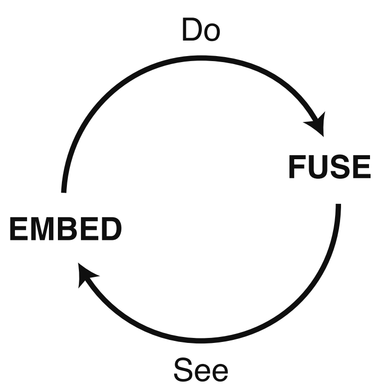
EXHIBIT 2

Observations

The best way to understand how visual calculating works in shape grammars, and why this isn't symbolic calculating, is to see what rules do when they're put to use. The left-hand side of a rule

$$A \rightarrow B$$

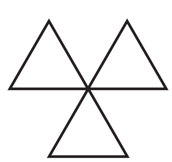
is key; it tells what to see in any shape C to which the rule applies. The shape A is an example that can be copied in some way to embed in C. Typically, this transformation is Euclidean, but it may be linear or something more general. The right-hand side of the rule tells what to do; B is usually an example of a shape that can be copied in the same way as A, to replace the copy of A that's embedded in C. Or B might be an instruction to make marks in a specified way or to throw paint as some artists urge, at least from the Renaissance on. Whatever it does, once the rule is tried, everything fuses to complete the embed-fuse cycle



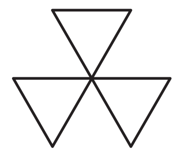
so that another rule, possibly the same one, can be applied from scratch, with no history/memory of anything that's happened before. What I do depends on what I see – now.

Three of my favorite examples provide the locus for this exhibit. In the trio, rules in shape grammars rely on embedding to exploit the inherent ambiguity in shapes that fuse. There are many surprises, all full of delight.

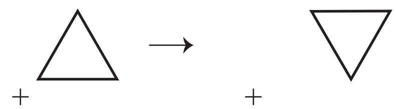
Example 1. Let's start with three equilateral triangles in a pinwheel



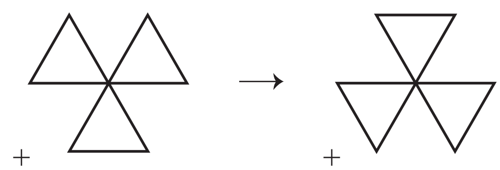
and rotate the pinwheel about its center



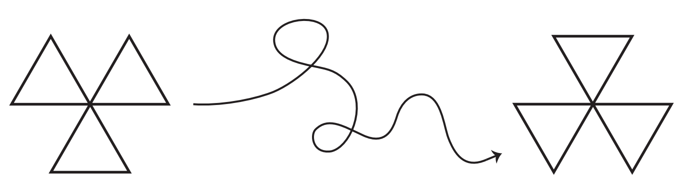
This is easy to do in the obvious way with the rule



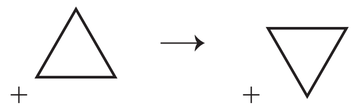
in the transformation schema $x \rightarrow t(x)$, that rotates a triangle about a vertex, pinned at the center of the pinwheel. I can use copies of the rule three times in the schema $x \rightarrow \Sigma$ F(prt(x)). Then, x is the pinwheel and prt(x) is each of its three triangles, so that different versions of F change (rotate) these parts separately in a coordinated sum. This describes the twisting action of the entire pinwheel in the rule



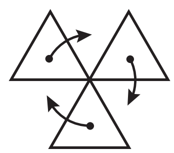
My description of this rule puts a pin at the center of the pinwheel to fix it in place as I rotate it, but there are alternative ways to track this movement in the rule. In fact, each of the triangles might have its own independent path, or divide into parts that move separately. Just looking at the rule, without knowing that it's the sum of three triangles and their rotations, I can't tell what motions are actually involved. Any will do that begin and end in the right way. Maybe the entire pinwheel takes a path like so –



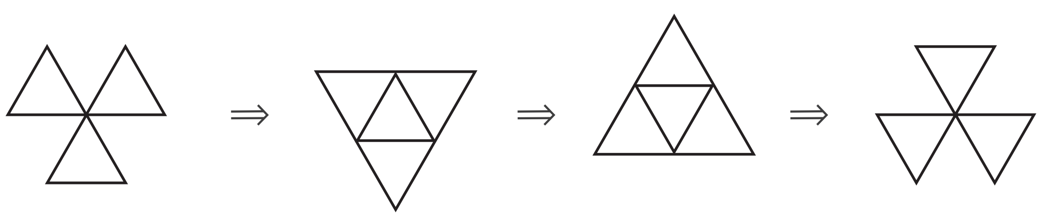
The two sides of the rule limit the transformation at its start and at its finish, without showing what happens going from one to the other. I can do this with other rules, usually in $x \rightarrow t(x)$, for the pinwheel and its parts in any way I please to spell out their motion in ever finer detail – inbetweening in a kind of animation. Now suppose I want to use copies of the rule



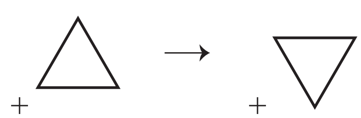
for F in the schema $x \rightarrow \Sigma$ F(prt(x)), to get the same result. This seems pretty unlikely, even spooky black magic. I guess strange stuff goes for more than particles and action at a distance – my new rule rotates a triangle about its center that's a fixed point, but the centers of triangles rotate as the pinwheel does



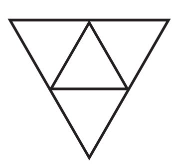
How can a rule that keeps the center of a triangle fixed/pinned in one place move it at the same time to another place? I said it was spooky. It sounds totally absurd – is this some new kind of paradox? Well, let's see what happens when I try. I can calculate just so –



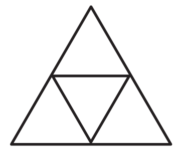
In the first and third steps, the rule



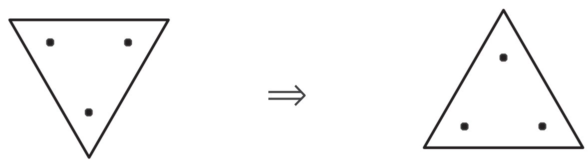
is used three times in the schema $x \rightarrow \Sigma$ F(prt(x)), and in the second step, it's used twice. The shape



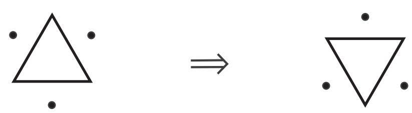
is produced from three triangles and changed as two, while the shape



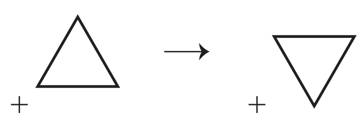
is produced from two triangles and changed as three. In the transition between these two shapes, the centers of the three triangles in the starting pinwheel are moved implicitly to the centers of the three triangles in its rotation – more precisely, when the large, outside triangle is rotated about its center



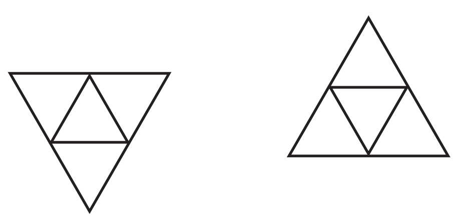
or reciprocally when the small, inside triangle is



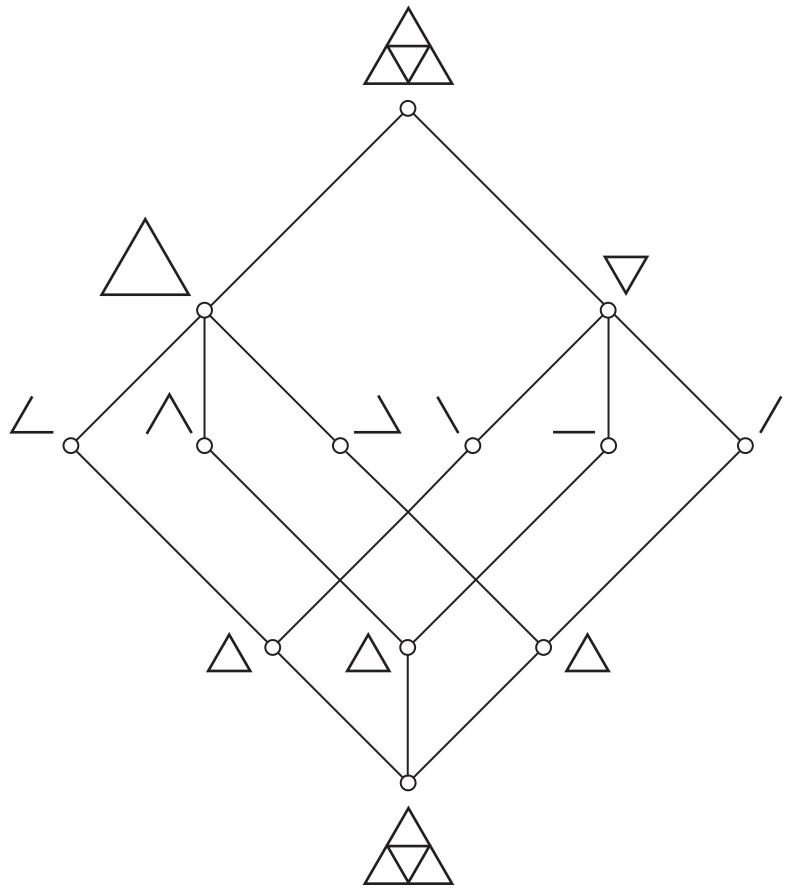
The three-step process from pinwheel to pinwheel is embed-fuse, embed-fuse, embed-fuse. This goes smoothly without a break, because the triangles in the second and third shapes fuse. As a result, the shapes are ambiguous with respect to the rule



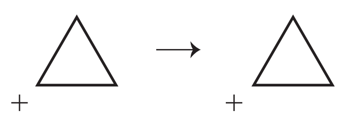
In retrospect, the rule shows how this works in topologies for the shapes



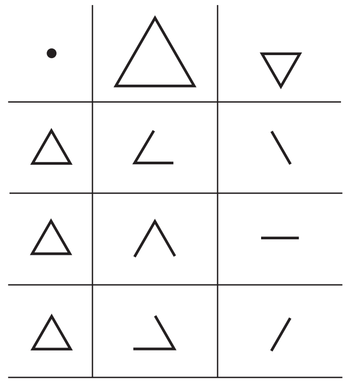
and the limiting pinwheels, to ensure continuity as I calculate.[1] But these relationships may be better summarized in another way – in this graph



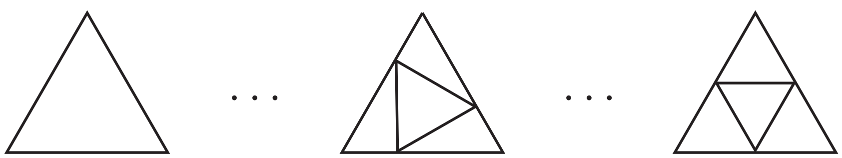
to reconcile incompatible descriptions. The top shape is two triangles in a tree defined using the identity for triangles



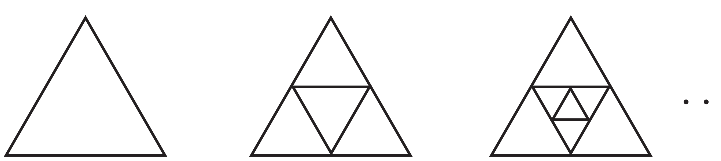
in the schema $x \rightarrow \Sigma$ F(prt(x)), and the bottom shape is three triangles, in terms of the same identity. Going from top to bottom in the graph shows how to combine the parts (units) at the six vertices of the middle tier, in order to change two triangles into three; going the other way around, three triangles are changed into two, combining the parts at the middle tier in a different way. The parts in the middle tier are the products of the triangles/parts distinguished in the trees at the top and bottom of the graph – these underlying (emergent/emanant) parts are required if I want to make sense of what I see in alternative ways, in non-hierarchical, contradictory/contrasting and incompatible/inconsistent views. The following table shows how these products are defined –



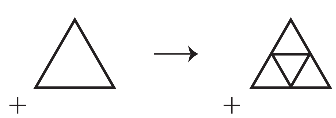
It may not feel that way, but seeing is a lot more complicated than it looks at first blush, especially when definite parts are involved. That’s why schemas and rules for visual calculating in shape grammars are so valuable; they let me calculate directly in terms of what I see, with no intervening structure to get in the way. There’s no reason to worry about underlying parts in visual analogies, or merging conflicting trees in graphs or defining topologies to ensure continuity for rules. Visual analogies and the like, given before I calculate, only make things hard; they’re defined meaningfully only after the fact, as a result of calculating – not before rules are tried, not before I’ve had a real chance to see as I please. My example also suggests others in a host of different ways, maybe first off, in the series



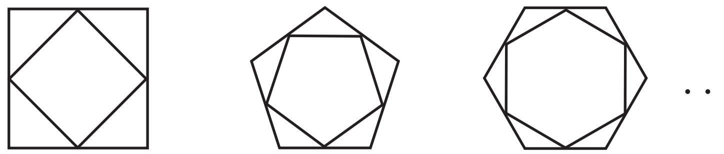
in which shapes vary parametrically as John McCarthy and Franz Reuleaux urge, or in the series



in which triangles are inscribed recursively using the rule



in the addition schema $x \rightarrow x + t(x)$. Other polygons work for all of this, as well – squares, pentagons, hexagons, etc.

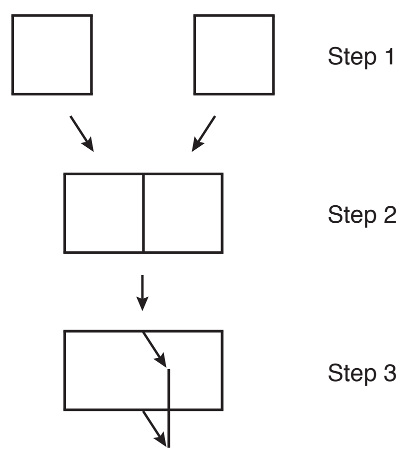


in an ongoing joust between pairs of polygons and multiple triangles. And irregular polygons are good, too – easy variations are everywhere I look. Perception is fluid, always in flux. There's no way around the embed-fuse cycle – it's the lifeblood of visual calculating.

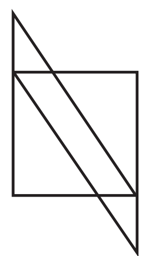
Example 2. Let's try my second example; it's a problem in object-oriented programming that was initially noticed a little more than two decades after it was already solved algorithmically, using shape grammars. (The original solution I have in mind is in *Pictorial and Formal Aspects of Shape and Shape Grammars* in terms of pictorial equivalence, subshapes, and reduction rules. No one reads it now – a few should.[2]) This bolsters my long-held view that art and design have a lot to contribute to computer science, or at least to calculating. Tellingly, the computer relationship doesn't seem nearly as strong going the other way around – so much for the wild claims and tiresome hype. But wait and see what my example holds. In his book *On the Origin of Objects*, Brian Cantwell Smith considers computers and intentionality. Sometimes in my weaker moments, I worry about the intentionality of his title that oddly nods to biology and Darwinian evolution. Nonetheless, Smith frames the issues nicely, when he "presses for a kind of representational flexibility that … object-oriented [computer] systems lack." This is evident in building information modeling (BIM) and parametric modeling, and the rest of computer-aided design (CAD). (For more on computers and AI in this vein, see fn 7.) In fact, representational flexibility is key, and not just for object-oriented systems. Surely, it's vital for poetry in the sweep of metaphor and all kinds of figuration ("a bush supposed a bear"), and for S. T. Coleridge's magical esemplastic power (imagination), when things fuse in order to re-create (re-divide) – "all objects (*as* objects) are essentially fixed and dead." Smith shows why this kind of representational flexibility is elusive for computers, in an entirely visual example –

> Many current systems are not only remarkably inflexible, but tend to hang on to ontological commitments more than is necessary. Thus consider the sequence of drawings shown [just below]. Suppose that the figure shown in step 2 was created … by first drawing a square, then duplicating it, as suggested in step 1, and then placing the second square so as to

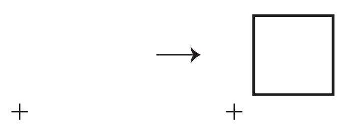
superimpose its left edge on the right edge of the first one. If you or I were to draw this, we could then coherently say: now let us take out the middle vertical line, and leave a rectangle with a 2:1 aspect ratio, as suggested in step 3. But only recently have we begun to know how to build systems that support these kinds of multiple perspectives on a single situation (even multiple perspectives of much the same kind, let alone perspectives in different, or even incommensurable, conceptual schemes). It should not be inferred from any of these comments [however] that these theoretical inadequacies have stood in the way of progress. Computer science does what it always does in the face of such difficulties: it makes up [ad hoc] answers as it goes along [debugging after the fact] – inventive, specific, and pragmatic, even if not necessarily well explicated.[3]



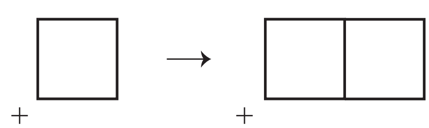
A big sigh – what a Sisyphean task. I guess explication isn't necessary. There's scant reason to figure things out once and for all, when you can patch them up, time and again as you need to, with cunning, after the fact as you go along. (Of course, this is ironic for computers, that try to predict what you're going to see.) Too bad there's so much to fix. In fact, endlessly more, and more that's likely to clash or interfere with what you've already done – patches on patches kept out of sight, in perpetuity. (The tragedy of false/fake answers is abundantly clear in Example 3.) Just what makes seeing so rigid for computers, when it seems so fluid to us? Maybe this is what Sisyphus really shows, that hidden cunning (debugging) is simply fruitless toil. Many try visual analogies (conceptual schemes) to cheat seeing, in the way Sisyphus cheats death in the myth; then what next? – to start over, rushing frantically to put in what's been left out, or to give up and insist on a single, "coherent" response that must be rigorously taught in rote recitation, and endlessly tested to make sure that everyone gets it right and vigilantly policed to keep it so. Either way, there's no end in sight. Or maybe this is a comprehensive plan for job growth and job security. And I thought computers were supposed to be labor-saving devices, the marvels of modern technology. Yes, to see things in new ways, to go on easily without breaks or gaps to bridge, is an intractable problem – a challenging conundrum that begs for a single solution that seems to be forever out of reach, when it's tried piecemeal in descriptions, part by part in visual analogies or equally, symbol by symbol. Somehow nothing ever seems to add up in the desired way. And it's a problem time and again today in computer science and AI, in surprising places when insight matters more than usual, if reasoning and logic are going to work – for example, in calculating the combined area of a square and a parallelogram



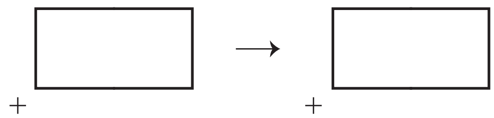
that are also two intersecting triangles. Maybe there's a straightforward way to find an answer that doesn't wait for difficulties (blind spots) to arise and multiply/compound with every change in perspective. In a shape grammar, I can define the two squares in step 1 of Smith's little example using the rule



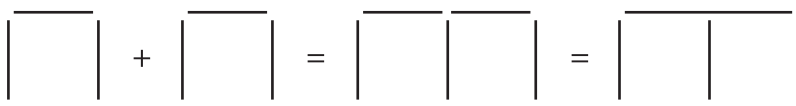
in the inverse $x \rightarrow prt^{-1}(x)$ of the schema $x \rightarrow prt(x)$ for parts, and the rule



in the addition schema $x \rightarrow x + t(x)$, to duplicate x in any way I choose, anywhere I want it to be, using any given transformation t. In $x \rightarrow prt^{-1}(x)$, x is the empty shape (a blank space), and $prt^{-1}(x)$ is a square (in fact, $prt^{-1}(x)$ can be any shape, because the empty shape is part of every shape); in $x \rightarrow x + t(x)$, x is a square, and t is typically a garden variety reflection or a translation that's the side of the square. This produces the shape in step 2 without any fuss. Then the identity

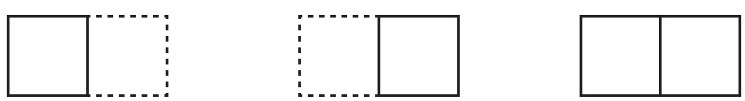


is enough to find the rectangle in step 3 – it simply embeds the rectangle after the two squares fuse. In fact, the rectangle and line are defined explicitly, when reduction rules are applied to lines (basic elements) to make the maximal (longest) lines that correspond to what a person trained in drafting the old-fashioned way would draw with a T-square and triangles. This is the smallest number of lines that describes the shape, a kind of visual Taylorism for drawing –

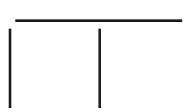


This highlights the difference between combining two squares as objects (symbols), and combining two squares as shapes. Objects combine/merge as elements in sets (visual analogies and spatial relations) or in other kinds of set-like structures, and they're preserved separately and are never tampered with; whatever parts there are, are simply different combinations (subsets) of distinct and independent objects.

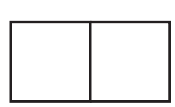
Two squares have only these parts



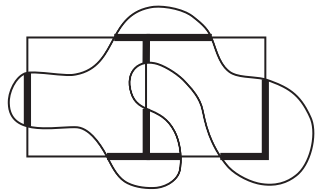
none of which are rectangles. When shapes combine in shape grammars, however, they fuse, and then, parts depend on rules and embedding. Two squares aren't defined explicitly in these five lines



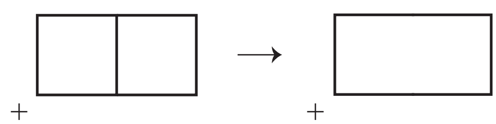
But the squares are there "figuratively," because I can embed their sides in the lines – try it with tracing paper in a visual proof to see how lines and segments match up. The shape



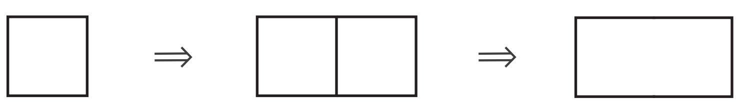
isn't just two horizontals and three verticals, or a rectangle and a line, or anything else I can say – I can see in it whatever I choose to see, whatever I can trace in its lines and their segments, whenever I try a rule. It's worth trying this to see what you get; it's always more than anyone ever assumes in advance, much more. No good counting in your head – it's what you see as you draw. Suppose I trace everything inside a simple curve –



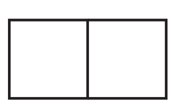
This may be a practical method to design building plans, complete with walls and rooms, windows and doors, etc. Designing isn't simply combining things in a lot of different ways and sifting through the results to find what you want; it requires seeing surprising things in other things, in the ones you're designing and in the ones you've copied – in a tree-trunk or clod of earth, in John von Neumann's Rorschach test or Oscar Wilde's beautiful form, in prior designs you're sure you know/understand. All of this takes insight and imagination, with embedding and shapes that fuse. Or I can try the rule



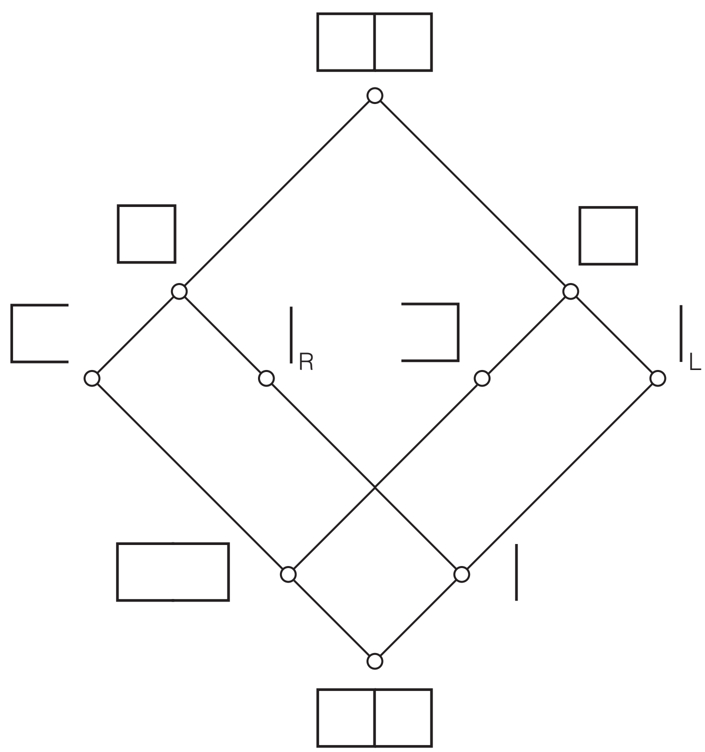
in the schema $x \rightarrow prt(x)$ to erase the middle vertical line to make a rectangle. In this case, there are two obvious steps; calculating goes like so –



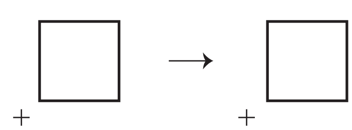
Once again, this relies entirely on the embed-fuse cycle – it's impossible to do without it. Two squares fuse, so that I can put in a rectangle or find other parts of any kind I like. Moreover, the alternative ways of seeing the shape



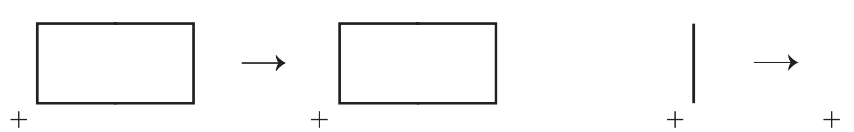
as side-by-side squares or a rectangle and a line, can be elaborated in a graph to merge distinct (“incommensurable” or at least topologically incomparable) trees, in the same way I used an identity in Example 1 to show how two triangles are actually three, and vice versa. (This kind of incommensurability and topological incomparability go for the trees in Example 1, as well.) But surely, squares aren’t a rectangle and a line. Isn’t that why computer systems in art and design (BIM, etc.) are so successful? They keep everything apart, never coadunate. Squares, rectangles, and lines are individual objects of three separate kinds/types that don’t interact, at least they shouldn’t for things to make any sense –



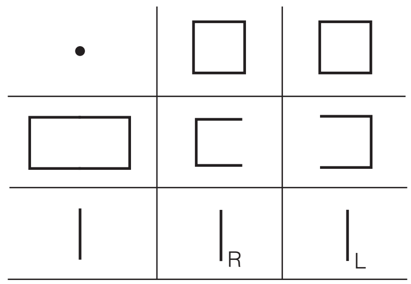
In my new graph, squares, rectangles, and lines are interchangeable. The identity



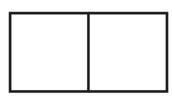
for squares defines the tree at the top in the schema $x \rightarrow \Sigma$ F(prt(x)). As required, the squares overlap – the right edge of the left one is the left edge of the right one. In the same way, the identities



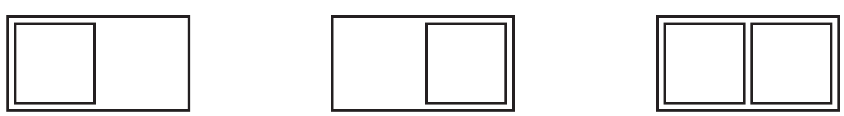
pick out the discrete rectangle and line. The superimposed (right and left) edges of the squares are represented as two lines in the single line that divides the rectangle. This is easy to see in the table of products for the parts picked out in the two trees –



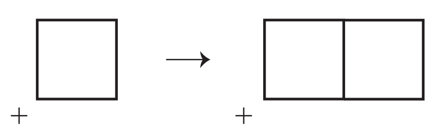
Of course, what's coherent isn't anything anyone can say, or coherently say, for sure – there are untold ways to see the shape



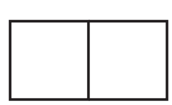
At the very least, there are building plans galore, traced with the identity for lines – but really, the identities for squares and rectangles are a better start. They do a lot more than Smith suggests, maybe more than reason and thought are ready to allow. Side-by-side squares are equally a rectangle and a square, and a rectangle and two squares –



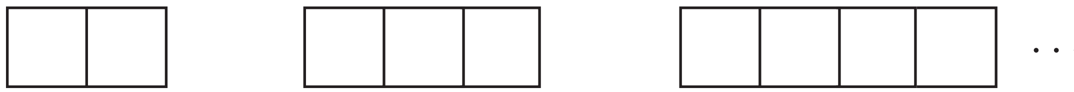
Neat, but who would have ever guessed? And why not use the rule



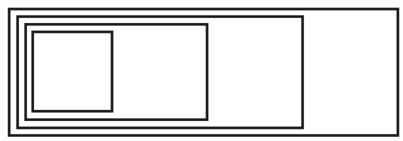
recursively? The shape



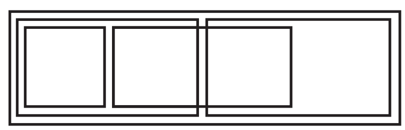
is merely the first in a series of shapes



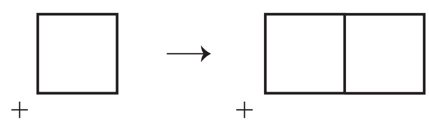
made up of dilating rectangles in arrangements like this



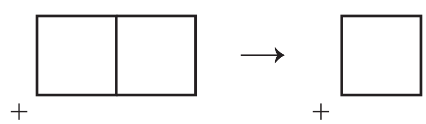
and in many other arrangements, as well –



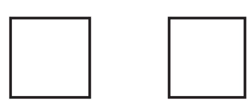
(It's worth it to spend a little extra time on the schema $x \rightarrow \Sigma\ F(prt(x))$, and the inverse of the rule



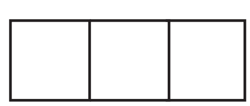
that is to say, the rule



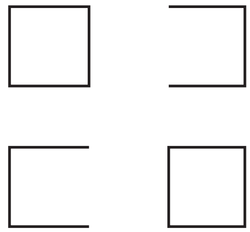
that erases one of two adjacent squares. I can use the inverse rule to insert gaps and spaces in rows of squares – for example, like this



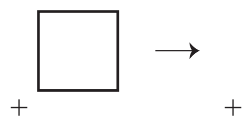
When x is the shape



and prt(x) is the pair of left squares and the pair of right ones. The inverse rule is applied to both ends at the same time, so that neither of the end squares has an interior side that's erased –

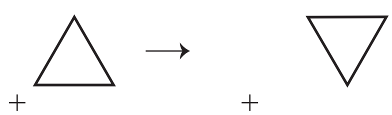


Clever, although the same thing happens elsewhere, sometimes with disappointing results – for example, the part schema x → prt(x) includes the erasing rule for squares

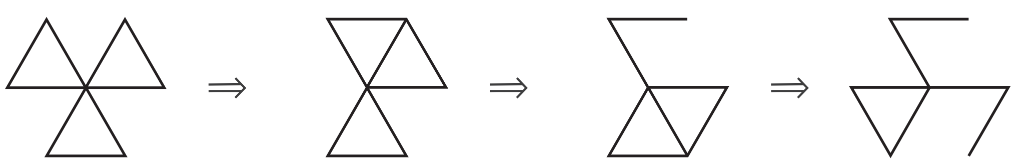


that takes out the middle square directly in the obvious way, to produce the shape

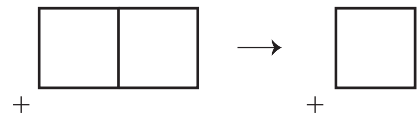
in which there are no squares at all. How can I ever fix that? Also for the pinwheel in Example 1, there are unwelcome surprises, when I try the rule



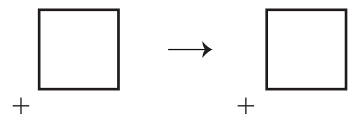
to rotate each of its triangles about a vertex, one after the other in the sequence



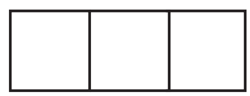
And I can go on with the last triangle, to end with it alone. If squares and triangles were objects/symbols, this wouldn't be a problem. Is there a way to keep parts the same as needed, without symbols that are invariant forever? In fact, the identities x → x fill the bill. They do an impeccable job of reconstruction and repair – simply put in the identity for anything you want to save, or that's to stay the same, with everything else you've already combined in x → F(prt(x)). Try it with the inverse rule



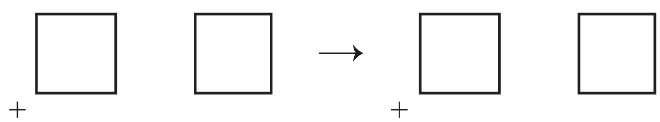
and the familiar identity for squares



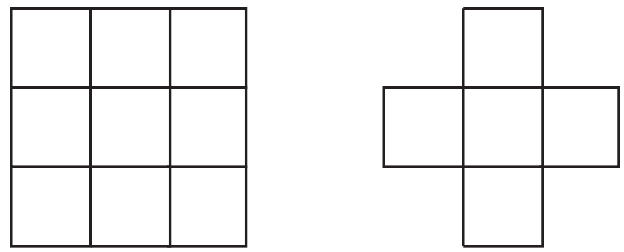
for one end of the shape



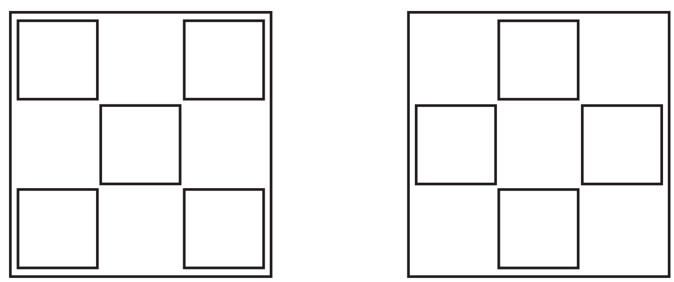
or the sum of the identities for the squares at both ends –



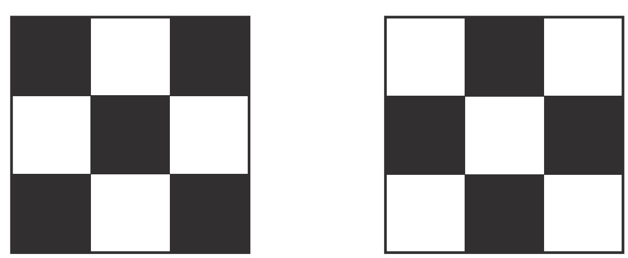
And this identity goes equally with the erasing rule for squares, too. It's a neat trick – because identities are incorporated in the rule that's defined in $x \rightarrow \Sigma\ F(prt(x))$ before the rule is tried, changes and corrections occur together in a single process that's fully integrated from the start. Sometimes, surprises are welcome, and sometimes, they're really not. Identities work for parsing, and now as well, to ensure that given/selected parts are invariant, at least for the time being. Identities keep parts intact, offsetting any unwanted side effects that may result as other rules are put into use – and the side effects needn't be known in advance, only what's to be saved.[4] Of course, other kinds of relationships for identities and rules are also possible to describe, parse, and produce shapes and pictures, using constructive and evocative devices in which perception and meaning interact and vary freely. This adds more to the algorithmic approach to aesthetics that I outlined in fn 20 of "Seven Questions." It's pretty amazing how much identities do.[5]) The way rows of squares are rectangles extends nicely to grids and similar forms



It’s always great fun to explore figure-ground relationships, effortlessly in varied grids; they’re easy to find everywhere, for example, in checkerboard patterns



so that presto chango, five squares are merely four. Then, there’s the coloring book schema $x \rightarrow x + b^{-1}(x)$ and the more inclusive schema $x \rightarrow x + b^{-1}(prt(x))$ to fill in boundaries –



There’s no end to schemas and rules. Visual calculating in shape grammars goes beyond what symbolic calculating allows in computers with visual analogies – conspicuously in object-oriented systems in CAD and BIM. Surely, “representational flexibility” is a praiseworthy goal, but a handful of coherent descriptions, whether Smith’s or yours or mine, is bound to be sorely incomplete when it comes to a Rorschach test or a beautiful form. Embedding and shapes that fuse, ask for much more – plenty for everyone, and then extravagant descriptions (prodigal ones) to try at will. There’s no telling what any of us will see next time that exceeds everything we’ve seen before, that’s indubitably coherent and makes perfectly good sense now (original participation) and retrospectively (final participation). This kind of personal freedom (representational flexibility) to see as you please is at once intuitive and open-ended – multiple perspectives abound – but it seems that there's never enough freedom in computers for art and design, or for very much of anything else when it comes to shapes and rules. In order to have enough, there must be too much. In shape grammars, too much is never less than all there is – the embed-fuse cycle brooks no limits.[6] Seeing is relentless; the winged eye never rests.

Example 3. My third example runs straight to the goal – it compares shapes and symbols directly, and shows how they differ. In fact, it comes from an earlier essay of mine on Wilde's critical spirit, with its key aesthetic formula to see things as in themselves they really are not – "*The Critic as Artist*: Oscar Wilde's Prolegomena to Shape Grammars."[7] I've modified this in a few places, so that it fits easily with "Seven Questions" and adds more to it, but mostly it's what I wrote before. In computer science (actually, in the theory of formal languages and automata), a rule in a generative grammar (Turing machine) is useless if it contains a "nongenerating" symbol or symmetrically, an "unreachable" one in a given vocabulary of symbols.[8] Unreachable symbols provide another way to show how shapes and rules work in shape grammars, and why shapes aren't symbols. In outline, the core idea is this – a symbol is reachable when it's in the initial string of a grammar or recursively, in the right-hand side of a rule that has only reachable symbols in its left-hand side. Rules with unreachable symbols can't be used to calculate; there's no way to apply them to any string I can get from the initial string using the other rules in the grammar, because every such string contains reachable symbols only. This doesn't seem very hard to check – maybe my initial string is the same symbol S lined up three times in this concatenation

SSS

and my rules are these five

$S \rightarrow AB$

$C \rightarrow BD$

$AD \rightarrow ABC$

SA → a

B → b

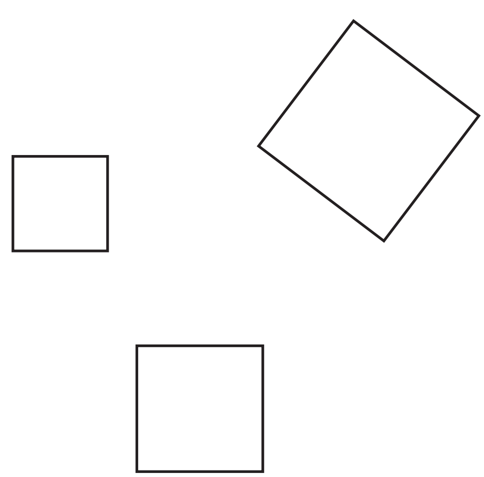
Starting with S as the first reachable symbol, and working recursively, I get

S, A, B

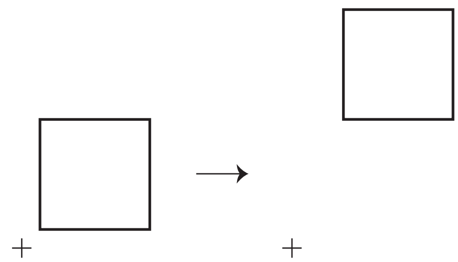
because A and B are in the right-hand side of the rule S → AB, that has a reachable symbol in its left-hand side, and then I finish with

S, A, B, a, b

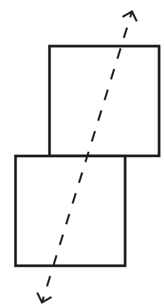
because a and b are in the right-hand sides of the rules SA → a and B → b, with the reachable symbols S, A, and B in their left-hand sides. The five reachable symbols are S, A, B, a, b, and the unreachable ones are C and D. So, C → BD and AD → ABC are useless rules that I'm free to delete – the grammar works the same when they aren't there. This is unremarkable for unreachable symbols; there's no way to try the rules in which they occur. But is this the case when symbols are shapes? Can I tell whether shapes are unreachable or not? Let's see how it goes if I try to discard useless rules in shape grammars. With shapes, there are bound to be some surprises. Suppose I start out with an initial shape made up of three squares, that corresponds to the initial string SSS with its three S's. Maybe the squares are arranged so –



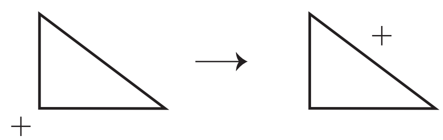
And suppose I calculate as usual in a shape grammar with the rule



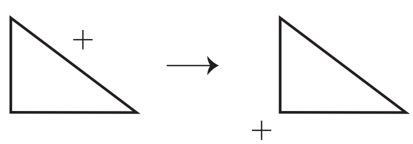
in the transformation schema $x \rightarrow t(x)$, that translates a square back and forth –



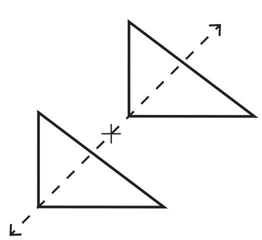
Of course, I can add other rules, too – maybe this one



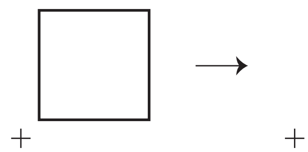
in $x \rightarrow t(x)$, and also from this schema, the inverse



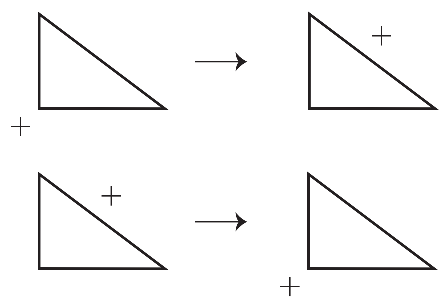
to translate a triangle back and forth, as well –



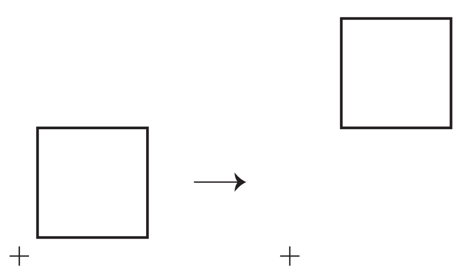
I need two rules for the triangle, because it's asymmetric, unlike the square. But does adding these two new rules do any good? The triangle in their left-hand sides isn't in the initial shape – obviously there are exactly three squares. I can use the erasing rule for squares



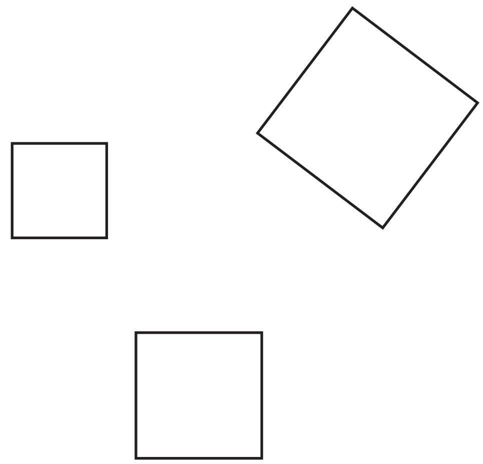
to make sure, eliminating them one at a time until nothing's left – isn't this the proper way to count things out, object by object? Nor is the triangle located at any place I'm able to see in the right-hand side of the rule that moves a square. Everyone agrees that triangles aren't squares – it's seems pretty clear, the triangle is definitely unreachable, and the rules



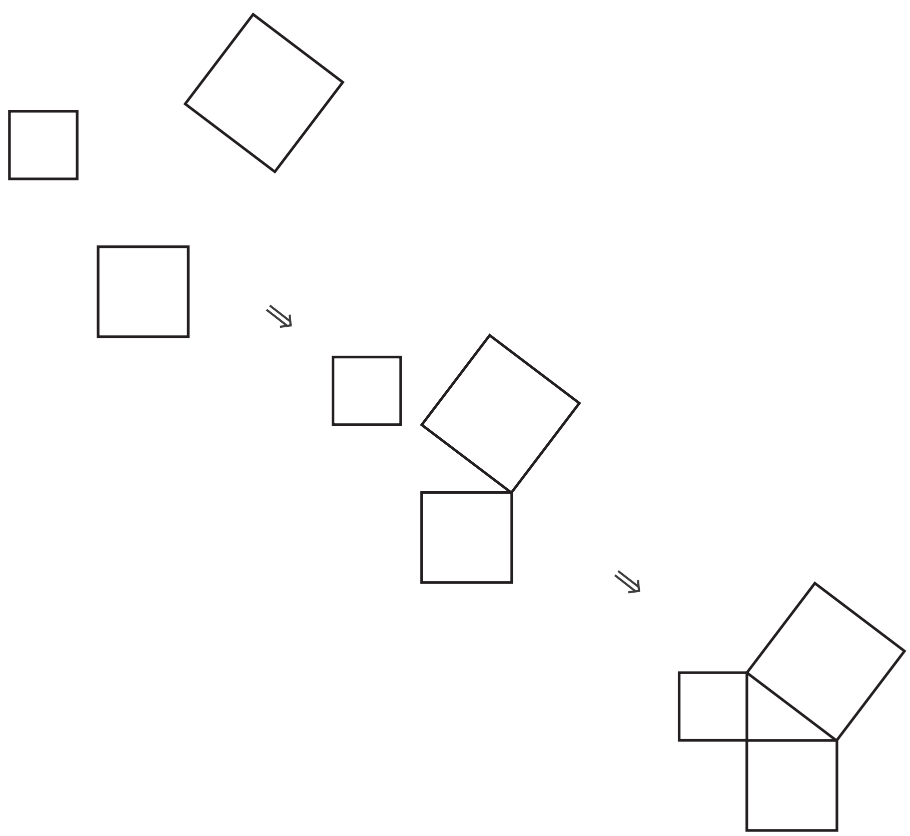
are undeniably useless. I guess my rules for triangles are a mistake. But let's see if it's prudent to say this, that it matches my ongoing visual experience when I calculate. If I try the rule



on the largest square and on the smallest square in the initial shape



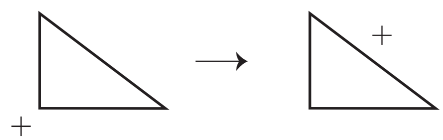
in that order, I get this series of shapes



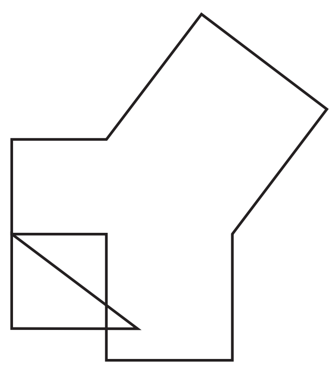
that ends with a Pythagorean figure – that unmistakable symbol of number, rationality, and usefulness. What would students learn in grade school and beyond, especially in STEM subjects, if there were no Pythagorean theorem about right triangles and their sides? It's the eternal formula that everyone knows by heart and can recite effortlessly. It's easy to say in words

$$a^2 + b^2 = c^2$$

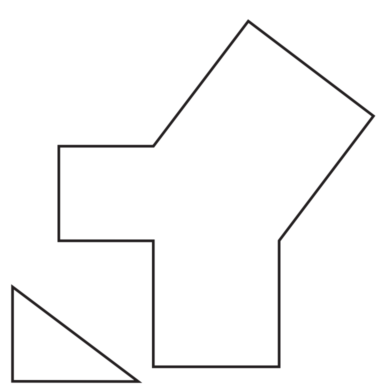
But this is merely symbols and numbers – my triangle is $3^2 + 4^2 = 5^2$. What made it appear as if by magic, when I started out with three squares, as counting proves? And in fact, God in Norbert Wiener's little, children's song in *Cybernetics* would agree that counting is the hallmark of definite things, so the squares must be there.[9] I guess the squares are arranged around the triangle – that's easy enough to see – but this isn't a binding distinction in shape grammars. I can embed (trace) the triangle in the Pythagorean figure, so it's there, too. No matter how I try to finesse this, I can apply the rule



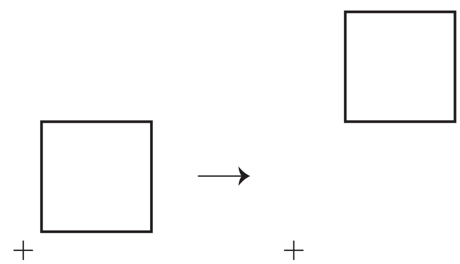
to translate the triangle and thereby calculate some more to get



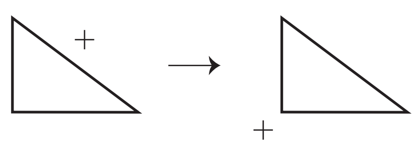
and this hardly seems useless. Just the opposite, it might be very useful. (I guess squares aren't objects in shape grammars, like the two squares in Example 2 that form a rectangle that can't be defined.) And I can go on in this manner to move the triangle again



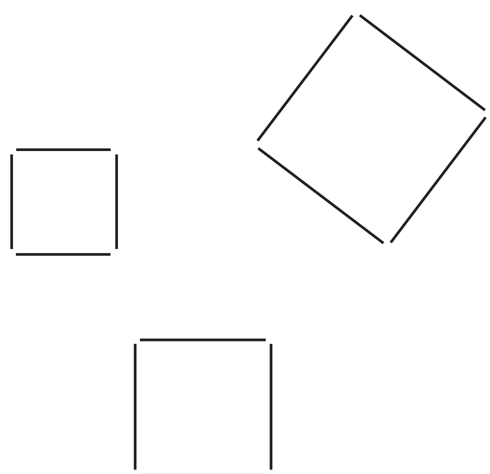
That's kind of strange – who would ever have imagined such a thing could happen? Now my first rule for squares



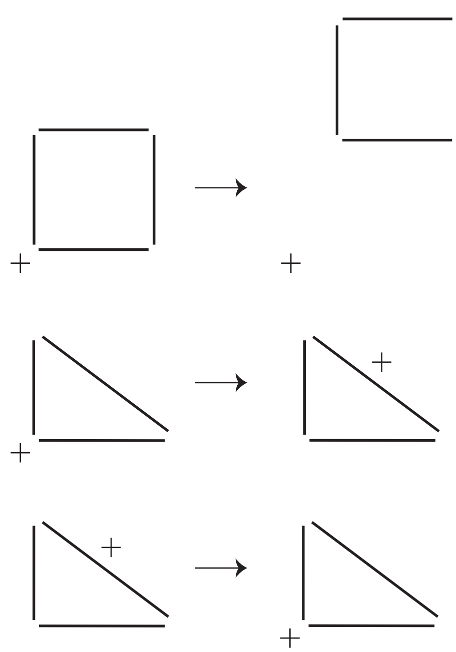
the one that was so useful before, is useless. I guess this goes for symbols, too – useful rules can become useless, but then it must be forever. I can use the inverse



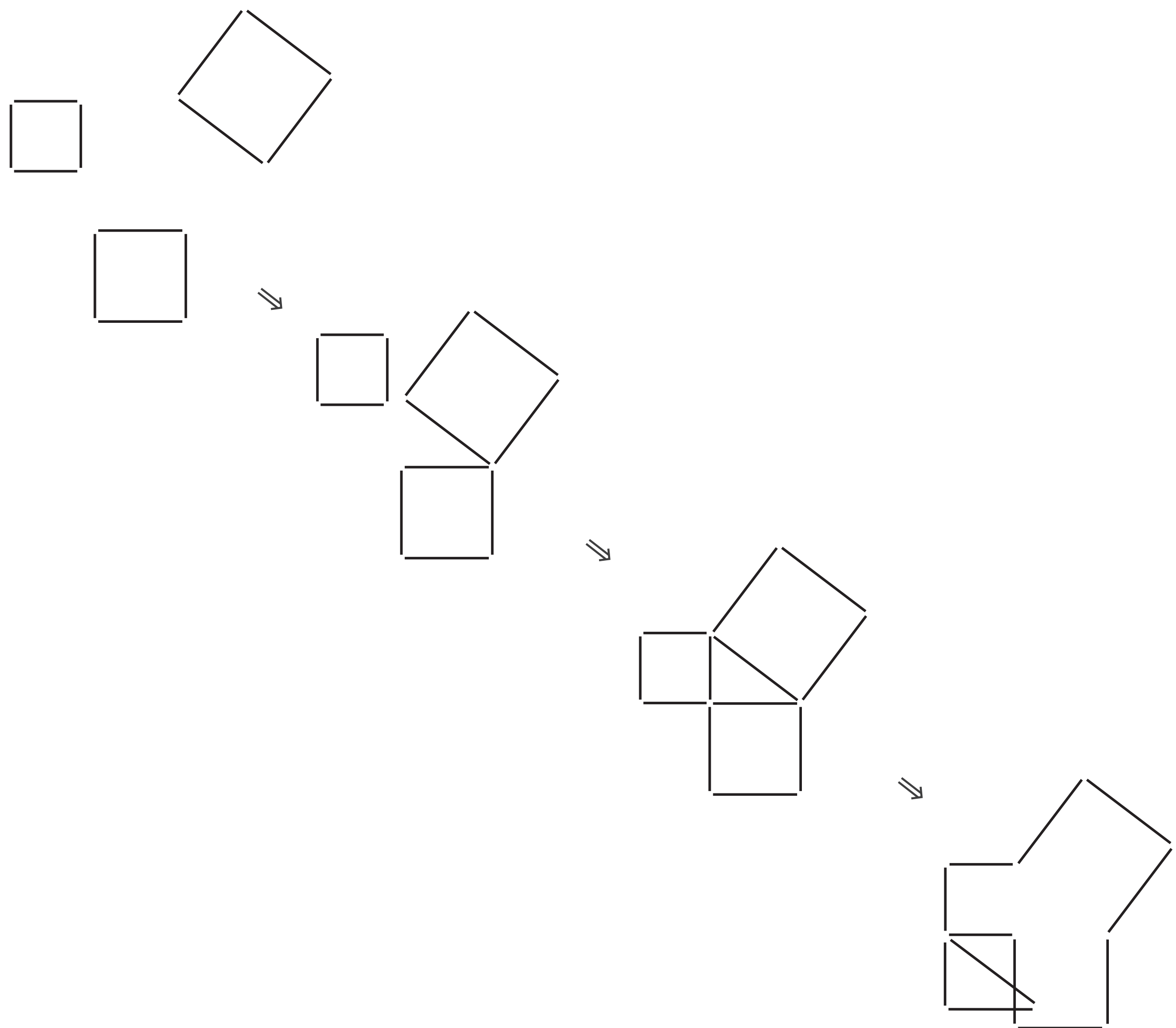
twice to retrieve three squares that pop out all at once, and then move them separately in multiple ways – eight ways for each one to be exact, in terms of how the symmetry of the square in the left-hand side of the rule for squares is modified by its right-hand side.[10] Useless/useful rules can turn into useful/useless ones, and then switch back again – in untold ways. (Surely, this is a marvelous source of original plots worthy of Wilde.) A trio of useless rules – two with an unreachable triangle and one with an unreachable square – that I thought I could discard without any qualms, are perfectly useful in obvious ways but at different times. Is this even possible? I must be missing something important – visual calculating and symbolic calculating can't be that different. Calculating is calculating, isn't it – just as a rose is a rose? This seems reasonable – everyone takes it for granted – but it's probably a mistake, especially when it comes to what I see. But maybe this is merely a misunderstanding that can be resolved rationally, thinking slow in logical steps.[11] Maybe squares and triangles are really four lines and three lines, respectively, as I was taught time after time in school. Why not use these visual analogies (spatial relations) in place of shapes? There must be a good reason for them, although my teachers were totally baffled when I asked for one – "why is George worried about this, when it's so obvious?" But I guess my teachers actually knew best – if squares and triangles have distinct sides, then the side of one can be the side of the other at the same time, in a kind of mutual exchange that defines both. It seems that lines can be like symbols (0-dimensional elements) in visual analogies for squares and triangles, and work just like units as I calculate – in effect to make visual calculating the same as symbolic calculating.[12] Being practical works wonders, taking things as in themselves they really are, at least the way they are, as they're rigorously taught in school – education is an admirable/necessary thing in an impossibly ambiguous world. This highlights the key difference between set grammars for spatial relations and shape grammars.[13] I've already tried shape grammars, but confused useful rules and useless ones. Maybe set grammars will work better, to avoid making the same mistake again. The initial shape, exploded just a little bit to separate symbols, is now these 12 lines



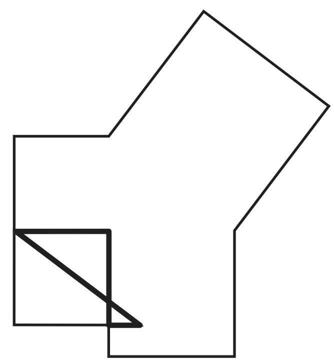
that are all units, and my three rules, also in exploded versions, are



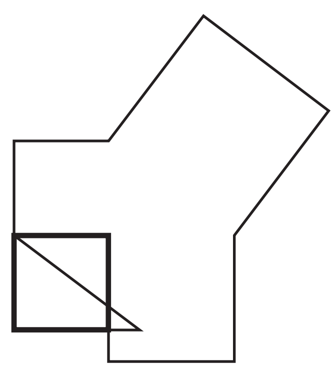
Lines are reachable – they're in my initial shape and the only symbols in my rules – and so, no rule is useless, even if it's hard to imagine how rules for triangles might be used. That's the reason to calculate – to find out. And when I do, everything works exactly right –



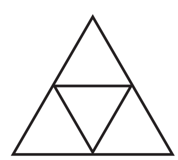
This is kind of cool. But are these visual analogies really a sure thing? They seem too good to be true. Does everything always come out right if I keep strictly to the lines that define squares and triangles? What can go wrong? Do I need to add more detail to be positive that these visual analogies are complete? How and when do I know this once and for all? Are there examples that don't work, even in theory?[14] Consider the little triangles – the so-called "emergent" ones



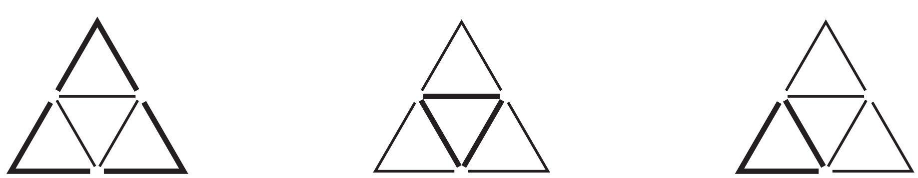
I should be able to move them with rules, as well. And it’s no problem in shape grammars with embedding, ironically because triangles are unanalyzed and undivided, and not lines (symbols) in a visual analogy – so much for thinking slow, with what I learned in school. And how about the emergent square



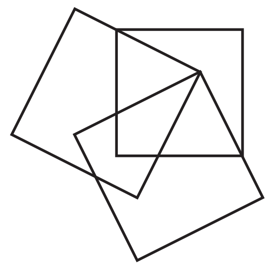
In fact with these surprises, my three rules with lines should all be useful at the same time. That’s something to be happy about. But I can’t move either of the triangles or the square because they don’t have sides – lines (symbols) put in by my rules – or they have fewer sides than they need. And suppose they did have sides, then what would happen to the lines in my three initial squares and in the emergent triangle they contain? How many additional rules are part of the mix to make any of this work? How many definitions of squares and triangles are there? All of a sudden, things seem to be getting awfully complicated – simply trying to move squares and triangles around in an effortless way. And what all of this shows is how truly different shapes and symbols actually are. Neither individual shapes nor shapes in rules can be described before I calculate, while symbols are given once and for all – fixed and independent forever. It may be that squares and triangles are made up of lines, but how do I know which ones? Can I find out just by looking at my initial shape and my rules? This is OK in generative grammars – symbols are known in advance. But it seems hopeless in shape grammars, because lines can be divided freely in lots of ways. I can't decide what I'll see before I've seen it, and I need embedding and rules for this. I have to calculate in shape grammars, to find out if the shape in the left-hand side of a rule is reachable or not – it's never enough to look at the initial shape and the other rules – and this makes it hard, no, it makes it impossible, to decide. There's nothing I can do before I try my rules, to find out what's going to happen next. Visual analogies and other descriptions aren't a reliable way to start. They rarely work and are usually misleading without calculating first without them. And then, what kind of calculating is this – surely, it's not calculating with symbols? There are many possibilities that are worth knowing – but for most of them, this means going on, not knowing beforehand they're there. There's nothing to learn in school that helps, not even recognized things like squares and triangles. For example, for the shape



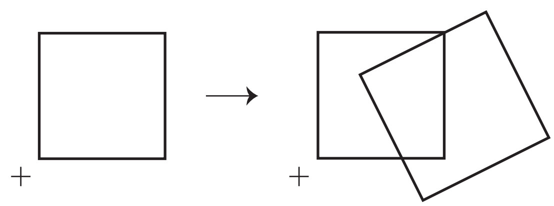
in Example 1 that's both two triangles and three, triangles are described in three distinct ways, as three angles, three sides, and an angle and a side. Go figure –



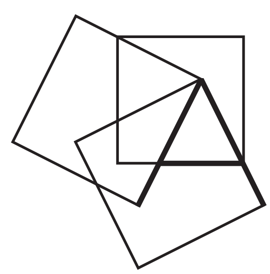
I guess that's finally it for my teachers – I had good reason to worry. It's exactly as Wilde puts it – "Education is an admirable thing, but it is well to remember from time to time that nothing that is worth knowing can be taught."[15] That's because what's worth knowing changes as I calculate, in a continuous process that's full of surprises. The use of embedding for shapes that fuse makes a huge difference – useless rules turn out to be useful. And maybe that's it for uselessness, too – it's not a distinction that makes much sense in shape grammars. It’s a cinch to cook up more examples like mine for squares and triangles, for any polygons – and equally in fact, for any shapes made up of linear elements.[16] Any shape or vocabulary of shapes can be used to make other shapes, sometimes in surprising ways. For example, three squares, a small one and two larger ones of the same size, are combined in this shape



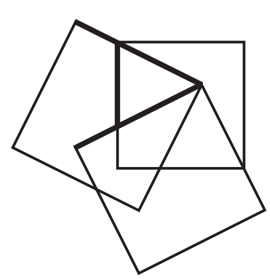
The rule is



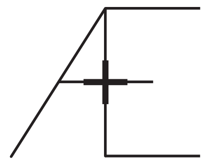
in the addition schema $x \rightarrow x + t(x)$, and it applies twice to a single square. And it’s a cinch that the lines in an A are embedded in the sides of the three squares –



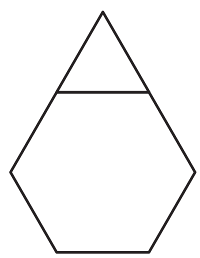
or is it two A’s that are embedded in the squares –



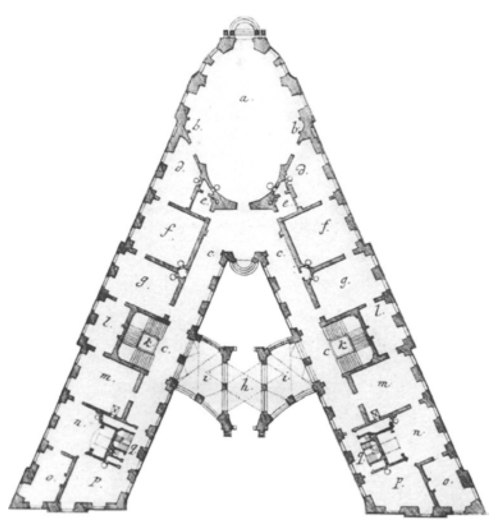
Moreover, the distinct symbols A and E contain the expression A + E in a similar way, no matter that neither A nor E is/includes +.[17] See for yourself –



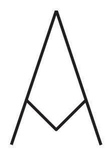
Symbols are shapes, too – or have an external presentation (sound, surface texture, etc.) that can be sensed/perceived – and as such, they're something to embed in shapes, or find in like phenomena. An A isn't just in three squares, but in two or more in many arrangements. And A pops out elsewhere in neat ways, for example, first in the remarks of an important philosopher, when he adds a triangle and a hexagon together, and wonders what this means –



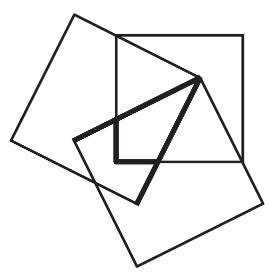
Then fancifully, it's in Johann David Steingruber's architectural ABC.[18] Steingruber puts rooms and walls together, with doors and windows, in elaborate building plans to make all the letters of the alphabet, here in an extraordinary "SECOND A" –



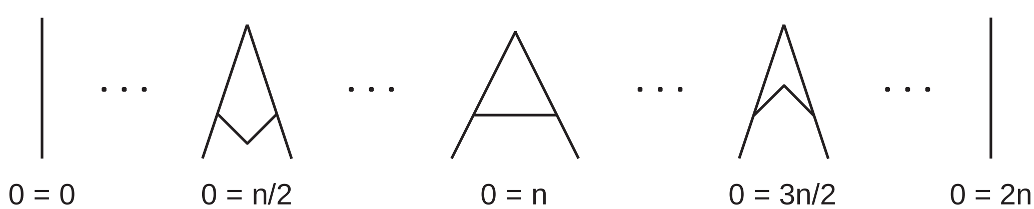
Which description of this shape fits it to a T, as in itself it really is – or is not? A building plan replete with rooms and walls, etc. represented as data/objects in BIM, the letter A, half a raven, right/left leg and partial beak (memory is lost when rules are tried, but it may matter in other ways), a philosophical puzzle, any combination of the four descriptions, none of them, or still additional ones? And once Steingruber’s A is seen, it’s impossible not to embed a copy (transformation) of it



as the third A in my three squares. An identity does the trick, to put in (find) the center of this symmetric trio, with overlapping A’s that are lined up as you please, maybe staggered and spaced in elevation, front to back, left to right, whatever you want –



This copy, though, is merely one possibility for a horizontal hinge that opens and closes, à la McCarthy and Reuleaux. Steingruber’s A varies in parametric profusion from greater than 0 to π in this way



(I can change any term, including the vertical at 0, into any of its successors in the schema $x \rightarrow \Sigma$ F(prt(x)), with two reciprocal rotations and a reflection in $x \rightarrow t(x)$. This divides the term and its successor with respect to x and t(x), each into four component parts/lines that are equal from term to term.[19] Notice, too, that the series extends symmetrically to 2π, when the hinge folds up into the top of the A. The same kind of switch also goes for Reuleaux’s series of nuts and bolts.) It’s uncanny – how the two kinds of A’s in my three squares are the same in one series. There are surprises galore whenever I open my eyes, and remarkable things to see – for example, the smaller M in Steingruger's A that's many other things, but surely, this is perfectly adequate for now. It seems that visual calculating in shape grammars isn't the same as symbolic calculating in Turing machines. In Turing machines, symbols are simply as in themselves they really are. Isn't this invariance what makes calculating possible in the first place – every string of symbols is easy to read as an elaborate sort of coded number. What would happen without invariant symbols? What if they weren't always the same? What if symbols were each and every one a Rorschach test? I guess this spells the end of Wilde's aesthetic formula with its warrant for ambiguity and unbridled change, and of pictures and poems, and art and design. It appears that calculating isn't seeing – unless of course, calculating is as in itself it really is not. How can this possibly be true? But that's what I've been trying to prove in "Seven Questions" with Coleridge, and von Neumann and Wilde, and in my expansive footnotes that tie in lots of others, and now, in this exhibit with my three examples to show the sweep of the embed-fuse cycle. Maybe there's a way to put all of this together in a single sentence –

Shapes aren't symbols, alone or in combination.

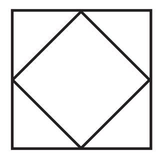
This is the quick of art and design. Shapes are seamless – they're divided again and again on the fly, in another way every time I try a rule in the embed-fuse cycle, to calculate with what I see. The ambiguity is what makes intuitive/personal experience possible, for example, in Wilde's modes of reverie and mood with countless changes and contradictions, and limitless delight.[20] Seeing – trying rules in shape grammars – goes on in ever shifting ways, that no one dares assay in advance. There's no chart, no map, no survey. My Pythagorean shapes, with squares and triangles that are reachable and not in a single shape grammar are, in fact, entirely commonplace, but unexpected just the same, even to the well-traveled eye –

> The three squares – just floating [drifting] down into the Pythagorean figure. And then those two little triangles, and the fourth [emergent] square. The first time you see the square and triangles, it's a total surprise. Once you've seen them and look again, there's an inevitability, a destiny to it. A Greek tragedy for calculating with symbols. The inevitable outcome of its tragic flaws.[21]

The Greek spirit is Wilde's original source for the critic/artist, who's armed with an aesthetic formula for different descriptions – Epicurean ones that are all true. I like this description of the Pythagorean figure because it contrasts shapes and symbols as the former lead to surprises that overwhelm the latter. And surely within Wilde's critical compass, reverie and mood go anywhere that embedding desires. Unexpected or not, tragedy is inescapable as long as symbols describe shapes fully, and once and for all in terms of what's previously known. No matter how much effort is put into this, shapes exceed any visual analogy (description) that's fixed in advance. Shapes are ineffably sublime – there's no saying what they really are as long as there's another rule to try in the embed-fuse cycle. But whenever I talk about how shapes and shape grammars are related to art and design in this way, someone invariably feels compelled to explain to all and sundry everything that I've misunderstood – and isn't this as it should be, isn't misunderstanding inevitable, being the starting point for seeing things as in themselves they really are not? Sometimes, my interlocutors have timeless lessons to teach about how things really are in pictures and poems, and sometimes, they betray sincere regret, as if I'm desperately lost in a kind of personal (psychotic) reverie that knows no bounds. Aficionados of the arts sigh and shake their heads – he just doesn't get it. But they can't say how or why, because then I could try and might succeed – or maybe not. Scientists and engineers aren't that much better, and they may be worse – that's not calculating, because you can't put it on a computer; you're just gaming the system; who told you, you could calculate like that? Science has some curious conventions that must be kept, but I prefer to see for myself. No matter who stands up, they're keen to recite tried and true slogans that lend comfort and support to the artist and the scientist alike, so that each can dismiss the methods and devices the other finds so dear. There's a thesis and an antithesis, and no synthesis. Yes, you can wear two separate hats, each at a different time; you can be a painter/poet on a Monday and then a physicist/mathematician on a Tuesday. But this frustrates and may even bar imagination to ensure two distinct cultures in one common world – art and science, and their vast unknown interstices never fuse in order to re-create, in unexpected divisions and precincts.[22] New insight is invariably lost in the incessant din of a rote chorus that repeats, yada-yada-yada, the kinds of things that everyone takes for granted –

Art is breaking the rules.

No doubt, this aesthetic rule is included in Wilde's aesthetic formula. But somehow the rule ensures that art and calculating are irredeemably apart, even if the formula is good for both. It seems that art breaks the rules, so it can't possibly be calculating, because calculating can only follow the rules – for computers, it's drag and drop, click-click, click, click, click. Maybe so for symbolic calculating and unreachable symbols, but this isn't so for visual calculating in shape grammars, where breaking a rule simply means trying another one, or using familiar rules in alternative (surprising) ways in the embed-fuse cycle. That's why Wilde's critical spirit provides such an invaluable prolegomena to shape grammars. The critic as artist invokes Wilde's aesthetic formula to see things as in themselves they really are not – every time with another rule. I'm free to see in this way, as well, in any way I want. I can draw two squares



and then see four triangles, or pentagons, hexagons, K's and k's – and in fact, right angle A's. There's no end to breaking the rules in shape grammars and visual calculating. Ambiguity and like uncertainties – change, contradiction, discontinuity, incompleteness, inconsistency, etc. – are the nub. Some of my protagonists in "Seven Questions" promote this, and some don't. But getting any of them to agree isn't the problem – I can keep them together with a little negative capability. Rather, the trick is to exploit ambiguity, as I calculate. This requires embedding and shapes that fuse, so that any rule works, no matter when I try it. Whether I'm consistent/coherent or not doesn't make even the slightest bit of difference – I'm free to go on. Nothing blocks my way. It's worth repeating – rules apply with no inherent memory, so I'm seeing for the first time, every time I look. And what I see – anything at all – corresponds to a rule. There's no way around this, because what I see relies on the embed-fuse cycle. Of course, the insistent question remains of whether I can predict the next rule I'll try or not. In answering this query, I invariably opt for not – picking a rule to use is exactly the same as putting an interpretation into a Rorschach test. Each of us calculates in a unique way – in von Neumann's words, as "a function of [our] whole personality and [our] whole previous history." But then, von Neumann's Rorschach test is Wilde's beautiful form – both are engaged personally with rules. It seems pretty sure that artist-computers (visual analogies) aren't enough, without the embed-fuse cycle in shape grammars. The rule I try, whatever it is, works now, whatever I see – and there's no reason to tailor any rule for this in advance. Nothing is left out to cheat seeing. Prior descriptions and tedious debugging aren't required to go on – Sisyphus is free at last. Rules forget what I've seen and may neglect any/all of my well-made plans; they needn't respect the past to work, or anticipate and plan for what's to come. Embedding and shapes that fuse let me calculate in an entirely open-ended process, to be myself fully. I can see exactly as I please. Shapes are as in themselves they really are not. This is Wilde's critic/artist at work – as an artist/critic.

[1] G. Stiny, *Shape: Talking about Seeing and Doing* (Cambridge, MA: The MIT Press, 2006), 296-301. The origin of this account is in, G. Stiny, “Shape Rules: Closure, Continuity, and Emergence,” *Environment and Planning B: Planning and Design* (1994): s49-s78. It was something of a surprise then, that continuity in shape grammars was defined retrospectively.

[2] G. Stiny, *Pictorial and Formal Aspects of Shape and Shape Grammars* (Basel: Birkhäuser, 1975), 152-162. But see also, G. Stiny, *Shape*, 186-188.

[3] B. C. Smith, *On the Origin of Objects* (Cambridge, MA: The MIT Press, 1996), 23-25. One of my students showed me Smith’s example with two squares, to prove that shape grammars were fatally flawed. He was stunned when I embraced the squares and the accompanying rectangle effortlessly – shape grammars weren’t what he thought they were, and his confusion wasn’t unique. It’s funny how things aren’t always what they seem to be at first blush. It was a “teachable moment” – where surprising opportunities arise to try something new, in the same way they do for shapes and rules in the embed-fuse cycle. But who said teaching was easy? He had already made up his mind, and wasn’t going to change it – so much for shape grammars and the untold surprises they hold for art and design.

[4] I was aware of gaps and spaces early on, and the kinds of problems they were apt to cause, in G. Stiny, ‘‘Kindergarten Grammars: Designing with Froebel’s Building Gifts,’’ *Environment and Planning B* 7 (1980): 409–462, 443-445. I used a painfully obvious and shamefully tedious kind of debugging as a way around this, and viewed side effects as a source of “unpleasantness” that might prove difficult to avoid. Looking back on what I was trying to do, my impatience seems totally unjustified, and my methods seem pretty ad hoc – restricted and wrongheaded, something to deny and then to quickly ignore. Once invariant parts are picked out to define identities in the schema $x \rightarrow \Sigma$ F(prt(x)), unwelcome side effects disappear – it’s no big deal to keep what you want. Of course, if you have any doubts about this – passing ones or not – it’s probably a better bet to give up on what you think you want, in order to see where side effects go. Isn’t looking for surprise and delight a vital part of art and design?

[5] Some wonder if I have any favorite rules – which ones are full of delight? When I’m asked, I invariably point to the identities in the schema $x \rightarrow x$. This is a neat way to highlight the distinction between object-oriented systems (computers) and shape grammars. On the one hand, identities aren’t worth a second glance; they play no role when it comes to calculating with symbols, and in fact, are better discarded, so as not to get in the way. Then on the other hand, identities strike at the quick of visual calculating, with embedding and shapes that fuse; at the very least, they

pick out parts to alter what I see, and keep the parts/shapes I can't live without. Identities show how ambiguity works when I calculate with shapes and rules.

[6] There are many arguments for this throughout G. Stiny, *Shape* – it's possible to start almost anywhere to find more you want. Owen Barfield distinguishes original and final participation in terms of figuration; they're both considered in A3 in "Seven Questions," 16.

[7] G. Stiny, "*The Critic as Artist*: Oscar Wilde's Prolegomena to Shape Grammars," *Nexus Network Journal: Architecture and Mathematics* (2016): 1-36. I asked a good friend of mine in computer science to read this. After he finished, he asked me when I was first aware of the squares and triangles in Example 3, and that they weren't symbols. I replied that I knew about them when I was doing my PhD at UCLA in the theory of formal languages and automata, nearly 50 years ago. I didn't mention this at the time, because I thought it might be embarrassing. He laughed – few today would care that there's an alternative way to calculate with shapes and rules in the embed-fuse cycle. Symbolic calculating is perfectly fine the way it is, when everything is possible in computers and AI with data and learning. But somehow, this misses the point of pictures and poems (art and design) entirely, where S. T. Coleridge's esemplastic power (imagination) and Wilde's critical spirit hold sway. It's next to impossible to believe that prior data and learning can ever go beyond seeing things as in themselves they really are – motionless, hopelessly snared in numbers and statistics, objects essentially fixed and dead. After all, that's what's crucial when computers and AI take over the useful functions in our everyday lives, from running our cities to cleaning our houses to driving our cars, etc. There's little room for imagination and Wilde's critical spirit when you're crossing a busy intersection, at the risk of life and limb. Computers and AI can do many things within their broad compass, but only beyond these bounds are things vital and alive – in the region of von Neumann's Rorschach test, in the region of beautiful forms in art and design. There's nothing to keep computers from making pictures and poems, and no doubt, computer art is already a known reality – it's just not for computers and AI that can only see things as in themselves they really are, solely in terms of visual analogies and corresponding structures in graphs, trees, topologies, etc. These are defined from past experience with no recourse to new perception – mere data and learning seek what's invariant and leave out surprise. This highlights the difference in sweep between symbolic calculating in computers and visual calculating in shape grammars – can anyone doubt the value of the latter in art and design?

[8] J. E. Hopcroft, R. Motwani, and J. D. Ullman, *Introduction to Automata Theory, Languages, and Computation*, 3rd ed. (Boston: Pearson Education Inc., 2001), 256-259.

[9] See "Seven Questions," 37.

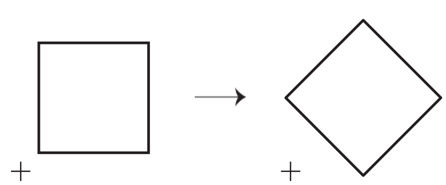
[10] G. Stiny, *Shape*, 265-268. This is a nice way to see how finite groups work in terms of La Grange's theorem for subgroups and their cosets.

[11] The popular distinction between thinking fast and thinking slow is key in Daniel Kahneman's behavioral economics and psychology, D. Kahneman, *Thinking Fast and Slow* (New York: Farrar, Straus and Giroux, 2011). Thinking fast is largely instinctive, intuitive, and perceptual (pre-cognitive); thinking slow is more rational and logical (cognitive). But much of what thinking slow offers relies first on thinking fast, to set the stage for description and representation. In visual calculating, thinking slow (logic) works best/only in retrospect, once thinking fast (seeing) is done. (Sometimes the practiced eye is frightfully slow, although neither rational nor logical. Maybe fast and slow are as in themselves they really are not.) Of course, thinking slow with visual analogies has yet to show its total worth in my Pythagorean example, to exhaust itself fully. That's to come. It's like getting a shot – it's fine for the season, unless you catch the flu. The vaccine is only a guess that may hit the wrong strains (visual analogies).

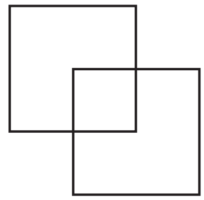
[12] This kind of dimensionality for shapes is discussed briefly in "Seven Questions," in fn 20. The canonical account is in, G. Stiny, *Shape*, Parts I and II.

[13] G. Stiny, "Spatial Relations and Grammars," *Environment and Planning B* 9 (1982): 113-114. It's a real mystery to me why I preferred shapes to sets and seeing to counting – no one else did in the 70's and 80's, given the bent for scientific method and technical rigor in architecture and design research. (For a really nice review of the spirit of the times, at least in the Martin Centre at Cambridge, see fn 12 in "Seven Questions.") In fact, I'm still surprised about this today – that I'd much rather look at things than say what they are, either in terms of visual analogies and spatial relations, or in terms of data and machine learning that make explicit visual analogies unnecessary. There's more to seeing than words. This can be a real problem, especially when you're expected to talk way before you've had a chance to gawk.

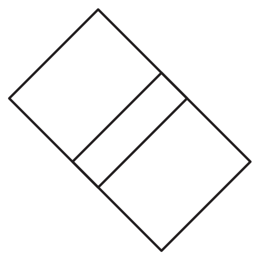
[14] Some straightforward examples of visual calculating in shape grammars, and a combinatorial proof that they can't be described/represented once and for all in terms of visual analogies (spatial relations or other kinds of computer structures in graphs, etc.) are given in I. Jowers, C. Earl, and G. Stiny, "Shapes, Structures, and Shape Grammar Implementation," *Computer-Aided Design* (2019): 80-92. The examples exploit the symmetry of squares, and their parallel sides. The rule



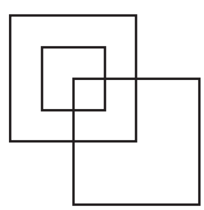
that rotates squares by $\pi/4$ around their centers is defined in the primary schema for transformations $x \rightarrow t(x)$. When this rule is applied to the shape



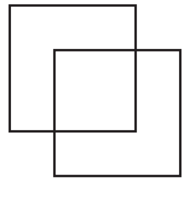
the sides of the three squares are structured in palindromic “words” or strings of incommensurable symbols/units. For b = a√2, the sides of the two large squares are segmented in the string ababaababa. This also divides the sides of the small square in the palindrome ababa, with the palindromic prefix/suffix aba. My rule is represented twice – once for the large squares, and again for the small one. The prefix/suffix is crucial when the large squares are rotated



so that their sides align and overlap. In the common piece, the sides match each to the other, head to tail – hence, aba. The kind of recursive structure-in-structure for palindromes in ababaababa at 1 (the large square),1/2 (the small square), and (2 – √2)/2 (the overlapping piece), however, is elusive in comparable shapes. In fact, such shapes are easy to define in indefinitely many ways, for example, when I add another small square in order to scale, either going up (putting in more squares) or equivalently going down (putting in smaller squares)



and when I push the large squares closer together along their common diagonal



even by the tiniest nano bit, but before the three squares merge/fuse into one. (The second shape is part of a series that

maps onto John McCarthy’s lines – cantilevers and crosses – and 100 years before, Franz Reuleaux’s nuts and bolts, where visual analogies vary continuously within fixed limits. McCarthy and Reuleaux are paradigmatic for parametric design.) Calculating runs smoothly without a hitch in the embed-fuse cycle with the rotation rule for squares and my pair of new shapes; it's impossible, however, with symbols that divide and segment from the start. This is probably no big surprise given how shape grammars work, although symbols are always ready for one more try and the promise of success the next time. To see segmenting loop endlessly can be mesmerizing, as it plays out in ever smaller and smaller divisions, and repeating patterns. There's no arguing with the results of combinatorial proofs, no matter that some of them spell misfortune and collapse for calculating in the usual way in computers – conspicuously in the object-oriented systems used/required in BIM and CAD. Of course, similar kinds of problems in object-oriented computer systems have already been considered in Example 2, in particular, for a square and a copy of it that form an undefined rectangle that's 2 x 1. For this kind of problem in all shapes, there's a positive solution with the use of reduction rules in the embed-fuse cycle; there's nothing to worry about in shape grammars. And in fact, that's the whole point of visual calculating, to be able to act on what you see – defined and not.

[15] O. Wilde, "The Critic as Artist," *Intentions*, in *The Artist as Critic: The Critical Writings of Oscar Wilde*, ed. R. Ellmann (Chicago: University of Chicago Press, 1982), 349.

[16] G. Stiny, *Pictorial and Formal Aspects of Shape and Shape Grammars*, 226-228. See especially, the equilateral triangle in three squares and the square in four equilateral triangles in Figure 2-35. The mystery of shapes isn't that there's so much in them, but that we see so little of it – and are usually more than happy to settle for this once we can say something about it. Words (visual analogies) make a big difference. Three squares are three squares; they're named and defined – 12 lines, four sides each – and that's all there is to it, even if my Pythagorean example and the A's just to come show otherwise. The part/role of artists (painters and poets) and designers, and shape grammars is to see beyond what's given and known.

[17] For an amusing discussion of this and how easy it is, even for the experienced thinker, to miss what's going on in verbal explication, instead of visual examples, see G. Stiny, *Shape*, 90-98. It's rollicking good fun to graph the equation $A + E = AE$, that is to say, $E = A/(A - 1)$, once you've seen that $A + E$ is the same as AE. It's neat that E and A are undefined for $A = 1$ and $E = 1$. Then, 0 and 1 are as in themselves they really are not, each equals the other – try the math; it's not hard to check.

[18] J. D. Steingruber, *Architectural Alphabet 1773*, trans. E. M. Hatt (New York: George Braziller, Inc., 1975). It's amazing what captures the observant eye – but why shouldn't letters (symbols) be tree-trunks and clods of earth?

[19] I used descriptions like this in a shape grammar for the *char-bagh* or fourfold garden – G. Stiny and W. J. Mitchell, "The Grammar of Paradise: on the Generation of Mughul Gardens," *Environment and Planning B* 7 (1980): 209–226,

222. Today, I would recast the grammar in terms of the schema $x \rightarrow \Sigma\ F(prt(x))$, for greater simplicity and improved clarity. The use of $x \rightarrow \Sigma\ F(prt(x))$ makes the grammar vastly easier than I initially thought – all of the complication disappears.

[20] I enjoy reading, rereading, and misreading the literary critic Harold Bloom, who tracks the influence of poet on poet in his fantastic "revisionary ratios." He notes in the preface of his latest book, H. Bloom, *Possessed by Memory: The Inward Light of Criticism* (New York: Alfred Knopf, 2019), that he's lost in reverie and past argument, and that this is where he stands on the verge of his tenth decade. (Bloom never tires of recounting his extravagant years.) Of course for the artist and the critic, the swerve from logical argument (visual analogy) to irony (re-description and open-ended representation) and reverie (re-constructed memory and evocation) is *de rigueur* – so it's something of a surprise that Bloom feels bound to affirm this in his writing, or at least to recognize it, as he moves into his 90th year. For Wilde and certainly in shape grammars, the use of irony and reverie is perfectly normal, simply business as usual, to trace the locus of pictures and poems. In shape grammars, wherever this goes is part and parcel of shapes and rules in the embed-fuse cycle.

[21] Personal e-mail correspondence with J. Gips (July 18, 2015).

[22] For a little more on my longtime aversion to hats that invariably rub and pinch, and feel like they're on even when they're off, see "Seven Questions," 15-16.