

Codes in Architecture

Werner Lonsing

Independent Researcher | Germany | sigradi2020@lonsing.com

Abstract

The design, development and writing software for computers has transformed from a niche technology into a means of production. Although computation, information technologies, and digitality have become part of the architectural design process, they are not considered as part of a code itself. By looking into history it can be shown that in architecture codes are already present in different forms. The codes found are used to mark the related periods and to point towards a future form of architectural code with computation involved.

Keywords: Architectural Code; Coding; Cartesian Architecture; Gothic; Renaissance;

INTRODUCTION

Code emerges in different forms and over time. Differentiated codes are means to transform ideas from creative minds into physical constructions. This concept is much older and broader than the expression 'code' reduced to 'computational code' suggest, although with the spread of computation in architecture the functions and roles of codes has become much more apparent. Computation, information technologies, and digitality are about to shape architecture as a whole, even though this rather broad approach is practically limited to the creation of new forms known as 'parametric design' and similar. The curtailment impedes the impact of the underlying techniques. Important aspects like smartness, interactiveness, responsiveness and other dynamic forms in buildings are not yet integrated into a holistic architectural code, assuming it exists. Therefor an important question this paper seeks to answer is, if there is something like an architectural code at all and if yes, could it be identified as such and how is it defined. While the first version of this text was focussed on coding as activity, it very soon became clear that the outcome of this activity, the code itself, is the matter that really needs attention. By looking into history it became evident that in architecture codes were already present in different forms and the codes laboriously found are sufficient to mark the period. From there some aspects of the upcoming architectural codes with computation involved are outlined.

CODES

'Code' originates from the latin noun 'caudex', which translates to 'tree trunk'. It transformed to 'codex' in the meaning of a book made of thin wooden strips coated with wax upon which one wrote (Houghton Mifflin Company, 2006, p.54). As 'code' it changed to a set signs and applied rules to transmit informations, and, as transmission over time, to fix and conserve them. The use of a code requires knowledge both of the signs and the rules applied to them. Only experts can use a code, and to become such an expert the code has to be learned. Then only these trained and knowledgeable coders can encode informations Without them encoded informations are useless. Knowledge limits the access. Examples are languages, literacy and especially secret codes.

The abstract concept of data coded as invariances to carry informations is a modern concept based on computation. Data can be stored on a medium, transferred and transformed, but has no meaning by itself. "Information is data that has been processed into a form that is meaningful to the recipient..." (Davis & Olson, 1985, p. 200). To process specific data on a designated computer a special set of instructions tailored to both data and machine is mandatory, the software program. "Source code (also referred to as source or code) is the version of software as it is originally written (i.e., typed into a computer) by a human in plain text (i.e., human readable alphanumeric characters)" (The Linux Information Project, LINFO). In essence code is a written set of instructions to process data on a computer. Writers of code as software are called coders, programmers or developers.

Architects are not supposed to write code, and if, it is some scripting inside a large software package. They never really code and never create a piece of software from scratch, make it running on a computer and tweak it on all ends, performance, UI, peripherals and design to make it better. Yet developing software is a creative process very similar to the design process of a building. The introduced constraints, in both cases of technical natures, need to be wrapped in an aesthetical presentation, although the graphics of a program on a screen are disguised as interface. Or are buildings interfaces? There is no need to argue further before looking at the nature of code. Both questions can be answered by looking at the canonical begin of architecture as discipline and the preceding period of the medieval builders.

The use of 'coding' as a noun in this paper strictly points towards the meaning of developing software on and for computers, while the usage of 'code' both as noun and verb covers a much broader range, but not 'architectural code' as comparable to something like a building code or understood as alteration in the meaning of a style so far.

STYLE

A word on style is needed, because a lot of attempts are made to identify the usage of computation as a specific style. While it is evident that often a change in code is reflected in a change of style, neither the assumption that the use of a computer already did change the underlying

code in architecture, as there is no real change yet, nor that a personal style found by computation claimed as an architectural style justifies the proclamation of a new style in general. Whatever new -ism is proposed, and parametricism is only one of them, is still premature. The line of work is not closed yet and an associated style can not be concluded like Vasari did it. New styles are not a concern of this paper.

However, there is probably an epoch or era coming to an end, a circumstance to observe. It opens the opportunity to conclude the past ages. The era since the invention of architecture and architects by Alberti and others starting the Renaissance up until the introduction of really computational architecture, which is about to emerge, can be regarded as a whole and likely characterized as a style, once the attributes are identified as common. This rather radical assumption is substantiated by the fact, that computers are already changing all aspects of our life. Therefor as a major undertaking of this paper the underlying codes in architecture are examined, probably identified and eventually differentiated against each other under the assumption, that different styles are supported by different codes.

DRAWINGS AND CODE

Before continuing with arguments about coding the subject of code in architecture itself needs to be examined: "The relationship between drawing and architecture is foundational yet paradoxical. As Robin Evans suggests, architecture could be defined by the struggle between the inherently two-dimensional plane of the drawing and the three-dimensional reality of space. Architects must fold the complexities of construction, materiality, and perspectival view into flat drawings while at the same time unfolding the abstract rationality of the drawing into much more complex space. For hundreds of years, architects primarily conducted their work in the parallel space of projective drawings: plan, section, and axonometric. With the advent of computer-aided design (CAD) software in the last few decades of the 20th century, the fundamental act of drawing was relatively unchanged despite the new and different toolset. Only as digital modeling and, more recently, building information modeling (BIM) have become the dominant methods used to design architecture has architects' relationship with drawing radically changed. After centuries of drawings being the essential mechanism to generate and critically engage architecture, we are at a moment where drawings might exist as only as an output of production" (Marcus & Kudless, 2018, p. 47). Both observations, the relation between drawings and buildings, and then the depiction of drawings as fading away due to computation need further exploration.

Although, there is no notion that drawings are code, it is exactly what they are. The relation between drawings and architecture is nothing else than a special case of the relation between code and information. Drawings as architectural plans are encoded architecture. The code carries the information out of the architectural mind into the being of a construction, no more, no less. The paradox between architecture and drawings can be resolved as the simple process of encoding and decoding architecture. The misinterpretation is probably caused by the quality of the code itself. As images architectural drawings are often regarded as art, as work on their own

despite the fact that their real purpose is the definition of a projected building.

Two questions are now at hand. At first, is it the only code for architectural purposes, or are there other codes? The second question is, how are the rules of the code in combination? Both questions can be answered by looking at the canonical begin of architecture as discipline and the preceding period of the medieval builders.

ARCHITECTURAL CODE IN HISTORY

Since during the Renaissance Vasari took up his stance against the medieval builders the architectural theory is divided into architecture and otherwise. The persons tasked to imagine and erect buildings were splitted into educated architects and, as if belonging to the tribe of the Goths, the preceding gothic builders.

BUILDERS' HIDDEN KNOWLEDGE

While the Renaissance has a well documented theoretical background, books are the hallmark of the epoch, the Gothic has not. "Over the past five hundred years, therefore, the logic of the Gothic design process has been less well understood, and less celebrated, than that of classical architecture" (Bork, 2014, p. 3). The expertise here is hidden, submerged or simply forgotten, although, there is no doubt about it, the builders never had a culture of secrecy. To disclose some of this knowledge for identifying the code of the builders and to avoid the almost 500 year old conflict builders vs. architects as theoretical dispute this paper looks briefly into the history of the detection of that medieval acumen by looking at several studies on the few passed on plans, documents and more and more importantly computational models reverse-engineering the buildings.

GOTHIC REVIVAL

The first mentionable reconsideration occurred as the revival of the Gothic in the nineteenth century. By completing unfinished cathedrals and building new gothically churches the overall interests were at that time rather in form and style than technology, while competing against other neoclassical styles. "Probably the best-known and most influential part of Viollet-le-Duc's architectural theory is that on restoration, which deliberately set out to put a building into an imaginary ideal state which in reality had never existed" (Kruft, 1994, p. 284).

LOOKING AT THE BUILDINGS

William Henry Goodyear, the son of the rubber pioneer, was the first to use technical equipment at the peek of his time to examine the buildings by itself. "A further innovation took place in 1910: Goodyear began to use a camera with an internal plumb line, built specially" (for him). (He) "accounted for the presence of these diverging uprights in medieval buildings" (Tallon, 2013, p.531). While his quest, the proof of the "Widening Refinement", that the Cathedral in Reims "was not constructed with perpendicular lines", was disputed and is now rejected, he was the first to discover, that the columns in Reims are moving outward. It was the first significant usage of advanced technology as an attempt to reverse engineer a medieval cathedral. During the sixties and seventies of the last century other scholars turned away from their books and started to look at the sources themselves.

Herbert wrote about John Harvey, who identified marks as templates in a dedicated room in the cathedral of York: "During the past three decades, scholars in medieval architecture have brought forward new and more detailed information about late Gothic architects and their design processes. ... (He) refutes earlier notions on the medieval architects that they were either anonymous masons working on the scaffold or clerics who practiced design as a sideline (Harvey 1972). Harvey insists that medieval architects functioned as full-time and highly skilled directors of construction, specially trained, well respected, and often celebrated for their work. He notes that medieval architects were trained from within the building crafts, usually in masonry or carpentry. He describes their early education in book learning, their years of craft apprenticeship, and their final four years of specialized training in drawing. In these final years '[their] main concern was to master and to memorize the very many problems in practical geometry involved in setting-out arch and vault voussoirs, tracery, and proportional design" (Herbert, 1993, p. 26).

COMPUTERS

The next generation used CAD to reconstruct the subjects to their research in much more detail and more comprehensive. They could correct their predecessors: "Harvey (1972) was convinced that the tracing house and its plaster floor were used directly for the creation of full-size templates. However, fresh observations made of the floor and, importantly, the rooms themselves suggest that this was not the case. In terms of the floors, neither York nor Wells provide consistent evidence of designs that could be used to produce templates for the cutting of stone. None of the designs are complete, even to a degree to allow production by symmetry" (Holton, 2006, pp. 1592-1593). Holton concludes: "Thus, it appears that the tracing houses at York and Wells functioned as design spaces as opposed to sites of direct template production, as viewed by Harvey (1972). These rooms served as a permanent area to experiment with new concepts, without wasting timber drawing boards or parchment" (Holton, 2006, p.1593). The capability of a 3D virtual model offered new observations like about the accessibility, e.g. how big a stone could be, to evaluate their findings in a broader context. While Harvey observed only the traces on the floor, Holton could see the room and its stairway from all views, too.

Handcrafted CAD was only an intermediate step to the now standard of point clouds created with laser scans. "The process of reverse engineering proportional systems of historic buildings has long been fraught with problems. One cannot assume, without knowing the specific conditions of acquisition, that existing plans are accurate enough to sustain the scrutiny necessary to resolve differences among potential proportional schemes" (Tallon, 2014, p.1), when Tallon compared different drawings and their methods of measurements at the cathedral in Bourges. Tallon, who really pioneered the technology, "was the first to use the scans to get inside medieval builders' heads. 'Every building moves,' he says. 'It heaves itself out of shape when foundations move, when the sun heats up on one side.' How the building moves reveals its original design and the choices that the master builder had to make when construction didn't go as

planned. Tracking this thought process requires precise measurements" (Shea 2015/2020).

INTERPRETATIONS

The work now continues by combining CAD and scans: "This article presents a geometrical analysis of Bourges Cathedral, based on the application of computer-aided design (CAD) techniques to the results of a recent and highly precise laser survey. This analysis reveals that the cathedral's original designer developed a tightly interlocking and strikingly unified design.[...] These results contribute to an ongoing debate about the use of 'ad quadratum' and 'ad triangulum' geometries in Gothic architecture, and they provide new evidence for the geometrical coherence of Gothic cathedral design. In methodological terms, meanwhile, this discussion demonstrates the potential of CAD-based geometrical analysis for the study of precisely surveyed medieval buildings" (Bork, 2014, p.1; Fig. 1). For the subject of this paper it is very clear that the medieval builders did have rules and the knowledge to execute them in order to erect buildings. Today scholars can prove it with their knowledge in combination with modern technology. As exhaustive as this research is, it is the only way to decipher these rules and identify them as code.

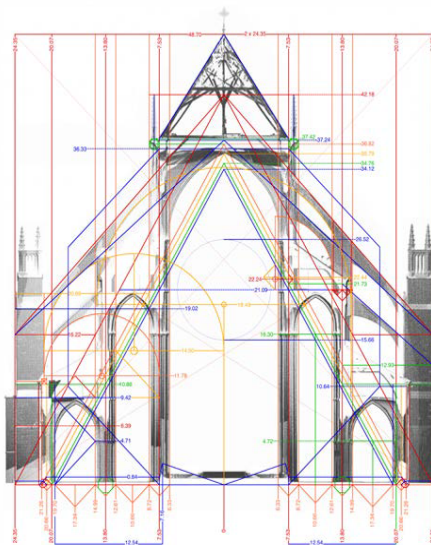


Fig. 1: The Geometry of Bourges Cathedral. Scan and drawings (Bork 2014; CC BY 4.0; <http://dx.doi.org/10.5334/ah.bz.s1>).

GOTHIC DRAWINGS, BOOKS & CODE

Although Vasari tried to sweep the builders under the carpet as illiterate, they did left their drawings and, although late and somewhat premature and incomplete, some writings. They are the only reports of their hands other than the monuments themselves.

DRAWINGS

Architectural drawings are now described as the invention of the builders during the thirteenth century. "It is no coincidence, though, that these drawings record the early history of a structure that would earn renown as one of the of the greatest of its age, since the use of drawings

enabled the members of the Strasbourg workshop to develop designs that would have been literally inconceivable to builders in previous centuries.[...]The rise of drawing-based design...was one of the most significant methodological developments in the history of medieval architecture" (Bork, 2005, p.2). Though the medieval builders invented the drawing as part of a building's design and developed it further geometrically and methodologically, both in means of drawing techniques and architecture, it was still the continuation of stone cutting and masonry and the inherited rules."However, Gothic design conventions governed the rules of the process more than the shape of the final product, which meant that the spatial relationships between building components varied far more widely in Gothic than in classical architecture" (Bork, 2014, p.2).

Others pointed to Christianity as yet another source of inspiration, where numbers for proportions were found by reading the bible. The impact of religion should be noted aside. "Almost from the outset, the Gothic evinced a scaleless, fractal character" (Akahane-Bryen, 2013, p.83). "In it we see a thoroughgoing emphasis on relative proportion and geometric interrelationships between constituent elements, and a radical lack of interest in absolute scale until the last moment in the design process, when some length had to be assigned to the module which governed the whole design" (Akahane-Bryen, 2013, pp.1-2). These rules were largely distributed person to person by showing and tell, without any means of preservation otherwise.

BOOKS

One of the first printed works of the modern theory of architecture in general – after the print of Alberti's *De re aedificatoria* in the previous year – was presented in 1486 by Mathes Roriczer. His 'Booklet Concerning Pinnacle Correctitude' ('*Büchlein von der Fialen Gerechtigkeit*') together with the 'Instructions' ('*Unterweisungen*') of Lorenz Lechler in 1516 and some few other writings are the only surviving theoretical instructions describing the builders' code. By examining some cathedrals as case studies Bork tried to "demonstrate that Gothic design methods involved the dynamic unfolding of geometrical constructions. This approach to design produced proportional relationships qualitatively different than those seen in the more static and module-based formal order of classicism" (Bork 2014). He pointed to "several important points about the use of geometrical proportioning strategies in Gothic architecture. They show, first of all, that centuries of sophisticated tradition informed the work of late Gothic authors like Roriczer and Lechler, even if their writings were not eloquent enough to compete with the work of their Renaissance rivals" (Bork 2014). Although they are the most important testimonials of the theory of the Gothic, "they ultimately prove frustrating to the modern researcher, since they fail to explain the origins of the complex dynamic forms." (Bork 2014). They are simply not comprehensive and conclusive. "So far, modern scholars have not been able to agree about the exact interpretation of this process...(T)he interpretation of the early publications by Roriczer vacillates, since these works concentrate on geometrical foundations and single motifs of late-Gothic design" (Hoppe, 2019, p. 571).

CODE

"Numerical and module-based thinking certainly played a role in Gothic design practice, but not to the exclusion of dynamic geometry. Instead, these were complementary strategies: sometimes geometrical constructions could be unfolded within modularly defined armatures, ... in other cases, modules could be combined to approximate geometrically determined proportions,... Most fundamentally, though, these examples begin to hint at the rich variety of geometrical planning strategies employed by Gothic designers, which deserve far more detailed and rigorous exploration than they have received to date. ...Enough good work has already been done ... to demonstrate that Gothic architecture embodied a complex procedurally based formal order whose conventions governed the dynamic unfolding of geometry, rather than fixed canons of proportion like those seen in classical architecture" (Bork, 2014, p.16). Drawn plans were auxiliary material, not the conceptual provider of the idea. Drawing as genuine part of the code was left to their successor, the theorist from the Renaissance, the then architects.

The medieval designers and builders had a decisive unique code. They had knowledge, rules, standards, traditions, strategies and means to communicate and distribute them. Their framework was part of the building-rather than the design-process. As actors they encoded their imaginary buildings and concepts on it in a different manner and in other forms than the later architects using standardized sets of drawings. Instead they established a direct line of production from the idea of a building to its construction without the detour of explicit plans. In this they had utilize a code similar to the eventual concept of a computational architecture with a direct digital line of production from a first visualization on site to an automated fabrication units without drawings at all.

RENAISSANCE

Covering the code of the Renaissance is much less demanding than that of the preceding period. The reason is simple: From the begin the architects of the Renaissance conveyed their whole design process in a body of theoretical work. There is nothing to decipher. It was laid out from the beginning in words and in drawings. They established themselves as literate and educated architects separated from the builders and artisan, still working with stone, bricks and mortar, and architecture as new denominations for the discipline. Only they can imagine a building, draw it and then give instructions. Since then the character of the profession has not changed very much through remarkable about five centuries now. Compared to the medieval builders with their easy going: "If you build it, they will come." mentality the architects had become part of a differentiated building process, which was mirrored in their products. The drawings as technical as they are, at a scale and with certain rules of lining, became almost naturalistic images of the conceived buildings. It was intended, because these types of drawings of imaginary architecture were meant as promises to the mostly uneducated building owners, who were no longer patrons. The perspective, in modern terms the rendering, was the icing, the oath on that promises.

By shaping architecture as a discipline the Renaissance introduced a couple of essential and distinctive elements:

the invention of the architect as individual, an explicit architectural theory and a common methodology based on the theory with ingredients like the use of models, the invention of the linear perspective and, a point rarely communicated, a dedicated drawing scheme composed out of floor plans, elevations from all sides and sections. A conclusive system of design was formed on top of the solid foundation of a comprehensive architectural theory. This design system, namely the model and the drawing scheme as its constituents will likely be identified as an effective architectural code in this paper.

CONCEPT OF IDEA AND THEORIES

The events leading to the first signature building of the Renaissance, the Florentine cupola, were triggered by Fioravanti's design in 1366. The complete set of data for the dome was coded as the model and stored in it. The information was preserved and carried through time without alteration. The latter part of the story is the well known completion of the cupola by Brunelleschi, who followed the design and since then is generally regarded as the first architect of the Renaissance. He also caused the first documented record of a perspective. The events of the design and construction of the dome triggered decisive inventions into the architectural design process, or by comparison to the builders, invented it at all. From this time on architectural designs were developed beforehand completely at a scale, architects were known by their names and their projects were documented as such.

At a broader angle beyond architecture, the philosophical concept of the idea is a momentous and copious source. "Ideas are among the most important items in Cartesian philosophy. ... As he says in a letter ... 1642, 'I am certain that I can have no knowledge of what is outside me except by means of the ideas I have within me.'" Descartes never produced any formal treatise or work dedicated specifically to the laying out of a theory of ideas. Even so, enough is included in (... his work...), that allows for a basic reconstruction of a theory" (Smith, 2019). The concept of an idea is one of the corner stones of the construct of thoughts as theoretical understructure of the architectural design process, as the idea turns into a very product by its formulation in code on some paper. Although the builders had the ideas of buildings as much as the architects of later periods, they did not know about it. They knew what and how they wanted to build, but the concept of ideas was not existing to them. Beginning with Brunelleschi and Alberti the architects became theorists on their own and thought about how to formulate the ideas of buildings they had in mind. How to draw plans, especially the invention of the central perspective, or the design and building of architectural models became the major subjects of the architects. Describing the idea of a building and formulate it theoretically as a complete architectural design became the essential task of the architect.

While the Renaissance was diverse from the beginning the general principles and common techniques were consolidated and did stabilize. "The governing principles of architecture had been set out by Alberti within the framework of an overall world order, by Francesco di Giorgio within a totally anthropometric context, and by Filarete and Francesco Colonna within the concept of a utopia. These conceptions were available in a great variety of interpretations, but were scarcely of use to an

architect with a commission to fulfil. This was the problem tackled by Sebastiano Serlio (1475– 1555/4). He sought to furnish practical rules for architecture, not for 'great minds' but so that 'any average [person] would be capable [of understanding them]'. He avoided theorising, and even brought it into some disrepute; with regard to perspective he wrote that he wished to impart only as much theory as an architect needed, ..." (Kruft, 1994, p.73). That practice certainly did help to canonize the rules of drawings representing architecture. It can be noted, that not the architecture of the Renaissance itself, but its presentations as ideas on paper combined with other factors like the invention of printed books propagated the new style almost unrivaled, as the builders could only refer verbally to their architecture or by their objects. Although their intent was obviously only the distribution of their designs they invented the first ubiquitous method of coding in architecture suitable to envision all building constructions as a fictional design.

THIRD DIMENSION

The presentation of a building's idea as a task challenged the proponents of the emerging architecture in a, except for models, unresolved manner, the third dimension. Its depiction as a property of buildings did become a central piece in the course of the formation of the architectural theory. Its most discussed achievement, the invention of the perspective, is a signature feature of the Renaissance.

MODELS

Models as technique to get a grasp on the third dimension were known before, "Aside from these examples, little evidence exists that architects regularly employed three-dimensional proportionally accurate models of proposed designs until the fourteenth century. Any use of scale models does not appear to have been important enough for architects to have made records of that use or to retain them for possible repairs"(Mindrup 2019). Once the late builders during the thirteenth and fourteenth century had built their models at larger scales to evaluate and demonstrate a design or construction, they were usually discarded or recycled. "At the beginning of the fourteenth century, however, the architectural model suddenly acquired a new significance among an administrative group of artisans and citizens acting as patrons for the reconstruction of the cathedral of Florence, Italy. [...] (T)he architectural model quickly became an ideal, easily comprehensible tool for communicating and recording the size and shape of complex three-dimensional forms; in some instances it also offered a reduced-scale demonstration of their method of construction" (Mindrup 2019). Additionally a certain level of abstraction was applied to them, as Alberti did note. He also pointed to the idea as ultimate source of the model by suggesting to make copies while preserving the idea in a principal or first one.

DRAWINGS

The emphasis was now on the drawings as an arrangement of lines. According to Alberti they have to be exact, at a scale and are composed to describe an object. By itself a line has no material, no scale or other physical properties, it has only a meaning as it signifies a change of state of the represented building projected on paper. From the rather simple thought of lines as visual representations, which in similar form might apply to the

drawings of the builders, Alberti developed his theory over the intersections of lines as edges to the surfaces and their perspectival deformations and then a sophisticated concept of raytracing. His perspectives are a technique or method to project objects out of the three-dimensional space onto the planar surface of a canvas. "Its central feature is the vanishing point, a single point toward which any set of parallel lines will seem to converge. If these lines are perpendicular to the picture plane, their vanishing point will be on the horizon, corresponding exactly to the position of the beholder's eye" (Janson & Janson, 2004, p.12,34). It was first described by Alberti. "A direct result ... was the little book on painting, *De pictura*, in which Alberti set out the new method, first formulated by Brunelleschi, of constructing three-dimensional space on a two-dimensional plane by the use of the *costruzione legittima* of monocular perspective, which was to dominate discussion about art for the next century" (Alberti, 1988, p.XII, Rykwert).

Orthogonal sections, horizontal as floor plans and vertical both as elevations and sections are the working horses of the presentations of an architectural design. In cases of elevations and floor plans the definition as sections is usually ignored and the theorists of the Renaissance failed about to discuss this matter in detail. Technically elevations are sections at a distance without slicing the building itself, while a floor plan is a horizontal slice through a building. Their appearance as such is undeniable since the drawings of Palladio. "[T]he architectural section breaks open contained space in order to show it as an elevation, forcing entry and revealing the interior to the distanced eye of the architect" (Evans, 2000, p.118). The architects did dissect their models in order to open them up to exhibit the interior. The theoretical work Alberti laid out in his books covers most aspects and the technical concept of visual graphics analogous to photography or computer graphics. Whether we like it or not, we have to acknowledge that the leap between the builders' and Alberti's theory is much bigger than the leap between his books and our theories in visual graphics.

CODE

Regarding the code the lines are its elements while the other components are related to the system of projection describing the rules. It appears, as if these rules, by contrast to the rules of drawings and certainly those of the perspectives, are never explicitly formulated, because all representations are abstract. They are referring solely to the form without any further, especially material information. However, combined with the new form of models dedicated to the ideas of the design instead to the construction it can be observed, that the third dimension in the presentation of a projected building is captured from two distinct sides, model and drawings. An outcome is a redundancy, meaning that either a model is created confirming to the drawings or plans are drawn after a model.

The success of this code is incredible. With the intention to spread their new ideas and promote them better than the builders before the architects developed exceptional features. At first, once a set of drawings is complete, the design of a building is perfectly described and its idea is manifested. Then the code is almost unidentifiable as such. By choosing naturalistic depictions of their building

the plan and drawings are considered as images or the like, but not as part of an abstract scheme. The final advantage is a direct result of the naturalistic drawings: The plans are easy to comprehend. Lay people, even children can identify almost all elements in the drawings.

ART OF CODING

Most of the accomplishments of the renaissance architects are well observed in architectural theory, but it seems as if their most influential and longest lasting achievement, the invention of the code, is still underestimated. Once established the code became ubiquitous. Almost without exception every building with an explicit formulated design is presented in this form. There are no other forms of code. Even modern CAD-systems follow the rules.

The referral to drawings as art like 'the art of drawing' is common enough and well accepted. But drawings are single drawings on a piece of paper, not the entire description of a building. Only the set of drawings as invented by the architects describe it thoroughly. If such a complete set is regarded as code and by assuming the drawings as artistic another inherent asset takes shape. By disguising their technical nature the elemental code appears as art. In fact the theorist developed their code as an art. It is by far their greatest achievement and can not be overesti-mated. For more than five hundred years their code is valid and still in use without considerable alterations, and as an art.

Table 1: The defining elements of the code:

Models Models are three-dimensional presentations at a given scale. They are complete. Fictional models as ideas are mandatory, physicals are not.
Completeness All drawings form a set, and this set of drawings is complete. There are floor plans of each level, elevations of all sides, and at least one vertical section through the building.
Planar All drawings are presented on a planar surface.
Scale All drawings are at a scale. If they form a set, all are at the same scale.
Orthogonal All drawing are parallel to the axes of the underlying cartesian coordinate system without exception. If necessary e.g. a wing building is attached to a main building at another as a rectangular angle, the coordinate system is rotated as a whole
Sections Sections are drawings as if the building is sliced vertically or horizontally at a given point. All plans except perspectives are sections.

CARTESIAN ARCHITECTURE

Still there are some minor problems. The code is neither recognized nor labelled as such. In addition some formal rules should be formulated. Because the carrying feature is the idea of a building, its imagination in an architectural mind, it seems consistently to use Descartes name similar to the well known Cartesian coordinates. Therefor the period beginning from the Renaissance up to the time when the code may become obsolete is likely to be named as 'Cartesian Architecture'. Its characteristics are solely defined through their code without any reference to some style of whatever detail has been attributed to architecture, because the buildings since then are all based on said code.

ARCHITECTURAL COMPUTER CODE

After identifying two historical architectural codes the focus is on computer codes, as it is a likely replacement.

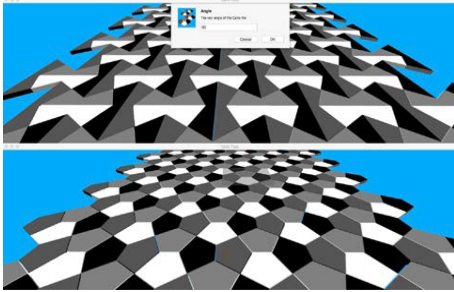


Fig. 2: Simpe program for Cairo-tiles with one exposed parameter. Source-code: <https://github.com/dialThat/CairoTiles>.

COMPUTERS IN ARCHITECTURE

Computational architecture requires computers. The question is, where is the computer and how does it define the architecture. The common case is the drawing machine inside an office mimicking the old cartesian design patterns. The computational design process by itself does not qualify, as it obeys the traditional concept. Nor does the majority of robotics fabrication, as the crafting of constructions by itself does not change the concept. These ideas can be discussed elsewhere, but there is only a marginal relation to an architectural code.

Only built objects with significant parts controlled by some computer to enhance the spatial environment with code developed by the mind of an architect are liberated from the Cartesian concept of idea, static drawing and compliant building. The concepts of these still few new spatial designs are differentiate by their use of auxiliary immersive devices. The first group is the field of responsive, adaptive, kinetic, reactive and/or performative architecture. The common characteristic are sensors and actuators, very often paired to change the physical appearance of a spatial construction, or the space itself. Pioneered by d'Estree Sterk (2003) as responsive architecture it already evolved into dissipative (Sprecher 2013) and living architecture (Beesly 2015) or morphogenetic (Menges 2012), biomimetic (El Ahmar & Fioravanti 2015) and similar designs. The second field comprises all architectures combining virtual immersive and real spatial experiences. In general the field of augmented and mixed reality (AR, MR), in architecture still considered as part of the design process, develops towards empathic computing (Billinghurst 2017). So far the only mentionable project here is polyrhythmic space (Ham 2019), though unfortunately only the single user behind the physical drums can completely appreciate it.

ONE PIECE OF SOFTWARE

Instead of lengthy theories a small piece of software has been developed in an afternoon to illustrate the implications of computer code outside those CAD-package. The standalone app simply creates Cairo-tiles depending on a set angle (Fig. 3). Graphics and UI are native software packages with an API, the language is Swift, depending on the underlying hardware, which

normally can only be chosen. As usual, the executable relies on some data, which are all hardwired into the code, colors, count, geometrical rules etc, except for the angle as sole exposed parameter. From a coder's point of view there is no real difference between data and parameters, as all parameters are data. The only difference is the extra load of work to handle the exposed parameters through the user-interface. Looking at the results it is very clear that the developed code, not the input on the exposed parameters injected by some user, creates the design according to the algorithm the developer did implement. In this sense the term 'parametric design' almost sounds like a joke. It was the developers decision what design to create and which parameters of the design are to be exposed. Even code snippets as parameters obey to this scheme. Without the executing software parameters make no sense at all. The expression 'parametric design' ignores this fact. If there is no software, there are no parameters. On the other hand by coding a design directly the distinction between data and parameter, when coder and user are identical, does not matter at all, but again, expression like 'data driven design' and similar ignore the impact of the code in the same way.

DEMONSTRATIONS AND PROJECTS

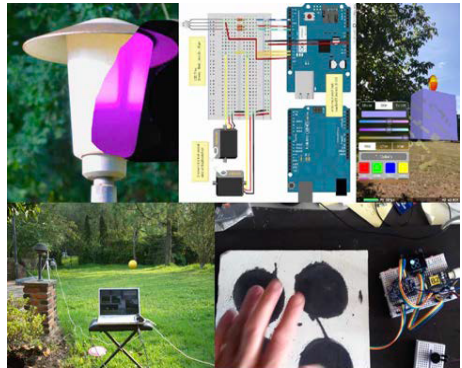


Fig. 3: Several coded projects by the author: 'Smart Street Light', 'Empathic lighting', 2 AR-systems and touchable concrete pads.

CONCLUSIVE STATEMENTS

As much as new rules and techniques can interfere with the architectural design process, developing the software as part of it can ignite something really new and become a design process on its own. We are facing computer programs not only as an auxiliary tool in the design's process but more and more as part of our buildings itself. Software is about to become architecture, whether we like it or no not. Therefore we must understand, that codes and coding are not bound to computation but to information and data. Our drawings as plans are at first code. Although we do not know yet what type of code will emerge due to computation, it is certain one will do. It will be created and established by people who can write in that code. If architects by themselves can master it, or if other disciplines trained in programming will move in, as we may anticipate it from previous times, is not decided yet and depends on us. We have to remember, that neither Brunelleschi nor Alberti were educated as architects when they started to replace the builders with

their brilliant code based on lines and graphics. Neither Behrens nor Gropius were, when Modernism took shape.

If we assume, that we as architects can produce software the design process may start with the layout of a software program which then transform into a building's design by either developing a special code, like we see it rudimentarily in parametric design, or become part of the design itself, being it as a technical entity in the realm of IoT and construction or as a virtual experience in a yet to come mixed reality spatial design.

EDUCATION

The task for the future is learning to code at all necessary levels. Use cases and applications in buildings must be formulated, discussed, working prototypes created and finally built. Otherwise people with their hand on the means of production, the code as capital, will drive the future of our architectural design. The step ahead of us is probably as steep as the step after the Gothic, when a wealth of abandoned projects had been left.

CODE AS ART

Architects are in general not coders and they are not aware of computer code as art like other artists, of them "most [...] seem to regard programming as an esoteric scientific discipline; they are keenly aware of its cultural mystique, envious of its potential profitability and eager to extract metaphors, imagery, and dramatic possibility from its history, but coding may as well be nuclear physics as far as relevance to their own daily practice is concerned. On the other hand many programmers regard themselves as artists. Since programmers create complex objects and care not just about function but also about beauty, they are just like painters and sculptors" (Chandra, 2014, pp. 1-2). The situation in architecture is even more delicate. Coding architects link computers by writing code for creating parts of buildings as architecture that represent the practical and aesthetical value of computation. It is valid for all the work of this kind, as it is coded art.

REFERENCES

- Alberti, L. B. (1988). On the Art of Building in Ten Books, trans. Joseph Rykwert, Neil Leach, and Robert Tavernor. Cambridge, Mass.: MIT Press, 1, 25.
- Belesky, P., Monacella, R. Burry, M. and Burry, J. (2015). A field in flux: exploring the application of computational design techniques to landscape architectural design problems. In ACADIA 2015 (pp. 194-201).
- Beesley, P. (2016). Epiphyte Chamber: Responsive Architecture and Dissipative Design. The Routledge Companion to Biology in Art and Architecture. New York: Routledge, 188-196.
- Billinghurst, M. (2017). The coming age of empathic computing.
- Bork, R. (2014). Dynamic unfolding and the conventions of procedure: Geometric proportioning strategies in Gothic architectural design. *Architectural Histories*, 2(1).
- Bork, R. (2005). Plan B and the geometry of Façade Design at Strasbourg Cathedral, 1250-1350. *The Journal of the Society of Architectural Historians*, 64(4), 442-473.
- Chandra, V. (2014). *Geek sublime: The beauty of code, the code of beauty*. Graywolf Press.
- Davis, G. B. and Olson, M. H. (1985). *Management information systems – Conceptual Foundations, Structure, and Development* (2nd ed.). New York: McGraw-Hill.
- d'Estrée Sterk, T. (2003). Using actuated tensegrity structures to produce a responsive architecture.
- Evans, R. (2000). *The projective cast: architecture and its three geometries*. MIT press.
- El Ahmar, S., & Fioravanti, A. (2015). Biomimetic-Computational Design for Double Facades in Hot Climates. *Proceedings of eCAADe 2015*.
- Goodyear, W. H. (1907). *The Widening Refinement in Rheims Cathedral*. Privately printed for the author.
- Ham, J., Woessner, U., Kieferle, J., & Harvey, L. (2019). Exploring the Affordances and Musico-Spatial Performance Opportunities of a Virtual Drumming Environment.
- Harvey, J. (1972). *The mediaeval architect*. London: Wayland.
- Herbert, D. M. (1993). *Architectural study drawings*. New York: Van Nostrand Reinhold.
- Holton, A. (2006). The working space of the Medieval Master Mason: the tracing houses of York Minster and Wells Cathedral. In *Proceedings of the Second International Congress on Construction History* (Vol. 2, pp. 1579-1597). Cambridge: Construction History Society.
- Houghton Mifflin Company. (2006). *More Word Histories and Mysteries: From Aardvark to Zombie*. Houghton Mifflin Harcourt.
- Hoppe, S. (2018). Translating the Past: Local Romanesque Architecture in Germany and its Fifteenth-Century Reinterpretation. In *The Quest for an Appropriate Past in Literature, Art and Architecture* (pp. 511-585). Brill.
- Janson, H. W., & Janson, A. F. (2004). *History of art: the Western tradition*. Prentice Hall Professional.
- Kruft, H. W. (1994). *History of architectural theory*. Princeton Architectural Press.
- Marcus, A. & Kudless, A. (2018). *Drawing Codes. Experimental protocols of architectural representation*. ACADIA, 46-55.
- Menges, A. (Ed.). (2012). *Material computation: Higher integration in morphogenetic design*. John Wiley & Sons.
- Mindrup, M. (2019). *The Architectural Model: Histories of the Miniature and the Prototype, the Exemplar and the Muse*. MIT Press.
- Palladio, A. (2002). *The four books on architecture*. Cambridge, MA (MIT Press).
- Shea, R. H. (2020/2015). Historian uses lasers to unlock mysteries of Gothic cathedrals. *National Geographic Magazine*. Retrieved from: <https://www.nationalgeographic.com/news/2015/06/150622-andrew-tallon-notre-dame-cathedral-laser-scan-art-history-medieval-gothic/>
- Smith, K. (2018). Descartes' Theory of Ideas. in Edward N. Zalta (ed.). *The Stanford Encyclopedia of Philosophy* (Winter 2018 Edition) Retrieved from: <https://plato.stanford.edu/archives/win2018/entries/descartes-ideas>.
- Sprecher, A., & Leblanc, F. (2013). Dissipative Architecture: The Informed Nature of Atopia. *Journal of Architectural Education*, 67(1), 27-30.
- Tallon, A. (2013). An Architecture of Perfection. *Journal of the Society of Architectural Historians*, 72(4), 530-554.
- Tallon, A. (2014). Divining Proportions in the Information Age. *Architectural Histories*, 2(1).
- Tallon, A. (2015). The Play of Daniel in the Cathedral of Beauvais. In *Resounding Images: Medieval Intersections of Art, Music, and Sound* (pp. 205-220).
- The Linux Information Project (LINFO). *Source Code Definition*. Retrieved from : http://www.linfo.org/source_code.html.