

---

# Vetruvio: Interactive Visualization of Cities

Interactions with alive cities

Master's Thesis submitted to the  
Faculty of Informatics of the *Università della Svizzera Italiana*  
in partial fulfillment of the requirements for the degree of  
Master of Science in Informatics  
Software Design

presented by  
Nicolas Lusa

under the supervision of  
Prof. Michele Lanza  
co-supervised by  
Yuryi Tymchuk

September 2015



---

I certify that except where due acknowledgement has been given, the work presented in this thesis is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; and the content of the thesis is the result of work which has been carried out since the official commencement date of the approved research program.

---

Nicolas Lusa  
Lugano, 11 September 2015



*To the very last person who reads this thesis*



"If you're not failing every now and again, it's a sign you're not doing anything very innovative."

Woody Allen





# Abstract

Cities are in constant evolution. Growth and changes in the town structure influence the city scheme. Cities are built and arranged with the aim to satisfy the population needs. Towns record data relevant to the city and its buildings in order to have functionalities overview. Two different formats are used to store data. The classical method expects the saved data to be on paper. Cities more prone to technology moved to a digital format. Data on paper is complex to handle. It requires significant physical space and human resources. When a document has to be consulted, relevant human effort is required and significant time is wasted in the search. Digital data is simple to handle. It requires little storage space and low human resources are needed to store it. The consultancy is immediate, interesting data is found automatically by a computer and presented to the user without any relevant human effort.

Both paper and digital formats present data in alphanumeric type with little or no visual feedback provided. Alphanumeric type is a data type that does not allow any interaction, only consultancy. With this kind of data it is difficult to have an overview over a town state. This issue influences negatively decision making over a city scheme.

This thesis introduces VETRUVIO, a tool which provides an interactive 3D representation of cities. Digital data of cities is brought into VETRUVIO which produces a representation of the city model providing interactions over the town. Examples of interactions are: searching the closest point of interest to a building, grouping buildings based on personal constraints, coloring the city based on hot spots. The tool allows users to inject data or modify properties of city components. It finally produces a visual feedback about the current city state.

VETRUVIO allows to hold control and awareness on a city state and on the arrangement of its components. The aim is to ease further planning on a city expansion through a visual representation of the data. VETRUVIO is a tool that creates a visual city representation which, along with the interactions provided, aims to enhance the town visualization.



# Acknowledgements

Not many years ago I came across for the first time with a tough professor who was used to remind me every week what I was doing with my life. Most people do not take seriously statements of this kind and so do I, but when it is a constant statement that is thrown at you every day, it eventually reaches you. For this reason I really want to thank this professor to have made me realize what I really wanted in my life.

I want to thank Prof. Michele Lanza who allowed me to work on this awesome thesis topic. I am glad I have been accepted as his master student and have been supervised by him. I am thankful to have been pushed to work hard on the bitter parts of the project.

I acknowledge Yuri Tymchuk for the help provided me in the project development phase and the patience in bearing with me for that period of time.

I would like to further extend my thanks to the whole REVEAL research group for the availability and willingness in discuss problems encountered during the project development. In particular I want to thank Tommaso Dal Sasso and Roberto Minelli who provided me assistance while developing in Pharo. Moreover I want to thank Luca Ponzanelli who never gave up on me and always pushed me when the moral was low.

My thanks have to be extended as well at Architect Nasso Marcello who followed the development of the project and provided useful advice.

I praise my family who supported me all the time and stood to me during the hardest time. In particular I want to thank my parents and my grandparents who made this possible. As well as my friends who encouraged me to never give up. A special thank is addressed to my comrades Andrea Ciani and Nicolas Latorre for the support over the last few years.



# Contents

<b>Contents</b>	<b>xi</b>
<b>List of Figures</b>	<b>xiii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Contributions . . . . .	2
1.2 Structure of the document . . . . .	3
<b>2 State of the Art</b>	<b>5</b>
2.1 History of city planning . . . . .	5
2.1.1 Pre-Classical . . . . .	5
2.1.2 Classical . . . . .	5
2.1.3 Middle Ages . . . . .	6
2.1.4 Renaissance . . . . .	8
2.1.5 Enlightenment . . . . .	9
2.1.6 Modern Urban Planning . . . . .	9
2.1.7 Urban Crisis . . . . .	12
2.1.8 New Urban Planning . . . . .	12
2.2 Current tools to visualize cities . . . . .	13
<b>3 Vetruvio</b>	<b>15</b>
3.1 Environment and frameworks . . . . .	15
3.2 Woden . . . . .	16
3.3 First Attempt: OSM 3D city model . . . . .	17
3.4 Vetruvio, the final prototype . . . . .	19
3.4.1 The structure . . . . .	20
3.4.2 Data gathering, reading and elaboration . . . . .	21
3.4.3 Provided interface . . . . .	22
3.4.4 Potential and limit . . . . .	24
<b>4 Use Cases</b>	<b>25</b>
4.1 How to access information . . . . .	25
4.2 General interactions . . . . .	28
4.3 Actions on single selection . . . . .	29
4.4 Actions on group selection . . . . .	31
<b>5 Reflections</b>	<b>35</b>

<b>6 Conclusions</b>	<b>37</b>
6.1 Summary . . . . .	37
6.2 Future work . . . . .	37
<b>Bibliography</b>	<b>39</b>

# Figures

2.1	Elburg town scheme in the Netherlands - 1830. . . . .	7
2.2	Palmanova representation of the seventeenth century. . . . .	8
2.3	Modern town Paris scheme. . . . .	10
2.4	Ebenzer Howard's diagram for urban growth through garden cities - 1902. . . . .	11
3.1	Representation of the Triumphal Arch area in Paris. . . . .	17
3.2	Triumphal Arch view almost at ground level. . . . .	18
3.3	VETRVIO starting screen. . . . .	19
3.4	VETRVIO architecture scheme. . . . .	20
3.5	From left to right: main interface, single selection and group selection tab. . . . .	23
4.1	Main tab interface and single selection interface. . . . .	26
4.2	Pharo inspector over a building. . . . .	27
4.3	Pharo inspector on a building model with inspection on properties field. . . . .	28
4.4	City colored based on altitudes. . . . .	29
4.5	City colored based on proximity to post offices. . . . .	30
4.6	Single selection view. . . . .	30
4.7	Real model of a selected building. . . . .	31
4.8	Scene with group selection over buildings with more than three floors. . . . .	32
4.9	City colored based on proximity to buildings with a single floor. . . . .	33





# Chapter 1

## Introduction

Cities nowadays appear to be well designed following architectural schemes. The aim to organize a city structure is to provide citizens with public services and infrastructures trying to enhance their daily quality life. An optimal management of city resources is difficult to achieve, but an efficient solution is eventually at reach. The fact that a city is in constant evolution, because of the appearance of new elements and disappearance of old ones, makes the structure of the city unstable and quickly prone to changes. These factors often break the optimal equilibrium that the city might have previously achieved. Decision making in these cases is not trivial. In order to avoid losses in efficiency a visualization, which provides a city overview, is required. A graphical city overview is often not available and this means that decisions might be taken without having a clear picture of the surrounding environment. The lack of clarity in data representation leads to decisions that might have a negative impact on the current city state affecting the life quality of citizens and in extreme cases the city public image.

The choice of a location is a further problem which arises when an unclear view of a town is provided. Many factors have to be kept into consideration. These factors depend on the entity that is looking for the location, which could be a family, a company or rather a public institution. Decide which location to pick is not a trivial task. Often an ideal scenario is not offered and therefore alternative options have to be considered. In order to analyze the available options it is required to collect a lot of data. Only once the data collected is analyzed it is possible to opt for the most suited option. This task is not trivial either and it happens that, due to missing information, the most efficient solution is not considered.

In order to face these issues the city collects data relative to buildings and their arrangement. The collected data is stored into different formats. Cities which are more toward technologies adopted a virtual representation of the data while others represent the data on paper. The digital data is more dynamic and less cost effective with respect to the paper version. Virtual data does not require much storage space and little effort is spent to store it. When data has to be consulted, the search of interesting documents happens automatically, typically a computer figures it out for the user. On the other hand data stored on paper requires relevant storage space, significant time is spent and human resources invested. When a document has to be consulted, time and human effort are required in order to find the data. The information provided by both these different formats is of alphanumeric type. Alphanumeric type allows consultancy but hardly provides interactions which gives a visual feedback to the user. The fact of not having a properly displayed city representation leads to unconscious mistakes. The ig-

norance of both the actual state and arrangement of the city may negatively influence decision making.

Tools that allows to visualize cities and their components into virtual environments are already available. These tools offer either precise or approximate representations of cities which are graphically entertaining. On the other hand there is lack of data availability. Given the huge data gap, poor interaction is present. Representations which does not provide interactions do not reduce the required effort to manage a city structure. Predictions on city evolution are hard to foretell, so is to plan a solution for upcoming changes. The lack of data and interaction in these representations imposes the city authority to find other means in order to plan decisions on the city scheme.

VETRUVIO aims to solve issues related to city visualization. Its goal is to provide a clean interactive visualization and to give the possibility to inject data in the elements, like buildings. The operations that the tool supports have the aim to give visual feedback over the city status to the user. Some examples of interactions provided are: points of interest hot spot coloring, possibility to search for buildings with predefined characteristics and group assignment for buildings.

VETRUVIO is thought to have a vast pool of users. In first place is planned to be used in architectural environment, private architects and architects working for public entities are the main target that VETRUVIO aims to. With its characteristics it is thought to allow architects to experiment in an approximated 3D representation the structure that they want to build, focusing on the strategic choice of the location based on the surrounding environment. VETRUVIO can also be used in different work environments, like estate agencies. VETRUVIO could operate as a virtual window over the city which shows to clients the proximity to points of interest or the available view from the interested building. Third Parties interested in opening a new activity could find a use into VETRUVIO. The collocation of activities based on proximity to competitors and in proximity of providers can be made with the tool.

Through some use cases scenario this thesis aims to show what are the capabilities of VETRUVIO. Third parties judgement toward VETRUVIO are exposed in order to highlight the potential and to prove that representations that this tool provides are interesting in different environments. Finally ends with a presentation of conclusions and possible future work that could be done to extend VETRUVIO.

## 1.1 Contributions

As previously mentioned VETRUVIO aims to solve the problem related to static data representation. It takes the city data and displays it as a virtual 3D interactive city model. VETRUVIO allows the live modification of data and provides different actions which can be performed on top of the city. The main contributions that VETRUVIO provides are:

- Interactive representation of a 3D city model. The thesis has taken as example Lugano city and builds a representation based on the provided data.
- Interactivity with cities and buildings. Color a city based on predefined hot spots or find closest points of interest from a building.
- Editable city components. Inject data into building models, changing data relative to the graphical representation (i.e. floor height) produces a live transformation of the element.

- Feedback based on the city state. Get data relative to buildings or get a more global picture coloring a city based on points of interest.

## 1.2 Structure of the document

- **Chapter 2** is about the state of the art. It resume the history of urbanism starting from early ages when first cities comes to life, ending with new urbanism. In a second place it talks about already existing tools which allow city visualization in a virtual environment.
- **Chapter 3** is the core of the document. It shows how VETRUVIO has been developed, the technologies adopted and decisions that have been made on the path to the final version. Finally discusses limits and potentials of VETRUVIO.
- **Chapter 4** proposes a walk through of features provided by VETRUVIO through four use case scenarios.
- **Chapter 5** exposes reflections based on third parties judgement on the tool.
- In **Chapter 6** the content of the document is discussed and conclusions are presented. Finally ends with a section about the future work where different paths that VETRUVIO could take are analyzed.



## Chapter 2

# State of the Art

### 2.1 History of city planning

This section resumes the evolution of town planning, starting from the pre Classical period up to modern and current urban planning with the aim to contextualize the main topic of the thesis. The history has provided several examples, out of those examples some are going to be discussed in this chapter. Those examples aim to show that planning a city scheme is not a trivial task but is instead a complex activity which is required in order to satisfy population needs.

#### 2.1.1 Pre-Classical

The first planned cities can be found in the pre-Classical and Classical periods which already had cities that were laid out based on some defined schemes, the majority though were still evolving organically. Examples of planned cities at that time can be found in different places. In particular the Minoan civilization on the island of Crete, the zone of the Tigri-Euphrates river system (nowadays Iraq and Kuwait), the Indian subcontinent (currently Pakistan and northwest India) and the Egyptians themselves were building cities based on predefined plans. According to archaeological retrievals the cities of Happa, Lothal, Dholavira and Mohenjo-daro are the earliest cities which are thought to be developed following some predefined plan, they are located between today India and Pakistan.[27] Those cities had a grid street layout with street hierarchy, houses were built in a way such that they were protected by the noise and many houses had their own water wells.[11]

#### 2.1.2 Classical

One of the first important characters for city planning is Hippodamus, a Greek philosopher who lived in the 5th century BC, first town planner and inventor of urban layout, also denoted as *father of city planning*[1]. His thoughts were based on having a city planned with an orthogonal urban layout divided into squared street blocks. Even though Hippodamus was credited to be the inventor of this layout, Egyptian retrievals, which also had similar layout, shows that this idea was already taken into consideration in earlier ages.[26] The Hippodamian plan was first applied in Greek cities, when the Greeks found colonies also those were to adopt the

Hippodamian plan. The most valuable example of idealized urban planning in the ancient Hellenistic world could be found along the coast of the Mediterranean sea where the mouth of Nile is located. Alexander the great ordered the construction of Alexandria to follow the traditional Hippodamus plan.

Not only Greek examples but also Hellenic examples inspired the Romans which themselves adopted regular orthogonal structures.[24] Romans based their schemes for city planning focusing on two different aspects which are the military defense and the civil convenience. The core of the scheme expected a central forum with city services with a rectilinear grid streets layout, everything contained in defensive walls. Streets were designed in order to avoid long travel times, two diagonal streets crossed the squared street grid passing through the central square. Moreover a river was usually present, it was meant to go through the city to provide water and transportation.[34] An example of town that preserves these characteristics is Turin. The logical scheme that Romans adopted expected the typical streets at right angles in a square grid which have the same width and length, with the exception of two roads which go respectively north-south and east-west that intersect in the middle to form the grid center. The presence of bridges where needed and the division of the city in insula. The insula is a city block formed by the division of four roads, it has a squared shape and typically was of 73 meters squared. Originally filled with buildings and alleys. The presence of defensive walls on the border of the city was also important in the Roman scheme in order to defend the city from attacks. At the point where the main roads and the walls collide there were gateways with watchtowers which were also placed along the walls for all the perimeter of the city. The surrounding area outside the walls was often used for farming purposes. These cultures were not the only one present with urban planning characteristics, also urban settlements of Celtic origin were present.[12]

The fall of urban culture comes with the fall of the West Roman empire, for 5 centuries not much was preserved in western and central Europe. Between 10th and 11th centuries urban culture starts to grow again thanks to political stability and economic improvement. Because of the population growth the urban culture was expanded through the extension of existing settlements and the creation of new towns.[2]

### 2.1.3 Middle Ages

In early Middle Ages urban development was characterized by fortresses. Usually the center of the city was placed on high ground while the rest of the city would mold following an organic plan. This fact is what brought irregularities of elevation contours. As previously mentioned starting from the 10th century we registered an increment in urban development, in particular already existing cities gets expanded and new one formed. This growth is what influenced the current situation on the shaping of Europe graphical structures. Most of the new towns did not grow much though, with the exception of few like Munich, Bern, Berlin and some more.[6] From the current state of the towns it can be deduced that back then they were carefully planned. Often shows regularity in the plan scheme with straight streets and houses of original same rectangular size.[3] An interesting example would be Elburg from the Netherlands. Funded in 1392 by Arent toe Bocop it preserved his scheme for long time. Figure 2.1<sup>1</sup> shows the scheme of the city based on the situation in 1830.

---

<sup>1</sup>By Wikiwim2000 CC BY 3.0 (<http://creativecommons.org/licenses/by/3.0>), via Wikimedia Commons, [https://commons.wikimedia.org/wiki/File%3AElburg\\_after\\_cadastral\\_plan\\_1830.jpg](https://commons.wikimedia.org/wiki/File%3AElburg_after_cadastral_plan_1830.jpg)

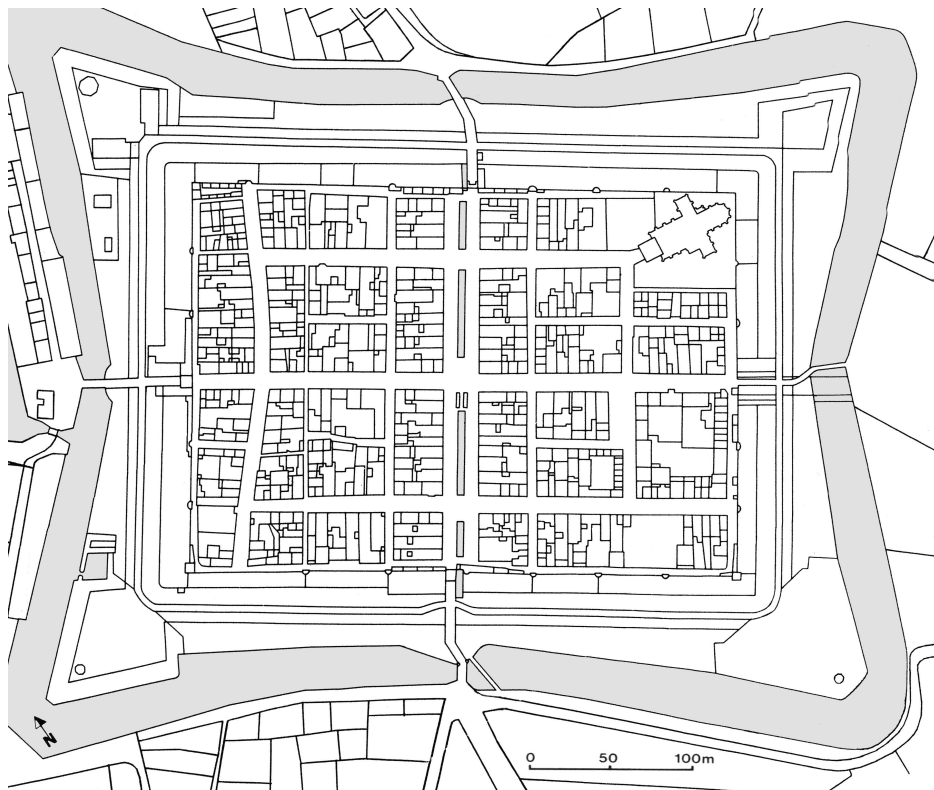


Figure 2.1. Elburg town scheme in the Netherlands - 1830.

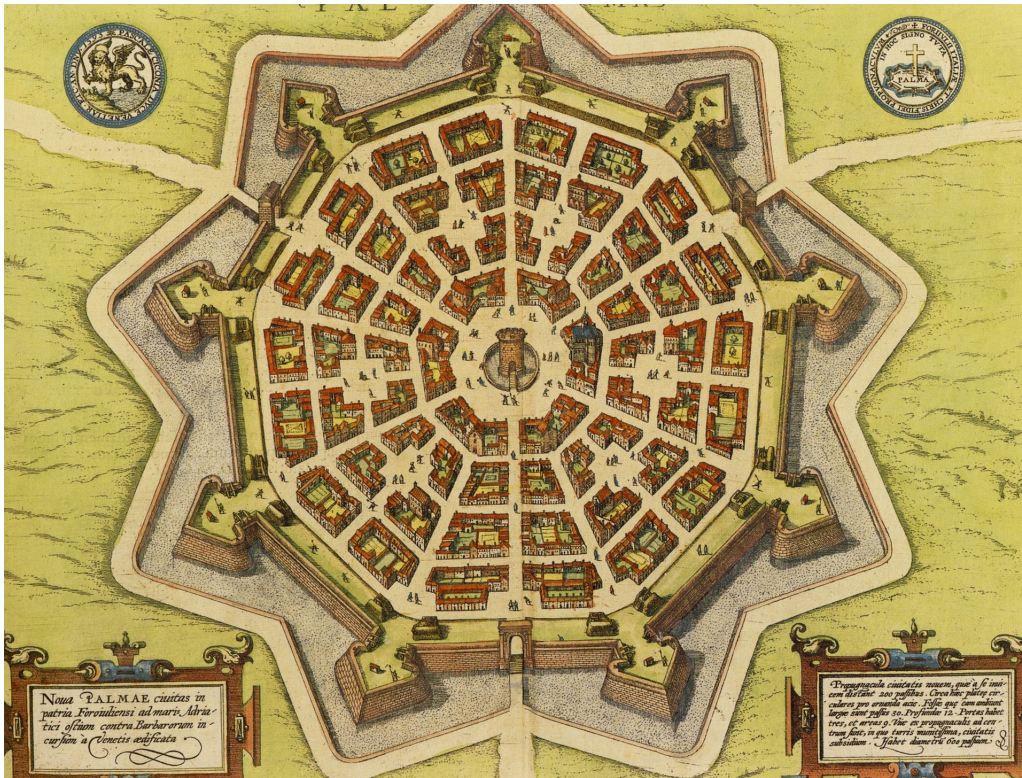


Figure 2.2. Palmanova representation of the seventeenth century.

After the Middle Ages comes the Renaissance with the impossibility to keep straight streets and in general the symmetrical orthogonal town plan. Therefore arise the deep depression around the 14th century which puts an end to the period of great urban expansion.

#### 2.1.4 Renaissance

One of the earliest model of new urban planning can be found in Florence, a star shaped layout for the star fort in order to resist the attack of cannons.[19] At that time it was widely imitated due to the cultural power of Florence. The Renaissance focused for one century and half on the same type, the star shaped city. The model was based on a center usually of military, communal or spiritual purposes outward which streets radially extended. In the ideal cities a building was in the center. An example of ideal city is Filarete's ideal city, building on Battista Alberti's *De reaedificatoria*, named Sforzinda. The city was thought to have a twelve pointed shape circumscribed by a circle. This process occurred in cities but not in industrial suburbs which were characterized by crowding and organic growth. Figure 2.2 shows a classical example of star fort based city. In this case we are looking at a representation of Palmanova of the seventeenth century, a town in the northeastern Italy, which nowadays still preserve the star shaped scheme.

After the bombardment of Brussels in 1695 when a large part of the city center was destroyed, Governor Max Emanuel proposed to reconstruct the city and completely change the



layout and architecture of the town. The idea was to have straight streets, uniform size buildings and a logical layout for streets. The residents and municipal though opposed and opted for a quick reconstruction of the city. In the reconstruction the old style was preserved but not kept identical. Despite the rapid work and the lack of financial means, several measures to improve traffic flow, sanitation and aesthetics of the city were taken.

### 2.1.5 Enlightenment

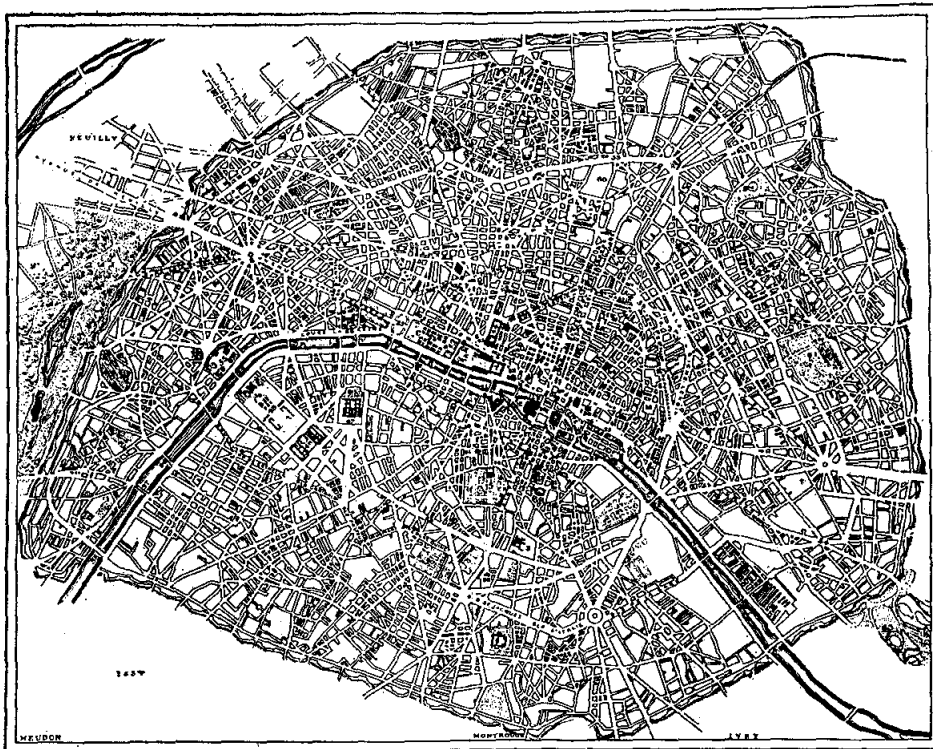
After the Renaissance began the time of the Enlightenment of Europe. This was a period symbol of attempts to redesign the capital cities as a showpiece for the nation. Often disasters were exploited to put in practice reconstructions following new schemes designed for the city and improves the city itself. An example can be found looking at London when in 1666 the great fire took place. After the fire many plans to change the city were proposed but no large scale redesign was achieved. Even though not many redesign took place there were made improvements on hygiene and fire safety. A different example is the earthquake in Lisbon in 1755.[31] King Joseph 1 of Portugal immediately rebuilt the city but following a new scheme which was based on razing entire sections of the city and build new streets. The reason behind this choice was that he wanted a new and perfectly ordered city. Similarly in Paris in 1852, Baron Georges-Eugene Haussman remodeled the Medieval street plan of the city. He removed old quarters and laid out a boulevard extending the old city limits further. He took into consideration both aspects of urban planning, the city center and the surrounding areas. He imposed regulations on building facades, public parks, waterworks, city facilities and public monuments.[20] Figure 2.3 shows how the scheme is structured for the town of Paris.

Another example worth mentioning is the plan to extend Barcelona. After the erasure of the city walls in 1854, Ildefons Cerda a Catalan engineer drew a plan based on scientific analysis of the city and modern needs. Credited for forging the term *urbanization* he tried to improve the health of inhabitants aiming to maximize the sunlight they received and assist social integration.[9]

### 2.1.6 Modern Urban Planning

In the 20th century we assist a paradigm shift and enter the time of the modern urban planning. Cities of the 19th century had grown a lot with the building style adapting according to private business concerns. The laissez-faire style of the government management of the economy gave start to a new liberalism that was at benefits of the poor and disadvantaged population. Around 1900 it started the development of urban planning to mitigate the consequences of the industrial age by providing citizen, especially workers, healthier environments.

The first major urban theorist was Ebenezer Howard, father of the garden city movement which took place in 1898. The core idea of this movement was to decentralize the working environment from the center of cities in order to provide a healthier living zone for workers. The core characteristics of the garden cities were the complete planning of the city, the decision to have communities surrounded by parks and the clear separation between residential areas, industrial areas and agricultural areas. These concepts were explained in a book published in 1898 by Ebenezer Howard titled *Garden Cities of To-morrow*, considered to be the most important book in the history of urban planning. The ideal garden city expected to have 32000 people in 6000 acres with a concentric patter and the presence of public parks. Transportation was characterized by six radial boulevards extending from the center. The garden city was



PLAN OF MODERN PARIS.

Figure 2.3. Modern town Paris scheme.

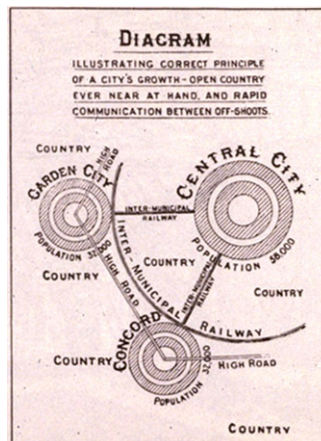


Figure 2.4. Ebenezer Howard's diagram for urban growth through garden cities - 1902.

planned to be self sufficient and once filled a new one would have been built nearby. The idea was to have a main garden city with satellite garden cities which would have been linked by roads and rails.[21] Figure 2.4 shows Ebenezer Howard's diagram used by himself to explain the concept that he had in mind for garden cities.

In the early 1900s urban planning finally becomes recognized as a profession. The first university course in urban planning was given by the University of Liverpool in 1909.[18] Also in the same year the *Town Planning Act of 1909* is written. It was meant to exploit the principles of garden cities to define a system of town planning for the country and to ensure specific building standards. In 1910 Thomas Adams becomes the first person being appointed as Town Planning Inspector. Four years later the Town Planning Institute is funded in order to study town planning and civic design. After World War One the concept of garden towns gets revised by Raymond Unwin who thought that the main city should be a garden city but the peripheral satellites were not supposed to be concrete and complete garden cities. In fact his idea was to have the main city to satisfy every need that the population had and make the smaller satellites benefit the services from the main city.[23]

After few years, starting from 1920, modernism ideas in urban planning started to rise. In 1922 Le Corbusier presented a scheme for a *Contemporary City* for three million people. The plan consisted in having six cruciform skyscrapers with walls of glasses meant to be used as offices at the center of the city. Those skyscrapers should have been set in large green areas. Always at the center of the city a huge transportation hub was planned, with highways inter-sections and at the top an airport. In order to enhance security and transportation efficiency the pedestrian were moved away from roadways. Starting from the center moving outward in a zig zag scheme blocks apartments were designed far back from the streets in green zones.[16] An important character for these years was Patrick Geddes who understood the importance of regional environment and relationship between social issues and town planning. He foresaw the emergence of huge urban conurbations. In 1927 he was commissioned to plan Tel Aviv. The plan he pulled out expected 40 blocks of 150 meters squared each with an inner small public garden. The roads to access was planned to be in a windmill configuration which resulted to be confusing for car traffic. The windmill configuration gave an undulating pattern to the streets, which exploited the north-south axis for commercial purposes and the east-west axis to catch

the sea breeze. Boulevards short cuts the system with public garden accessing the sea front. Basically he found a simple but efficient way to modernized the classical fixed grid pattern. The plan revealed to be a success.

After World War Two ended, Ebenezer Howard's planning concept gets adopted in large scale. For the first time with the reconstruction of London the decentralization issue gets addressed. The displacement of population and the employment was necessary to rebuild with a desired density. Over one million people got displaced into satellite suburbs, rural towns and new towns.[25] New cities brought housing improvement with the consequence of road constructions and comprehensive urban renewal projects. United States and many other European states had success with new towns as well.

### 2.1.7 Urban Crisis

In early 1970s modern schemes were sapping vitality from the community and therefore got blamed for multiple problems and began the decline of modernist planning. The construction of cheap uniform tower blocks ended in most countries. Some of these kind of buildings got replaced with other building types.[32] At this point planning was focusing on individualism and diversity in society and economy.

### 2.1.8 New Urban Planning

Nowadays movements in urban design aim to create sustainable urban environments with long lasting structures, buildings and great livability. The New Urbanism is based on the walkable urbanism form, how friendly an area is to walking. This approach aims to successfully reduce the environmental impacts by altering the built environment to create and preserve cities that support sustainable transports. Designed such that few driving miles are required for people living in urban neighborhoods in order to have a lower environmental impact with respect to those who lives in sprawling suburbs.[17] It also embraced the adoption of *Circular flow land use management* to promote sustainable land use patterns. In sustainable construction, a recent movement is worth mentioning, *New Classical Architecture* which is based on smart growth, walkability, architectural tradition and design.[22] It is in contrast with modernist, globally uniform architecture and opposing solitary housing estates and suburban sprawl.[29] New Urbanism is criticized for focusing too much on transport and excessive individual mobility. The real problem is about the entire urban metabolism of the city, while auto-mobility is less than the half of the overall ecological footprint. It is argued that land use planning can do little to achieve sustainability without regulating the design and technologies. The distances and the density are not important, the total metabolism of the development is what determines the environmental impact. At this point the emphasis is moved from sustainability to resilience, with a spacial scope from city to the urban region.[8] The action to compact the urban form is a difficult and slow process. In the end with the new global situation, the low density growth and the climate change going on, it would be wiser to focus on resilience of whole cities regions retrofitting the existing sprawl for sustainability and self efficiency and investigating in green infrastructures.[7]

## 2.2 Current tools to visualize cities

This section goal is to find what are the interesting features that a tool, which creates 3D interactive cities representations, should provide. Already existing tools which provide cities representations are available. In this section we are going to introduce some of these tools with the aim to compare them with VETRUVIO.

Handling data relative to cities is not an easy task. In order to make this activity more flexible and reduce the effort required most cities started moving data from paper into a digital format. Digital format allows data to be managed by computers, drastically reducing human effort. Digital formats can be used to inject data into tools. *Esri*<sup>2</sup> and *AutoCAD*<sup>3</sup> provide specifics over digital formats, allowing creation of files that present a predefined structure which is thought to be read by designed tools. There are different representations that can be provided by those tools. Most common representations are graphical cities models which can either be in two or three dimensions. Since VETRUVIO provides a 3D city model, we are going to focus on tools which provides a similar visualization.

The first program that we want to look at is *ViziCities*<sup>4</sup>. *ViziCities* is a free and open source project that allows the visualization of cities in a 3D model. It is designed to run in a browser, developed in JavaScript with the support of WebGL via Three.js a popular 3D visualization library which provides representation to the data. Data is gathered from OpenStreetMap, a free mapping service and other databases. In the city simulation it is possible to move around and extrapolate information about the surrounding environment. The London example provides an overview over available interactions. Examples of provided interactions are:

- Feedback over London subways, provided in real time.
- Flooding danger zones can be highlighted.
- Crimes based on districts.

*ViziCities* is a project which provides static representation of cities with visual feedback. Right now *ViziCities* does not allow, as a user, to inject data into the map.

*ArcGIS*<sup>5</sup> is another project which provides graphical representation to digital data. *ArcGIS* is a platform developed by *Esri* that allows the creation of maps and 3D models which can be shared. There are several actions that can be performed on maps created by *ArcGIS*. The main purpose to provide graphical representation of traffic patterns, demographics, customers and competitive data. The final target of *ArcGIS* is to allow the user to perform analysis based on current data and predictive modeling.

There are several entities which adopts *ArcGIS*, one of those is Geneva city<sup>6</sup>. From the simulation of the city it is possible to interact with buildings and get information that qualifies them. No further interaction is available. This makes analysis difficult to be performed.

In our opinion the tools presented have great potential which is not completely exploited. The fact that entities like Geneva invest resources in the city representation means that there is interest in these tools. We believe that the limited interaction and impossibility to inject data reduces the potential of the tool. VETRUVIO aims to provide both graphical city representation and interactions with its components, allowing modification of building information.

---

<sup>2</sup><http://www.esri.com/>

<sup>3</sup><http://www.autodesk.it/products/autocad/overview>

<sup>4</sup><http://vizicities.com/>

<sup>5</sup><http://www.arcgis.com>

<sup>6</sup><http://www.arcgis.com/apps/CEWebViewer/viewer.html?3dWebScene=37db2aa792b94cb08f19d2f5ce230da3>



## Chapter 3

# Vetruvio

This chapter presents VETRVIO, a tool which provides interactive 3D cities representations. Environment and framework involved in the development of the tool are discussed in a first section. Afterwards a brief introduction over Woden, VETRVIO graphic engine, is provided. An alternative implementation of the tool, which uses OpenStreetMap, is shown and explained why discarded. Finally the latest version of the tool is presented and discussed.

### 3.1 Environment and frameworks

The environment that has been selected to develop VETRVIO is Pharo, a Smalltalk dialect. Pharo is an open source implementation of the programming language and environment Smalltalk. One of the main characteristics of Pharo is strong live programming. This feature provides immediate object manipulation, live update and hot compilation.[5][4]

The main factor that led to the choice of Pharo out of the vast pool of programming languages resides in VETRVIO needs. The aim is to provide a city representation that is manageable at any time. Thanks to its strong live programming Pharo allows manipulation of objects at any time. Although Pharo offers strong live programming there are other aspects that influenced the choice. It is a technology which is in continuous evolution, provides the opportunity to inspect objects and provides a powerful debugger. The Pharo community is active and willing to interact with its members. Constant support is provided through the mailing list in case of needs over relevant developed projects. On the other hand we had to consider drawbacks of this technology. Even though the community is active it is pretty small, if compared with other communities. The framework and tools used in Pharo, like Pharo itself, are under constant evolution and it might happen that sudden changes in a framework have negative impact on other packages which depends on it. These issues undermine the entire system stability.

In the end we decided for Pharo since we evaluated the positive factors being worth the disadvantages.

VETRVIO has been developed with the use of Woden as a graphic engine. Woden is an external package that allows the 3D creation of objects in OpenGL. Woden offers many features among which there is no possibility to draw polygons. Therefore an extension for Woden in order to draw convex polygon with the aim to create accurate building representations was planned. Eventually the possibility to represent the exact building perimeter has been

implemented. The implementation of convex polygons for the top and bottom side of the shape have been avoided due to time issues.

Another relevant package is Spec. Spec is a project that works on top of Morph and allows graphical interface creation with relative little effort.[30] In VETRUVIO it has been used to create interaction menus.

Fuel is an object-serialization framework. It takes objects and serializes them into fuel format.[13][14] In this way it is possible to achieve data persistence. When data is serialized with fuel a file with the information of the objects is created. This file contains information about the serialized object. When the file is deserialized the data relative to what was previously serialized is moved to the environment as its original state was. Fuel is used in VETRUVIO to provide a persistent format for buildings and bus stops data.

XMLSupport<sup>1</sup> is a relevant framework used in VETRUVIO. It is a project that handles the reading and parsing of an xml file. It takes an xml file which is parsed and creates a document object that contains elements objects representing the various components of the xml file. In VETRUVIO it allows to read the input file, which is an xml file that respects Esri specifications[15], and extracts buildings data.

Other packages have been used for different purposes. OSProcess<sup>2</sup> is a package which allows to send messages to the terminal and execute commands. It is used in VETRUVIO to open google maps in a browser with a mark on the selected building coordinates. NeoJSON<sup>3</sup> is a package that allows to read and produce JSON files. It is used to read the answer provided by OpenStreetMap servers when request of data is made.

## 3.2 Woden

Woden<sup>4</sup> is a 3D graphic engine for Pharo in continuous evolution. It has been chosen as graphic engine to draw the city as a 3D model in VETRUVIO. It allows the creation of predefined 3D shapes in an efficient way.

Although the presence of many features in Woden the possibility to draw polygons was not provided. Therefore it has been extended allowing the representation of real buildings perimeters. The drawing of both top and bottom face of the shape was avoided due to time issues. In the beginning the idea was to draw the exact perimeter of the building, but with the current state of the framework it is impossible due to performance issues. In fact Woden is not optimized to handle several objects with different shapes. When several objects that have different shapes are drawn, it becomes impossible to navigate through the city model due to massive latency.

These issues led us to the decision which opts for an approximative representation. Buildings shape has been designed as a bounding box of the real perimeter.

---

<sup>1</sup><http://www.squeaksource.com/XMLSupport/>

<sup>2</sup><http://pharo.gemtalksystems.com/book/PharoTools/OSProcess/>

<sup>3</sup><https://ci.inria.fr/pharo-contribution/view/Books/job/EnterprisePharoBook/lastSuccessfulBuild/artifact/NeoJSON/NeoJSON.pdf>

<sup>4</sup><http://woden.ronie.cl/>





Figure 3.1. Representation of the Triumphal Arch area in Paris.

### 3.3 First Attempt: OSM 3D city model

A first attempt was designed to involve OpenStreetMap<sup>5</sup> as a database from where buildings data is taken. OpenStreetMap is a project which has the goal to create a free editable map of the entire world. In this map there is information available for several buildings in the most known places. The data collected in OpenStreetMap is at disposal for free. The data we are interested to inject in VETRUVIO has to provide buildings characteristics. OpenStreetMap provides data relative to buildings perimeter but does not provide any data relative to heights. Only recently OpenStreetMap started to build 3D representations of cities. The data at disposal about this project and buildings is restricted to a small area.

Since this version uses real buildings perimeters as representation, it is not possible to draw a relevant area of a city due to a performance issue. This implementation is still available in the project. It can be run giving as input coordinates of any place in the world, if OpenStreetMaps provides data it displays the buildings. Figure 3.1 shows an example of a representation with OpenStreetMap data. The image shows the surrounding area of the triumph arch in Paris.

Due to performance issues, as discussed in the previous section, it is not possible to represent a massive number of buildings. The result, even though only a small amount of buildings is drawn, is notable. It is really precise with a small margin of error.

<sup>5</sup><http://www.openstreetmap.org/>

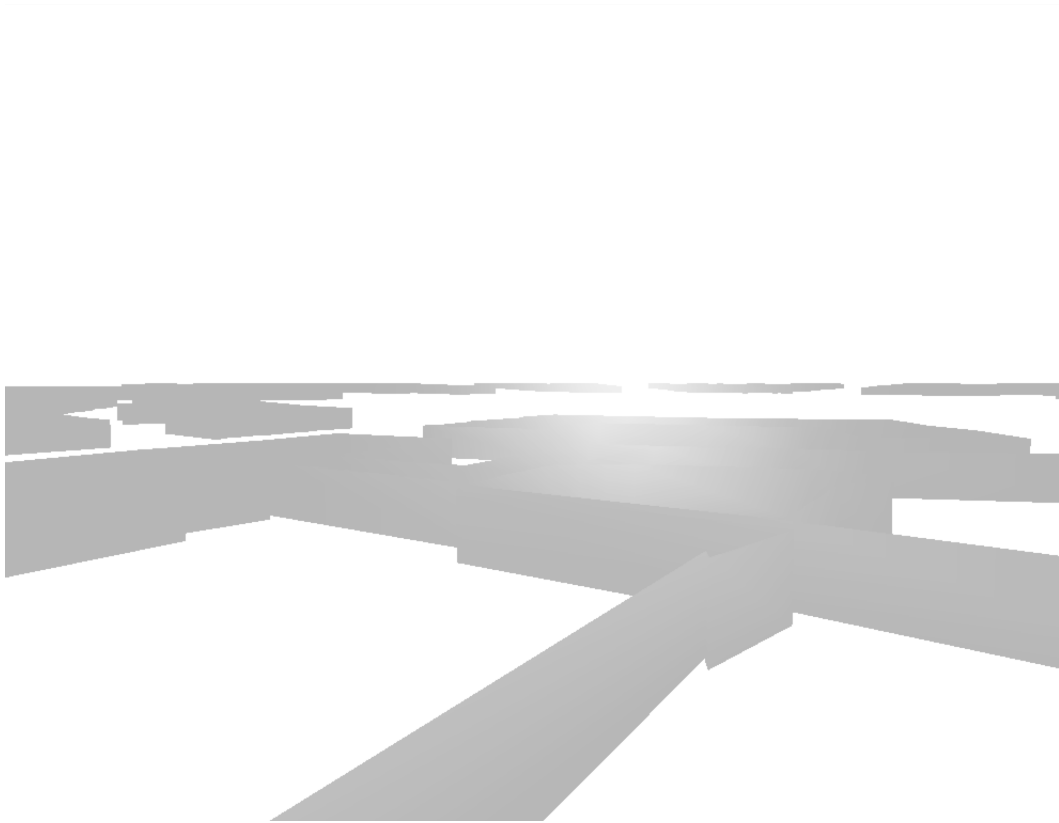


Figure 3.2. Triumphal Arch view almost at ground level.

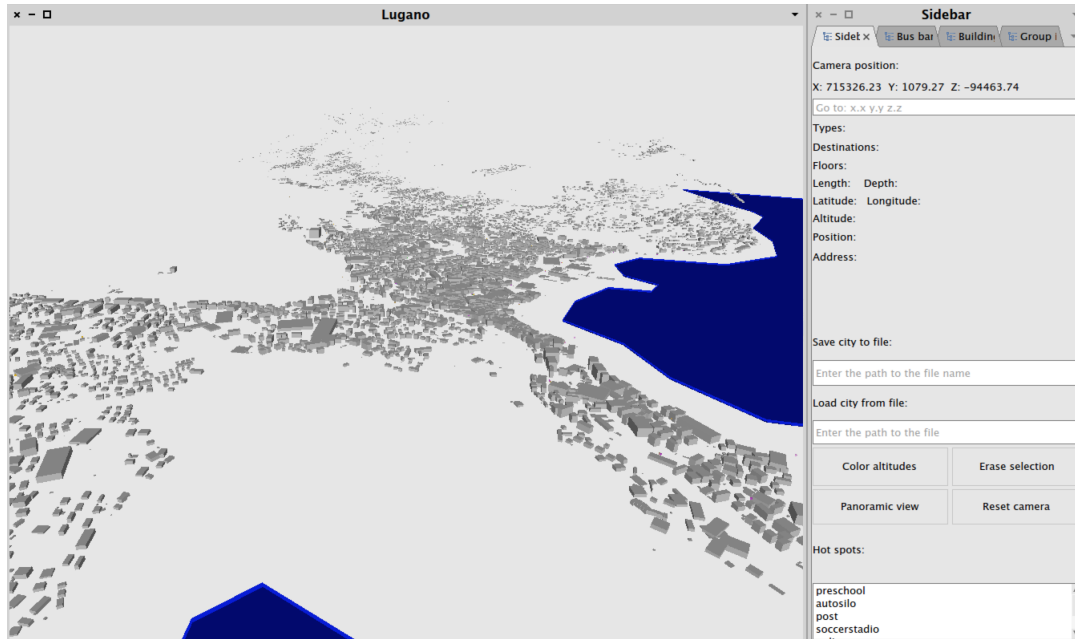


Figure 3.3. Vetrivio starting screen.

Figure 3.2 shows a snapshot taken almost at ground level. The distance to buildings with respect to figure 3.1 is drastically reduced. When too close to buildings a sense of confusion arises. It is difficult to distinguish the different buildings. This uncertainty is enhanced by the fact that every building has same height. Since the data is not available the same default value is imposed to every building.

Finally we concluded that this was not the implementation which VETRUVIO should sustain. The impossibility to draw large numbers of buildings and the lack of data imposed us to evaluate other options.

### 3.4 Vetrivio, the final prototype

Since the first attempt using OpenStreetMap<sup>6</sup> cannot provide enough data to create an interesting city representation a drastic change for what concern the data type has been made. In order to have more data about buildings to work with the option of Esri xml file, provided by Lugano city with data relative to the town buildings, has been accepted. Figure 3.3 shows how the final prototype of VETRUVIO looks like. The main window where the city is shown can be found on the left, alongside the tab based interface.

The upcoming sections illustrate VETRUVIO and its development. The first part proposes the VETRUVIO architecture. Afterwards a section that presents some data types encountered during the tool development is present. Followed by a description of the interface provided to the user. Finally it concludes with the analysis of potentials and limits of the tool.

<sup>6</sup><http://www.openstreetmap.org/>

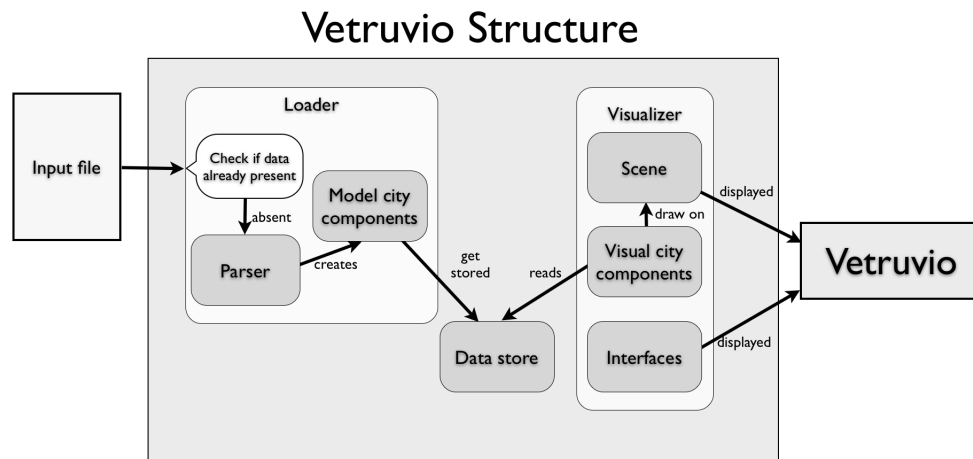


Figure 3.4. Vetrivio architecture scheme.

#### 3.4.1 The structure

The structure that VETRUVIO implements is composed by two main entities. The loader which handles loading operations of data into the system, and the visualizer which manages the visualization and interactions. Figure 3.4 shows a simplified version of VETRUVIO architecture. The input file is taken into the loader which check if data is already loaded into VETRUVIO. If no data is stored into the system the information present in the file is extracted and saved into a singleton container. The saved data is a list of building models. If the container had already data stored in it the loader avoids to retrieve the data from the file since it is a time expensive operation. Once data is available the visualizer takes it and represent it into visual objects which are drawn on the canvas. Finally it produces the user interfaces and everything is displayed to the user.

A more detailed overview of the structure can be provided by looking at the objects that compose VETRUVIO. These objects can be divided into groups:

- City loader.
- Model city components.
- Visual city components.
- City visualizer.
- Interface components.
- Utility objects.

City loader contains classes which are used to read and load data into the system. The loader takes a path to a file as input. If persistent data is already loaded into the system the loading phase is prevented in order to avoid waste of time. On the other hand, if there is no persistent data already into the Pharo image, data is loaded from the given input file. An xml parser<sup>7</sup>

<sup>7</sup>XMLSupport Pharo library

reads the data and creates a structure with the retrieved data. The information which is extracted is relative to buildings perimeters and buildings heights. With this data building models are created. These models are stored in a singleton object into the Pharo image.

Model city components contain objects which represent the city buildings and bus stations. These objects are used as a mean to store and extract the data relative to the element. Building objects hold more data with respect to bus stations. The data that can be found in a building model is relative to the properties that qualify the building. Some of the default properties are: perimeter, depth, coordinates, usage of the building, number of floors. This data is organized into a dictionary. Bus stops objects contain little data, only position, coordinates, name, and line are present.

Models are used to create the visual city components. The visual components are objects which provides a 3D virtual representation of models. They are defined such that Woden simply takes them and add them to the scene. Differently from the models, the visual objects implement functionalities which allow the interaction with the object in the canvas. They communicate with the interface and between themselves, providing visual feedback to the user. Moreover they offer the possibility to modify the model of the element through the apposite interface.

The visualizer is the object responsible to visualize the scene. It takes models, creates the visual objects and represent them on the canvas. Furthermore, the visualizer communicates with the objects allowing interactions over the city. Finally the visualizer creates the interfaces which are allowed to communicate with it in order to provide commands from the user.

Interfaces are composed by several objects. With *Spec*, a Pharo package for windows creation, a tab based interface has been created, through which the user can interact with the city. The interface is divided into four different tabs: main, bus stops, single selection and group selection tab. Tabs have been divided based on functionalities. Every tab is composed by multiple interfaces components which are either a set of interface components or recipient for content. Content can assume different aspects. In *VETRUVIO* the most used are labels, buttons, input text and lists. Labels are used to provide feedback of alphanumeric data. Buttons allow users to run interactions. Input text are present when an interaction requires input. Lists are used when an interaction has multiple fixed inputs which have to be selected.

Finally in the utility objects group three distinct object types can be distinguished. Two of those are dedicated at the consistency of the data. Two singleton objects that contain respectively building models and bus stops models are defined. These objects provide an option to save the data with the use of *Fuel*, a binary serializer for Pharo. It is possible to load data from a fuel file format so that it is not required to wait several minutes to load data from an xml file. This feature allows with few operations to quickly save eventual modifications that have been made to the city. It is possible to save the actual state of the city model and load already existing city models. The third object that falls into this category is the converter. It converts data relative to the position of the elements from CH1903 to WGS84.[10]

### 3.4.2 Data gathering, reading and elaboration

During the developing process of *VETRUVIO* various data types have been encountered. This section explains the role of most relevant types in *VETRUVIO*, and eventual transformations performed on data.

The data type taken as input is of xml type. It is used in the process to load buildings data. The data respects the XML Schema of the Geodatabase proposed by Esri.[15] This specific data

format is dynamic, scheme structure can be differently configured. The structure encountered presented relevant data about buildings, perimeter and number of floors.

At the beginning of the file an header which describes the content is defined. It regulates fields and types as well as values range, tolerances in the coordinates system and other set ups parameters. After the header it follows a long record of elements which represent buildings. Structure and constraints, of record elements are defined in the header. The record presents several data, out of this data VETRUVIO finds use only on buildings perimeter and floors number. Other data has been ignored since qualified as meta data with no high relevance for the project.

The data is assimilated in VETRUVIO through the loader, which uses *XMLSupport* from Pharo to read and parse the xml file. The parser creates objects based on xml tags, producing a nested sequence of xml elements. Once the list of records is extracted it is possible to iterate through it and grab the desired data. Once the data is selected, for each record a new building model, which gets stored into a singleton container, is created.

CH1903 is another data type of relevance that has been elaborated during the development of VETRUVIO. CH1903 is the Swiss projection coordinates system. It uses an Oblique Mercator on a 1841 Bessel ellipsoid. [10] An Oblique Mercator is an oblique conformal cylinder projection.[35] Together with the 1841 Bessel ellipsoid, a reference ellipsoid of geodesy with base at the old observatory in Bern, gives meaning to the projected coordinates.[33] The transformation computations of the data from CH1903 to WGS84 is provided from the Swiss confederation. The precision with this transformation is of respectively 1 meter and 0.1". In order to derive the transformationa specific formulae has to be applied. The formulae can be found on the website of the Federal Office of Topography swisstopo<sup>8</sup>.

### 3.4.3 Provided interface

VETRUVIO aims to create interactive 3D cities representations. In order to have interactions that may suggest interest in users, it is required to have a flexible system which allows customization of objects. In order to achieve this feature in the most flexible way the access to the code should be provided. Since this is not a user friendly feature, a tab based interface is at disposal of the user. The proposed interface is composed by four tabs arranged based on functionalities.

The main interface shows information about buildings on mouse over. Actual position of the camera with the possibility to change it is also provided. Furthermore it offers two coloring scheme for the city, one colors based on altitudes, the other colors based on buildings destinations designed as hot spots.

The bus stops tab allows users to highlight different lines stops. This is possible through a provided list of available bus stops.

The single selection interface shows more detailed information about the selected building compared to the basic information displayed in the main tab. The detailed data contains indices of building data comparison. Moreover it allows to add the selected building to a group and offers interactions to find out where the closest points of interest are. Minor features are available and will be discussed more in details in use case scenarios along with a more detailed overview of the data presented in this interface.

The last interface is the group selection tab. It consider the multiple buildings selection and allows operations to be performed on top of them. It is possible to use selected groups as hot spots and color the rest of the city based on them. Otherwise it is possible to perform hot spots

---

<sup>8</sup><http://www.swisstopo.admin.ch/>

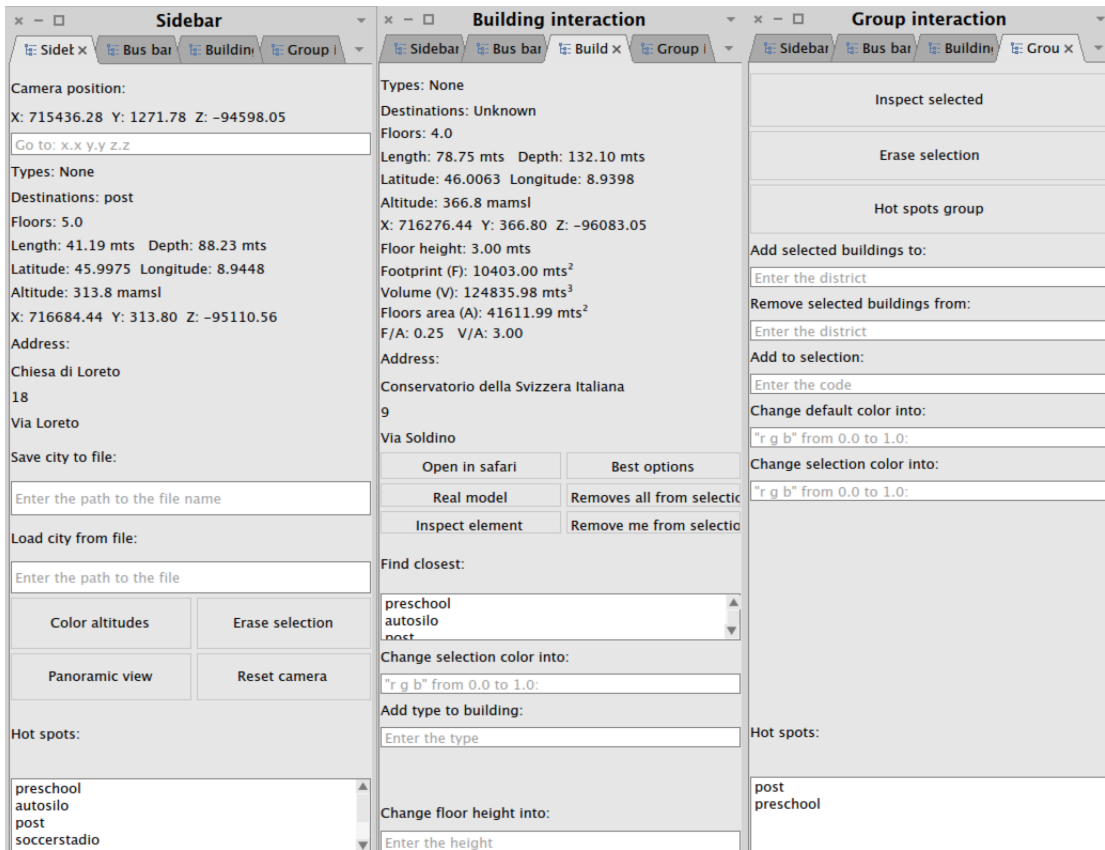


Figure 3.5. From left to right: main interface, single selection and group selection tab.

only on selected buildings with some predefined destinations acting as centers. Also allows the user to add or remove buildings from types and gives the possibility to customize selection and default color of buildings.

These interfaces are designed such that the possible interactions are provided to the user in a user friendly environment.

#### 3.4.4 Potential and limit

In our opinion the tool has great potential. Right now there are not many tools which represent cities in 3D and even less are able to allow interactions and injection of data. We think that the possibility to take plain alphanumeric data and gives it a virtual representation which provides visual feedback is the success key of VETRUVIO.

Right now VETRUVIO is a prototype which could be further developed. With some more implementations like a ground representation it could be possible to inject more data which is not related to buildings and bus stops, but also actual streets, rivers, bridges, green zones and tunnels could be included. Once other elements are available in the representation further interactions could be implemented raising the level of visual feedback provided by the tool.

The current limit of VETRUVIO is strictly related to data availability and to the graphical engine. We did not manage to find accurate data to inject in the example of Lugano, therefore we were limited to make interactions with the available data. Since not all data was available for each building, where data was missing we had to fill in with a predefined default value. The graphical engine is another limit since it does not allow an efficient 3D representation of polygons. This issue led us to turn down graphical accuracy and opt for a bounding box representation of buildings perimeter.



# Chapter 4

## Use Cases

In this chapter some of the possible actions that can be performed into VETRUVIO are presented. The aim of this chapter is to provide a practical introduction to VETRUVIO, through four different use cases. The first use case shows the available options to gain access to building data. In the second use case a description of the interactions available in the main interface tab is presented. In the third use case interactions over a selected building are discussed. Finally the last use case presents actions that can be computed over a group of buildings.

### 4.1 How to access information

The first use case introduces the available options to access building models data. There are different actions which provide access to stored data into buildings models.

The most intuitive and practical method is through mouse over event. Figure 4.1 shows two different interfaces. The left interface is the main tab interface. It contains data related to buildings, which becomes visible once the mouse goes over an object in the scene. The first field of the displayed data is denoted as *types*. It provides the list of groups in which the building is present. Groups are defined by the user and are used to categorize buildings which, conceptually, have characteristics in common. The next field in the data is denoted as *destinations*. It is used to represent the usage of the building. A building can have multiple destinations. Examples of destinations are: *post*, *firefighters*, *police* and *preschools*. The possibility to add other destinations is provided to the user. Other available data is the number of floors that the building possesses. The number of floors is displayed if available, otherwise a *N/A* is shown. The size of buildings is available for consultancy, it is measured in length and depth both expressed in meters. Real world coordinates are represented in latitude and longitude with decimal values. The altitude is expressed in meters above mean sea level and influences the position of the element in the space. The system coordinates which are in CH1903, influence the position on the x,y axis. Finally the data concerning the address of the street where the building is located. Address data is taken from OpenStreetMap.

The building model information can also be accessed through other channels. Another possibility is with building selection. When a building is selected the single selection tab is updated displaying detailed information about the model. The information displayed is more precise with respect to the previous case. Figure 4.1 shows two tabs present among the interfaces. The right interface is the selection tab. Among the detailed information different values and

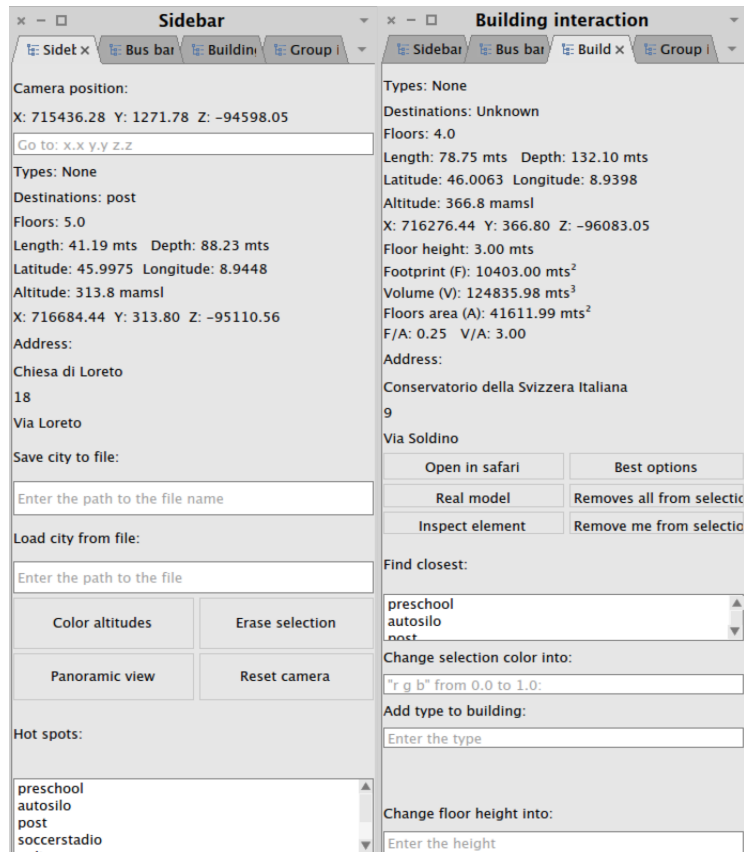


Figure 4.1. Main tab interface and single selection interface.

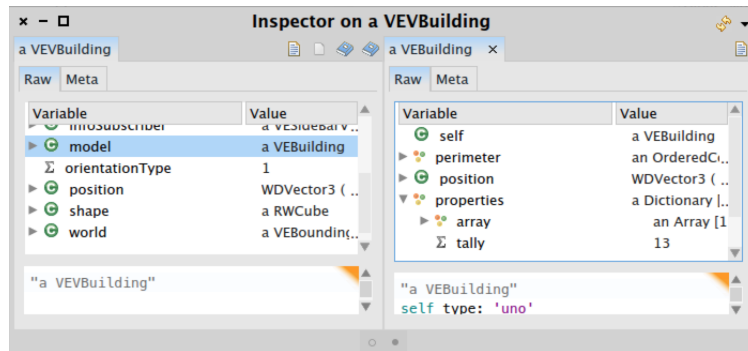


Figure 4.2. Pharo inspector over a building.

indices are found. Building floors height expressed in meters is one of those. Footprint, which is the area of the building floor, is expressed in squared meters. Building volume is presented in cube meters. The sum of the floors area in squared meters. In the case that any of this data is unavailable a *N/A* is found at the place of the real value. Similarly for the *F/A* index which represent the portion that the ground area covers compared to the total floors area. Finally the last value available is the *V/A* index which is obtained dividing the volume for the total area.

A third more accurate but less intuitive access for the data is through inspection of a selected item. Pharo allows the possibility to inspect elements and *VETRUVIO* uses this feature on both single and group selection. In order to inspect the currently selected object is enough to simply press the inspect element button in the building selection interface. In this way a window with the Pharo inspector on the currently selected building appears. Figure 4.2 shows how the inspector looks like. The inspector can be divided into two sections. The left side contains fields available in the inspected object. When a field is selected, on the right side, appear details about the inspected field. Depending on what is stored in the field we can have different visualizations. In the case proposed in figure 4.2 the model contains an object with fields. A slightly different visualization would have been provided in the case of a list or dictionary.

It is possible to run code directly from the inspector. This is possible from the small blank space below the specification of the object. The code directly interacts with the object. In this way it is possible to send messages to the object while it is alive. The navigation through the object and its components in the inspector window is possible. Clicking on a field opens the inspection on the field object, as shown in figure 4.3. In this case a dictionary containing properties of the building model is shown.

Similarly for group selection, pressing the button inspect selection opens an inspector window containing the list of selected items which can be navigated through.

These are the different approaches that can be taken in order to access the data stored into building objects. The first two approaches presented are thought for a simple values retrieval. The last shown example where we interact with the Pharo inspector, is more appropriate if a precise understanding and analysis of the object is required.

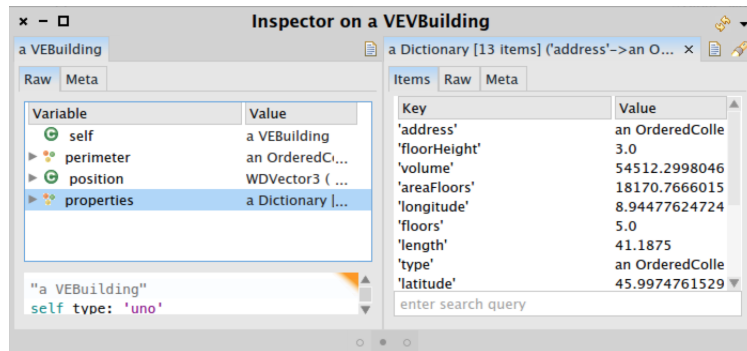


Figure 4.3. Pharo inspector on a building model with inspection on properties field.

## 4.2 General interactions

This use case aims to introduce some interactions that are available in VETRUVIO. In particular it highlights interactions that can be performed on cities from the main tab interface. The actions described in this use case target the entire city rather than specific objects.

Figure 4.5 shows the interface on the right side. It contains information relative to the camera position and building on which the mouse cursor is. The camera can be moved through two different actions. A first possibility is through the default keys when the focus is on the scene. *W*, *A*, *S*, *D*, *SPACE*, *Z* are the keys which allow the camera to move. Mouse click and drag changes the orientation of the camera. It is also possible to move the camera through the apposite text bar which can be found in the toolbar. If some coordinates are provided the camera would instantly move there.

It is possible to save the city components models. This through the apposite input bar present in the toolbar. It requires a full path to a folder and a file name (which typically ends with the extension *.fuel*). Data is going to be saved into the specified file. This feature allows to keep a copy of the city components models, which can be loaded later into the image. In order to load the saved file the user has to specify in the loading text bar the full path to the file that contains the information about the city model. The data in the specified file will be loaded into the Pharo image and relaunching VETRUVIO will display the city based on the loaded data.

Buttons which offers different interactions have been provided in the interface. The button that states *Color altitudes* colors buildings based on a logarithmic scale over altitudes. Figure 4.4 shows how the city looks like once the button is pressed.

*Erase selection* button is a default-state button. Its aim is to simply deselect eventual selected elements and erase eventual color applied on buildings. Another utility available to users is the possibility to reset the camera into the default position or to move it into a favorable position which provides a pleasant sight of the town. The *Panoramic view* button moves the camera high over the city in order to provide a better view over the surrounding area. The last button, *Reset camera* brings the camera to the initial position.

Figure 4.5 shows an example of color scheme applied to the city. The city is colored based on post offices. They act as hot spots while the remaining buildings become colored based on the distance to the closest hot spot. This is possible to be achieved by simply selecting, from the hot spots lists presented at the very bottom of the interface, the point of interest which should act as hot spot. The list contains all the different destinations present in the city. It is possible

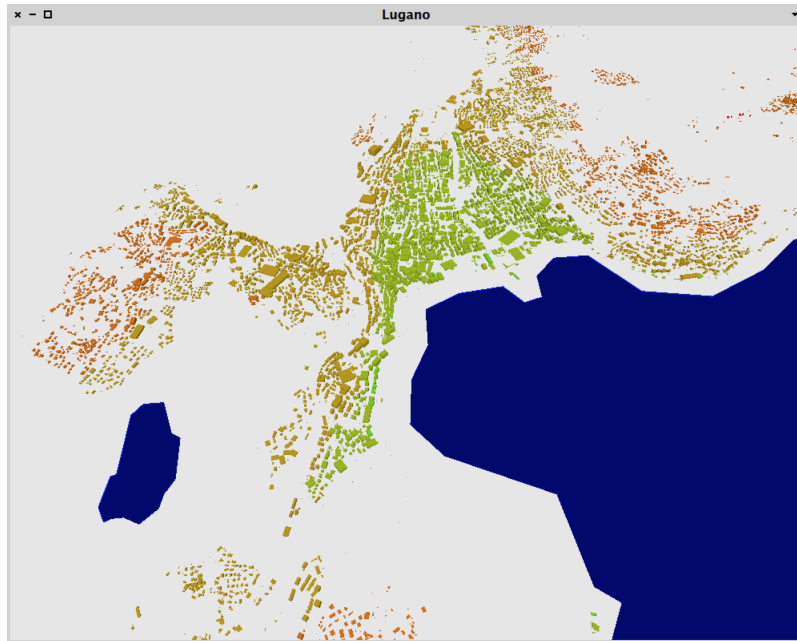


Figure 4.4. City colored based on altitudes.

to select multiple destinations present in the list and combine hot spots. This will produce hot zones where the proximity to the selected destinations is minimal.

### 4.3 Actions on single selection

This use case shows what actions can be performed on top of a selected building. The interactions proposed are accessible through the apposite single selection tab in the provided interface.

Figure 4.6 shows the single element selection interface alongside the scene, from which is possible to distinguish selected elements highlighted in yellow. The interface on the right contains data of the selected building and provides interactions that can be performed. The data presented is more specific with respect to the data of the previous case.

The interactions available on this interface target single buildings and not the entire city. Some of these features are provided through the use of buttons. The *Open in safari* button opens a safari browser window on google maps with a mark on the selected building coordinates. With this feature the location of the building on google maps is performed with a minimal effort. The *Best options* button highlights the closest building for each destination such that all destinations are present at least once. This feature allows to see where are the closest points of interest with respect to the current location. The *Real model* button offers the possibility to see how the real perimeter of the building looks like. As shown in figure 4.7 a window with the representation of the building perimeter is available. The button *Removes all from selection* resets all types of selection. Both group and single selection are erased as well as the eventual color on buildings, restoring it to its original state. The *Inspect element* button opens an inspector window of the visual building object. For more details about this utility

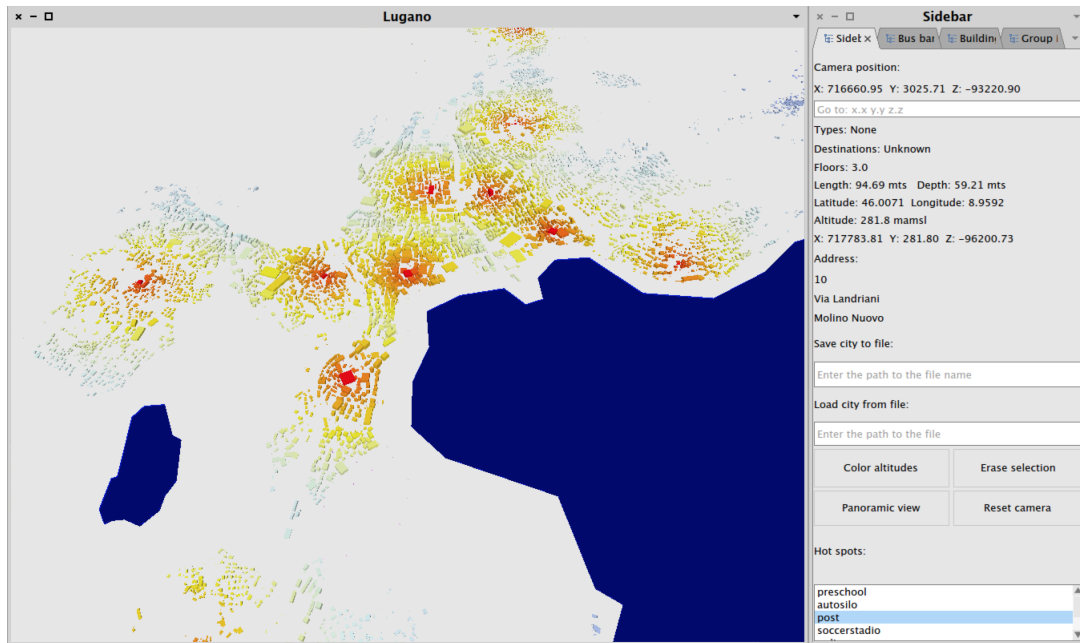


Figure 4.5. City colored based on proximity to post offices.

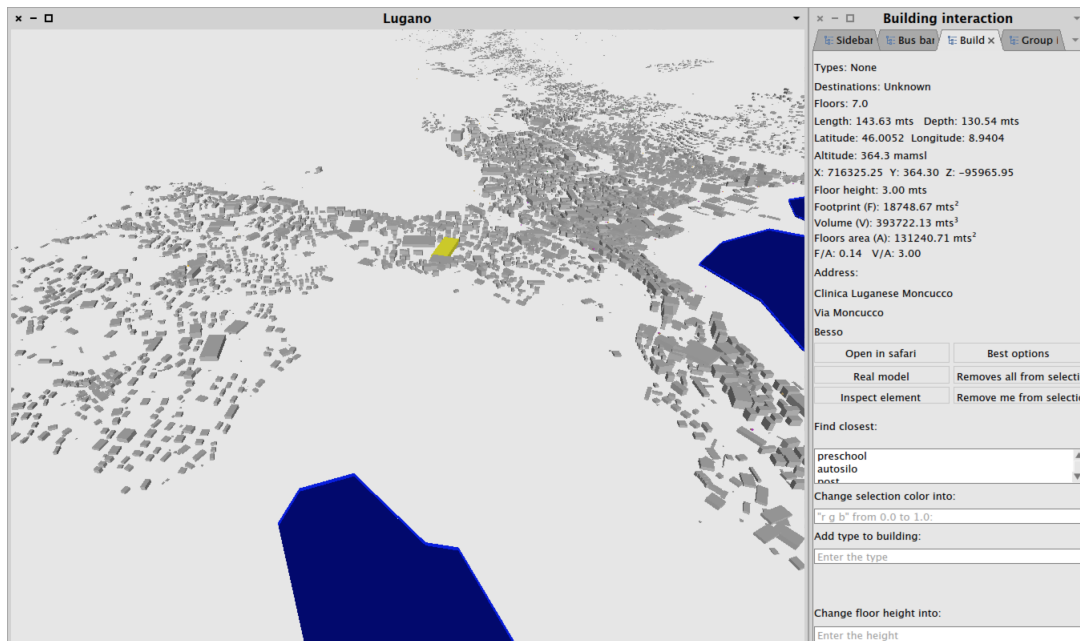


Figure 4.6. Single selection view.

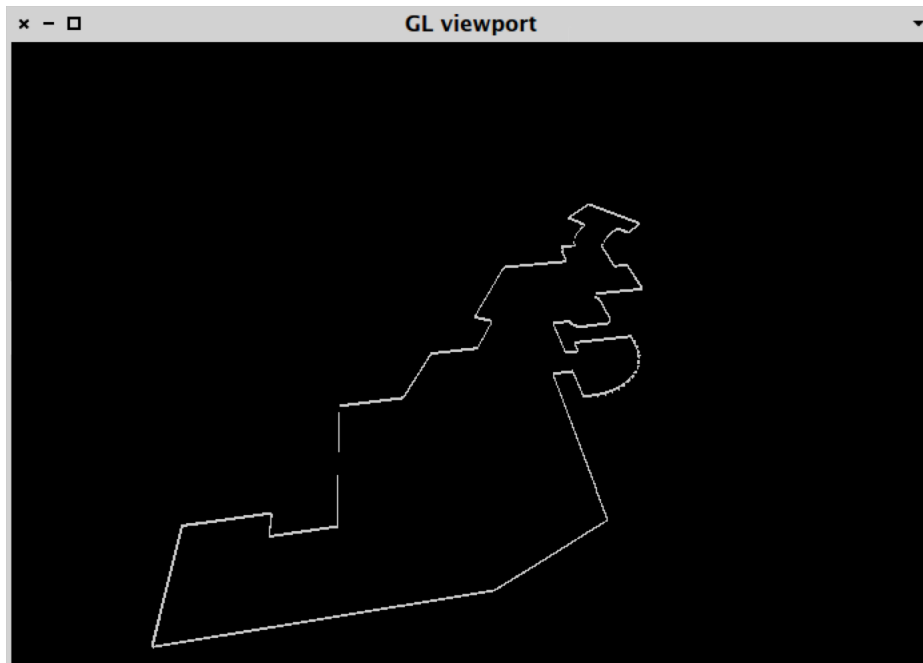


Figure 4.7. Real model of a selected building.

button refer to the first use case. The last button *Remove me from selection* removes only the current object selected from both single selection and group selection. It also restores the color of the building to the default one.

A different kind of interaction is possible through the list of destinations given in the toolbar. Selecting a destination or multiple of those, highlights the closest buildings to the selected element with at least one of the chosen destinations.

Furthermore is possible to change the color for the selection. It has to be specified in *RGB* format in range from 0 to 1. The possibility to add types to the building through the apposite input bar is also available. Type buildings are intended to be tags which denotes buildings, used to create groups which conceptually share characteristics. The last input bar has the aim to provide the user a quick method to change the building floor height. Since this value was not available in the gathered data, the default value of 3.0 has been set. Modifying the floor height makes the building shrinking or getting taller live.

## 4.4 Actions on group selection

This use case shows what is possible to perform with buildings groups and different actions which are provided in the group selection tab interface.

Figure 4.8 shows, on the right, the interface proposed to interact with buildings groups. On the left, there is a scene where it is possible to spot selected buildings, highlighted in yellow.

There are different interactions that can be performed with a selected buildings group. The first proposed is through the button called *Inspect selected*. Pressing the button a new Pharo inspector window opens with information related to the selected buildings. A more detailed

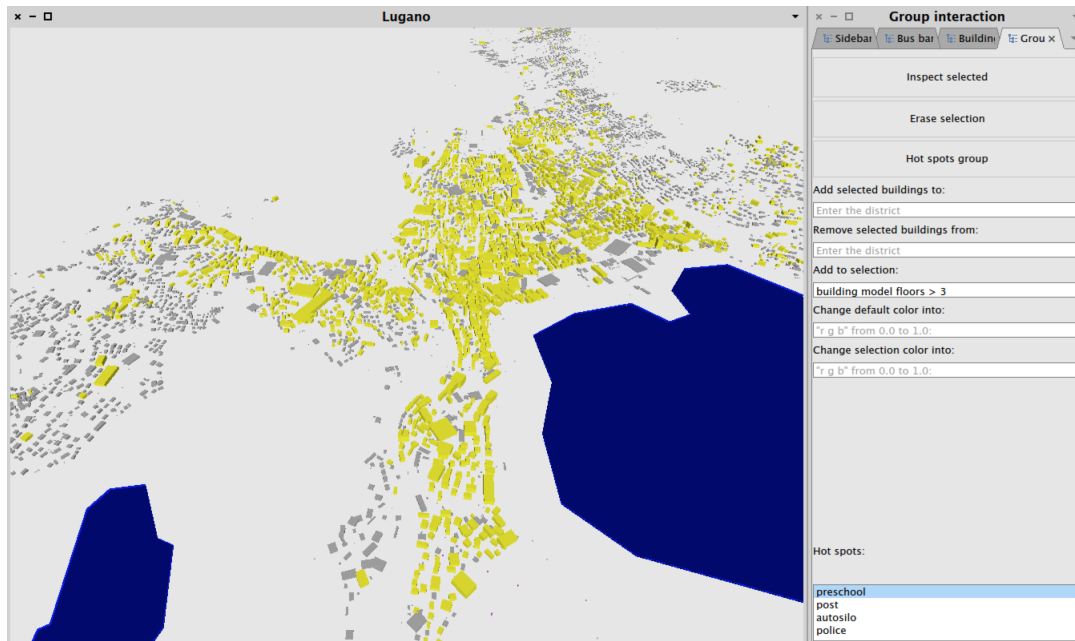


Figure 4.8. Scene with group selection over buildings with more than three floors.

explanation can be found in the first use case. A second action can be performed through the button that states *Erase selection*. Clicking the button performs a reset over the whole city. The buildings color is set back to default and selections are erased. The last button offers a different interaction which makes the selected buildings operate as hot spots. The city is colored based on distance from the closest of the designed hot spots. Figure 4.9 shows an output of the interaction. Buildings in red are hot spots which were previously selected. This example considered all buildings with a single floor to be hot spots.

A sequence of text input bars is proposed in order to provide input from the user. Each of these text input bars offers different actions to perform on the city. The first allows to assign a specific type to all selected buildings. For example in our case, as shown in figure 4.8, the selected buildings have more than three floors. Having these buildings selected it makes sense to assign a type that reflects their common characteristic. Therefore a string like "*Floors > 3*" would qualify this group of buildings properly. Similarly the next input bar takes a string representing a type present among buildings and removes the type from the selected buildings which own it.

The *Add to selection:* input bar offers the possibility to search and select among city buildings. Through a boolean expression it is possible to express constrains that a building has to respect in order to be selected. Figure 4.8 shows an example of code that can be applied in the search. The keyword to refer a building is *building*. It refers to the visual building object. Through the visual building object it is possible reach the data relative to the model of the building and access the different properties in it to compare with our constrains. The result is a set of buildings added to the current selection.

The next action which is possible to perform in this view is a simple change of default building color. If it is changed, when the reset of the city occurs, buildings will refer to the



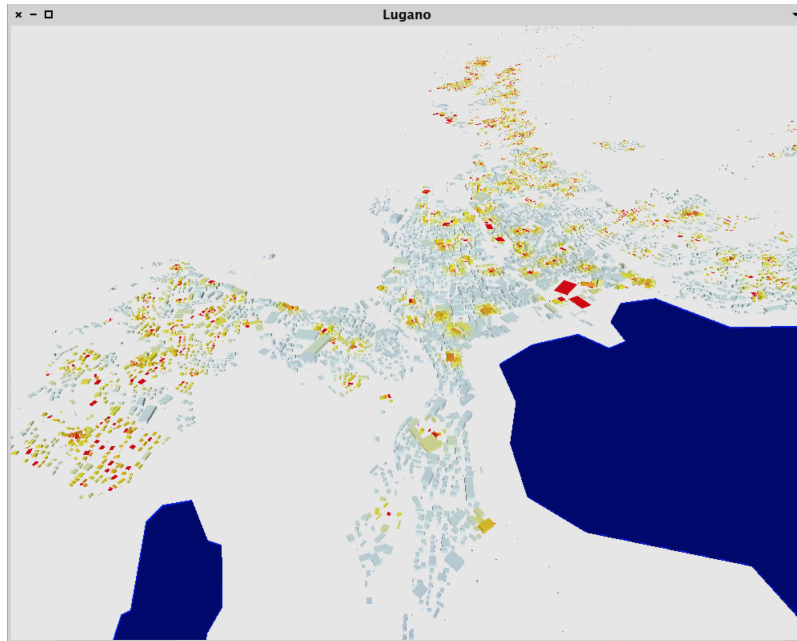


Figure 4.9. City colored based on proximity to buildings with a single floor.

new default color. This is not an information that is kept in the model but directly in the object hence when the city is saved this information is not going to be saved. Similarly the last input bar allows to change the building selection color. Through a *RGB* scale from 0 to 1. This information is not persistent either and every time *VETRUVIO* is ran it is going to get the standard default value.

At the very bottom a list of point of interest, which are present among the selected buildings, can be found. The hot spot algorithm can be ran focusing only on the selected buildings. The hot spots are the selected points of interests from the list and get a result similar to the hot spots interaction described in the second use case.



## Chapter 5

# Reflections

This chapter resumes external judgments toward VETRUVIO by third parties interviewed during the tool development. VETRUVIO has been proposed to third entities and opinions about the role that VETRUVIO could cover in a real work environment have been collected. The choice to show VETRUVIO has been made over three different targets. The aim was to target who could potentially be interested into an application of VETRUVIO. Moreover a person who already experienced the development of similar tools has been asked to express his thoughts about the project.

The first person whom VETRUVIO was presented to is a private architect. We thought private architects to be the main target, since VETRUVIO allows the modification of a city and provides interactions which through visual feedback facilitate the understanding of the city scheme. The project was presented with a walk through of features, trying to show the dynamism and interactions that VETRUVIO offers. As expected a first remark was about the lack of data relative to buildings. Not every building has precise data, therefore some values have an approximation to a default value. Another remark was concerned the precision with which buildings are represented. We discussed the possibility to graphically show buildings basis and suspended buildings. We concluded that, due to unavailability of data, both basis and suspensions, could not be represented. It also emerged the interest in having comparison of data among the single building. Different indices have been added such that values would be automatically computed when a building is selected. Finally we concluded that the lack of data is a huge burden for VETRUVIO and that in a case where more data is available the tool would potentially become interesting for a professional use.

The second entity approached was a Lugano city architect and a data IT responsible. From them emerged different remarks. Obviously also from their side the negative feeling about the missing data was perceived. It emerged that other data related to buildings, like heaters type, could be interesting to be put inside the model. They also expressed the interest to divide the city into sectors in order to assign restraints on the constructions (e.g., building height, distance between neighbors). The remark on which they were concerned the most was, surprisingly, about buildings aspect. They were expecting to have a precise and detailed representation with textures on at least buildings with major importance. With the combination of existing features it is possible to divide buildings belonging to the town into sectors. On the other hand it is not possible to provide yet an accurate representation of buildings due to the inefficiency of the graphic engine in drawing polygons. The conclusions were that the project

was undoubtedly interesting from a public entity point of view. The reorganization of data into a digital format that provides an interactive 3D city representation was appreciated. On the other hand we had a negative feedback over the graphical engine efficiency and lack of data.

The last meeting took place with someone who already experienced the development of a similar project. This person developed a tool to represent cities based on thermal activity. We have shown him the project looking for a feedback and talked about the problem of the missing data. The feeling about VETRUVIO was positive, recognized as an interesting project with potential. Also from this person side a negative remark toward the 3D drawing framework was made. We figured out that on both our side and his side emerged the problem of the lack of data which, for a program that has to highly rely on data, is a huge burden.

As a final reflection it is possible to state that the project was of interest from every party and that they agreed on the lack of data present in the model. Moreover, from each entity we got more specific remark related to their work environment and aimed to lead VETRUVIO toward their branch. In the end we can conclude that VETRUVIO has good potential as long as data is available.

# Chapter 6

## Conclusions

### 6.1 Summary

This thesis contributions can be summarized in:

- A 3D city representation in Pharo.
- Interactivity with cities and city elements exploiting live programming.
- Editable city components with the possibility to make the data persistent.
- Feedback based on the city state. Get data relative to buildings or get a more global picture coloring a city based on points of interest.

Based on the interest emerged from the previous chapter where reflections about third parties judgements toward VETRUVIO have been discussed, it is possible to state that there is potential in VETRUVIO. What raised the interest in the tool was the interactive representation of the data. On both public and private sides an interest in the graphical representation with interaction has been shown. There is still work that has to be performed on both data collection and VETRUVIO. The lack of data availability is a huge burden for VETRUVIO which bases its interactions on the data loaded into the model. On the other hand VETRUVIO can still be improved in different aspects.

### 6.2 Future work

The future work on VETRUVIO depends on the direction that it will take. Since the user pool of VETRUVIO does not include only work environments with strictly same needs, depending on which direction it takes, different implementations could be performed. There are also missing features of common interest.

In first place the approximative representation of buildings is a clue point on which every third party agreed. A graphical improvement with the addition of details would enhance the representation clarity. The accurate buildings representation is a sought feature which would improve the sight that VETRUVIO provides over the city but on the other hand would not change interactions over the town. The city model is missing a ground representation. Even with an approximate ground representation additions of city elements could take place. Elements like

streets, green zones, rivers, tunnels and many others would allow further interactions, like fastest route calculation, to be implemented.

In the case where VETRUVIO would lead toward public institutions and private architect studios the requirements of further implementations would be strictly related to constructions. For example the integration of data related to the construction constraints of a zone would suggest interest. Depending on the zone when a new building is virtually built it would be wise to know what are the restraints in the area (e.g. building height, distance from the closest building, dangerous zones). With this feature architects could foresee eventual issues that might arise due to technical restraints.

It is possible to improve artificial intelligence in VETRUVIO and implement pattern recognition over cities schemes. This feature could be of use if the tool aims to provide possible scenarios of city evolution. Based on a city scheme automatically recognized it could be possible to estimate a prediction of the city evolution and represent such evolution on a interactive 3D model in VETRUVIO.

Finally VETRUVIO could be improved on the interface provided to the user. Facilities on data modification and data injection could be added as well as new visual feedback actions over the city.

# Bibliography

- [1] Aristotele. *Politica* ii.
- [2] R. Bartlett. *The Making of Europe. Conquest, Colonization and Cultural Change 950-1350*. Penguin UK, 1994.
- [3] M. W. Beresford. *New towns of the Middle Ages: town plantation in England, Wales, and Gascony*. Lutterworth, 1967.
- [4] A. Bergel, D. Cassou, S. Ducasse, and J. Laval. *Deep Into Pharo*. Lulu. com, 2013.
- [5] A. P. Black, O. Nierstrasz, S. Ducasse, and D. Pollet. *Pharo by example*. Lulu. com, 2010.
- [6] W. Boerefijn. *The foundation, planning and building of new towns in the 13th and 14th centuries in Europe. An architectural-historical research into urban form and its creation*. PhD thesis, Universiteit van Amsterdam, 2010.
- [7] D. Bogunovich. *Urban sustainability 2.0: Resilient regions, sustainable sprawl and green infrastructure*. 2012.
- [8] D. Bogunovich. *From planning sustainable cities to designing resilient urban regions*. 2014.
- [9] J. Busquets. *Barcelona, the urban evolution of a compact city*. Harvard University, 2005.
- [10] S. Confederation. *Approximate solution for the transformation ch1903 wgs84*. 2005.
- [11] R. Davreu. *The World's Last Memories*. Reader's Digest, Sydney, 1978.
- [12] A. Demandt. *Die Kelten*. Verlag Ch. Beck, 1998.
- [13] M. Dias, M. M. Peck, S. Ducasse, and G. Arévalo. *Clustered serialization with fuel*. In *ESUG International Workshop on Smalltalk Technologies 2011*, 2011.
- [14] M. Dias, M. M. Peck, S. Ducasse, and G. Arévalo. *Fuel: A fast general purpose object graph serializer*. *Software Practice and Experience*, 2011.
- [15] Esri. *Xml schema of the geodatabase. An ESRI Technical Paper*, 2008.
- [16] N. Evenson. *Le Corbusier: The Machine and the Grand Design*. 1969.
- [17] R. H. Ewing and G. Anderson. *Growing cooler: the evidence on urban development and climate change*. ULI Washington, 2008.

- 
- [18] S. S. Fainstein. Urban planning. *Enciclopedia Britannica Online*, 2015.
- [19] S. Gideon. Space, time and architecture. *The Growth of a New Tradition... Mass*, 1941.
- [20] M. Girouard. *Cities & people: a social and architectural history*. Yale University Press, 1985.
- [21] B. Goodall. *Dictionary of Human Geography*. Puffin, 1987.
- [22] T. A. Group. Beauty, humanism, continuity between past and future. 2014.
- [23] P. Hall. *Cities of Tomorrow*. Blackell, Oxford, 2002.
- [24] W. Harris. Invisible cities: the beginning of etruscan urbanization. In *Atti del Secondo Congresso Internazionale Etrusco*, Rome, 1989.
- [25] L. Hill. *New Towns: Their Origins, Achivements, and Progress*. London, 1963.
- [26] B. Kemp. *Ancient Egypt: Anatomy of a Civilization*. Routledge, 2006.
- [27] B. A. Kipfer. *Encyclopedic Dictionary of Archeology*. Springer, New York, 2000.
- [28] W. R. L. Londe and J. R. Pugh. *Inside Smalltalk: Volume 1*. Prentice-Hall International, Inc., 1990.
- [29] A. I. of Architects. Issue brief: Smart-growth: Building livable communities. 2014.
- [30] B. V. Ryseghem, S. Ducasse, and J. Fabry. Spec: A framework for the specification and reuse of uis and their models. In *ESUG International Workshop on Smalltalk Technologies*, 2012.
- [31] N. Shradly. The last day: Wrath. *Ruin, and Reason in the Great*.
- [32] M. Smith. *British Town Planning and Urban Design*. Longman, Singapore, 1997.
- [33] T. Vincenty. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. *Survey review*, 23(176):88–93, 1975.
- [34] P. Vitruvius. *The Ten Books on Architecture*. Cmabridge: Harvard university Press, 1914.
- [35] Q. Yang, J. Snyder, and W. Tobler. *Map projection transformation: principles and applications*. CRC Press, 1999.