The use of visual programming interface for structuring a generic digital framework in a City Information Model (CIM) workflow

Eugênio Moreira[1], João Victor Mota Alexandrino[1], Vinícius Fernandes Muniz[1], Daniel Ribeiro Cardoso[1]

[1] Federal University of Ceará, Fortaleza, Brazil
eugeniomoreira@daud.ufc.br; joaovictormotaa@arquitetura.ufc.br; viniciusfmuniz@arquitetura.ufc.br; danielcardoso@daud.ufc.br

Abstract. One of the great challenges for the urban planner/designer is to establish strategies to deal with the increasing complexity of the contemporary city. Among the possibilities of dealing with it, the emergence of the so-called City Information Models (CIM) has presented itself as a promising direction. This paper seeks to contribute to the problem by describing a way to structure a CIM and proposing the creation of a computational application called Carcará, a plugin for a visual programming interface capable of reading and writing to a georeferenced database, allowing the creation of representations not only of the built space, but also manipulations of its semantic characteristics and calculation of a variety of metrics.

Keywords: City Information Model, Urban planning, Interface design, Parametric analysis, Urban Indicators

1 The challenges of urbanism in the third modern age

Ascher (2010) proposes an interpretation of the history of cities considering the process of technical, social and spatial division of production, linking their dynamics to their potential for transportation and storage of goods, technologies and people. The advance of technologies providing changes in this system of mobility and storage would then be responsible for the evolution of cities, which *crystallize and reflect the logics of the societies that host them* (Ascher, 2010, p.20). Understanding how these logics are established would be the role of Urbanism. For the author, the process of modernization of cities is not gradual and linear, but rather is marked by revolutions caused by specific and particular historical conditions involving dynamics of individualization, rationalization, and social differentiation. From his point of view, we would be in the third modern age of cities, where information and communication technologies play a key role in changing the material base of contemporary society – a perception that is also shared by Castells (2007).

Given this dynamic, cities become complex systems that can no longer be explained from the metaphor of mechanical systems (which tend to static equilibrium), but rather by the metaphor of biological systems, where negotiation strategies and competition among different actors can reach certain moments of dynamic equilibrium (Batty, 2007; Batty, 2009). This poses some challenges for urbanism, which can no longer be guided by static big plans that establish top-down control mechanisms to guide the system over long periods of time. Otherwise, urban planning must be guided by the production of conflict negotiation devices in a *heuristic, iterative, incremental, and recursive cycle* (Ascher, 2021, p.83). As the complexity of the urban system and the tasks involved in managing and planning it increases, the need to create strategies to deal with this (i.e. the creation of tools that allow not only representing it, but experiencing it at some level) grows. City Information Models (CIM) present themselves as an important step in this direction.

The purpose of this article is to present the development of a specific computational application – Carcará –, a plugin for a visual programming interface that is part of a CIM framework. In section 2, we will discuss a conceptual framework for a City Information Model within which the creation of the computational application in question is relevant. In section 3, the structure of the tool is presented, showing its parts and functionalities, as well as some works already done that use the tool as a framework. Finally, in section 4, the results obtained are critically evaluated and future developments are envisioned.

2 City information Modelling: a theoretical background

2.1 A Concept in Dispute

Information modeling is not a new theme, being present in the field of Computer Science since the first data processing systems in the 1950s (Mylopoulos, 1998). Information models refer to representations of portions of the real world through symbolic structures, replicable in a computational environment, organized from a particular system of categories, that is, endowed with their own ontology (Mylopoulos, 1998; Guarino, 1998).

The idea of information models in the field of disciplines that deal with design first appears in Architecture with the structuring and consolidation of Building Information Models (BIM), having the first mentions of the term appearing in the 1980s (Eastman et al., 2014). Following the consolidation of this field, Khemlani (2005) is pointed out as one of the first authors to refer to the term CIM, assuming that it would be possible to develop a technology that integrates the different urban structures and services just as BIM integrates the different aspects of construction. This starting point generates a stream of research that assumes that structuring a CIM would lie in some extension of the capabilities of BIM platforms in dealing with georeferenced data, with a strong emphasis on the integration of BIM and GIS technologies (Zhu et al., 2018; Arroyo Ohori et al., 2018; Zhu & Wu, 2021).

The proposal, while raising some interesting points, starts, however, from a conceptual misunderstanding. The knowledge domain of the building differs significantly from the knowledge domain involving the city. Direct adaptations, taking into account only the change of physical scale (and not of complexity scale) are problematic and can lead research results to applications that can do little to aid the task of planning cities (Moreira & Andrade, 2018).

Following a different path, we can point out the City Induction research, reported in depth in Duarte et al. (2012). There, we notice a concern in the creation of specific ontologies to describe the city and urban environment domain, as well as an urban development methodology. For the ontology of the urban environment, the authors build on the seminal work of Lynch (1960) and propose classes (sub-ontologies) based on morphological elements of the city: networks, blocks, zones, landscape and focal points. For the urban development methodology, they propose the structuring in phases, each one based on specific theories, namely: problem formulation – based on Alexander et. al's (1977) pattern language, solution generation – based on Duarte's (2005) discursive grammars, and evaluation – based on the calculation of performance metrics, such as those of Hillier & Hanson's (1989) spatial syntax or Berghouser-Pont & Haupt's (2009) density (Fig 1). For each of the phases, we have the development of derived research, respectively Montenegro (2015) – formulation, Beirão (2012) – generation, and Nourian (2016) – evaluation. A more updated version of the general concepts is presented in Beirão & Duarte (2018).

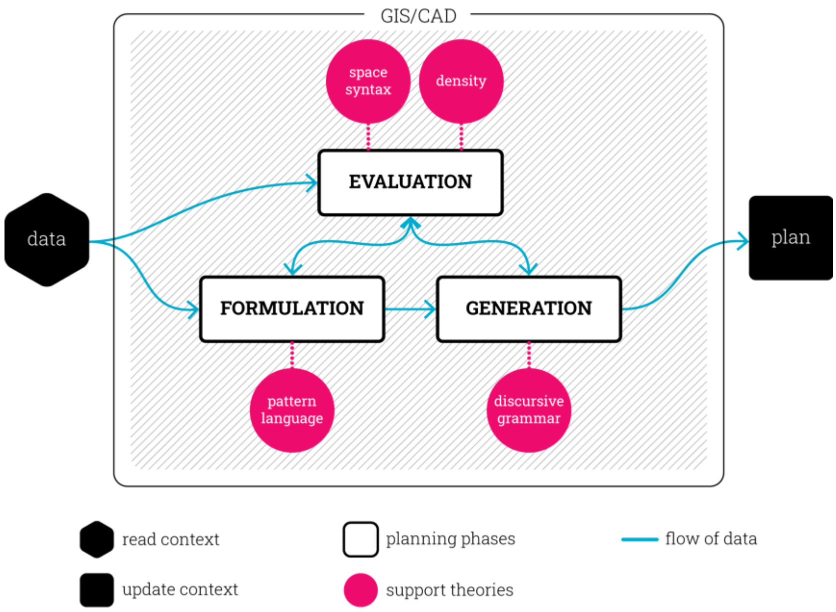

Figure 1. Structure of the ontology for the planning process thought for the City Induction project. Source: Made by the authors based on Duarte et. al. (2012, p.84)

2.2 Choosing a Path and Extending it

The research presented here chooses to continue the vision initiated by the City Induction project for considering it more aligned with the concept of a information model applied to the city. From the technological point of view, to achieve the vision established by the cited authors, it is useful to integrate a set of existing computational applications, allowing the incorporation of several representations of the urban space from the collected data, producing new data from performance metrics (enhancing the perception of the urban environment) and being able to elaborate propositional scenarios. Such scenarios can also be analyzed from the same performance metrics, making possible the comparison between the various states of the system (either the initial state or proposed states), becoming a device that can help in the negotiation between the different stakeholders.

The starting point of such an implementation is the creation of a Database (DB), and it is useful that it is hosted on a remote and accessible server. To organize and manage the data and access, the use of a Relational Database Management System (RDBMS) is extremely convenient. This type of application brings potentialities such as: a) management of different user profiles with different levels of access, b) capacity to manipulate and store a wide range of data types, proving useful in the management of data obtained from various sources, making participatory processes viable, c) functionalities that allow management from a single repository and remote access, maintaining the consistency of the DB and ensuring the access of all stakeholders involved to updated information, avoiding data duplication, and d) possibility of becoming a connector element between various design and analysis platforms, allowing the adaptability of the system to specific demands, as an incremental approach in its implementation (Gil et. al. 2011; Moreira & Andrade, 2018).

Starting from the last potentiality point, it is possible to list a series of useful applications to the urban planning process and that can find in a RDBMS an interoperability node. The use of Geographic Information Systems (GIS) allows not only the feeding of the DB, but also the execution of topological analysis and some geometric manipulations, being a powerful tool in the creation of data visualization through the elaboration of maps. The use of Integrated Development Environments (IDE), especially in notebook format, allows the implementation of data set analysis through Data Science techniques with the use of predictive models, data mining, and chart visualization. The integration with Statistical Analysis Software (SAS) brings functionalities for processing large datasets to perform descriptive statistics, regressions, cluster analysis, non-parametric tests and other procedures, also working on the understanding of urban information through the creation of charts. The connection with Web Applications (WA) makes it possible to build interfaces usable by the population in general, either to display portions of the existing and produced data, or to collect the impressions on scenarios proposed in the elaboration of urban plans. In another way, there is the possibility of integration with existing platforms through data scraping techniques, allowing the capture of spatially localized data through social networks, for example.

The connection with an algorithmic modeler, however, shows itself as one of the greatest potentials for the proposed CIM environment, where the use of Visual Programming Interfaces (VPI) in combination with CAD platforms constitutes an interactive design tool, at the same time showing itself as an environment where several analysis processes can be modeled. A VPI presents itself as an excellent tool for design professionals, providing a programming interface that does not demand the learning of a specific syntax, bringing the creation of algorithms closer to the creation of process diagrams, a type of representation that is familiar to this kind of professional.

An illustrative diagram of the proposed system can be seen in Fig. 2. So far, the research depicted here has adopted as platforms for proof-of-concept PostgreSQL (RDBMS), QGIS (GIS), Jupyter Notebook with Python (IDE), IBM® SPSS Statistics (SAS) and Rhinoceros (CAD) + Grasshopper (VPI) suite as algorithmic modeler.

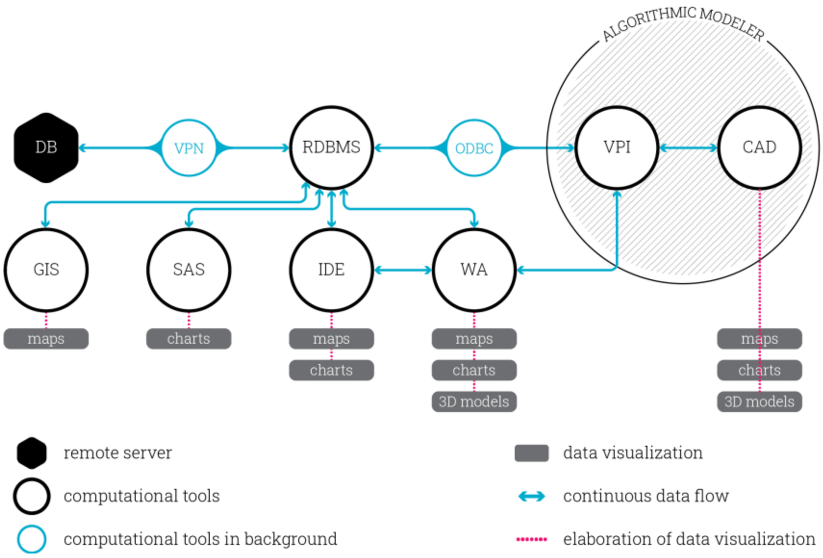

Figure 2. A proposed structure for a CIM system. Source: made by the authors.

While proving to be an important tool, the algorithmic modeler adopted has some limitations that have been identified as important to address. Grasshopper has a very active community of developers, with many add-ons for various tasks. Among these is the Slingshot! plugin by Nathan Miller, which establishes communication between VPI and relational database managers through ODBC protocols, which could be used to communicate with a PostgreSQL structured database. In fact, this is the option used by some works pointed out here, such as Beirão (2012), Beirão et. al. (2012) and Duarte & Beirão (2018). However, the functions provided by Slingshot! are related to direct database queries using SQL (Structured Query Language), and it does not have any specific tools to deal with georeferenced data or geometry processing. Otherwise, it requires the user to know a specific textual syntax (SQL), which kills one of the main potentials of a VPI.

In order to face this limitation, we present Carcará, a Grasshopper plugin that identifies some actions typical of the use of georeferenced databases in urban planning processes and encodes them as functions in VPI, allowing the user to work with the proposed CIM platform within the algorithmic modeler without the use, at first, of any specific textual syntax.

3 Putting the Carcará to fly

Carcará uses the concept of design patterns (Gamma et. al., 1995) and establishes a series of problems typical of the use of databases in urban planning processes, establishing reusable object-oriented functions. These functions have so far been structured in 5 thematic groups: Modelling, Queries, Utilities, Dataviz and Surrounding Analysis (Fig. 3).

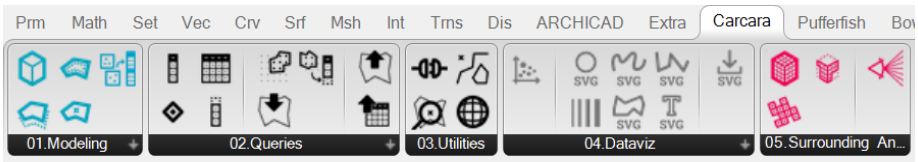

Figure 3. Screenshot of the Carcará tools tab. Source: made by the authors.

The "Modelling" group presents some functions that have been identified as recurring in the construction of three-dimensional urban models. The "Building Meshes C#" function, for example, constructs a prismatic building volume from footprint and height data in polygonal mesh format, separating the outputs into "ground faces", "side faces" and "rooftop faces", using textual syntax in the C# module. The task of creating a prismatic volume is not difficult to implement with standard Grasshopper functions by associating nurbs surfaces. However, to create representations of buildings in an urban fabric, we often need to create thousands of volumes, which is very inefficient to do with surfaces, which justifies the choice of polygonal meshes, combined with the use of textual syntax in C#, which makes the processing faster. Otherwise, we chose to separate the different enclosures, since different analyses make use of only some portions of the built volume, such as visibility analyses (which take into account only the side faces) or solar energy production potential analyses (which need to separate the rooftop faces to delimit the evaluation space). Other functions present in this group are: "Identify Duplicate Polylines", since it was identified that the official mapping bases frequently have entity duplication errors; "Offset Python", which uses the python module to build an offset function that is less prone to errors than Grasshopper's default function; Point Inside Polygon, useful for creating associations between entities that contain others, such as lots and buildings; Sort By Container, which rearranges a list of elements based on a list of "containers" (e.g., rearrange a list of lots by the blocks they belong to or a list of building footprints by the lots they belong to).

The "Queries" group is the heart of Carcará, being responsible for establishing the communication between VPI and the database. In a relational database, the data of an entity is stored in a column, which belongs to a table, which is inside a schema. To access any data, an SQL statement is required, following a well-defined structure. The functions in this group start from the idea of separating from this statement structure and variables, turning the latter into user-visible inputs. Structure and variables are joined through Grasshopper's native text concatenator function, forming a query. This is passed to a function written in the C# module for communication with the database via the ODBC driver, returning the expected result. The whole set is encapsulated in a cluster, creating a Carcará function. A schematic representation of this structure can be seen in Fig. 4.

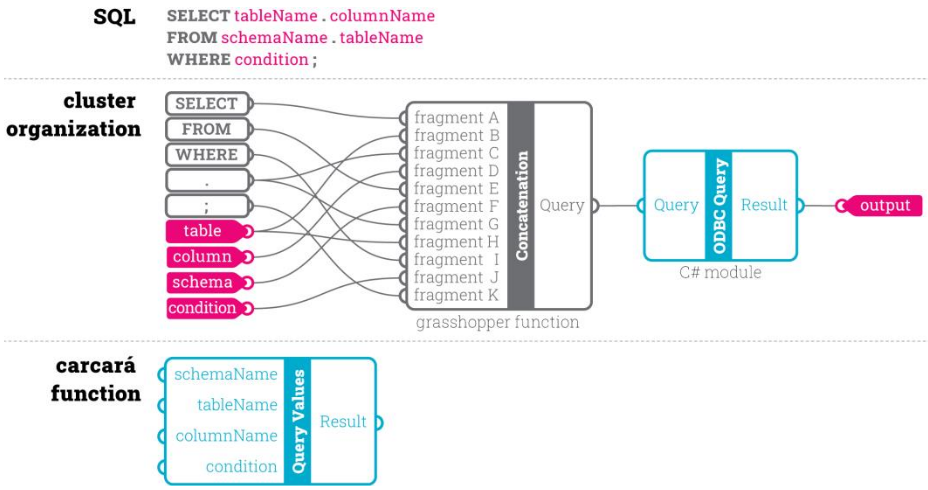

Figure 4. General scheme for creating a function from the "Queries" group. Source: made by the authors.

Another problem is in the translation of geometric entities. In a PostgreSQL database, storing georeferenced data requires the use of the PostGIS extension, which brings with it some extra functions for dealing with this type of object. In the table translation, the geometry becomes another column of alphanumeric data and is translated into the hex-encoded well-known-binary format, which is not readable by Grasshopper. Thus, functions involving the query of geometries use a translation function from the hex-encoded WKB format to a coordinate string of the constituent points of the geometry. This string is processed with Grasshopper's text tools, making it possible to reconstruct the queried geometry, be it a point, polyline, or polygon. A scheme explaining the translation can be seen in Fig. 5.

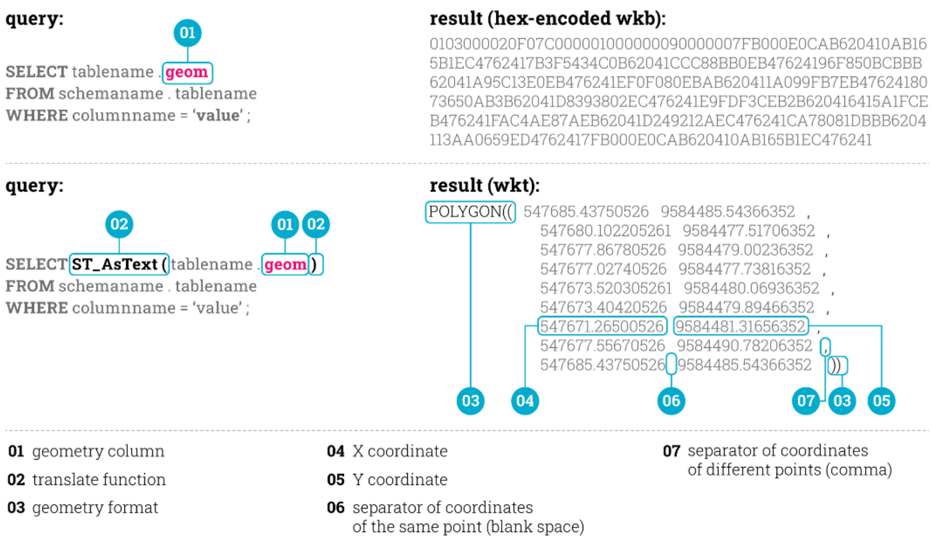

Figure 5. Translation of geometric data. Source: made by the authors.

The third group, called "Utilities", will bring some components that can be used in conjunction with the functions from the "Queries" group, making it possible to get certain input data already formatted for use. In this group, there are functions such as: "Connection String", which builds a structured text string with data about the connection to the database, such as username, password, IP address, and others; "Find Correction Parameters", which can be used to obtain the coordinates of a reference point in the study area, to be used later in queries to correct positions in the generated model; "Geometry Types", which lists the possible types of geometry in a GIS; and SRID, which brings a list of the most used SRIDs.

The fourth group, "Dataviz", brings some graph building functions into the grasshopper environment. In the current state of development, there are 2 subgroups. The first is intended for building graphs on the Rhino screen and features a scatterplot function, which is fully configurable in its appearance. The second sub-group is intended to allow the generated graphs to be saved directly in SVG format files, contributing to the creation of visualizations of the data sets and can be used directly for reporting. The functions in this group are a set of composers (which translate distinct geometries into the corresponding SVG code) and an exporter, which feeds in the data from the composers, builds the contents of the file, and saves it to a folder on the user's computer.

Finally, the fifth and last group (so far) is called "Surrounding Analysis" and is composed of a set of tools that allow the calculation of visibility indexes of objects of interest immersed in an urban context. The tools are the result of research aimed at establishing metrics to evaluate the ambience of built cultural heritage and can be seen in more detail at (Moreira et al., 2019).

Taken together, the Carcará tools provide a generic framework for modeling the urban environment within an algorithmic modeling platform, allowing a number of appropriations for work with different themes. Some research that has already made use of the tool (in previous versions) are: Filipe et. al. (2019), for the evaluation of the quality of public space; (Moreira, 2018), for the evaluation of the ambience of the built cultural heritage; Lima et. al. (2019), Lima et al.(2020) and Lima et. al (2021), for representing informal settlements and calculating indicators that allow their participatory planning; and Passos Filho & Cardoso (2019), in an application of machine learning for computational bypass in the generation of environmental comfort indicators.

In fig 6 we can see an example of application of the system. In session 01, we have an isometric perspective of the built volumes of 5 neighborhoods in the city of Fortaleza, Ceará, Brazil, where the height information was translated into a color gradient ranging from cyan (lowest value) to magenta (highest value), passing through yellow (average value). For this example, the DB was queried, accessing information of all neighborhoods in the city, retaining the 5 shown based on their official names and using the geometry of their boundaries as a spatial filter for the information of all built volumes in the city, with their respective height data. In session 02, an information layer containing location data of all public open spaces in the city was queried, and it was possible to measure the distance from these to each built volume, selecting the distance to the nearest one, establishing a very simple analysis. The list with these distances was translated with the same color gradient as in session 01, and the result applied to the volumetry.

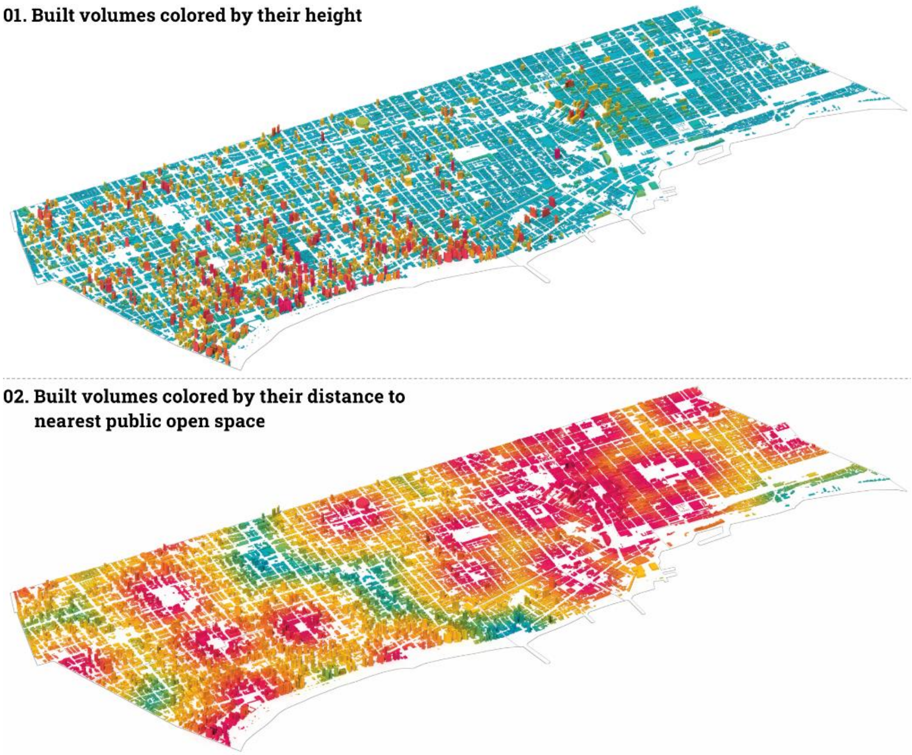

Figure 6. Results obtained with the use of Carcará. Source: made by the authors.

4 Discussion

The discussion about the concept of City Information Modelling and its applications is still a bit far from reaching some consensus. Until now, some experiments have been made in order to build some tool that operates within this conceptual framework, without yet arriving at something of consolidated use. Thus, we understand that the work presented here is promising as it tries to take advantage of a set of existing tools, establishing protocols for their communication. However, although the use of an algorithmic modeler opens an extremely broad field of application, either in the modeling of analysis processes, or in the construction of generative systems for the production of solution scenarios, some limitations can be pointed out.

The Rhino+Grasshopper platform is quite widespread in the field of product design and Architecture, having still timid applications in Urbanism. In fact, the system is not prepared to deal with geographic entities that, depending on the reference system adopted, may have coordinates that exceed 6 digits, which is transformed into scientific notation by Grasshopper, creating inaccuracies in the representation and requiring the creation of certain workarounds.

On the other hand, making a critical analysis of the theoretical line adopted, it is possible to notice little emphasis given to strategies for creating visualizations with more visual qualities, which is quite present in works that investigate the CityGML protocols in BIM+GIS integration. Such representations, although of little use for analysis processes, are very important in communicating with non-technical sectors, with the general population.

As future developments, the research presented here intends to further investigate the urban environment ontologies proposed in Duarte et. al (2012), Beirão (2012) and Beirão & Duarte (2018) with a view to designing organizational structures for the database, which will allow queries to become more intuitive. We also intend to expand some groups with new tools, especially the "Dataviz" group, with the incorporation of new types of charts, allowing greater integration between this type of representation and three-dimensional or cartographic ones.

References

Alexander, C., Ishikawa, S., Silverstein, M., & Jacobson, M. (1977). *A pattern language:* Towns, buildings, construction (36. print). Oxford Univ. Press.

Arroyo Ohori, K., Diakité, A., Krijnen, T., Ledoux, H., & Stoter, J. (2018). Processing BIM and GIS Models in Practice: Experiences and Recommendations from a GeoBIM Project in The Netherlands. *ISPRS International Journal of Geo-Information*, *7*(8), 311.

Ascher, F. (2010). *Os novos princípios do urbanismo* [The new principles of urbanism]. Romano Guerra.

Batty, M. (2007). *Complexity in City Systems*: Understanding, Evolution, and Design. Working Papers, 117.

Batty, M. (2009). Cities as complex systems: Scaling, interaction, networks, dynamics and urban morphologies. Em R. A. Meyers, *Encyclopedia of complexity and systems science* (p. 1041–1071). Springer.

Beirão, J. N. (2012). *CItyMaker*: Designing grammars for urban design. Delft University of Technology, Faculty of Architecture, Department Architectural Engineering+ Technology, Department of Urbanism.

Beirão, J. N., Montenegro, N., & Arrobas, P. (2012). *City Information Modelling: Parametric urban models including design support data*. Actas da Conferência Internacional PNUM 2012, 1122–1134.

Beirão, J., & Duarte, J. P. (2018). Generic grammars for design domains. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, *32*(2), 225–239.

Berghauser-Pont, M. Y., & Haupt, P. A. (2009). *Space, density and urban form.* Technische Universiteit Delft.

Castells, M. (2007). *A sociedade em rede* [The Rise of the Network Society] (R. V. Majer & F. H. Cardoso, Trads.). Paz e Terra.

Duarte, J. P. (2005). A discursive grammar for customizing mass housing: The case of Siza's houses at Malagueira. *Automation in Construction*, *14*(2), 265–275.

Duarte, J. P., Beirão, J. N., Montenegro, N. F. S. de C., & Gil, J. (2012). City Induction: A model for formulating, generating, and evaluating urban designs. In S. M. Arisona, G. Aschwanden, J. Halatsch, & P. Wonka, *Digital Urban Modeling and Simulation* (p. 73–98). Springer. http://link.springer.com/chapter/10.1007/978-3-642-29758-8_5

Eastman, C., Teicholz, P., Sacks, R., & Liston, K. (2014). *Manual de BIM:* um guia de modelagem da informação da construção para arquitetos, engenheiros, gerentes, construtores e incorporadores. Bookman.

Filipe, S., Farias Filho, J. A., Cardoso, D., & Beirão, J. N. (2019). Modelagem da Informação para a Qualidade do Espaço Público. *Anais...* XVIII Encontro Nacional da Associação Nacional de Pós-Graduação e Pesquisa em Planejamento Urbano e Regional, Natal.

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). *Design patterns: Elements of reusable object-oriented software*. Addison-Wesley.

Gil, J., Almeida, J., & Duarte, J. (2011). *The backbone of a City Information Model (CIM): Implementing a spatial data model for urban design*. RESPECTING FRAGILE PLACES - 29th eCAADe Conference Proceedings. 29th eCAADe Conference, Slovenia.

Guarino, N. (1998). Formal Ontology and Information Systems. Em N. Guarino (Org.), *Formal Ontology and Information Systems: Proceedings of the First International Conference (FOIS'98)* (p. 3–15). IOS Press.

Hillier, B., & Hanson, J. (1989). *The Social Logic of Space*. Cambridge University Press.

Khemlani, L. (2005). Hurricanes and their Aftermath: How Can Technology Help? *AECbytes*.

Lima, M. C., Cardoso, D., & Freitas, C. (2019). Informal Settlements and City Information Modeling. *Proceedings...*, 10. https://doi.org/10.5151/proceedings-ecaadesigradi2019_414

Lima, M. Q. C., Freitas, C. F. S., & Cardoso, D. R. (2021). Invisible informality and the contribution of information modeling to data-based urban regulation. *Gestão & Tecnologia de Projetos*, *17*(1), 199–217.

Lima, M. Q. C., Moreira, E., Farias, S., & Freitas, C. F. S. (2020). A data-driven approach to inform planning process in informal settlements. *Blucher Design Proceedings*, 516–521.

Lynch, K. (1960). *The image of the city*. MIT PRESS.

Montenegro, N. F. S. de C. (2015). *CityPlan: Contributo para o desenvolvimento de uma metodologia e ferramenta computacional para apoio ao desenho urbano* [Doutorado em Urbanismo]. Universidade de Lisboa.

Moreira, E. (2018). *Modelando a percepção:* O ambiente do patrimônio cultural edificado na regulação da forma urbana [Dissertação (Mestrado em Arquitetura e Urbanismo e Design)]. Universidade Federal do Ceará.

Moreira, E., & Andrade, D. (2018). Construção e Cidade: Diferenças ontológicas na elaboração de um conceito de CIM. Em A. M. G. Souza, R. E. Baeta, & N. V. de A. Junior (Orgs.), *Anais...* (Vol. 8, p. 12003–12021). FAUFBA.

Moreira, E., Cardoso, D., & Beirão, J. N. (2019). Modelagem da informação e métodos quantitativos a serviço da preservação da ambiência do patrimônio cultural edificado. *Livro de Actas*, 417–431.

Mylopoulos, J. (1998). Information modeling in the time of the revolution. *Information Systems*, *23*(3–4), 127–155.

Nourian, P. (2016). *Configraphics: Graph theoretical methods for design and analysis of spatial configurations*. Delft University of Technology, Faculty of Architecture, Department Architectural Engineering+ Technology, Department of Urbanism].

Passos Filho, J. A. A., & Cardoso, D. (2019). Machine Learning para a acessibilização de análises em escala urbana. *V!RUS*, *19*.

Zhu, J., & Wu, P. (2021). Towards Effective BIM/GIS Data Integration for Smart City by Integrating Computer Graphics Technique. *Remote Sensing*, *13*(10), 1889.

Zhu, J., Wright, G., Wang, J., & Wang, X. (2018). A Critical Review of the Integration of Geographic Information System and Building Information Modelling at the Data Level. *ISPRS International Journal of Geo-Information*, *7*(2), 66.