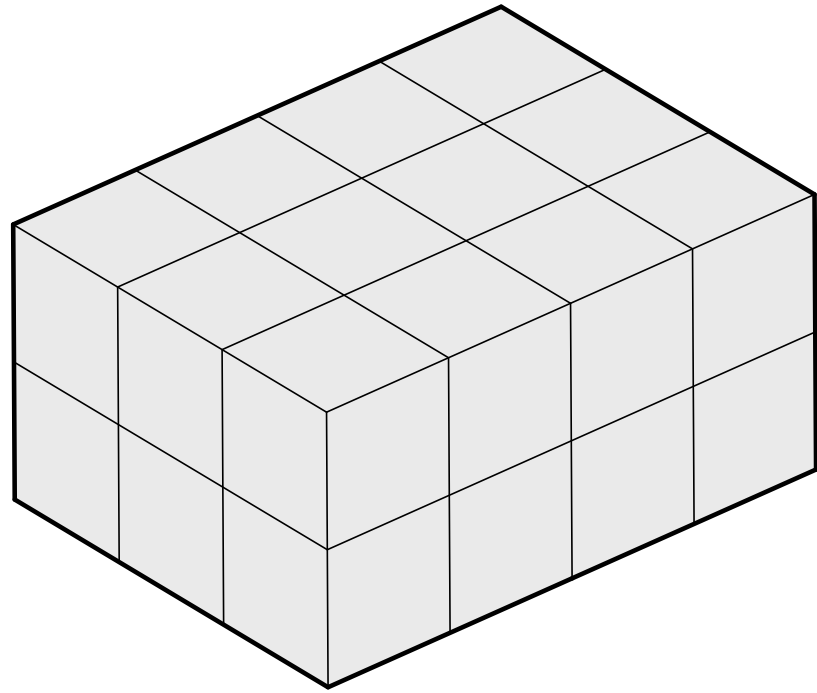
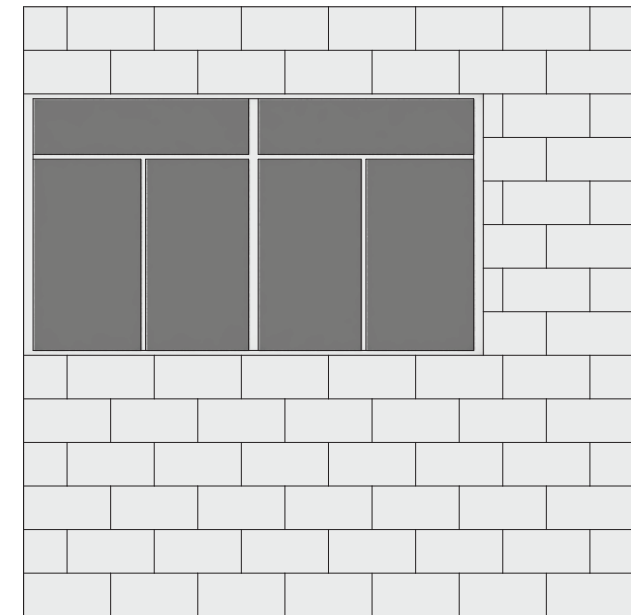
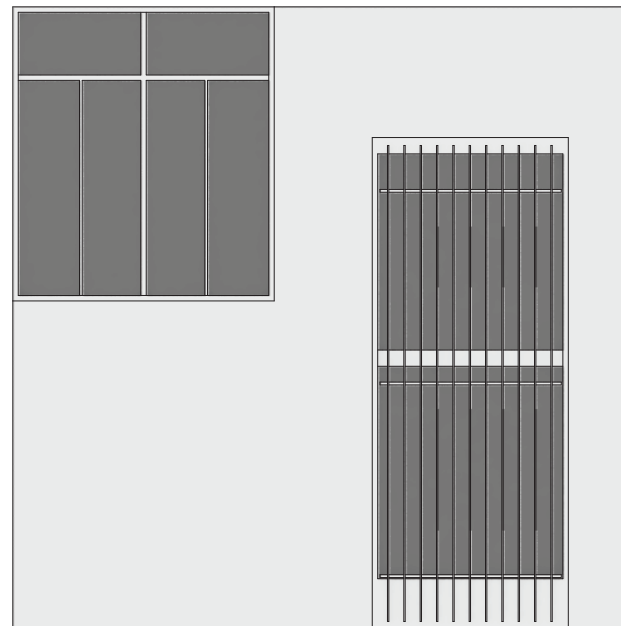
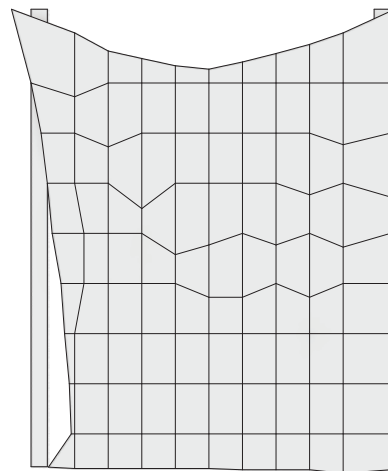
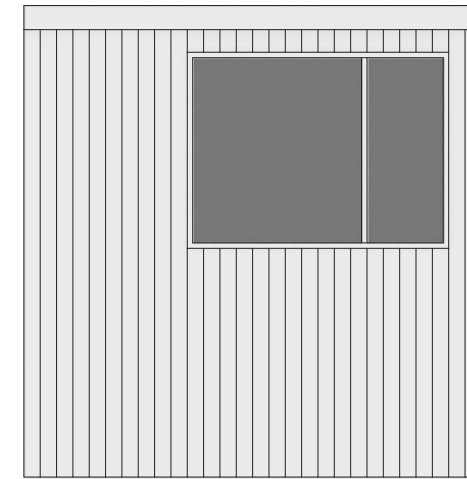
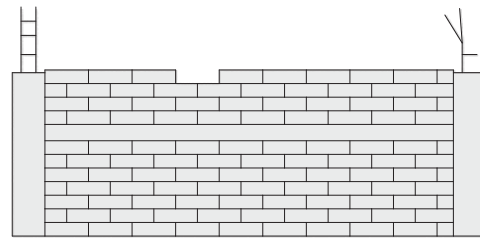
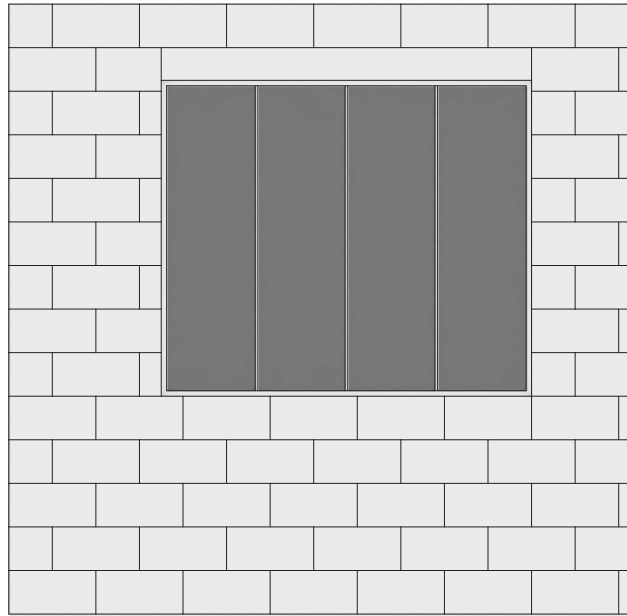


Fig 37 : Weisstein, Eric W. "Elementary Cellular Automaton." Wolfram MathWorld. Accessed December 16, 2015. <http://mathworld.wolfram.com/ElementaryCellularAutomaton.html>.



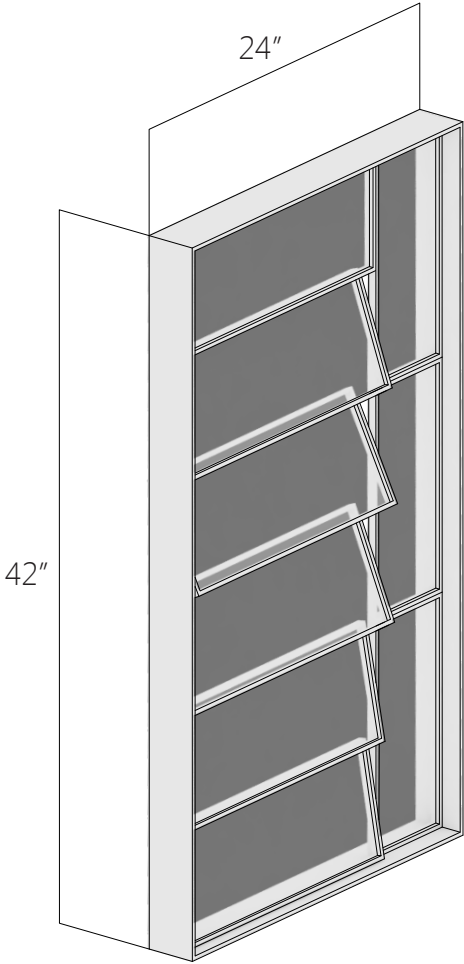
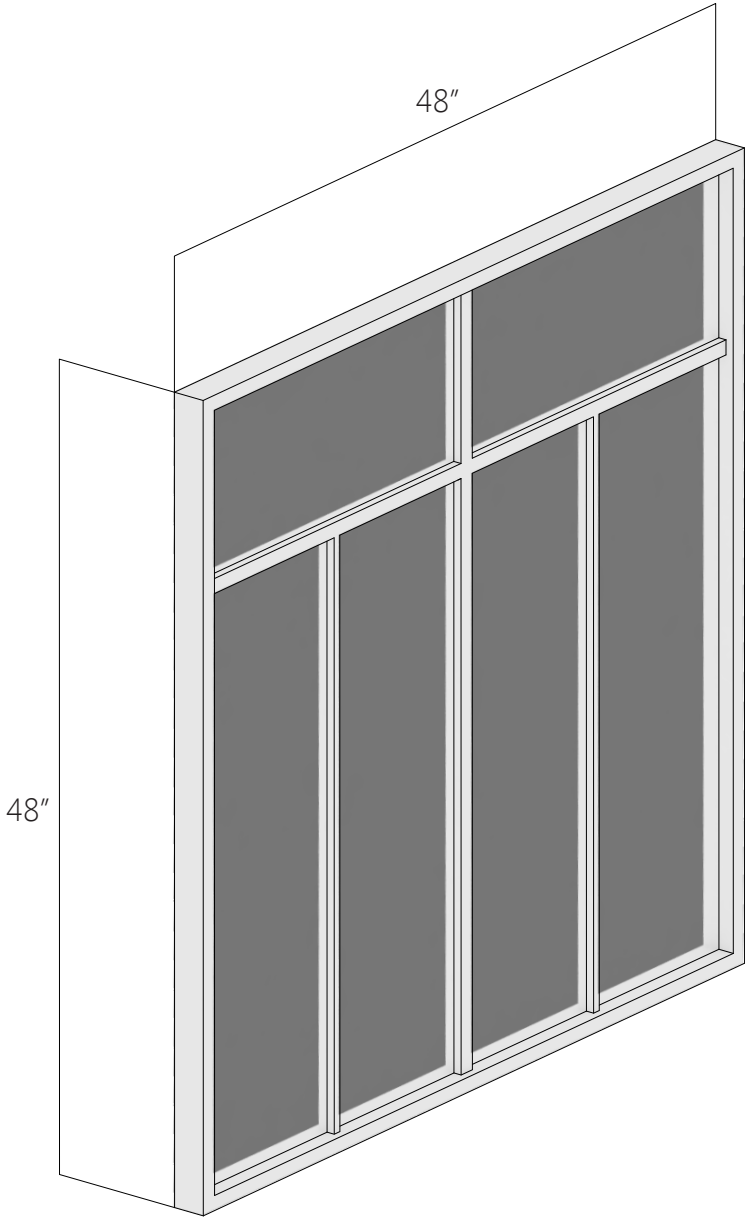
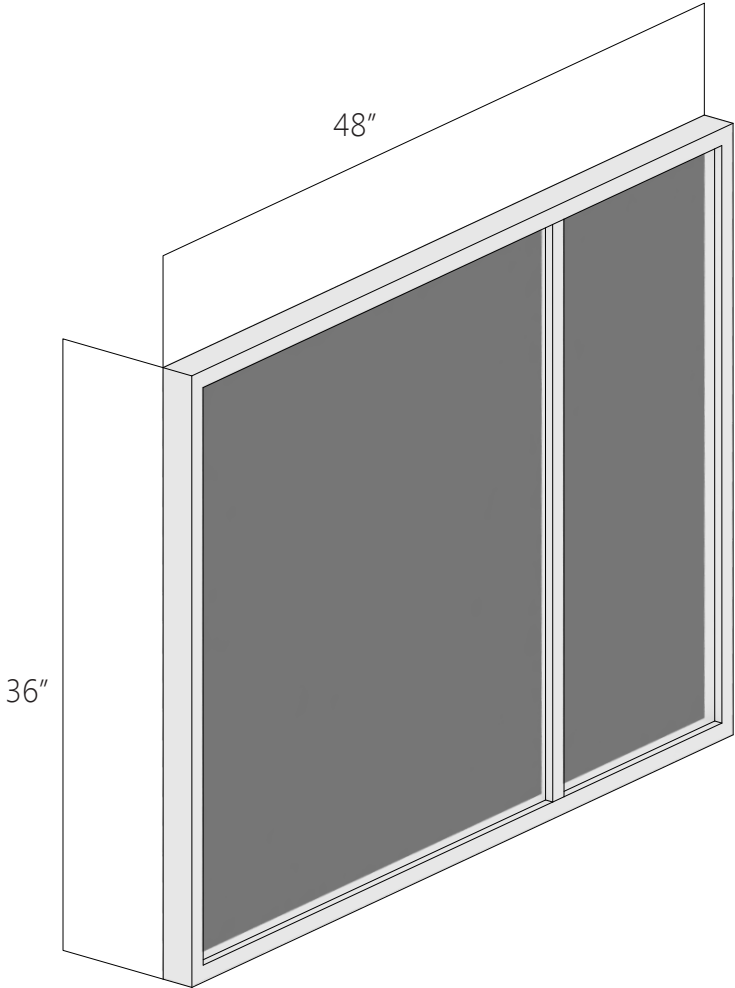
Customization | Wall

Due to the nature of their interaction with the structure or overall bay divisions of a building, must be customized according to each panel. Based on the material chosen, construction methods or assembly walls could vary widely.



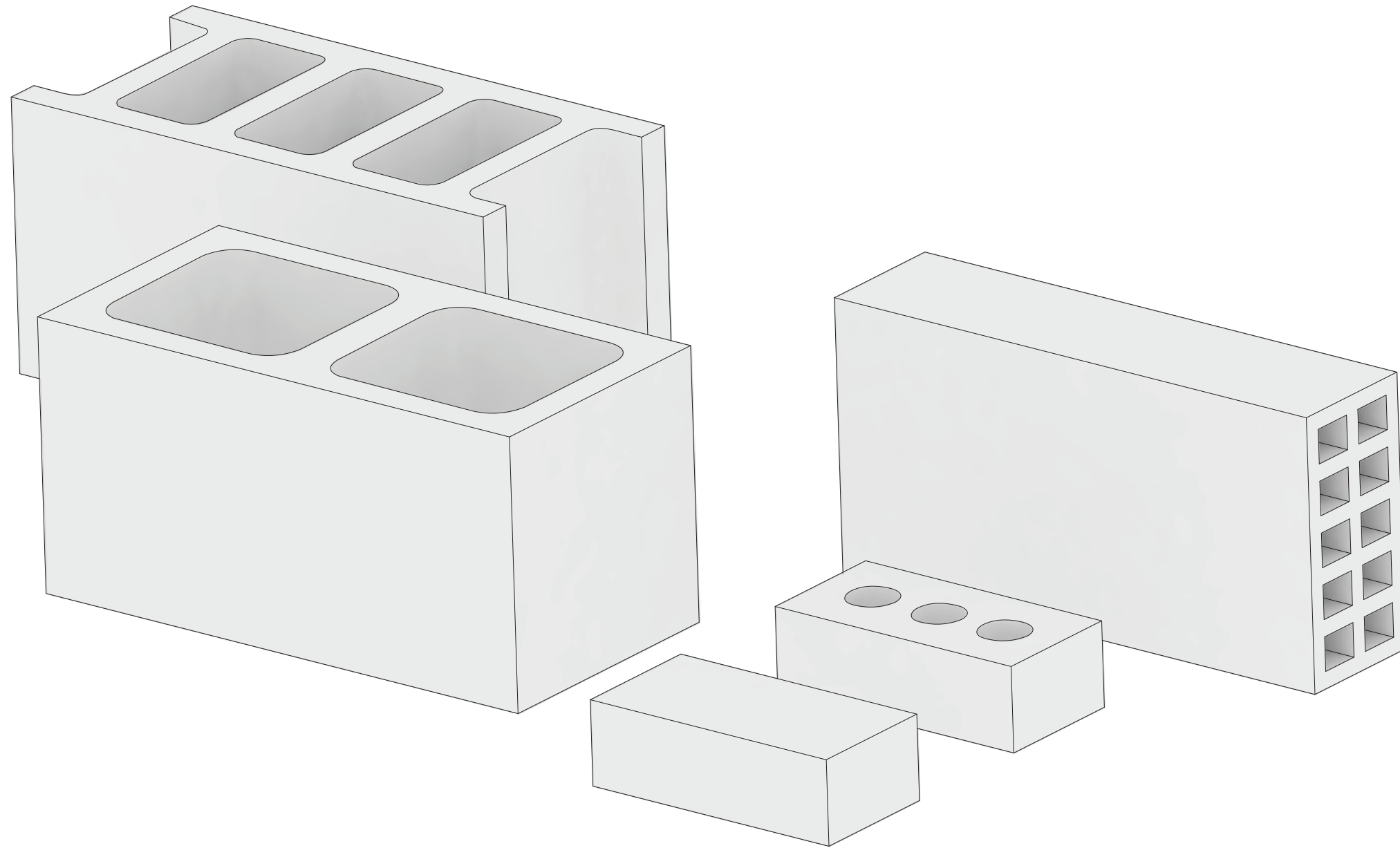
Customization | Window

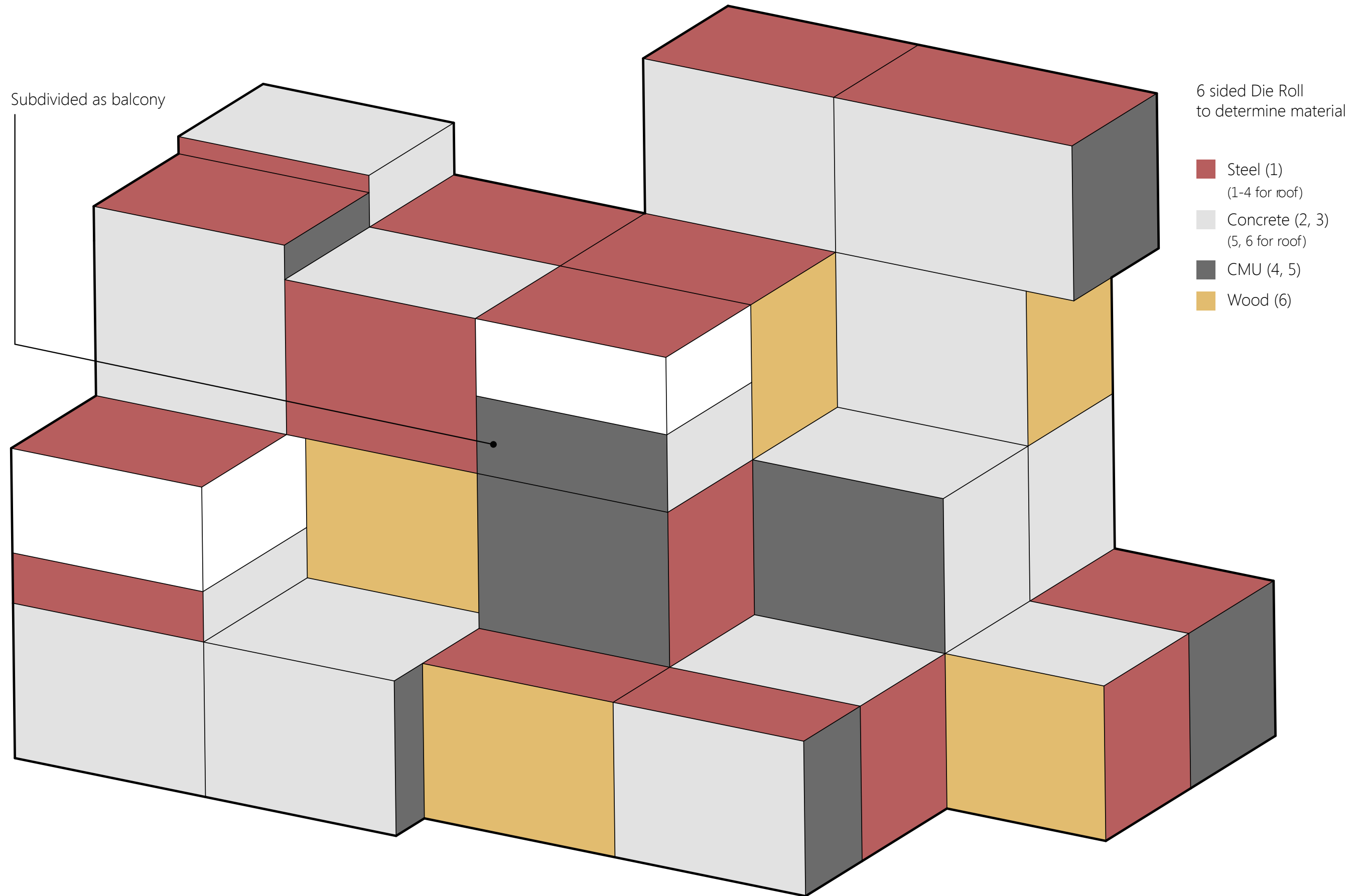
Come in varying shapes and sizes, forms can be easily generated, but a library of 10-20 options might work just as well and still allow for enough variance.

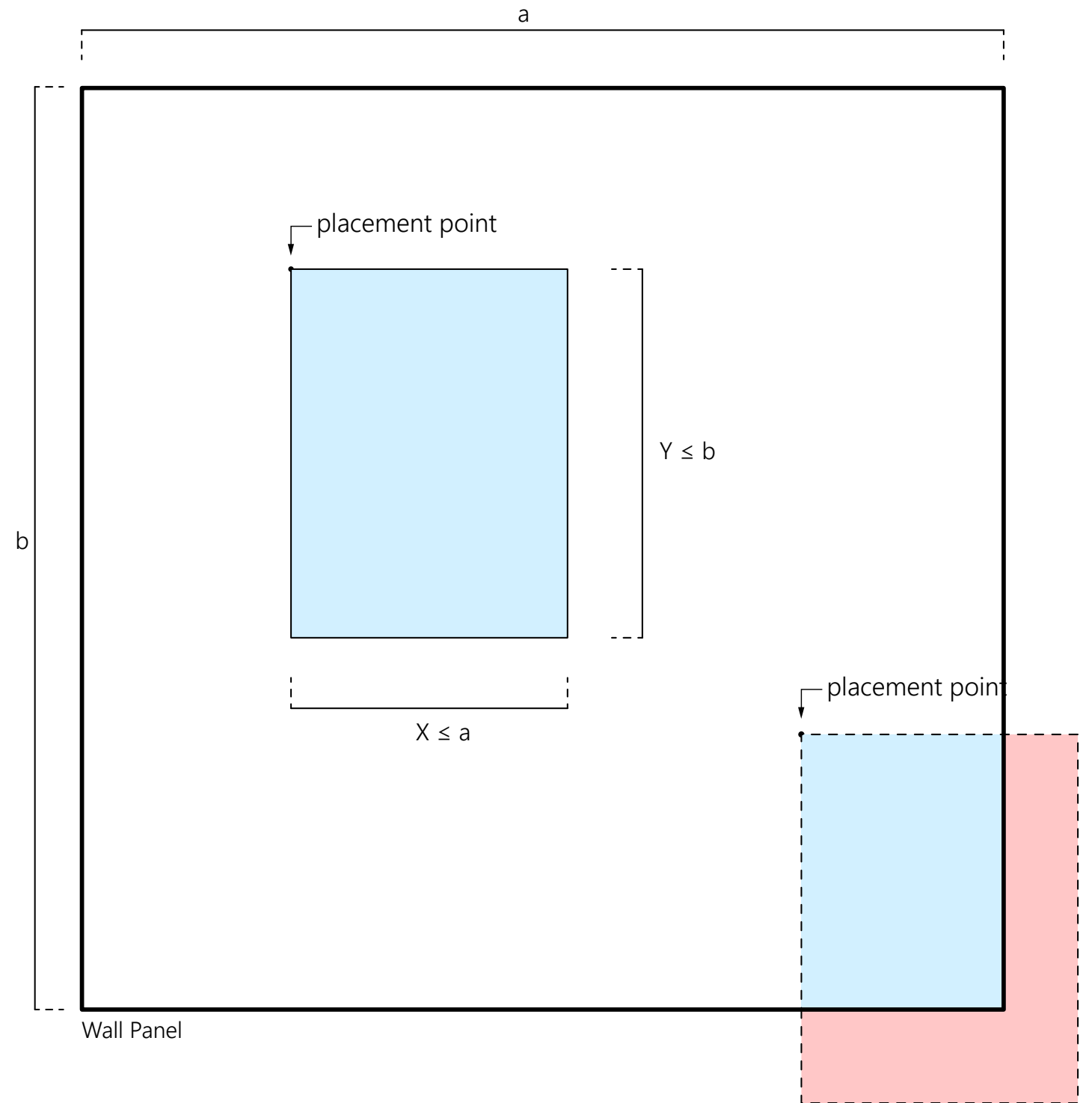
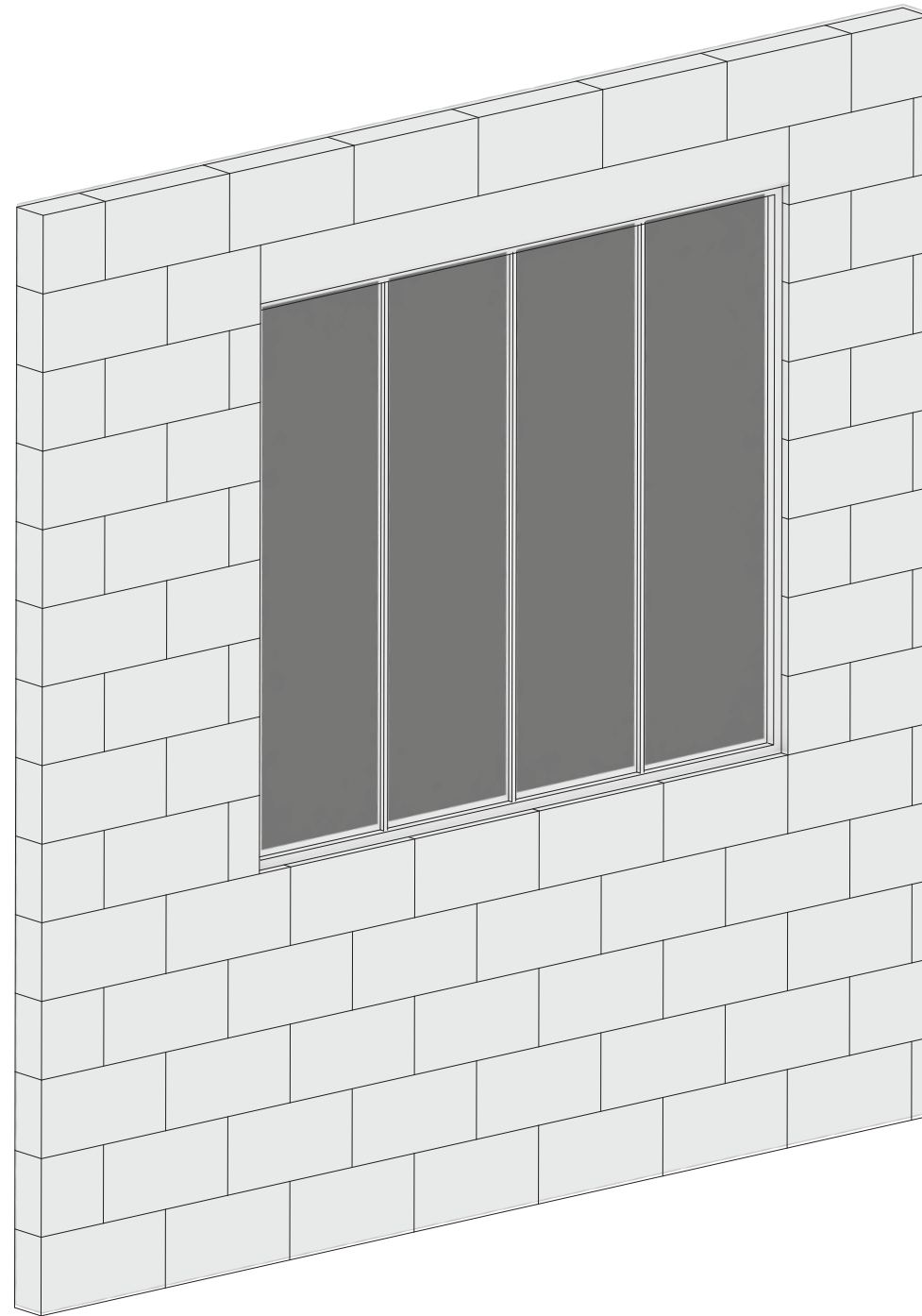


Customization | Brick

Easy to customize and work with due to modular nature and repetition. Can be understood as panels when modeling in order to preserve computation effort if needed.







If the window is positioned outside the bounding box the window should not be placed

Therefore, (placement point) $\leq ((a-X), (b-Y))$ in order to keep the window within the panel



Citations

Books:

McGuirk, Justin. *Radical Cities: Across Latin America in Search of a New Architecture*. London, England: Verso Books, 2014.

Brillembourg, Alfredo, and Hubert Klumpner. *Torre David*. Zurich, Switzerland: Lars Muller, 2013.

Angelil, Marc, and Rainer Hehl, eds. *Cidade de Deus; Working with Informalized Mass Housing in Brazil*. Berlin, Germany: Ruby Press, 2013.

Gouverneur, David. *Planning and Design for Future Informal Settlements*. New York, NY: Routledge, 2015.

Zappulla, Carmelo, Cristian Suau, and Alenka Fikfak. "The Pattern Making of Mega-slums on Semantics in Slum Urban Cultures." *Journal of Architecture and Urbanism*, December 23, 2014, 247-64.

Simpson, Mariana Dias. "Urbanising Favelas, Overlooking People: Regressive Housing Policies in Rio de Janeiro's Progressive Slum Upgrading Initiatives." *Development Planning Unit*, paper no. 155 (August 2013).

Girard, Greg, and Ian Lambot. 1993. *City of darkness: life in Kowloon Walled City*. [Chiddingfold]: Watermark.

Leach, Neil. "Swarm Urbanism." *Architectural Design* 79, no. 4 (July 2009): 56-63.

Mass, Winy, ed. *The Vertical Village*. Rotterdam, Netherlands: NAI Publishers, 2012.

Alexander, Christopher. *Notes on the Synthesis of Form*. Cambridge, MA: Harvard University Press, 1966.

Johnson, Steven. *Emergence: The Connected Lives of Ants, Brains, Cities*. New York, NY: Scribner, 2001.

Websites:

Peeble, Anthony. "Rust Faction Towns." *Imgur*. Last modified September 25, 2015. Accessed December 16, 2015. <http://imgur.com/a/9GWu7>.

"Project Showcase." *Planet Minecraft*. Last modified 2013. Accessed December 16, 2015. <http://www.planetminecraft.com/project/>.

"Game Development." *StackExchange*. Accessed December 16, 2015. <http://gamedevstackexchange.com/>.

"New Minecraft model of Sinay Market." *Block by Block*. Last modified July 4, 2014. Accessed December 16, 2015. <http://blockbyblock.org/>.

"Press Sheet." *No Man's Sky*. Accessed December 16, 2015. http://no-mans-sky.com/press/sheet.php?p=no_man%27s_sky.

Germano, Tom. "Self Organizing Maps." *Worcester Polytechnic Institute*. Last modified March 23, 1999. Accessed December 16, 2015. <http://davis.wpi.edu/~matt/courses/soms/>.

"Self-organizing map." *Wikipedia*. Last modified June 2011. Accessed December 16, 2015. https://en.wikipedia.org/wiki/Self-organizing_map.

Weisstein, Eric W. "Elementary Cellular Automaton." *Wolfram MathWorld*. Accessed December 16, 2015. <http://mathworld.wolfram.com/ElementaryCellularAutomaton.html>.