Chapter 2
Ontologies and Procedural Modelling

Abstract This chapter consists of a literature review regarding the use of ontologies on virtual environments and the procedural modelling solutions that have been proposed with focus in two approaches: (1) the production of virtual hollow buildings, uniquely composed by outer facades; and (2) the production of virtual traversable buildings, with interior divisions included. The integration of ontologies and semantics in procedural modelling is also addressed in each one of the referred approaches.

2.1 Ontologies on Virtual Environments

Over the years, several authors [12–14] have defined and characterized "ontology" while others [15–17] were concerned with its applications in fields such as information systems and engineering. All of them inspired Guarino [18] and Chandrasekaran et al. [19] in the formulation of their own concepts about these knowledge organization structures.

According to Guarino [18], an ontology aims to describe a certain entity using a particular system of categories. In some areas like engineering—such as Artificial Intelligence—an ontology is established through a set of concepts and respective meanings (i.e. vocabulary) with a relation structure that intends to characterize a certain reality. The author also refers the increasing importance of ontologies for computer science, focusing in the information systems area, in which ontologies can play an important role due to their straight relation with components such as databases and program objects and application programs.

Chandrasekaran et al. [19], shares a similar vision. The author states that this kind of structures intends to represent a set of facts related with a particular domain through the organization of the integrating knowledge concepts. This organization arises from the analysis over the domain fields, in which the following authors' assumption should be considered:

> Weak analyses lead to incoherent knowledge bases.
>
> —Chandrasekaran et al. [19]

T. Adão et al., *Ontology-based Procedural Modelling of Traversable Buildings Composed by Arbitrary Shapes*, SpringerBriefs in Computer Science,
DOI 10.1007/978-3-319-42372-2_2

One of the most interesting features of ontologies is the possibility of sharing knowledge. This promotes a re-usability and standardization. For example, a building can share characteristics common in several architectonic styles. Despite their appearance on the different styles, a window, a door and a wall are transversal elements to the majority of the existing architectonic styles. The representation of particular cases (e.g. Manueline window, mesquite entrance, skylight) are extensions of the generic ones.

> Shared ontologies let us build specific knowledge bases
> that describe specific situations.
>
> —Chandrasekaran et al. [19]

Besides re-usability, ontologies are considered, by Chandrasekaran et al. [19], abstract structures. The combination of these two features are completely compatible with the design of data models and object oriented programming classes (towards the considerations made by Guarino [18]). Thus, the ontological analysis of a certain domain field can be suitable for integration in areas such as computer science and software engineering, for both project stages: requirement analysis and algorithmic development.

Considering the notions left by Guarino [18] and Chandrasekaran et al. [19] at an abstract level, one can infer that the use of ontologies can be extended to other areas—besides information systems or software engineering—namely the ones involving virtual representations, as some authors have already shown (e.g. [20–22]). The following subsections will expose some of the works that successfully applied ontologies to regulate virtual representations in different scenarios and contexts. CityGML standard will also be discussed, due to its relevance in the context of virtual environments representation.

2.1.1 Virtual Representations Based on Ontologies

The integration of ontologies in some of the works developed around virtual representations will be presented in this subsection. Although the concept of ontology has been explored for a while, this way of structuring knowledge is still largely used by several authors today.

Pittarello and De Faveri [23] were concerned with the lack of semantic description across the considerable amount of virtual environments available on the Internet, which prevented a proper interaction with them. To get around this issue by enhancing the models with semantics, the authors proposed a solution capable of relating geometric primitives—X3D and VRML—with semantic class objects through the so called *MetadataSet* nodes. An independent scene ontology was also incorporated to establish a set of relations used for the description of a certain domain (for example, a wall can be contained inside the room).

One year later, an approach involving surgical models for computer-assisted neurosurgery was proposed by Jannin and Morandi [20]. The aim of this work is to provide a visual framework for planning surgeries, improving human–computer interfaces—specifically, for the computer-assisted surgery systems—thus formalizing the surgical knowledge and practice. The presented framework relies in a surgical ontology which establishes the concepts and relationships belonging to the surgical work domain—extensible to other areas of medicine—and a supporting software that describes the surgical procedures.

For setting up urban environments, a work that employs ontologies was proposed by García-Rojas et al. [24]. Their parametric system allows common users—nonexperts in the virtual reality field—to prepare 3D scenes through an on-demand configuration. To achieve this, a visual programming paradigm, supported by a proper ontology, allows the organization of a 3D scene components.

A pervasive system, proposed by Lee et al. [25], combines ontology-based context-awareness with adaptable augmented reality. In short, the framework considers several aspects such as user preferences, device profiles and security to augment personalized virtual models, which must in accordance with a given acquired context. Context-awareness is provided by three ontologies. The first ontology holds the general concepts related with the pervasive environment. A second ontology organizes knowledge about users' device profile (related with the mobile device capabilities) and preferences (a set of user options). At last, a social ontology maps users activities related with information shared in the web, for further re-utilization. The authors exposed three system applications as examples: ubiquitous home visualization and simulation, ubiquitous car services and ubiquitous engineering collaboration.

The strength of ontologies was once again highlighted by ShapeAnnotator: a system developed by Attene et al. [21] that allows the classification of 3D virtual model meshes in a certain knowledge domain. The system provides a set of tools for model segmentation that can be manipulated by a user in order to easy and properly link the model parts to the domain knowledge, formalized by an ontology. The usefulness of ShapeAnnotator framework was discussed in two scenarios. The first one explains the potentialities of the tool in supporting the creation of human models (avatars) for Multi-massive online role playing games (MMORPGs) and virtual worlds, such as Second Life. The second scenario focuses the collaborative e-manufacturing of 3D products taking advantage from the abstractness and re-usability provided by the ontology.

In the museums context, another system assisted by ontologies that aims the collaborative annotations of 3D museum artifacts through web-based services, was proposed [26]. This system—entitled Harvesting and Aggregating Networked Annotations (HarvANA)—promotes the participation of communities in the cultural enrichment and improves museum objects indexation. One of the most relevant features of this system is the flexible ontology-based categorization—called of folksonomy—which optimizes tagging proceedings among the communities.

In the context of urban planning and management, Martins [27] proposed an urban ontology to overcome the issue of data heterogeneity among municipalities that use different geographic information data sources. The main idea is to establish a common data model and provide an unified platform for data sharing between municipal technicians. Thereby, the author supported a significant part of his work in the CityGML standard—addressed in the next subsection—to develop a set of data models, which intend to reflect the different urban elements. For example, the proposed building model establishes a structure which includes a building, building parts, rooms, openings such as doors and windows and several boundary surfaces like walls, grounds, ceilings and furniture.

A couple of years later, Colledani et al. [28] proposed the integration of several heterogeneous software tools for manufacturing activity design over the same platform, using ontologies for uniformization purposes. This is implemented through the so-called Virtual Factory Framework, which is composed by several components: data and knowledge, Semantic Virtual Factory Data Model (VFDM), Semantic Virtual Factory Manager (VFM), decoupled virtual factory modules and the real factory interface. In this case, the VFDM is the abstraction layer holding the ontology that can extend products and define manufacturing processes. The rest of the components complete the system by ensuring the connection of the framework to the external applications, through special framework connectors designed for integration.

Recently, Flotynski [29] developed the Semantic Modelling of Interactive 3D Content (SEMIC), which employs a method for the modelling of knowledge, rather than the modelling of virtual content itself. SEMIC employs a method that consists in the mapping of 3D content into semantic classes, which are related with each other, in order to establish object relations for a given domain. The creation of 3D content consists in a set of well-defined steps that includes the design of the concrete semantic representation containing properties related with the 3D content, the mapping of specific domain concepts and the design of a conceptual semantic representation of 3D content, for arbitrary 3D creation purposes. These steps, as the author refers, require the intervention of different skilled professionals such as content developers, domain experts and content consumers. This aspect might suggest that this system is somewhat complex regarding 3D virtual models creation process.

Another recent approach addressing simulations in virtual environments was proposed by Béhé et al. [22]. The authors presented a framework for interactive multiagent-based simulations in virtual environments, adapting ontologies as a core notion to ease the simulation design and re-usability. Thereby, simulations are configurable through semantic modelling. This is used to describe the different aspects of the simulation, namely agent behaviours, surrounding environments with physical objects, scheduling for operation progress and the results of agent actions and interactions.

The differences between Geographical Information Systems (GIS) and BIM were addressed by Mignard and Nicolle [30], who developed a system called SIGA3D. This system takes advantage of ontologies, to provide interoperability between construction and urban management. So, information about buildings and geographic data can be managed together, in the same structure. The system also promotes collaboration between facility managers, aiming the enrichment of knowledge models since the designing stage to buildings' recycling.

The last work reviewed in this subsection is Virtual Collaboration Arena (VirCA): a collaborative virtual/augmented reality framework that enables testing and training events in the context of manufacturing systems, through several practical scenarios [31]. Such scenarios are mounted through web-based applications and interfaces that provide mechanisms for using and extending virtual reality content. Interaction with the referred scenarios is provided by cyber devices, also known as CDs, that enable the manipulation of their objects. The ontology concept acts here: a semantic manager layer ensures the bidirectional communication between CDs and virtual scenes, factoring on the available capabilities and requests—in terms of allowed actions/functionalities—supplied by the scenes' ontologies.

Summing up, one might conclude that ontologies have made a significant contribution to the success of several works, some of them referred in this subsection. This way of organizing knowledge to regulate processes has revealed robustness and flexibility in several works that require data representation through virtual models in a wide variety of contexts such as simulation, industrial manufacturing, collaboration, urban planning and others. However, in the specific context of 3D urban modelling, there is a well-defined urban knowledge structuring standard named CityGML. The next subsection will address some of the most relevant features of this standard.

2.1.2 CityGML: A 3D Urban Environment Standard

CityGML is, perhaps, the most important effort for the standardization of 3D urban representations [32, 33]. These guidelines, proposed by the Open Geospatial Consortium (OCG), intend to provide a widespread XML-based format for the geometric and semantic representations of city components. The standard covers the building entity including its inner and outer components. Kolbe [34] explains the CityGML support to this crucial city element, by presenting also an abstract class named of *_AbstractBuilding*. This is the mother class that derives to *Building* and *BuildingPart*. *Building* holds sets of *BuildingPart* which can be seen as groups of structures that take advantage of the recursive relation with the *_AbstractBuilding* in order to support a wider range of structural rearrangements. For example, a given building may hold a stack of floors and a castle might be constituted by a set of horizontal distinguishable parts such as towers, curtain walls or gatehouses. Buildings and building parts can be represented in terms of constraints by another important class which is the *BoundarySurface*. This one can derive to specific boundaries such as *WallSurface* or *RoofSurface*. Furthermore, this boundaries may hold a set of objects spawned from *_Opening* class, as for example, doors and windows. Finally, *Room* class is intended to support the inner compartments inside the building and building parts. The diagram that describes the building knowledge organization is depicted on Fig. 2.1.

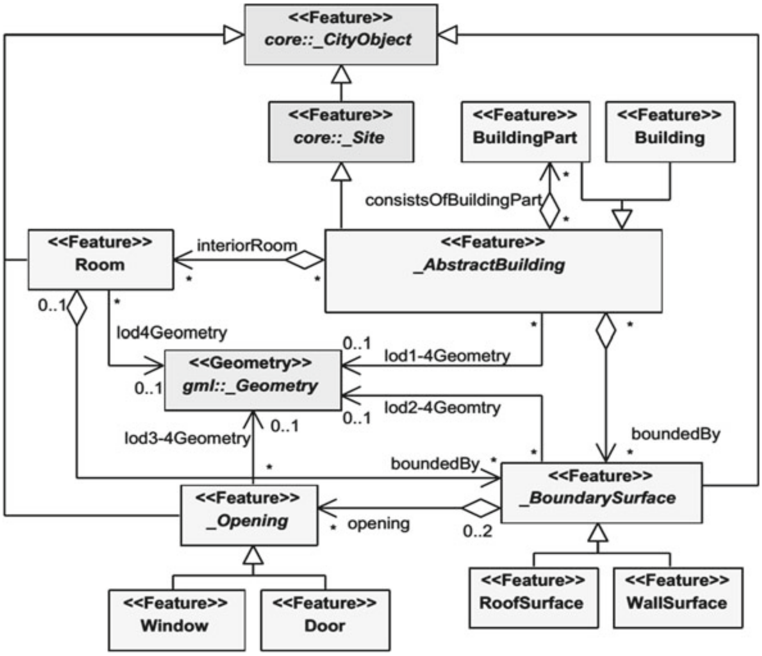

Fig. 2.1 UML diagram showing a simplified excerpt from the CityGML building model [34]

The specification of buildings is also extensively addressed by Industry Foundation Classes (IFC): a data model standard widely used by the construction industry for project purposes, in the context of BIM process [35]. However, the quantity of addressed civil construction and architectural domains [36] is excessive and technically complex to describe an urban environment oriented for areas such as videogames, 3D cinema or archaeology. The overload of dispensable information—not perceptible to the eye during representation—can even, in some cases, harm the objective of the virtual building representation, depending on usage context (unnecessary heavy and time consuming renderings, graphical lag, heavy processing, etc.). To conclude and considering such aspects, CityGML seems to be more suitable for designing general urban environments that target virtual content creation for areas as the ones that were aforementioned.

Besides the various applications of ontologies on virtual environments, some procedural modelling works also use them to regulate structures' generation (e.g.: [9, 11, 37–39]). The aforementioned works will be presented during the following pair of sections, which are reserved for an extensive analysis to procedural modelling, including the generation/reconstruction of virtual extensive urban environments and traversable buildings.

2.2 Procedural Modelling of Virtual Urban Environments

In this section, the production of virtual environments concerning the generation/reconstruction of exteriors will be presented. These environments are mainly composed by street networks and buildings, exclusively represented by outer facades.

2.2.1 L-Systems for Procedural Modelling

L-System was introduced by Lindenmayer [40] and adopted by Parish and Müller [1] to generate an extensive virtual environment considering its exterior layout. This technique uses an alphabet of symbols combined with a set of production rules. The process starts with an initial set of symbols that are iteratively replaced by other symbols until the final string is obtained. This final string is then used to generate shapes through a transformation mechanism. Parish and Müller [1] used this technique in two steps: one for generating a street network and other for producing buildings that can be described by several formats, provided by extrusion operations.

XL3D modelling system [41] also incorporates a geospatial L-System. Streets and blocks coordinates provided by a database populated with real data were used in combination with the L-System production rules, to generate the virtual downtown of Porto, Portugal.

2.2.2 Detailing Facades Through Split Grammars

Split grammar was introduced by Wonka et al. [2]. This technique relies on a grammar that operates in the context of a shape in order to produce 3D layouts. There are two types of rules in this grammar: splitting rules to replace geometries and conversion rules to produce transformations upon geometries. Shapes marked to be splitted or converted can be categorized in two classes: terminal and nonterminal. The process starts with an initial shape. A split grammar operates iteratively to force the shape to undergo several nonterminal states, until its final form. This process is considered finalized when all shapes in a pool reach their final state. Split grammar was applied to improve the details of building facades.

Later, Larive and Gaildrat [42], developed a wall grammar for buildings' generation, based on the split grammar of Wonka et al. [2]. This wall grammar integrates a technique to produce 3D buildings with exterior facades. The process encodes several elements—such as building footprint and height extracted from GIS, user specifications and high-level features for the building exterior appearance—into grammar rules that are applied to create a building with extruded walls representing outer facades. Differently from [2], this grammar operates on walls instead of shapes. Finally, a straight skeleton technique [43, 44] is used to generate the building's roofs.

2.2.3 Semi-automatic Digital Reconstruction of Old Buildings Considering GIS Data-Based Topology

A preserved area of Nicosia city was digitally reproduced through a partial automatic method which combines geographic data of building bases, building classification and style-concordant building components, such as doors and balconies [45]. Two distinct processes integrate the method. The former consists in photographing particular elements of the city to model realistic virtual building components. The later is a rule-based automatic process that starts by comparing the outline of each building—obtained from GIS data—with templates to determine building topology. Then, the ground edges are transformed into 3D walls. The previously modelled virtual building components are applied to each wall accordingly with the building topology classification and wall space available. Finally, using a straight skeleton computation approach [43, 46], the roof of each building is properly produced and applied.

2.2.4 Random Extrusion of Floors

Greuter et al. [47] proposed a system to generate pseudo-infinite cities. The generation of road networks is based on a regular grid, globally adjustable. The generation of buildings is achieved by combining geometric primitives—each set corresponds to a floor plan—that varies from floor to floor. The process starts with an elementary set of geometries, at the building top. Then, an extrusion forms a volume and primitives are added to a subsequent level. The process is repeated until reaching the building ground level.

2.2.5 Feature-Based Decomposition of Facades

Finkenzeller et al. [48] were capable of creating highly detailed realistic facades using floor plan modules (aggregations of 2D shapes at the ground level), coarse structures (volumetric shapes based in the floor plan modules), a procedural decomposition method for subdividing features in those coarse structures (e.g. windows, doors, frames and others) and also a geometry factory to handle the representational purposes. The referred work preceded another system that produce buildings with complex facades, proposed by Finkenzeller and Schmitt [49]. In this system, designers must provide high-level requirements such as building coarse, type and style to produce highly accurate 3D buildings. They can also change facades parameters after building's generation, as the system is prepared for recomputing such modifications. A more mature version of this later system was presented by Finkenzeller [50] and it was used to produce the virtual model of the University of Karlsruhe. The realism of the exposed models is impressive. However, such level of detail requires spending between few minutes to about two hours per model, accordingly with the author.

2.2.6 Computer Generated Architecture for Buildings Production

Computer generated architecture (CGA) is another methodology related with outer facades generation, which relies on a rule system provided by shape grammars [3]. The process initiates with the creation of a mass model that constitutes the exterior format of a building, including the roof. This mass model can be seen as a merge of volumetric shapes that can variate in scale, rotation and usage portion. In the next step, facades are created to properly cover the mass models. The final step of the process increases the detail in doors and windows and also accommodates building ornaments. The grammar used in this technique—an extension of the split grammar—is sequential (similar to the Chomsky grammars addressed by Sipser [51]) and enables a large-scale production of buildings with different styles. A suburbia model of Beverly Hills was produced and depicted along with a procedural reconstruction of Pompeii, generated with 190 manually encoded CGA shape rules.

CGA was also applied in some works aiming virtual reconstructions [52–55]. Müller et al. [52] used it with the purpose of reconstructing Puuc-style buildings, that are similar structures to the ones found in Xkipché, México. In their work, the authors created a grammar to fulfil the architectonic requirements of the referred buildings. Thus, accordingly with the typical design of these structures, the grammar defines the building as following: first, the base is defined; then, middle walls are addressed considering building accesses; upon these walls, a middle-layer designated by medial moulding is produced; the last rules define frieze, cornice moulding and completion ornaments.

The same methodology was also applied by Dylla et al. [53] who produced a 3D reconstruction of ancient Rome (Fig. 2.2) through the combination of manually designed structures and procedurally generated buildings, in the same virtual environment. The class of each element defines what kind of approach is needed. For known positions, dimensions and design, class I elements are loaded from models created using a commercial computer aided design (CAD) software. If some information is missing, class II elements are generated procedurally using CGA shape methodology.

Besuievsky and Patow [54] developed their own CGA shape rules to reconstruct historic buildings and urban environments for serious games. They use as input 2D data provided by GIS with corrective mechanisms to deal with map issues, like distortion. This input allows the acquisition of relevant features such as building outlines, for further extrusions, forming mass models. The production of building facades is made through a user-friendly tool which hides the grammar to improve the easiness of use. The application of this methodology produced some interesting results on the virtual reconstruction of Carcassone's old town, in France. The tool's flexibility was demonstrated through the virtual reconstruction of two other cities with different architectonic styles: Nantes of France and Girona of Spain.

Tepavčević and Stojaković [55] combined developed a reliable mathematical model that combines fuzzy logic and probabilistic calculations to produce stochastic

Fig. 2.2 Virtual reconstruction of ancient Rome made by Dylla et al. [53], using class I models that were manually produced and also class II models, procedurally generated with a CGA shape rules set in order to overcome the lack of information

CGA shape rules, that are used to generate realistic Neo-Gothic chapels. Each set of rules defining a chapel starts by specifying a building lot and a mass model. The mass model is decomposed in three main parts: apse, nave and tower. Apse and tower mass model parts are replaced with appropriate shapes and the subsequent steps will detail the model until the final 3D form.

2.2.7 Building Generation Based on Facade View Acquisition

A work focusing the decomposition of building facades through image analysis and shape grammars was proposed by Müller et al. [56]. Their system starts by processing a facade image and extracting its elements using automatic operations, through a top–down hierarchical process. For example, a given facade image is divided into floors which are, in turn, subdivided into tiles that are then partitioned into smaller rectangles. Afterwards, a stage that consists in matching the last rectangles with a library of 3D architectural elements, takes place. The whole process results in a tree shape that is encoded into a shape grammar, which holds the definition for facade representation.

A similar research line was followed by Koutsourakis et al. [57], who proposed a framework capable of producing 3D models from a single facade image. Its inputs are a parametric shape grammar and a rectified image of a single building facade. A tree-based process takes over the generation, collecting a set of rules that regulate it and, then, a Markov Random Field (MRF) formulation optimizes the parametric rules to produce the buildings' final aspect. Later, Simon et al. [58] developed a system that uses shape grammars and facade image classifiers to generate 3D buildings. A combination of procedural modelling, statistics and image processing led to their solution. Both of the previously referred works are extensively addressed on Simon's Ph.D. thesis [59].

2.2.8 Digital Map-Based Generation of 3D Buildings with Multiple Roofs

Sugihara and Hayashi [60] proposed an automatic solution focused on the production of virtual buildings with multiple roofs, considering building footprints provided by digital maps. Using a system to express polygon angles and sort vertices (clockwise), their method is capable of splitting a building footprint into rectangles, that are used to determine roof branches and constitute the base shape for roof creation. The presented results demonstrate a set of buildings automatically generated, each one holding a set of gable roofs unified by branches. Moreover, complex roofs were also produced in order to fulfil the requirements for the reconstruction of an ancient japanese temple and also a pagoda.

2.2.9 City Modelling Procedural Engine (CMPE)

Carrozzino et al. [4] developed an engine that uses a set of input elements such as aerial photographs, vector and raster maps or even text descriptions, to produce extensive urban environments. At the beginning, streets data and block footprints are automatically extracted from the input maps. Then, a 2D road network is produced, followed by its 3D representation. 3D blocks and related buildings are subsequently generated. The buildings are produced considering user specifications or pseudo-random definitions. Further manual interventions for the refinement of the virtual buildings in post-production are also supported. A clean visualization of the whole scene is provided through the XVR rendering engine, incorporated in the CMPE.

2.2.10 Procedural Generation 3D (PG3D) Solution

A solution for procedural generation of extensive urban environments was presented by Silva and Coelho [5]. They opted by a strategy that consists in storing the instructions for modelling urban elements and geographic data in the same spatial database. The instructions for buildings' modelling are stored in database native language, pointing to the shape grammar rules. Each set of these rules can be seen as instructions to guide the geometric generation process of elements such as buildings, roofs or balconies. The presented results include the digital reproduction of some locals of Porto city (Portugal) with a considerable degree of resemblance (see Fig. 2.3). The extended version of this work can be found in the master thesis of Silva [61].

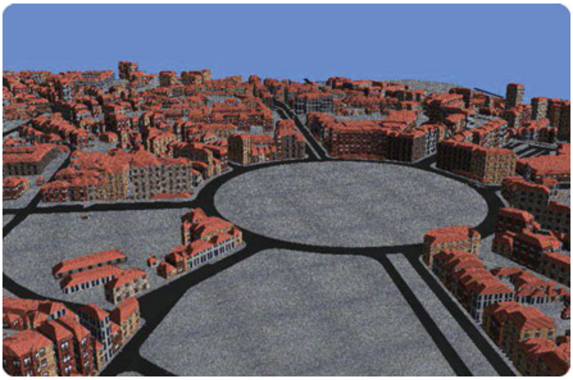

Fig. 2.3 Boavista roundabout produced by PG3D [5]. An available set of information was considered to generate the virtual model of this urban area, that has considerable degree of resemblance

2.2.11 Ontology-Based Generation of Urban Environments and Building Exteriors

The generation of ontology-based virtual urban environments was an area that few explored. In the current subsection, these works will be addressed.

2.2.11.1 Ontology-Based Procedural Modelling to Recover Cultural Heritage

An ontology-based solution was proposed by Liu et al. [11]. The authors embraced the challenge of recovering the cultural heritage of ancient China. To accomplish such challenge, a city generator was developed, capable of producing virtual models based on an ontology and on user input: a grammar for building definitions. Moreover, a style checker was implemented to avoid generation inconsistencies, such as buildings upon streets. Their work is also one of the few cases of an extensible ontology application that covers other architectonic styles. This system followed their previous work [10] in which a semantic-based modelling system was proposed. The objective was to improve users' focus on its specific implementation, while the geometric details are encapsulated by the semantic elements such as walls, doors and windows. In short, a user propagates the semantic information of a building using a XML format and then a document type definition verifies the XML conformity. Finally, in case of success a procedural modeller produces the geometry according with a user demands.

Recent works [37, 38] include some additional features. Yong et al. [37] reported the improvements made to the previous semantic-based solution [10] that intended to overcome some noted issues regarding procedural modelling, such as the lack of annotations for digital architectural heritage which also impacts in the identification of procedural rules for digital reconstruction of missing monuments. Such issues triggered the proposal of an approach that puts together semantics, machine intelligence, data mining and automatic annotations. Later, a granular ontology approach was suggested by Liu et al. [38] to allow a collaborative ontology design based on the sub-concepts provided by users of different expertise areas.

2.2.11.2 Semi-automatic Generation of Ontology-Based Building Facades

Bellotti et al. [39] proposed a statistical algorithm for the procedural generation of urban areas, capable of producing virtual buildings composed by several ontology-based facade components, considering georeferenced layouts and template styles statistically selected. The referred ontology is used to organize and relate several architectonic elements of facades such as windows, doors or roofs. The authors used the algorithm for the generation of urban environments in the context of cultural heritage promotion and in a 3D movie. Both were presented to users who rate positively the reconstruction, despite the absence of architectonic details, provided by elements like balconies or porches.

2.3 Procedural Modelling of Virtual Traversable Buildings

Besides the focus on urban environments and buildings outer facades, several other approaches address the procedural modelling of 3D traversable buildings. In the next subsections, they will be reviewed.

2.3.1 LaHave House: An Automated Architectural Design Service

Rau-Chaplin et al. [62] developed an automated architectural service to provide a collaborative way of plan and design modern houses, foreseeing the interaction between architects and final users (clients or service consumers). A design engine (that generates over 100,000 different house designs), a customization component (for the final user) and a building configuration tool are the three main components of this architectural service endowed with some automatic capabilities to process house designs.

2.3.2 Procedural Generation of Buildings Using Graphs and Expansion Algorithms

Martin [63] presented a procedural algorithm to generate residential units. Considering a grammar and user-defined constraints, the process starts by generating a graph

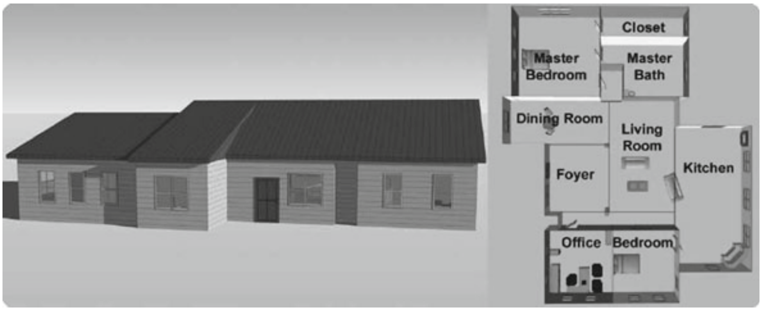

Fig. 2.4 An house with interiors produced by Martin's approach [64]: first, a graph system is used to connect and place rooms and then those rooms are increased in a fixed area using a Monte Carlo algorithm

in which nodes represent rooms and edges represent the connections between rooms. Then, those rooms are distributed within a footprint and, finally, a Monte Carlo algorithm expands them until equilibrium is reached. Figure 2.4 depicts the result of the author's approach. In his thesis, Martin [64] discusses graph-based techniques to generate structures of connected rooms and to place them within a given area. The process of room expansion is also explained in detail. Regardless of the similarities with the Monte Carlo algorithm, it seems that this designation was replaced by square bubble growing algorithm.

A similar solution—based on seed and growth approach—was proposed by Long [65]. This system considers as input an area for feature placement that can be rectangular or non-rectangular. Still, it is confined to shapes formed exclusively by right angles. The process of area fill includes the determination of feature types (shape variety) and also their placement and adjustment in the available space. The author argues that his technique is more effective than squarified treemaps approach (addressed in a later subsection).

2.3.3 Building Indoors Generation Using Constructive Solid Geometry Algorithms

Bradley [66] proposed a semi-automatic methodology to produce traversable buildings which considers two types of input: American Standard Code for Information Interchange (ASCII) files with heuristics and room definitions and building outline. Some 2D and 3D Constructive Solid Geometry (CSG) algorithms are used to deal with division of the building outline into rectangular cells, walls extrusion, placement of door and windows, among other operations. The outputs results are 3D traversable buildings devoid of any details.

2.3.4 *Lazy Production of Virtual Building Interiors*

The concept of real-time generation of interior divisions was exploited by Hahn et al. [6]. The plans of each floor are generated through a random division of the floor into rectangular divisions and hall passages. The division process starts by defining a temporary region which is then divided in smaller temporary regions and built regions. This is repeated iteratively until each region becomes a built region. The authors also implemented some architectonic rules to ensure the proper generation of the final geometry.

2.3.5 *Interior Rooms Generated Through Voronoi Diagrams and Constrained by Convex Layouts*

Dahl and Rinde [67] developed an algorithm that generates rooms inside polygons representing building limits (Fig. 2.5). It receives a set of specifications for building generation such as constraint walls, windows and doors and also a couple of parameters defining region types and room types. Then, it mounts the building skeleton with a mandatory corridor for layouts with large dimensions and creates regions for grouping sub-regions or final rooms. These last elements are generated using a weighted Voronoi diagram that spreads rooms inside regions considering the desired room weights. Meanwhile, a room graph is created in order to connect rooms and then, the room types are defined accordingly with the input parameters.

This approach generates traversable buildings considering irregular shapes as constraints. However, some issues were identified: the impossibility of managing the number and size of rooms to be generated; the confinement to the generation of structures disregarding geometric holes; finally, the absence of visual details such as textures.

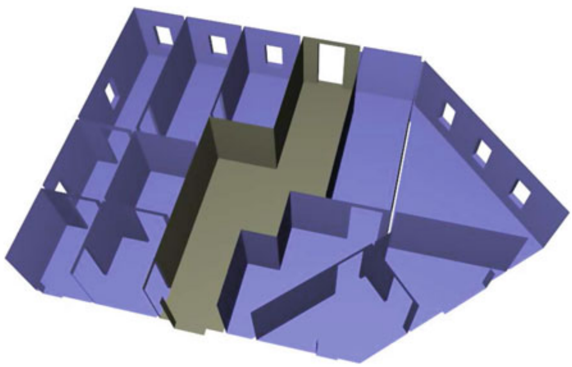

Fig. 2.5 Dahl and Rinde [67] used a Voronoi diagram to subdivide an irregular polygon into rooms

2.3.6 Rule-Based Generation/Reconstruction of Buildings

Rodrigues [68, 69] proposed a rule-based method capable of generating portuguese houses (regulated by [70]) and reconstructing ancient roman houses (regulated by Maciel [71]). Several steps such as room graph definition, floor plan composition using shapes and extrusions considering doors and windows lead to the 3D model achievement. Rodrigues et al. [72] revealed, in detail, the methods and processes carried out to generate roman houses, which include spontaneous L-systems, multi-layer graphs defining containers and also room connections, among other relevant procedural operations. Later, Rodrigues et al. [73] extended their work to provide virtual building models in several formats (for example, X3D and VRML). The improvement foresees the integration with virtual platforms like Second Life. The aforementioned works culminated on a doctoral thesis [74] that has some images depicting virtual reconstructions of roman houses like the ones that can be seen in Fig. 2.6.

2.3.7 Squarified Treemaps for Virtual Buildings Generation

The squarified treemap [75] is a subdivision strategy (Fig. 2.7) adapted for the generation of buildings with interiors. The strategy consists in splitting rectangular areas considering a set of weights and the following key rule: in each division it must be ensured that the aspect ratio has the closest value to 1. Marson and Musse [7] applied it to subdivide rectangular building footprints into functional zones and then rooms. The final step is the placement of a corridor to connect the unreachable rooms. Mirahmadi and Shami [76] used the same method with some optimizations at the corridor placement step, to increase the realism of the architectural designs.

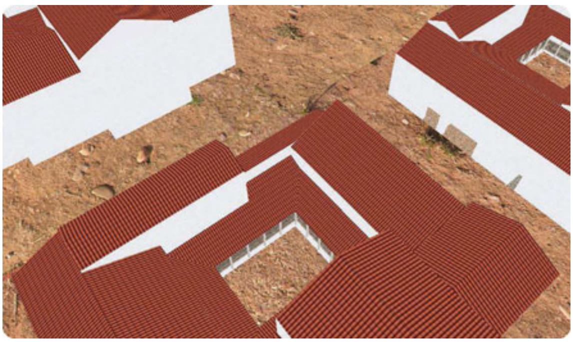

Fig. 2.6 Virtual environment depicting roman houses [74]. A set of reconstruction rules is used to guide the cultural heritage recovering process from the floor plan stage to the complete 3D building model

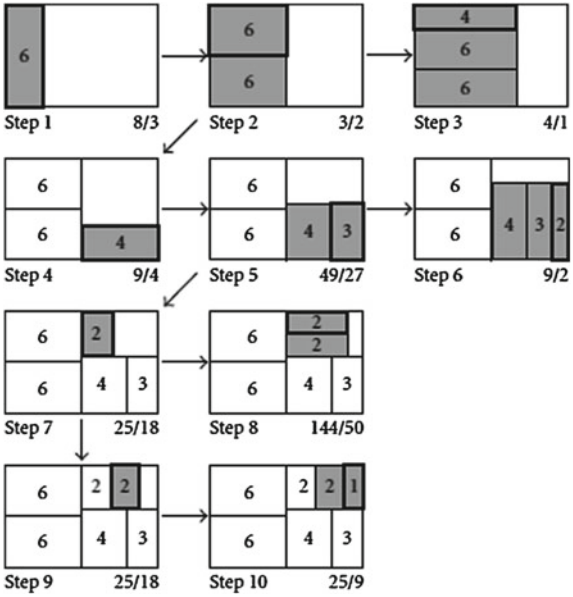

Fig. 2.7 Squarified treemaps operation [75]. Considering the following sequence of rectangular areas 6, 6, 4, 3, 2, 2 and 1, a descendant sorting algorithm optimizes the process which tries to arrange the areas inside a rectangular container, in order to find the aspect ratio with the closest value to 1, in each iteration

2.3.8 Residential Buildings Generation Based on Bayesian Networks

An approach for generating virtual models of residential buildings, focusing the production of floor plans highly based on architectonic knowledge, was presented by Merrell et al. [8]. In this approach, a set of high-level requirements must be provided by an user and then expanded into an architectural program, that consists in a bubble diagram providing a list of rooms, respective sizes and connections. The architectural program—compatible with the referred requirements—is provided by a bayesian network that contains 120 of these structures manually encoded. Next, the floor plan is determined through stochastic optimization that occurs using operations such as wall sliding and room swapping. Afterwards, the 3D model generation takes place, considering a selected template that aims to garnish the building with an appearance style.

2.3.9 Grid Approach Focusing Floor Plan Generation

A method inspired in geometric grids used by architects to aid the manual drawing of floor plans was proposed by Lopes et al. [77], who developed a grid-based algorithm to allow the placement and expansion of rooms. Taking into consideration a few user inputs (for example grid size, list of room, respective types and dimension), first, the functional zones are determined (e.g. public and private zones) as it was also suggested by Marson and Musse [7]. Then, rooms are placed in the proper zones and expanded using the grid approach. The placement step defines the appropriated position of a room in the grid. Then, growth methods based on cells filling are applied to make the rooms expand through the available area. At last, the connections are processed accordingly with connectivity requirements or definition rules to complete the floor plan.

2.3.10 Generative Modelling Language (GML) for Virtual Buildings

GML—acronym for generative modelling language in the specific context of this work (different from Geographic Modelling Language)—is an imperative programming language used to define geometric structures based on split grammars [78], that also supports the generation of building interiors. The available operations include the creation, modification and termination of scopes and also relative and absolute subdivisions. The tool effectiveness was demonstrated through a case study that consisted in the reconstruction of the University of Technology in Graz, Austria, which involved several steps ranging from the identification of superstructures (subparts of the same building) till the linkage of floors using staircases. More information about this language can be found in [79].

2.3.11 Component-Based Modelling of Virtual Buildings

Leblanc et al. [80] proposed a tool that requires programming skills to support the component-oriented modelling of virtual buildings. Each component can be seen as a geometric element of the building (2D or 3D shape), composed by faces or regions. The referred programming tool allows some operations upon components such as attribute alteration (add, modify or delete), component connection (consists on linking a component coordinates system to another component region) or creation (that includes, for example, instantiating, slicing, splitting, extruding or roofing components for geometric transformations or decomposition). Despite the high-level of freedom, a sequence of steps is suggested towards the attainment of the expected virtual models: space partitioning into roofs, storeys and rooms; extrusion of interiors and exteriors; placement of architectural elements such as windows, doors and balconies; and placement of furniture.

2.3.12 Producing Virtual 3D Buildings from Pre-designed Floor Plans

A tool for the expeditious production of 3D virtual buildings, including interiors and outer facades, considering scanned floor plans among other input information such as photos, room areas, location and surroundings was proposed by Santos et al. [81]. At a preparatory stage, some user inputs have to be provided in order to accomplish a few operations such as the floor plan vectorization through digital decal drawings and the indication of staircases. Subsequently, the entire geometry providing 3D visualization is produced considering the following steps: wall extrusion, placement of doors and windows, inclusion of interior furniture and, finally, roofs creation. Furthermore, some virtual elements are generated out-of-doors to integrate the building surroundings. Other interesting features that worth to highlight are, for example, the rule-based furniture placement, the realistic texturing and the (manual or automatic) creation of virtual visitations.

The prototype Building Model Generator (BMG) was another proposal that appeared even earlier than the previously mentioned [82]. The prototype receives as input 2D floor plans, previously developed in a commercial CAD software. Then, the floor plans are properly converted to a compatible BMG format and the prototype posteriorly detects and corrects small geometrical inconsistencies. Those floor plans are analyzed in order to extract rooms and portals. Further steps include the extrusion of walls and also the proper placement of doors and windows. An interactive editor is provided to allow some adjustments on building elements, including materials. Finally, there is a complementary tool—a staircase generator—that enables the proper placement of staircases. The final result is a virtual 3D model of a building with connected floors.

2.3.13 Ontology-Based Generation of Traversable Buildings

The ontology-based modelling of traversable buildings was also addressed by a few authors. The current subsection will expose each work.

2.3.13.1 Virtual World Grammar for Automatic Generation of Virtual Worlds

An ontology-based virtual world (VW) generator focusing institutions was proposed by Trescak et al. [9]. The case study provided by the authors is an auction system, represented by a set of activities: admission, item registration, auction, auction info and also entrances and exits. Then, a VW Grammar is constructed, based on an ontology. The referred ontology comprises both activities and a shape grammar. An object mapping is also performed in order to relate ontological objects with the proper shape grammar responsible for its representation in the virtual world. After the completion of VW Grammar, a set of heuristics, validations and evaluations regulate the generation of the institution, that is made in two main steps: floor plan production and 3D virtual model transformation.

2.3.13.2 Framework of Procedural Techniques

Tutenel et al. [83] proposed an inclusive framework that produces virtual buildings using several procedural techniques. Building floor plan definitions must be provided using declarative language instructions, that trigger a sequence of steps including the loading of the floor plans and the selection of the techniques that should be used in each phase of generation. For example, lot (3D building gross model) generation can be made through CGA shape grammar and the floor plan might be produced using a grid approach like the one presented by Lopes et al. [77]. Some regulation mechanisms are applied to guide the whole process aiming, for example, the assignment of a convenient procedural technique for each building element to be generated and the avoidance of functional conflicts (bad placement of objects, such as regular room windows in bathrooms).

2.4 Summary

This chapter started by presenting the versatility of ontologies applied in a wide variety of solutions incorporating, 3D virtual models to serve areas such as medicine, industry, cultural heritage and design. Then, standard knowledge-based representations for virtual urban environments were addressed: CityGML was referred and shown along with the building definition proposed by this standard. Moreover, the procedural modelling solutions were extensively documented, including the generation/reconstruction of virtual urban environments and traversable buildings. Each topic had a few works that also consider ontologies.

Summing up, there is a wide variety of procedural modelling works addressing the generation of virtual buildings' interiors with distinct approaches: tools based on shape grammars [62], CSG algorithms [66], expanding rooms [64], grid-based approaches [77], squarified treemaps [7, 76], rule-based solutions [74] with architectonic awareness [70, 71], generative modelling languages [78, 80] and others. However, most of them only deal with rectangular shapes or geometries uniquely formed by right angles. Alternatively, Dahl and Rinde [67] addressed the production of virtual buildings composed by convex shapes with a Voronoi diagram approach. However, some issues remain to be addressed, including: (1) the lack of support to holes in the middle of floor plans; (2) the apparent inappropriateness of the rooms' generation approach when the control over rooms' features (such as geometric constraints and arrangements) is required; and (3) the absence of textures that hamper the visual distinction of, for example, room types.

Considering the referred issues, a novel procedural modelling methodology will be presented along with a building ontology—for regulation purposes—inspired in the exposed ontology works and in the CityGML standard. The referred methodology contains a process that relies in some of the aforementioned procedural modelling solutions, specifically in the floor plan generation and the 3D transformation. The refinements made during the development of this methodology will be progressively presented until the final solution, that supports the generation of buildings constrained by arbitrary shapes, with rooms limited by a configurable number of inner walls. This solution also intends to enable the generation of virtual buildings in different architectural contexts, taking advantage from the integrated building ontology. General purpose buildings belonging to the generic ontology, T2 houses complying with portuguese architectural rules and buildings based on roman architecture—specifically *domus*—are addressed as case studies and for demonstration purposes. Moreover, a stochastic approach regarding the random generation of virtual generic buildings will be presented as a way of automating the virtual models production using this methodology.

References

1. Parish, Y.I.H., Müller, P.: Procedural modeling of cities. In: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '01, pp. 301–308, New York, NY, USA, 2001. ACM (2001). ISBN 1-58113-374-X. doi:10.1145/383259.383292
2. Wonka, P., Wimmer, M., Sillion, F., Ribarsky, W.: Instant architecture. ACM Trans. Graph. **22**(3), 669–677 (2003). ISSN 0730-0301. doi:10.1145/882262.882324
3. Müller, P., Wonka, P., Haegler, S., Ulmer, A., Van Gool, L.: Procedural modeling of buildings. ACM Trans. Graph. **25**(3), 614–623 (2006). ISSN 0730-0301. doi:10.1145/1141911.1141931
4. Carrozzino, M., Tecchia, F., Bergamasco, M.: Urban procedural modeling for real-time rendering. In: Proceedings of the 3rd ISPRS International Workshop 3D-ARCH, vol. 2009 (2009)
5. Silva, P.B., Coelho, A.: Procedural modeling for realistic virtual worlds development. J. Virtual Worlds Res. **4**(1) (2011). doi:10.4101/jvwr.v4i1.2109. https://journals.tdl.org/jvwr/index.php/jvwr/article/view/2109/5541
6. Hahn, E., Bose, P., Whitehead, A.: Lazy generation of building interiors in realtime. In: Canadian Conference on Electrical and Computer Engineering, 2006. CCECE '06, pp. 2441–2444, May 2006. doi:10.1109/CCECE.2006.277767
7. Marson, F., Raupp Musse, S.: Automatic real-time generation of floor plans based on squarified treemaps algorithm. Int. J. Comput. Games Technol. **2010**, 7:1–7:10 (2010). ISSN 1687-7047. doi:10.1155/2010/624817
8. Merrell, P., Schkufza, E., Koltun, V.: Computer-generated residential building layouts. ACM Trans. Graph. **29**(6), 181:1–181:12 (2010). ISSN 0730-0301. doi:10.1145/1882261.1866203
9. Trescak, T., Esteva, M., Rodriguez, I.: A virtual world grammar for automatic generation of virtual worlds. Vis. Comput. **26**(6-8), 521–531 (2010). ISSN 0178-2789. doi:10.1007/s00371-010-0473-7
10. Liu, Y., Xu, C., Pan, Z., Pan, Y.: Semantic modeling for ancient architecture of digital heritage. Comput. Graph. **30**(5), 800–814 (2006). ISSN 0097-8493. doi:10.1016/j.cag.2006.07.008. http://www.sciencedirect.com/science/article/pii/S0097849306001294
11. Liu, Y., Xu, C., Zhang, Q., Pan, Y.: The smart architect: scalable ontology-based modeling of ancient chinese architectures. IEEE Intell. Syst. **23**(1), 49–56 (2008). ISSN 1541-1672

12. Borgo, S., Guarino, N., Masolo, C.: An ontological theory of physical objects. In: Proceedings of Qualitative Reasoning 11th International Workshop, pp. 223–231 (1997)
13. Noy, N.F., Hafner, C.D.: The state of the art in ontology design: a survey and comparative review. AI Mag. **18**(3), 53 (1997)
14. Richard Benjamins, V., Fensel, D.: The ontological engineering initiative (ka)2 (1998)
15. Wand, Y., Weber, R.: An ontological model of an information system. IEEE Trans. Softw. Eng. **16**(11), 1282–1292 (1990)
16. Weber, R., Coopers & Lybrand: Ontological Foundations of Information Systems. Coopers & Lybrand accounting research methodology monograph. Coopers & Lybrand and the Accounting Association of Australia and New Zealand (1997)
17. van Heijst, G., Schreiber, A.Th., Wielinga, B.J.: Using explicit ontologies in kbs development. Int. J. Hum.-Comput. Stud. **46**(2), 183–292 (1997). ISSN 1071-5819. doi:10.1006/ijhc.1996.0090. http://www.sciencedirect.com/science/article/pii/S1071581996900907
18. Guarino, N.: Formal ontology and information systems. In: Proceedings of the First International Conference (FOIS'98), pp. 3–15. IOS Press (1998)
19. Chandrasekaran, B., Josephson, J.R., Benjamins, V.R.: What are ontologies, and why do we need them? IEEE Intell. Syst. Appl. **14**(1), 20–26 (1999). ISSN 1094-7167. doi:10.1109/5254.747902
20. Jannin, P., Morandi, X.: Surgical models for computer-assisted neurosurgery. Neuroimage **37**(3), 783–791 (2007)
21. Attene, M., Robbiano, F., Spagnuolo, M., Falcidieno, B.: Characterization of 3d shape parts for semantic annotation. Comput.-Aided Des. **41**(10), 756–763 (2009)
22. Béhé, F., Galland, S., Gaud, N., Nicolle, C., Koukam, A.: An ontology-based metamodel for multiagent-based simulations. Simul. Modell. Pract. Theory **40**(0), 64–85 (2014). ISSN 1569-190X. doi:10.1016/j.simpat.2013.09.002. http://www.sciencedirect.com/science/article/pii/S1569190X13001342
23. Pittarello, F., De Faveri, A.: Semantic description of 3d environments: a proposal based on web standards. In: Proceedings of the Eleventh International Conference on 3D Web Technology, pp. 85–95. ACM (2006)
24. García-Rojas, A., Gutiérrez, M., Thalmann, D.: Visual creation of inhabited 3d environments. Vis. Comput. **24**(7–9), 719–726 (2008)
25. Lee, J.Y., Seo, D.W., Rhee, G.: Visualization and interaction of pervasive services using context-aware augmented reality. Expert Syst. Appl. **35**(4), 1873–1882 (2008). ISSN 0957-4174. doi:10.1016/j.eswa.2007.08.092. http://www.sciencedirect.com/science/article/pii/S0957417407003818
26. Hunter, J., Gerber, A.: Harvesting community annotations on 3d models of museum artefacts to enhance knowledge, discovery and re-use. J. Cult. Herit. **11**(1), 81–90 (2010)
27. Martins TMP: Ontologia urbana para ambiente virtual colaborativo no contexto do planeamento e gestao municipais. Master's thesis, Faculdade de Engenharia da Universidade do Porto (2011)
28. Colledani, M., Pedrielli, G., Terkaj, W., Urgo, M.: Integrated virtual platform for manufacturing systems design. Proc. CIRP **7**, 425–430 (2013)
29. Flotynski, J.: Semantic modelling of interactive 3d content with domain-specific ontologies. Proc. Comput. Sci. **35**(0), 531–540 (2014). ISSN 1877-0509. doi:10.1016/j.procs.2014.08.134. http://www.sciencedirect.com/science/article/pii/S1877050914010990. In: Proceedings of the Knowledge-Based and Intelligent Information & Engineering Systems 18th Annual Conference, KES-2014 Gdynia, Poland, September 2014
30. Mignard, C., Nicolle, C.: Merging bim and gis using ontologies application to urban facility management in active3d. Comput. Ind. **65**(9), 1276–1290 (2014). ISSN 0166-3615. doi:10.1016/j.compind.2014.07.008. http://www.sciencedirect.com/science/article/pii/S0166361514001432. Special Issue on The Role of Ontologies in Future Web-based Industrial Enterprises
31. Galambos, P., Csapó, Á., Zentay, P., Fülöp, I.M., Haidegger, T., Baranyi, P., Rudas, I.J.: Design, programming and orchestration of heterogeneous manufacturing systems through vr-powered remote collaboration. Robot. Comput.-Integr. Manuf. (2014). ISSN 0736-5845. doi:10.1016/j.rcim.2014.08.012. http://www.sciencedirect.com/science/article/pii/S0736584514000738

32. Gröger, G., Kolbe, T.H., Czerwinski, A., Nagel, C.: OGC City Geography Markup Language (CityGML) Encoding Standard. Technical report, Open Geospatial Consortium (2008)
33. Gröger, G., Kolbe, T.H., Nagel, C., Häfele, K.-H.: OGC City Geography Markup Language (CityGML) Encoding Standard. Technical report, Open Geospatial Consortium (2012)
34. Kolbe, T.H.: Representing and exchanging 3d city models with citygml. In: 3D Geo-information Sciences, pp. 15–31. Springer (2009)
35. Liebich, T.: Ifc4–the new buildingsmart standard (2013)
36. BuildingSMART: Industry foundation classes release 4. http://www.buildingsmart-tech.org/ifc/IFC4/final/html/ (2013). Accessed May 2015
37. Yong, L., Mingmin, Z., Yunliang, J., Haiying, Z.: Improving procedural modeling with semantics in digital architectural heritage. Comput. Graph. **36**(3), 178–184 (2012). ISSN 0097-8493. doi:10.1016/j.cag.2012.01.003. http://www.sciencedirect.com/science/article/pii/S0097849312000040. Novel Applications of VR
38. Liu, Y., Zheng, X., Tang, F., Chen, X.: Ontology design with a granular approach. Expert Syst. Appl. **41**(10), 4867–4877 (2014). ISSN 0957-4174. doi:10.1016/j.eswa.2014.02.019. http://www.sciencedirect.com/science/article/pii/S0957417414000827
39. Bellotti, F., Berta, R., Cardona, R., De Gloria, A.: An architectural approach to efficient 3d urban modeling. Comput. Graph. **35**(5), 1001–1012 (2011)
40. Lindenmayer, A.: Mathematical models for cellular interactions in development ii. simple and branching filaments with two-sided inputs. J. Theor. Biol. **18**(3), 300–315 (1968). ISSN 0022-5193. doi:10.1016/0022-5193(68)90080-5. http://www.sciencedirect.com/science/article/pii/0022519368900805
41. Coelho, A., Bessa, M., Augusto Sousa, A., Nunes Ferreira, F.: Expeditious modelling of virtual urban environments with geospatial l-systems. In: Computer Graphics Forum, vol. 26, pp. 769–782. Wiley Online Library (2007)
42. Larive, M., Gaildrat, V.: Wall grammar for building generation. In: Proceedings of the 4th International Conference on Computer Graphics and Interactive Techniques in Australasia and Southeast Asia, GRAPHITE '06, pp. 429–437, New York, NY, USA, 2006. ACM (2006). ISBN 1-59593-564-9. doi:10.1145/1174429.1174501
43. Felkel, P., Obdrzalek, S.: Straight skeleton implementation. In: Proceedings of Spring Conference on Computer Graphics (1998)
44. Eppstein, D., Erickson, J.: Raising roofs, crashing cycles, and playing pool: applications of a data structure for finding pairwise interactions. In: Proceedings of the Fourteenth Annual Symposium on Computational Geometry, SCG '98, pp. 58–67, New York, NY, USA, 1998. ACM (1998). ISBN 0-89791-973-4. doi:10.1145/276884.276891
45. Dikaiakou, M., Efthymiou, A., Chrysanthou, Y.: Modelling the walled city of nicosia. In: Proceedings of the 4th International Conference on Virtual Reality, Archaeology and Intelligent Cultural Heritage, VAST'03, pp. 61–70, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association (2003). ISBN 3-905673-08-8. doi:10.2312/VAST/VAST03/061-070
46. Aichholzer, O., Aurenhammer, F.: Straight Skeletons for General Polygonal Figures in the Plane, pp. 117–126. Springer (1996)
47. Greuter, S., Parker, J., Stewart, N., Leach, G.: Real-time procedural generation of 'pseudo infinite' cities. In: Proceedings of the 1st International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia, GRAPHITE '03, pp. 87–ff, New York, NY, USA, 2003. ACM (2003). ISBN 1-58113-578-5. doi:10.1145/604471.604490
48. Finkenzeller, D., Bender, J., Schmitt, A.: Feature-based decomposition of facades. In: Proceedings of the Virtual Concept (2005)
49. Finkenzeller, D., Schmitt, A.: Rapid modeling of complex building façades. Institut für Betriebs-und Dialogsysteme, Universität Karlsruhe (TH), Germany (2006)
50. Finkenzeller, D.: Detailed building facades. IEEE Comput. Graph. Appl. **28**(3), 58–66 (2008). ISSN 0272-1716. doi:10.1109/MCG.2008.50
51. Sipser, M.: Introduction to the Theory of Computation. Course Technology, Boston (1996)
52. Müller, P., Vereenooghe, T., Wonka, P., Paap, I., Van Gool, L.: Procedural 3d reconstruction of puuc buildings in xkipché. In: Eurographics Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST), pp. 139–146. EG (2006)

53. Dylla, K., Muller, P., Ulmer, A., Haegler, S., Fischer, B.: Rome reborn 2.0: a case study of virtual city reconstruction using procedural modeling techniques. In: Proceedings of Computer Applications and Quantitative Methods in Archaeology (2009)
54. Besuievsky, G., Patow, G.: Procedural modeling historical buildings for serious games. Virtual Archaeol. Rev. **4**(9), 160–166 (2013). ISSN 1989-9947. http://varjournal.es/doc/varj04_009_23.pdf
55. Tepavčević, B., Stojaković, V.: Procedural modeling in architecture based on statistical and fuzzy inference. Autom. Constr. **35**(0), 329–337 (2013). ISSN 0926-5805. doi:10.1016/j.autcon.2013.05.015. http://www.sciencedirect.com/science/article/pii/S0926580513000824
56. Müller, P., Zeng, G., Wonka, P., Van Gool L.: Image-based procedural modeling of facades. In: ACM Transactions on Graphics (TOG), vol. 26, p. 85. ACM (2007)
57. Koutsourakis, P., Simon, L., Teboul, O., Tziritas, G., Paragios, N.: Single view reconstruction using shape grammars for urban environments. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 1795–1802. IEEE (2009)
58. Simon, L., Teboul, O., Koutsourakis, P., Paragios, N.: Random exploration of the procedural space for single-view 3d modeling of buildings. Int. J. Comput. Vis. **93**(2), 253–271 (2011)
59. Simon, L.: Procedural reconstruction of buildings: towards large scale automatic 3D modeling of urban environments. Theses, Ecole Centrale Paris, July 2011. https://tel.archives-ouvertes.fr/tel-00637638
60. Sugihara, K., Hayashi, Y.: Automatic generation of 3d building models with multiple roofs. Tsinghua Sci. Technol. **13**, 368–374 (2008)
61. Silva, P.B.: Modelacao procedimental para desenvolvimento de jogos de computador. Master's thesis, Faculdade de Engenharia da Universidade do Porto (2010)
62. Rau-Chaplin, A., MacKay-Lyons, B., Spierenburg, P.: The lahave house project: towards and automated architectural design service. In: Proceedings of the International Conference on Computer Aided Design (CADEX-96), pp. 62–66. IEEE (1996)
63. Martin, J.: Algorithmic beauty of buildings methods for procedural building generation. Computer Science Honors Theses, p. 4 (2005)
64. Martin, J.: Procedural house generation: a method for dynamically generating floor plans. In: Symposium on Interactive 3D Graphics and Games (2006)
65. Long, J.H.: Room and feature placement for procedurally generated indoor environments. Ph.D. thesis, University of Abertay Dundee (2011)
66. Bradley, B.: Towards the procedural generation of urban building interiors. Ph.D. thesis, M.Sc. thesis, Game Programming, University of Hull (2005)
67. Dahl, A., Rinde, L.: Procedural generation of indoor environments. Master Thesis with shared authorship, Charmers University of Technology (2008)
68. Rodrigues, N., Dionísio, M., Gonçalves, A., Magalhães, L.M.G., Moura, J.P.: Rule-based generation of houses. Comput. Graph. Geom. **10**(2), 49–65 (2008). http://cgg-journal.com/2008-2/05/index.html
69. Rodrigues, N., Dionísio, M., Gonçalves, A., Magalhães, L.G., Moura, J.P., Chalmers, A.: Incorporating legal rules on procedural house generation. In: Proceedings of the 24th Spring Conference on Computer Graphics, SCCG '08, pp. 59–66, New York, NY, USA, 2008. ACM (2008). ISBN 978-1-60558-957-2. doi:10.1145/1921264.1921279
70. RGEU: Regulamento geral das edificações urbanas, decreto n. º 38382 (2008)
71. Maciel, M.: Vitrúvio - Tratado De Arquitetura. Ist Press (2006)
72. Rodrigues, N., Gonzaga Magalhães, L., Paulo Moura, J., Chalmers, A.: Automatic reconstruction of virtual heritage sites. In: Proceedings of the 9th International conference on Virtual Reality, Archaeology and Cultural Heritage, pp. 39–46. Eurographics Association (2008)
73. Rodrigues, N., Magalhães, L., Moura, J., Chalmers, A., Santos, F., Morgado, L.: Archhousegenerator—a framework for house generation. J. Virtual Worlds Res. **2**(5) (2010). ISSN 1941–8477. https://journals.tdl.org/jvwr/index.php/jvwr/article/view/846
74. Rodrigues, N.: Rule-based generation of virtual traversable architectural-period houses. Ph.D. thesis, University of Trás-os-Montes e Alto Douro (2010)

75. Bruls, M., Huizing, K., van Wijk, J.: Squarified treemaps. In: Proceedings of the Joint Eurographics and IEEE TCVG Symposium on Visualization, pp. 33–42. Press (1999)
76. Mirahmadi, M., Shami, A.: A novel algorithm for real-time procedural generation of building floor plans. CoRR, abs/1211.5842 (2012)
77. Lopes, R., Tim, T., Smelik, R.M., de Kraker, K.J., Bidarra, R.: A constrained growth method for procedural floor plan generation. In: GAMEON'10, November 2010. http://graphics.tudelft.nl/~rafa/myPapers/bidarra.GAMEON10.pdf
78. Hohmann, B., Havemann, S., Krispel, U., Fellner, D.: A GML shape grammar for semantically enriched 3d building models. Comput. Graph. **34**(4), 322–334 (2010). ISSN 0097-8493. doi:10.1016/j.cag.2010.05.007. http://www.sciencedirect.com/science/article/pii/S0097849310000749. Procedural Methods in Computer Graphics Illustrative Visualization
79. Havemann, S.: Generative Mesh Modeling. Ph.D. thesis (2005). http://deposit.ddb.de/cgi-bin/dokserv?idn=977813207
80. Leblanc, L., Houle, J., Poulin, P.: Component-based modeling of complete buildings. In: Proceedings of Graphics Interface 2011, GI '11, pp. 87–94, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2011. Canadian Human-Computer Communications Society. ISBN 978-1-4503-0693-5. http://dl.acm.org/citation.cfm?id=1992917.1992932
81. Santos, D.S.S., Dionísio, M., Rodrigues, N., Pereira, A.: Efficient creation of 3d models from buildings floor plans. Int. J. Interactive Worlds **2011**, 1–30 (2011). doi:10.5171/2011.897069
82. Lewis, R., Séquin. C.: Generation of 3d building models from 2d architectural plans. Comput.-Aided Des. **30**(10), 765–779 (1998). ISSN 0010-4485. doi:10.1016/S0010-4485(98)00031-1. http://www.sciencedirect.com/science/article/pii/S0010448598000311
83. Tutenel, T., Smelik, R.M., Lopes, R., de Kraker, K.J., Bidarra, R.: Generating consistent buildings: a semantic approach for integrating procedural techniques. IEEE Trans. Comput. Intell. AI Games **3**(3), 274–288 (2011). ISSN 1943-068X. doi:10.1109/TCIAIG.2011.2162842

http://www.springer.com/978-3-319-42371-5

Ontology-based Procedural Modelling of Traversable Buildings Composed by Arbitrary Shapes
Adão, T.; Magalhães, L.; Peres, E.
2016, VIII, 122 p. 42 illus., Softcover
ISBN: 978-3-319-42371-5