Using design patterns to build and maintain the Rule of Law

Laurence Diver • me@laurencediver.net

Abstract

Law is, and has, an architecture. This article investigates that architecture by reference to the idea of ‘pattern languages’, as described in Alexander, Ishikawa, and Silverstein, *A Pattern Language* (OUP, 1977). Such ‘languages’ are combinations of design patterns, operating at multiple levels of abstraction and interlinking with one another to form a ‘fabric’ that constitutes the broader practice or enterprise. Applied to the legal domain, we can similarly identify patterns that operate at various levels, brought together into the idealised pattern language of legality and the Rule of Law. The article attempts to deconstruct the Rule of Law into its components. It presents two visual schemas showing those patterns and their dependencies, an initial narration of a selection of them, and example of how to ‘read’ the language. The goal is to provide a mechanism to assess which Rule of Law elements are present (or not) in a given instance, and also to identify what was absent in cases where we feel the ideal has failed to live up to its promise. It is also intended to contribute to the development of 'legal tech', by offering a mechanism of translation between Rule of Law values and software development practice. Ultimately, the goal is to foster legal and digital architectures that respect, sustain, and even strengthen legality and the Rule of Law.

Keywords: rule of law, pattern language, design patterns, computational law, legal technology, code as law

1. Introduction

Law is, and has, an architecture. This is not just metaphor, though it can be a helpful idea when viewed in that way.[1] It consists of various parts that work together to produce the phenomenon of 'law', for better and for worse. It is a highly complex system, many elements of which are innocuous or even invisible, to the point that we tend not to think of them as essential parts of what makes law and legal practice what they are, or perhaps even notice them at all, at least until something goes wrong.[2]

The goal of this article is to begin identifying the discrete building blocks of law's architecture, by reference to the idea of design patterns and pattern languages, drawn from Alexander et al (1977). By viewing law as an architecture of interlocking and interdependent elements and patterns, we might be better able to diagnose circumstances where the central ideal of the Rule of Law is not being met.[3] According to the pattern language approach, each element and pattern of practice is identified and explicitly described in a uniform way, including the function it has, problem it solves, or capacity it affords. These elements can be combined into higher-level constellations of patterns that describe the necessary ingredients of a particular legal practice, operation, or fact, independent of its specific content. The hope is that this deconstruction/reconstruction approach might enable us to see more clearly what

[1] Fuller, for example, refers to his principles of legality, as "those laws respected by a carpenter who wants the house he builds to remain standing and serve the purpose of those who live in it" (Fuller, 1977, p. 96). Radbruch's antinomian characterisation of law is perhaps even more apposite, where the three elements of justice, legal certainty, and political purposes push against one another in a constant tension that holds law aloft, rather like the buttresses of a cathedral (Radbruch, 2006, p. 6ff).

[2] Work in law and media has explicated the idea of the role of material artefacts in law and legal practice. See for example van Dijk (2015) and Vismann (2008) and the discussion in the introduction to this collection (Koulu & Pohle, forthcoming).

[3] As will be set out below, this article takes an expansive view of the Rule of Law, encompassing or at least bringing alongside it the ideas of legality and legal protection. But the basic approach to translation does not require one to hold such a view.

has gone wrong in a given circumstance where the Rule of Law ideal has not been met, and thus what needs to be done to avoid such failures in future.

This first benefit allows for real-world problems to be considered according to the normative architecture of law, to identify what is missing or has been poorly executed in practice. A second benefit relates very specifically to digital technologies, particularly ‘legal tech’. Such technologies can be assessed, in advance, by reference to the normative architecture of law, to see how they effect or affect the elements and practices that underpin the ideal operation of the Rule of Law, for better or worse. It might be that their introduction is necessary to uphold or bolster the patterns that the Rule of Law relies on. Conversely, it might be that they are incompatible with certain patterns, meaning that their introduction ought to be avoided or prohibited in those specific contexts.

An effective translation between law’s architecture and the design of the artefacts and infrastructures that facilitate it can help in developing new interpretative methods between law and design (cf Koulu and Pohle (2021); COHUBICOL (2019)). We understand the law in light of the (technological) context, and technology in light of the legal context, with the goal of fostering legal and digital architectures that respect, sustain, and even strengthen legality and the Rule of Law. The ultimate vision is to articulate the core building blocks of law, their dependencies, and the conditions that make the whole enterprise possible, at a level of abstraction that is intelligible outside of legal philosophy circles — most obviously for the public, but also crucially for developers building digital systems that are destined either to be targets of regulation or to be involved in its administration and practice.

The article continues as follows. First, I set out one conception of the Rule of Law (accepting that this is a contested concept), and how its values are co-constituted by certain practices and (technological) conditions of possibility. I then sketch the theory of design patterns, highlighting the conceptual elements of it that are particularly salient to the cross-disciplinary approach I am proposing.

From there, I begin an initial sketch of a Rule of Law pattern language, including (1) a hierarchical schema of the elements and patterns of the language and their relationships to one another, and (2) an enhanced visual graph representation that shows their dynamic (co-)dependencies. I follow that with some discussion of the questions that arise from this approach, before setting out some of the potential benefits of approaching law in this way. Filling out the many details of this novel approach will require more work than is possible in a single article, but hopefully this contribution provides an initial blueprint.

1.1 The ideal of the Rule of Law

Viewing law as an architecture implies consciously identifying the elements that make it up. By doing this, we are empowered to think about the roles those elements play, and the nature of the relationships between them. This is not just a conceptual exercise of describing the law, or the conditions that are necessary for its existence (although it is that too). It is a normative exercise: it forces us to consider what elements are necessary in *any* architecture that can support and sustain the Rule of Law.

Once we have that groundwork in place, we can then identify in a given circumstance (1) whether or not those necessary elements are present, (2) whether they are operating as expected, and also (3) what might happen if their current method of operation, or 'mode of existence', evolves or changes entirely (cf. Latour (2013), chapter 13). An example of the latter would be a shift from social and institutional processes and artefacts to computational processes and artefacts.[4] Adopting this approach might enable us to be more specific about what aspects of the law are and are not working as well as they might, and to identify

[4] For a relevant discussion see P. McBride, '1. Introduction: foundations and scope' in McBride & Diver (2024).

which of them might benefit from such changes (and which should be preserved as they are).[5]

While law is and has an architecture, it also has a guiding ethos, at least in societies following the Western liberal mould. There are conflicting views on this point, of course, both about what precisely that ethos is and how well it is reflected in practice. But without digressing into a review of the critical legal studies literature, it is enough to say that one hope for this work is to begin to address some of the scepticism about the distance between Rule of Law ideals and how they are (not) implemented 'on the ground'.[6] As we will see, the aim is to make the ideal more workable in the real world, while maintaining a pragmatic view about what is actually possible. Before tackling any of that, I take as my starting point the normative position that the ethos of the Rule of Law is what animates, underpins, and sustains law's architecture, which at its core is about providing protection against the arbitrary exercise of power.

1.2 Law's architecture as an expression of legal intention

The ethos of the Rule of Law sets the backdrop of the whole enterprise of law and legal practice. It provides an initial perspective or way of looking at things that is (or ought to be) the starting point for the design and the operation of the elements that comprise law and its operation. The elements of law's architecture

[5] This normative view — of the conditions of possibility for a legal system worth having — differentiates the approach in this article from existing attempts to develop ontologies of legal concepts for use in software systems or legal informatics (see for example the Legal Knowledge Interchange Format (LKIF) (Hoekstra et al., 2007)). The latter already presume the existence of a legal system, and seek to describe what goes on within it, rather than identifying the foundational conditions that made it possible in the first place.

[6] As suggested above, the approach set out here is not limited to the conception of the Rule of Law that I set out in this article — it could in principle be applied as a means of translation to any conception of law. The point is to describe the normative architecture — whatever one considers that to be — in such a way that the presence or absence of its elements can be identified and responded to accordingly.

are not natural or physical facts that have just happened by chance to result the phenomenon of law; given the range of possible mechanisms of governance that are possible within the physical and social world we inhabit, the ideal(s) of the Rule of Law are the result of conscious choices that — albeit constrained by physical and technological reality — were nevertheless shaped by an ethos that was and is only one of many possibilities (cf. Latour (2013, Chapter 2) Fish (1989); Gadamer (2013, p. 283ff)). That ethos informs not just the elements of the legal architecture, such as legal texts and courtrooms, which in themselves are somewhat inert, but also the relationships between those elements. This is true both in terms of their conceptual relation to one another (e.g. a judge to a court, or a legislative rule to a particular circumstance), and in terms of how they fit into a broader legal practice over time.

1.3 Imperfections in the Rule of Law ideal

The Rule of Law as an ideal is clearly not realised in every case, and perhaps not in very many cases at all, if one considers all the instances where power is exercised arbitrarily that might otherwise have been restricted by legal means. This gives many people cause to reject the Rule of Law as idealised, utopian, or impossible, especially given the many power imbalances that pervade society, and the prevailing economic consensus that does not incentivise the provision of genuine legal protection to many.

It is possible, though, to recognise the imperfections in how the Rule of Law operates in practice, and at the same time to believe in its essential value both as a guiding concept and a force that is at least capable of constraining arbitrary power.[7] If we accept that starting point, we should develop approaches

[7] Cf. e.g. the Marxist historian E.P. Thompson's characterisation of the Rule of Law as a 'universal good', despite criticisms that it is a tool designed for and by the bourgeoisie (Thompson, 1975). See also Bańkowski (1993).

that can identify when, where and how it falters or fails, and that provide concrete, practical mechanisms for ameliorating or solving those deficiencies.

1.4 What about other conceptions of the Rule of Law?

This article consciously adopts a particular notion of the Rule of Law as its starting point, but there is no necessary link between the methodology of developing a pattern language and any particular conception of the Rule of Law. If the elements of the Rule of Law that are identified here are incomplete, incorrect, or the reader has a different understanding of what constitutes the Rule of Law, there is no obstacle to developing a different set of elements and the patterns that flow from it. So while the particular ethos of the Rule of Law adopted here is key to the goal of the article, the hope is that the language pattern methodology can in any case become a valuable mechanism for translating values from the world of legal theory into the world of design practice.

2. Patterns in architecture

Law is an architecture consisting of interconnected parts, unified under a guiding ethos that defines the nature of and relationship between those parts. This basic idea resonates closely with the notion of a *pattern language*, set out in Alexander et al.'s *A Pattern Language: Towns, Buildings, Construction* (1977, p. x). The idea of a pattern language is to provide a means of reaching "the quality without a name", a kind of animating spirit which elsewhere Alexander describes as

> …the root criterion of life and spirit in a man, a town, a building, or a wilderness. This quality is objective and precise, but it cannot be named… in order to define this quality in building and in towns, we must begin by understanding that every place is given its character by certain patterns of events that keep on happening there. (Alexander, 1979, pp. ix–x)

Alexander et al.'s goal is that those who design and build public and private spaces can embed this animating ethos in the environments they create.

A Pattern Language describes the concept of a pattern language and sets out the 253 patterns that make it up, derived from the authors' years of experience in building and planning work. Those patterns each follow the same form:

> Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice. (Alexander et al., 1977, p. x)

These individual sub-patterns are like templates for a discrete problem, providing a description of the issue and a principled mechanism for its solution. For example, the pattern NATURAL DOORS AND WINDOWS (number 221) specifies the problem like this (Alexander et al., 1977, pp. 1046–1049):[8]

> *Finding the right position for a window or a door is a subtle matter. But there are very few ways of building which take this into consideration.*
>
> In our current ways of building, the delicacy of placing a window or door has nearly vanished… Now look at the windows and doors in most buildings made during the last 20 years… It is almost always a rigid construction system, combined with a formal aesthetic, which holds these windows in such a death grip.

The pattern then includes some principles of application, before concluding with some references to further relevant patterns:

> Therefore:
> On no account use standards doors or windows. Make each window a different size, according to its place…
> Make the windows smaller and smaller, as you go higher in the building.

[8] In this article I follow Alexander et al.'s practice of writing pattern names in capital letters, followed (where relevant) by their associated reference in brackets.

> Fine tune the exact position of each edge, and mullion, and sill, according to your own comfort in the room, and the view that the window looks onto — LOW SILL (222), DEEP REVEALS (223).
>
> …

This example is heavily abridged due to limited space, but hopefully it gives some idea of the structure of a design pattern: first the problem is summarised, followed by a more in-depth discussion, which might include references to relevant literature. The basic requirements for the solution are then listed, followed by a conclusion that connects the pattern to others that are optional or necessary for its implementation.

2.1 Combining individual patterns into a whole

Individual patterns can be combined into larger aggregated patterns, potentially in multiple concentric stages. As the set of patterns builds up, they ultimately come together into the conceptually unified whole of the pattern language:

> Each pattern is connected to certain "larger" patterns which come above it in the language; and to certain "smaller" patterns which come below it in the language. The pattern helps to complete these larger patterns which are "above" it, and is itself completed by those smaller patterns which are "below" it. (Alexander et al., 1977, p. xii)

Subsets of the pattern language can also be combined to form smaller languages that represent elements of the larger whole, for example a porch on the front of a house (Alexander et al., 1977, pp. xxxv–xxxvii):

> …any small sequence of patterns from this language is itself a language for a smaller part of the environment; and this small list of patterns is then capable of generating a million parks, paths, houses, workshops, or gardens. For example, consider the following ten patterns […] This short list of patterns is itself a language: it is one of a thousand possible languages for a porch, at the front of a house. (Alexander et al., 1977, p. xxxv)

As we saw above, the description of a pattern includes a list of the other patterns which it relies upon or contributes to. For example, in the pattern for a porch just mentioned, the authors suggest as related patterns PRIVATE TERRACE ON THE STREET, SUNNY PLACE, OUTDOOR ROOM, SIX-FOOT BALCONY, PATHS AND GOALS, CEILING HEIGHT VARIETY, COLUMNS AT THE CORNERS, FRONT DOOR BENCH, RAISED FLOWERS, and DIFFERENT CHAIRS (Alexander et al., 1977, p. xxxv). Some of these then refer on to further patterns, not included in the porch example, presumably because they were not desired in the final result (RAISED FLOWERS, for example, is connected in the pattern language to SITTING WALL, TERRACED SLOPE, STAIR SEATS, BUILDING EDGE, and GARDEN WALL (Alexander et al., 1977, p. 1133)).

Ultimately this creates a web or graph of interrelated patterns, in an almost hypertext-like fashion. The patterns can be combined semi-arbitrarily to create novel artefacts that nevertheless reflect the overall tenor of the pattern language: one can "generate an almost unimaginably large number of possible different smaller languages, for all the different projects you may choose to do, simply by picking patterns from it" (Alexander et al., 1977, p. xxxviii).

A Pattern Language provides a conceptual and practical blueprint for planning and building cities, including all the component parts that go into them at each level of design and construction, from major infrastructural elements down to the aesthetic elements of an individual room in an individual building. These can even be used in the planning of specific practical building tasks, and so are not limited to philosophical or aesthetic discussions of the elements in question.

The pattern language therefore provides guidance at both abstract and granular levels of city planning, and at various points on the spectrum in between. This can be seen in the form of the language itself, where the earlier the pattern comes in the language, the more abstract or conceptual it is (e.g. INDEPENDENT REGIONS (1) and MOSAIC OF SUBCULTURES (8)), while the later the pattern comes in the language the more concrete or material it is (e.g. GARDEN SEAT (176) and WINDOWS WHICH OPEN WIDE (236)).

3. Patterns in law

Taking Alexander’s basic premise, we can begin to apply the same conceptual apparatus to the architecture of law. As argued above, law consists of different parts that build on each other in interlocking and concentric ways, coming together as a whole into what might be presented as a pattern language of the Rule of Law. As a normative value, the Rule of Law manifests in many different ways and in many different legally-recognised contexts, but it is always geared toward the same fundamental goal of providing protection against the arbitrary exercise of power. We saw above how pattern languages are intended as a means of capturing and reflecting a certain spirit in the contexts in which they are developed and used; in the same way, we might think of the central spirit that animates the practice of law, and which should in turn orientate the design and deployment of the processes, tools and artefacts that are used within that practice.

Taking this further, the idea is to break down the Rule of Law into the set of patterns and elements that the concept and its ‘spirit’ implicitly or explicitly rely upon. Like the city planning pattern language, these building blocks are to an extent hierarchical, with more fundamental or atomic elements/patterns providing the necessary underpinning for the ‘higher’ patterns that rely on them. For example, the fundamental ‘software’ of law is language (Gaakeer, 2019, p. 4), and its basic operation is the performative speech act, which brings new legally-recognised states of affairs into being (Diver et al., 2021; MacCormick, 2007; Austin, 1962). Speech acts rely on text as a source of externally specified requirements that define what counts as a valid speech act (e.g. the rules under which a couple can be married, a contract concluded, or a crime deemed to have been committed). These requirements in turn rely on the interpretability of those texts, since in any democratic constitutional state citizens cannot be governed solely by brute force or un-resistible or un-contestable ‘technological management’ (Brownsword, 2016). That means the capacity to interpret what legal norms mean in a given context is an essential quality of such states, and in turn the Rule of Law ideal. We can start to see how law has many internal dependencies whose connections to one another are what produces the complete architecture.

3.1 Necessary conditions for the Rule of Law

The interconnection of these elements is fundamental to the conceptual substance of the Rule of Law. This is true whether or not one thinks in terms of patterns or pattern languages. Without their presence — even if it is only a background presence that tends not to be explicitly acknowledged by those involved — what will be happening may *look* like law, but it will in fact be something else (e.g. technological management, cybernetic control, or simple brute force).[9] The practice appears legal on the surface, but in fact it is not compatible with legal values or processes at a deeper level. This latent incompatibility could potentially remain covered up for some time.

This perspective on the minimum necessary conditions for the Rule of Law resonates with what Alexander et al. say about interrelatedness within their pattern language, where the parts make up the whole, while the whole informs the parts:

> This is a fundamental view of the world. It says that when you build a thing you cannot merely build that thing in isolation, but must also repair the world around it, and within it, so that the larger world at that one place becomes more coherent, and more whole, and the thing which you make

[9] This is clear in the way that some people seem to think of law as consisting of no more than a list of (usually criminal law) rules that simply require to be followed. For a relevant discussion see ‘The ecology of code versus the ecology of law’ in McBride & Diver (2024, pp. 78–80).

> takes its place in the web of nature, as you make it. (Alexander et al., 1977, p. xiii)[10]

This is the idea of the 'hermeneutic circle', which we see reflected in MacCormick's description of law: "[t]he parts are fully comprehensible only as elements of the whole, and the wholeness of the whole therefore bears on the character of its elements" (MacCormick, 2007, p. 5; Gadamer, 2013, Chapter 4).

The articulation of the Rule of Law as a pattern language consisting of interrelated and interconnected patterns and sub-patterns is intended to emphasise that the Rule of Law need not be just an abstract concept or a vaguely desirable but unattainable goal. It is manifested through concrete, identifiable components and patterns of practice that can be made real to a greater or lesser degree. At least some of those components are essential to what law *is*. They are constitutive of its character and *must* be present, whether or not we pause to think in philosophical terms about what they do. Being clear about which of the patterns or components have this quality is an essential aspect of understanding and then actively sustaining the concept and character of the law. It also provides a benchmark against which to assess the appropriate contribution of computation to sustaining that character.

3.2 Generative and procedural patterns

Let's return to the discussion above of dependencies between patterns. If text and performative speech acts are fundamental building blocks of law, we can step one level up the ladder of abstraction to think about what higher patterns these building blocks in turn make possible.[11] In order to make sense of those

[10] The reference to a 'world' resonates with idea of law as a *Welt*, or a shared institutional 'world' with its own concepts and logics that separate (but don't isolate) it from the broader societal context within which it is embedded, and which it ought to serve (cf. Hildebrandt, 2015, p. 50ff).

[11] On the role of abstraction in understanding, see e.g. 'How we know what we know' in Hayakawa (1941).

patterns I propose to group them into two broad categories: *generative patterns* and *procedural patterns*.

On the one hand, generative patterns are intended for the creation of legally-recognised 'institutional facts', such as legal norms, contracts, corporations, and legal persons (MacCormick, 2007). The facts that arise from them are static, in the sense that they become part of the set of objects that are recognised by the legal system, and have different types of right, duty, and agency attached to or flowing from them. This group of patterns is generative because they have *legal effect*, which means they create, alter or extinguish legally recognised (or recognisable) facts.

On the other hand, the law is not just a bag of legally recognised facts. It is also — fundamentally — about procedures that manipulate those facts, and the legal persons who create, adapt, own, exchange, use, and extinguish them. Disputes about what those legal persons have done, and whether and how their rights have been recognised and respected, are an essential part of law: if there were no disputation of that kind, there would be no need for law. Procedures must therefore exist that can facilitate the fair articulation and resolution of such disputes to the satisfaction not just of the parties (at least one of whom will of course be left unsatisfied) but of the community as a whole. Such procedures are every bit as fundamental to law as are legal norms and institutional facts (Waldron, 2011). The set of procedural patterns are those which facilitate such conflict resolution.

Note that generative and procedural patterns are co-constitutive, which means they cannot be strictly delineated. Generative patterns provide the rules and facts that govern procedure, while procedural patterns provide the interpretative processes that explicate the meaning of those rules and facts, as well as providing authoritative closure to disputes about those meanings. They exist in a kind of dynamic symbiosis, as we shall see later.

3.2.1 A hierarchy of patterns

In Alexander's theory, patterns are connected hierarchically. Similarly, the Rule of Law relies on both procedural and generative patterns, which in turn rely on the atomic pattern of the speech act, which is reflexively reliant on the pattern of norm-creation where the requirements to perform the act can be found. This is a hierarchical but also concentric or multi-dimensional system; the higher patterns rely on the lower patterns while simultaneously the lower parts rely on the higher.[12] In this sense the pattern language is fractal, or recursive (Iba & Isaku, 2016, p. 3); larger patterns (e.g. LEGAL PERSONHOOD) rely on other patterns, such as INSTITUTIONALITY and NORM CREATION, as well as on STATEHOOD and (democratic) LEGITIMACY.

Some of these are themselves pattern languages that contain a multitude of patterns that exist within other societal domains.[13] The pattern language proposed here could therefore be extended beyond law and into other fields (e.g. political science, political economy, or philosophy of technology). This might indeed be one of its strengths: breaking down a view of law as 'fenced off' from the society it is supposed to serve and the other domains on which it relies (such as design and computer science), while at the same time maintaining an internal point of view, and commitment to, the specific values the Rule of Law represents.

In the Rule of Law pattern language, as in the city planning context, larger patterns come above smaller ones, but they rely on each other as elements of a

[12] One could also approach this from a historical perspective, akin to Alexander et al.'s discussion of the accretion of patterns over time (since a city does not just suddenly appear in one step; its patterns coalesce over time to facilitate the overarching policy envisaged in the pattern language). Similarly, the Rule of Law did not appear out of nowhere, or suddenly following a single moment, but evolved alongside the technological conditions and patterns of practice that made it possible (Ong, 2012; Eisenstein, 2012; Hildebrandt, 2015). This could be a valuable application of Alexander's ideas.

[13] As discussed in the introduction to this collection (Koulu & Pohle, forthcoming), there is a passing connection here with Luhmann's work in systems theory, however his use of the binary legal/illegal coding for operations in the legal 'world' would seem to flatten precisely that which the design pattern concept brings to life. See Luhmann (2004).

dynamic architecture. For example, Alexander et al. discuss the ACCESSIBLE GREEN pattern, which is connected to the larger patterns SUBCULTURE BOUNDARY, IDENTIFIABLE NEIGHBOURHOOD, WORK COMMUNITY, and QUIET BACKS, and the smaller patterns POSITIVE OUTDOOR SPACE, TREE PLACES, and GARDEN WALL (Alexander et al., 1977, p. xii). Each of these is a re-usable element that can (and does) appear in patterns other than ACCESSIBLE GREEN.

By analogy, a given practice or operation in law — for example creating a contract, annulling a marriage, or passing a sentence — will be facilitated by a pattern that entails a slightly different set of elements in the Rule of Law pattern language. Some will be present in most or all instances (e.g. TEXT), while others will depend on the particular operation that is being pursued, for example whether the matter is criminal or civil, and therefore what generative and procedural patterns are implicated.

To give an example from the schemas developed in the following sections, forming a contract involves LEGAL EFFECT, INSTITUTIONALITY, NORM CREATION, VALIDITY CONDITIONS, CONSENSUS IN IDEM, CAPACITY, COMPLIANCE, RIGHTS, DUTIES, and TEXT and/or DIGITAL TEXT. Procedurally speaking, contracts also entail the existence of an AUTHORITATIVE PROCEDURE, as well as EMPIRICAL EVIDENCE and EVIDENTIAL STANDARDS. If an alleged breach is contested, further components are implicated: CONFLICT RESOLUTION, DETECTION, CONTESTATION, EMPIRICAL EVIDENCE, and the CAPACITY TO INTERPRET the contents of the agreement.

These are the patterns and elements that are set out in the Rule of Law pattern language below. One can ‘read’ them from the perspective of the pattern language: when a conflict arises, its RESOLUTION (A) relies on DETECTION (B), which in turn relies on INTERFACES (G) that facilitate the communication and gathering of EMPIRICAL EVIDENCE (F). CONFLICT RESOLUTION (A) also relies on CONTESTATION (C), which in turn relies on the CAPACITY TO INTERPRET (H) the terms of the agreement. ENFORCEMENT (D) of the contract and/or the judgment of the court relies on

STATEHOOD (I), which in turn relies on DEMOCRATIC LEGITIMACY (L) and the state's MONOPOLY OF VIOLENCE (M). And so on.

The invocation of these patterns is necessary regardless of the substance of the particular contract. This is because although all contracts are necessarily different in some crucial respects, at a certain level of abstraction they are also always the same, because they all contain the building blocks of jurisdiction, personhood, speech acts, text, interpretability, etc.

As we saw above in Alexander et al.'s discussion of the porch, these subsets of the pattern language can thus form discrete languages for particular legal purposes. To paraphrase Alexander, "this small list of [legal] patterns is then capable of generating a million [legal persons, rights, duties, contracts, trusts, or statutes]."[14] Every time we encounter a legal dispute, or seek to implement a legal operation or create a legal institutional fact, we can check to see whether or not the requisite patterns were present and correct.

3.3 A Rule of Law pattern language: an initial schema

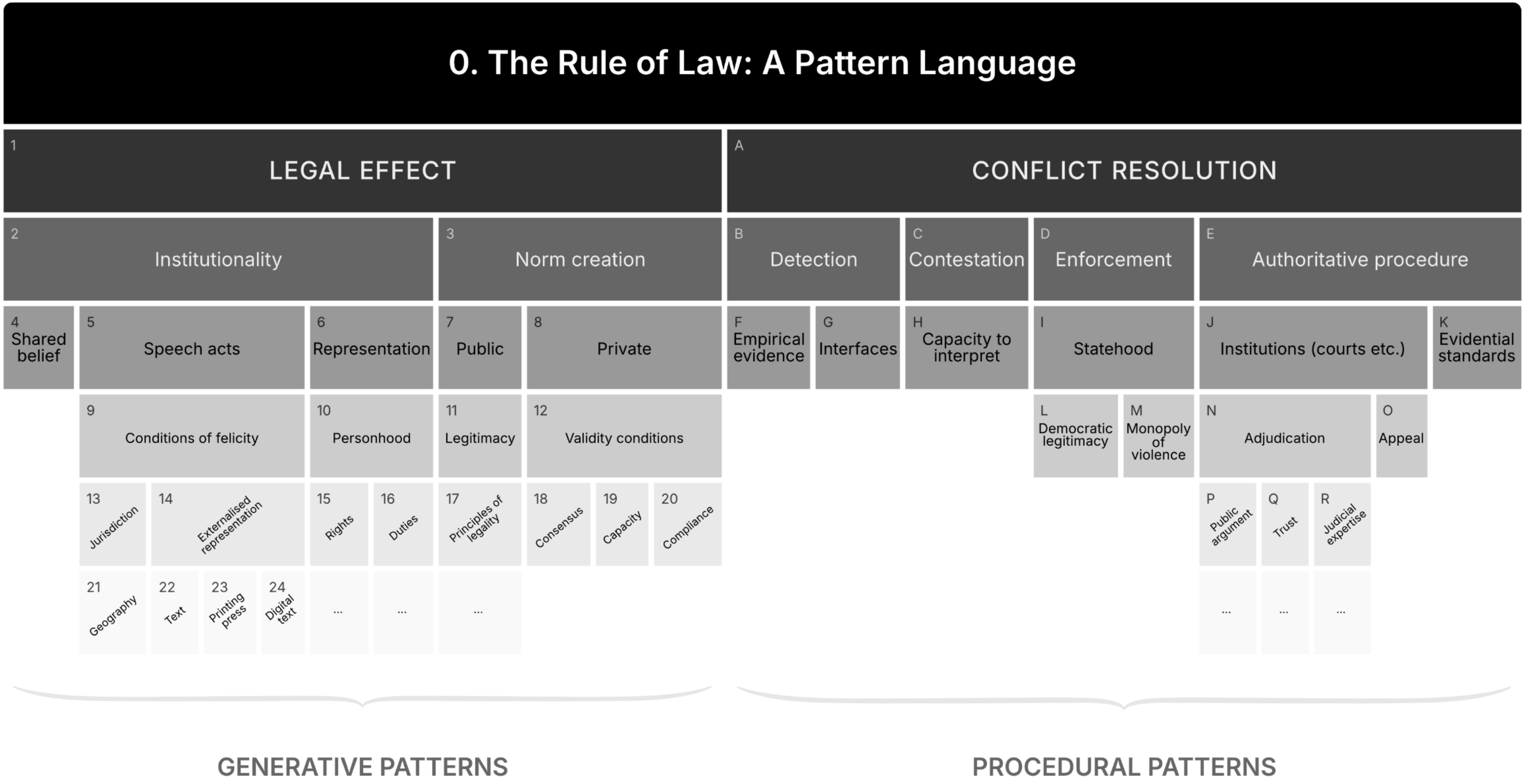
We now turn to an initial attempt to develop a schematic representation of a Rule of Law pattern language. The purpose of this representation is to demonstrate the elements and patterns of the Rule of Law and the relationships between them. Once this initial work is done, it is then possible to narrate the individual patterns themselves.

3.3.1 A hierarchical schema

The schema in Figure 1 shows the various elements and patterns of the Rule of Law from the 'larger' at the top to the 'smaller' at the bottom. As we saw in the previous section, these are composable such that the higher patterns rely on the

[14] Adapted from Alexander et al. (1977, p. xxxv), which refers to "a million parks, paths, houses, workshops, or gardens."

ones that come lower (or, put differently, the lower, more concrete patterns are what make the higher, more abstract patterns real and/or possible).



0. The Rule of Law: A Pattern Language

1 LEGAL EFFECT

A CONFLICT RESOLUTION

2 Institutionality

3 Norm creation

B Detection

C Contestation

D Enforcement

E Authoritative procedure

4 Shared belief

5 Speech acts

6 Representation

7 Public

8 Private

F Empirical evidence

G Interfaces

H Capacity to interpret

I Statehood

J Institutions (courts etc.)

K Evidential standards

9 Conditions of felicity

10 Personhood

11 Legitimacy

12 Validity conditions

L Democratic legitimacy

M Monopoly of violence

N Adjudication

O Appeal

13 Jurisdiction

14 Externalised representation

15 Rights

16 Duties

17 Principles of legality

18 Consensus

19 Capacity

20 Compliance

P Public argument

Q Trust

R Judicial expertise

21 Geography

22 Text

23 Printing press

24 Digital text

...

...

...

...

...

...

GENERATIVE PATTERNS

PROCEDURAL PATTERNS

Figure 1. A hierarchical schema of a Rule of Law pattern language

The idea here is to map out the conditions that are necessary for the Rule of Law to operate in its ideal form, identifying them at a level of abstraction that helps us articulate the normative essence of the Rule of Law, without going any further. It would not be helpful, for example, to continue to deconstruct the Rule of Law to articulate more (literally) atomic patterns of particles, or the chemical processes that go on in the courtroom or in contracting parties' brains. To do so would contribute nothing to the task of highlighting what makes this architecture specifically *legal*, and subsequently what effect different (technological) conditions might have on that architecture.

There are some limitations to the hierarchical schema presented in Figure 1, however. While it aims to identify the building blocks that are requirements of the Rule of Law, in this form it seems somewhat reductive of the dynamism inherent in the relations that exist between those building blocks. Some elements are optional, and do not sit easily alongside one another in a strict hierarchy.[15] A more dynamic representation might therefore be helpful.

3.3.2 A graph schema

The visualisation above follows a hierarchical tree structure. Although this is easily intelligible, it also obscures some of the relationships between the different elements that are more horizontal than vertical/hierarchical. For example, the results of the procedural patterns will often result in new legal norms, and so to that extent they are also generative. More generally, as noted earlier there can be no strict delineation between generative and procedural patterns, since they

[15] For example, DIGITAL TEXT (24) and the PRINTING PRESS (23) might plausibly be considered sub-patterns of TEXT (22) (my thanks to one reviewer for highlighting this). The affordances of the former two are however more far reaching than simply providing another medium of representing text: in many ways they provide the technological foundations of the law in a way that text *alone* does not (see e.g. Hildebrandt (2015)). This makes placing them in hierarchical relationship to one another even more difficult, perhaps emphasising the necessity of the more dynamic and reflexive representation that the next section explores.

are co-constitutive, connecting to one another in a permanently unfolding dynamic relationship. To try to represent this, Figure 2 takes inspiration from Iba and Isaku (2016) to present a distributed graph that shows both (1) the hierarchical relationships (i.e. dependencies between the lower-level patterns/elements and the higher-level elements of the Rule of Law) and (2) the conceptual or background dependencies that connect within and between the generative and procedural sets.

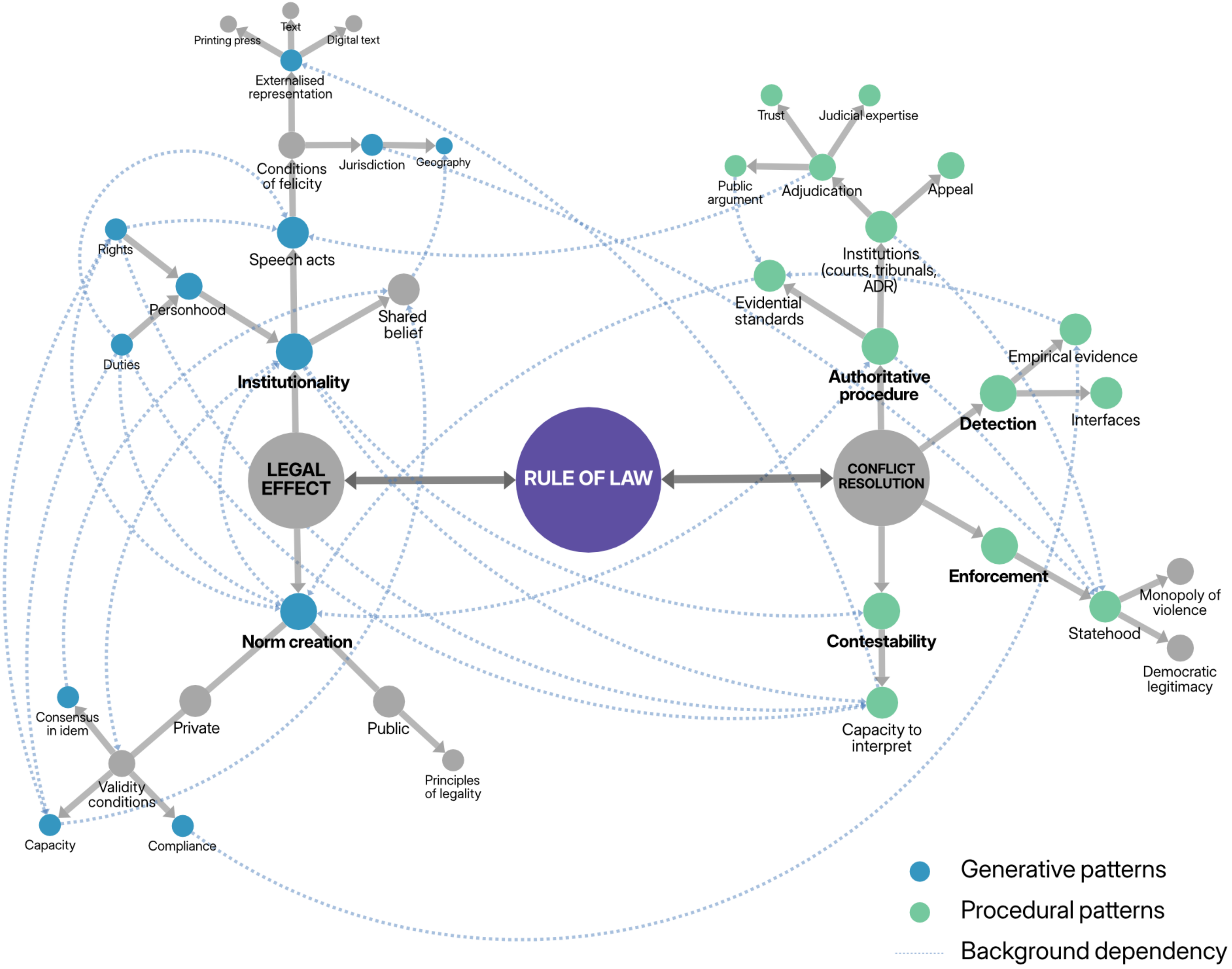


Figure 2. A graph schema of a Rule of Law pattern language

This schema gives a better sense of the dynamism inherent in the architecture of the Rule of Law. The complexity is evident, as are the dependencies between the parts (the dotted arrows can be roughly read to represent ‘depends on’). This representation should also make clear that the goal here is not to *simplify* the law in the course of deconstructing it, but rather to *highlight* its complexity, and how its ‘moving parts’ connect to and depend on one another in perhaps non-obvious ways.[16] This is relevant to the question of introducing computation, discussed later.

To demonstrate how such a representation enables us to ‘see’ the dynamic structure of the law, we can alter the graph to isolate a subset of the patterns involved in a particular legal operation, for example the creation of a contract:

[16] Again, the argument is not that this particular visualisation is a perfectly accurate representation — just as with the Rule of Law itself, views will differ on what are and are not components of it, and how they relate to one another. Instead, the aim is to explore what kinds of understanding are afforded by representing law in this way, both visually and in the narrated patterns below.

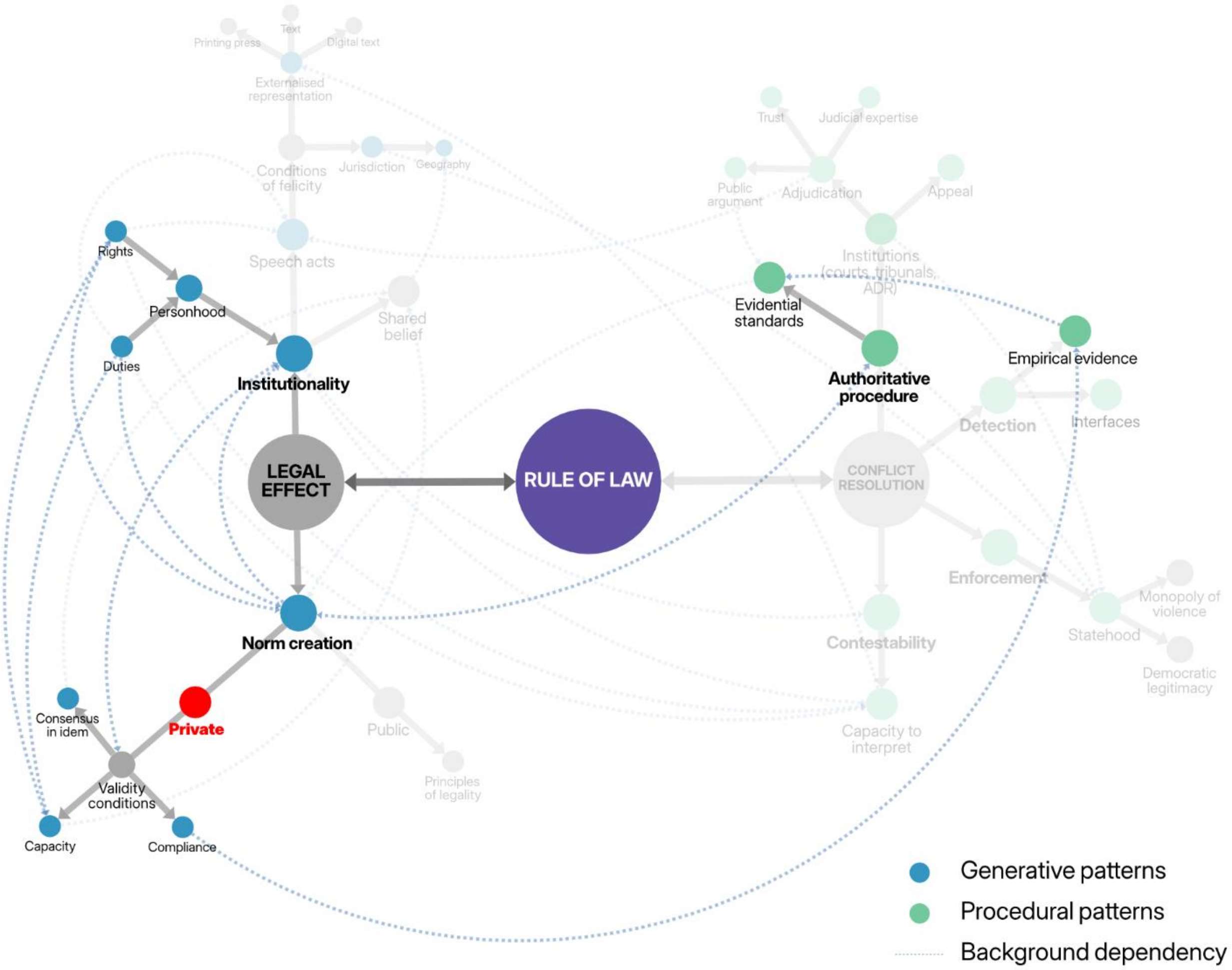


Figure 3. Isolating the patterns and elements relevant to contract formation

Figure 3 isolates the NORM CREATION (PRIVATE) pattern, and some of the patterns that it depends on. This highlights the higher and lower dependencies within the language, which we can narrate: on the generative side, NORM CREATION (PRIVATE) depends in the first instance on VALIDITY CONDITIONS, which rely on CONSENSUS between the parties, CAPACITY TO CONTRACT, and the possibility of COMPLIANCE. The RIGHTS and DUTIES that arise from the contract are also conceptually dependent on the idea of CAPACITY. Those in turn are dependent — in the 'background' — on legal PERSONHOOD, which itself is dependent on INSTITUTIONALITY (as are NORM CREATION and VALIDITY CONDITIONS in the first place). On the procedural side, we see that an AUTHORITATIVE PROCEDURE is reliant among other things on EVIDENTIAL STANDARDS, which is reflexively dependent on NORM CREATION, while COMPLIANCE is conceptually dependent on EMPIRICAL EVIDENCE and, in turn, EVIDENTIAL STANDARDS.

As mentioned above, at a certain level of abstraction all contracts are the same, and so this isolated subset of the whole language is also a representation of all contracts. This is valuable when it comes to building software intended to implement such legal concepts — the necessary dependencies, and therefore what features and affordances the system must have, are immediately clear.

3.4 Some example patterns

We can now consider how the actual patterns might be articulated. How these are finally specified will of course be key; the essential goal is to distil a crisp representation of the pattern and the problem it solves, in line with Alexander's approach outlined above.

This will set the groundwork for a translation of the patterns into technical artefacts and infrastructures, and an assessment of whether or not this has been effective. The more tractable the Rule of Law pattern language is to application in the design context, the more likely we are to encourage legal protection by design at the level of code. What follows are two indicative examples of patterns in the language, starting with the core or overarching pattern that sets the scene.

> **Pattern 0: THE RULE OF LAW**
>
> Power is unevenly distributed in society. This frequently results in (i) the direct oppression of individuals and collectives, (ii) the suppression of their humanity, dignity, and freedom, and (iii) a reduction in their capacity to order their lives according to their own values.
>
> …
>
> Therefore:
>
> (i) The exercise of power must be subject to constraint;
>
> (ii) Those mechanisms of constraint must be recognised by that same community, based on shared fundamental values and mutual expectations;

(iii) The extent of the community is delineated according to its recognised JURISDICTION (13), commonly based on a shared culture and GEOGRAPHY (21);

(iv) ENFORCEMENT (D) of those constraints must take place only following a pre-defined AUTHORITATIVE PROCEDURE (E) that has DEMOCRATIC LEGITIMACY (L) and the confidence of the community;

(v) Those within the community must be able to understand the constraints to which they are subject in advance of them being enforced;

…

The above is of course an outline intended to demonstrate the approach; any final explication of these conclusions would draw upon the wealth of literature on the elements and purposes of the Rule of Law.[17] However, it is crucial to bear in mind the level of abstraction that is most relevant, given the pattern's position within the pattern language. More detailed or complex aspects of the Rule of Law will be covered in the appropriate sub-patterns, and so there is no need, and indeed it would be harmful to the methodology, to attempt to reflect the entirely of the theory of the Rule of Law within this first pattern.

Pattern 1: LEGAL EFFECT

The exercise of power will always be more readily available to those who have the resources to exercise brute force — most obviously the state (via the police and armed forces, incarceration, weaponry), but also the community (mob and vigilante 'justice') and commercial enterprise (manipulative practices, private policing, excessive protection of private property, inequality of arms in contracting, 'externalities' whose costs are borne by the public, etc.)

…

In order to achieve THE RULE OF LAW (0), there must be a means of exercising power other than by brute force, to afford those without the latter the protection and empowerment that THE RULE OF LAW (0) promises.

[17] We already see in the above the echoes of numerous theorists' and institutions' explications of the basic requirements of legality, from across a wide range of theoretical perspectives (Council of Europe, 2016; Bingham, 2011; Raz, 1979; Fuller, 1977, p. 49ff).

…

Therefore:

(i) Relations of power must be articulable in a way that does not rely on the capacity of the parties involved to exercise brute force;

(ii) Parties must be afforded the CAPACITY TO INTERPRET (H) what their RIGHTS (15) and DUTIES (16) are in a given circumstance;

(iii) Parties must be afforded to chance to CONTEST (C) the interpretations by others of their RIGHTS (15) and DUTIES (16);

(iv) Any method of representing RIGHTS (15) and DUTIES (16) must at any given time afford the CAPACITY TO INTERPRET (H) after-the-fact;

…

Space constraints limit what can be developed here. Significant work will be needed to fill out and refine the full set of patterns, and it will need to take into account both the significant literatures around each of them and the Rule of Law lens that will shape how they are articulated for this purpose. Nevertheless, this brief foray into articulating textual Rule of Law patterns, combined with the narration of pattern dependencies in the previous section, will give an indication of how these abstracted representations can foster greater explication and hopefully understanding of the architecture of law.

4. What can we do with a pattern language for the Rule of Law?

Once we have developed a set of patterns that reflects both the relevant elements and the ethos of the Rule of Law, there are various ways the pattern language might ultimately be put to practical use. These can be roughly divided into ex post and ex ante categories. What follows is an initial outline, followed by a brief consideration of how this perspective might help to shape real-world software development toward the goal of achieving legal protection by design.

4.1 Ex post: diagnosing what went wrong

In a given circumstance where the Rule of Law does not operate in the ideal way, we can use the schema to identify which element (i.e. which pattern) was improperly represented or followed in practice. This narrows down the scope of the problem, allowing us to assess in a more fine-grained way what exactly went wrong. For example, in the contract example above, one can imagine concluding that because there was a failure to afford access to evidence of compliance, the validity of the 'contract' was undermined from the outset.[18] This in turn has a bearing on the status of the parties involved and their capacity to do something about it — a capacity that the Rule of Law is intended to afford.

This ex post 'diagnostic approach' might also allow for broader conclusions to be drawn about whether the problem is recurring, which might give insight into more fundamental issues in the legal system. Such legal 'antipatterns' are "just like a pattern, except that instead of a solution it gives something that looks superficially like a solution, but isn't one" (Koenig, 1998).[19] There are many hypothetical examples: in the contract example above, it might be presumed by the developer of the computer system implementing the contract that including certain types of user access and logging were sufficient to afford an understanding how it operated ex post. This might not accord with the requirements of legal process, however, and so the response might be to mandate the storing and accessibility of *specific kinds* of computer evidence that are more closely compatible with the procedural needs of the law.[20] Another

[18] In pattern language terms, the dependency between COMPLIANCE (20) and EMPIRICAL EVIDENCE (F) was broken, which undermined NORM CREATION (PRIVATE) (8).

[19] There is a close connection here between the idea of an antipattern and the point made above in the section on the necessary conditions for the Rule of Law: just because something looks like law, does not mean that it is compatible with legal values or processes, and it may take some time for that incompatibility — the antipattern — to become apparent.

[20] Such a mandate could have had some bearing on, for example, the Post Office Horizon scandal. There, the Horizon computer system's prohibition of access to evidence that would have supported their arguments was one factor that led to hundreds of sub-postmasters being

example might be to focus on accessibility and logical comprehension in order to address problems in the clarity of legislative drafting.[21] Appropriate responses to Rule of Law failures might also lie outside the legal system per se, for example the role of political will in prioritising the funding of legal aid as a prerequisite for access to justice.

At this point we might also be better equipped to consider the potential role of legal technology: to what extent could the failure in question be addressed by the introduction of computation? For example, if the issue is lack of access to legal materials, a problem that likely contravenes point (v) in THE RULE OF LAW (0) above, higher-quality digitalisation, archival and search infrastructure might ameliorate this, for example by facilitating access via different types of media and devices, lower (or no) financial barriers to access, better accessibility for those with visual or cognitive impairments, and even automated translation for those who do not speak the relevant language.

At any rate, if the solution to the antipattern lies within law, then it is for lawyers and those working in the legal system to adapt to solve it. If the solution lies outwith the legal system, then although lawyers might not be able directly to respond to it, the problem has at least been diagnosed.

4.2 Ex ante: Rule of Law by design

We can think of the ex post approaches sketched above as responses to existing problems. But the pattern language approach might also contribute to the avoidance of antipatterns developing in the first place. This could be especially

erroneously convicted of fraud — a manifest failure of the Rule of Law, given the scale of the harm wrought and the amount of time it has taken to begin to address it. See Marshall (2020).

[21] See recent work on ‘law smells’ (Coupette et al., 2023), drawing an explicit analogy with the software development concept of ‘code smells’ and the idea that the intelligibility, comprehension and maintainability of (both legal and software) code might be improved by the use of tried-and-tested patterns in order to avoid embedding technical debts that will be costly later on.

useful with respect to legal technologies, as a means of anticipating their impact and assessing the values they in fact promote (which may be at odds with those they are advertised as promoting, or those they *ought* to promote according to the ethos of the Rule of Law).[23] One could ask the provider of a legal tech system which pattern their product is intended to promote, improve, or strengthen. If they were unable to justify their system in these terms, this might be a signal that the system will not contribute to a strengthening of the Rule of Law and/or that it is designed in pursuit of a different value.

By the same token, a system might be interrogated as to its potential negative impact on Rule of Law patterns. We might ask, for example: to what extent does the blockchain platform uphold the CAPACITY TO INTERPRET agreements, and the DEMOCRATIC LEGITIMACY of state-backed ENFORCEMENT?[24] How does the prediction of judgment system maintain the practices of PUBLIC ARGUMENT and ADJUDICATION?[25] Does the automated benefits system provide INTERFACES that allow for DETECTION and the CONTESTABILITY of calculations that are incorrect, or that are insufficiently sensitive to the particularities of individual cases?[26] To what extent does the method of formalising legislative rules reflect the PRINCIPLES OF LEGALITY and the dependencies of LEGAL EFFECT that in turn facilitate CONFLICT RESOLUTION?[27] And so on.

[23] This general theme is reflected in 'Typology of Legal Technologies' (Diver et al., 2022), which aims to highlight the difference between on the one hand what legal tech providers claim their systems do, and on the other what they *actually* do, and how that potentially impacts on the fundamental character of legal normativity.

[24] On the relationship between technological design and the law's reliance on the capacity to interpret, see Diver (2021).

[25] See for example Medvedeva & McBride (2023).

[26] For an illuminating discussion of the UK's automated benefits system through the lens of the Rule of Law, see Mears and Howes (2023). See also the discussion of street level bureaucracy in the context of legal automation in McBride & Diver (2024, pp. 73–75).

[27] Such as CONTESTABILITY, the CAPACITY TO INTERPRET, and ADJUDICATION.

4.3 Translating Rule of Law patterns into software design patterns

Design patterns have a long history in design and software development (Gamma, 1995). They articulate best (or at least accepted) practice, reduce the time and thinking required to implement a recurring solution, and can facilitate the reflection of societal values in the design of real products.[28]

Although design patterns are a common feature in software development practices, they are also not entirely alien to the law. In the legislative drafting world, for example, patterns of rule construction are used for various scenarios that appear recurrently in the translation between policy and legislative texts. The Scottish Government's Parliamentary Counsel Office, for example, produces guidelines setting out 'common legislative solutions' for provisions that frequently appear in legislation, such as licensing schemes, duties of collaboration between bodies, and powers of entry (Parliamentary Counsel Office, 2018). Similar to the patterns in Alexander's work, these are articulated in terms of a description of the problem, the elements of the solution, and examples in real legislation where the pattern has been followed. As these common patterns of legislative language become embedded over time, practitioners and citizens can develop an understanding and reasonable expectation of how they will operate, bolstering the legal certainty dimension of the Rule of Law.

With time, one can envisage a body of practice developing that facilitates this kind of translation between the law and software engineering. The identification of clusters of patterns that represent legal concepts and operations might enable the development of software design patterns that are properly sensitive to the dependencies of the legal patterns. This can in turn highlight what might currently be blind spots for those who attempt to develop legal technologies: the requirement to afford EMPIRICAL EVIDENCE, or the CAPACITY TO INTERPRET, or INSTITUTIONALITY, for example, can quickly highlight what can

[28] On the development of software design patterns intended explicitly to reflect societal values, see for example Hussain et al. (2018) and Friedman & Borning (2002).

feasibly be facilitated by computer, and what cannot. Taking the blockchain example above, if the technology had been considered through this lens from the outset, the idea of a 'smart *contract*' would probably not have gained much traction — or would at least have been better framed as no more than a programming script — because the dependencies of NORM CREATION (PRIVATE) are so evidently absent.

Legal design specifications, and limits, are thus implicit in the pattern language, ready to be explicitly surfaced and articulated. This means that the proper role of computation can be delimited according to which patterns are appropriately susceptible to computational representation and processing, and which are not.

By moving up and down the ladder of abstraction that is implied by the Rule of Law pattern language, one can address different audiences in a way that is always coherent with the overarching goals of the language, and therefore with the Rule of Law. From a practical development perspective, the idea of implicit limits is also relevant: a software engineer might not need to understand the philosophical intricacies of the higher, more abstract patterns of the language (e.g. INSTITUTIONALITY, SPEECH ACTS, LEGITIMACY) but can nevertheless have confidence that their work is facilitating compliance with Rule of Law values through the proper implementation of the 'lower', more concrete patterns that those 'higher' patterns rely on (e.g. TEXT, EMPIRICAL EVIDENCE, INTERFACES). Practical software design patterns would need to be developed that properly reflect the legal patterns, of course, and the more concrete the latter are, the more effectively a practical design pattern might be derived from them.

5. Conclusion

This article proposes the application of Alexander et al.'s pattern language concept to the Rule of Law. It views the latter as an ethos that both frames the practice of law and implies a set of 'architectural' conditions that are necessary for its very existence. From that starting point, it deconstructs the Rule of Law to identify the elements and patterns of practice that create those conditions, which provides a conceptual basis for ensuring that those conditions are respected, sustained and strengthened.

From a legal perspective, this enables a more nuanced consideration of where Rule of Law ideals are not being met. Instead of rejecting the whole concept as unworkable or utopian, as some critical theorists might, we can instead see which specific pattern or element of the Rule of Law is failing, and direct efforts to resolve the problem. From a technological perspective, the approach allows for more fine-grained ex post assessment of digital systems that potentially impact on legal protection. It also provides a potentially useful extension to software requirements specification to ensure the affordances of computer systems are compatible from the outset with the normative expectations and requirements of the Rule of Law.

Momentum is building behind the application of concepts from software development to legal practice, including the use of reusable patterns. At the same time, however, the ideas here are intended to operate at a structural level, highlighting the fundamentals that make the Rule of Law possible, even to non-technical audiences. The point of creating a pattern language for the Rule of Law is not to oversimplify what the Rule of Law is, or to give the green light to inappropriately computationalised forms of law. The goal is in fact the opposite: in making the Rule of Law a more tractable concept for software developers and computer scientists, their understanding and practice can move closer to law, rather than the converse.

The architecture of the Rule of Law is not a given. A Rule of Law pattern language can be simultaneously abstract and practical, representing in the same place both the conceptual foundations of law and the practicalities of its implementation. The ultimate goal is to provide both lawyers and developers with a resource that highlights the commitments that need to be made, and sustained, in order to build and maintain the Rule of Law.

Acknowledgements

I’d like to thank fellow participants at the University of Helsinki Legal Tech Lab’s workshop on legal design patterns for some very stimulating discussion and for providing helpful comments on an earlier draft. I’d also like to thank colleagues in the COHUBICOL project for their helpful feedback. Any remaining errors are mine.

Conflict of interest

No potential competing interest was reported by the author(s).

Data availability

Not applicable.

References

Alexander, C. (1979). *The timeless way of building* (8th printing). Oxford University Press.

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

Austin, J. L. (1962). *How to do Things with Words*. Oxford University Press.

Bańkowski, Z. (1993). Don’t Think About It: Legalism and Legality. In M. M. Karlsson, Ó. Páll Jónsson, & E. M. Brynjarsdóttir (Eds.), *Rechtstheorie: Zeitschrift für Logik, Methodenlehre, Kybernetik und Soziologie des Rechts* (pp. 45–57). Duncker & Humblot.

Bingham, T. H. (2011). *The Rule of Law*. Penguin Books.

Brownsword, R. (2016). Technological management and the Rule of Law. *Law, Innovation and Technology*, *8*(1), 100–140. https://doi.org/10.1080/17579961.2016.1161891

COHUBICOL. (2019). *A new legal hermeneutics*. Counting as a Human Being in the Era of Computational Law. https://www.cohubicol.com/about/new-legal-hermeneutics/

Council of Europe. (2016). *Venice Commission Rule of Law Checklist* (CDL-AD(2016)007-e).

Coupette, C., Hartung, D., Beckedorf, J., Böther, M., & Katz, D. M. (2023). Law Smells. *Artificial Intelligence and Law*, *31*(2), 335–368. https://doi.org/10.1007/s10506-022-09315-w

Diver, L. (2021). Computational legalism and the affordance of delay in law. *Journal of Cross-Disciplinary Research in Computational Law*, *1*(1), Article 1.

Diver, L., Duarte, T., Gori, G., Hildebrandt, M., & Van den Hoven, E. (2021). *Research Study on Text-Driven Law*. COHUBICOL. https://publications.cohubicol.com/research-studies/text-driven-law/

Diver, L., McBride, P., Medvedeva, M., Banerjee, A., D’hondt, E., Duarte, T., Dushi, D., Gori, G., van den Hoven, E., Meessen, P., & Hildebrandt, M. (2022). *Typology of Legal Technologies*. Counting as a Human Being in the Era of Computational Law (COHUBICOL). https://publications.cohubicol.com/typology/

Eisenstein, E. L. (2012). *The Printing Revolution in Early Modern Europe* (2nd ed.). Cambridge University Press. https://doi.org/10.1017/CBO9781139197038

Fish, S. (1989). Working on the Chain Gang: Interpretation in Law and Literature. In *Doing What Comes Naturally: Change, Rhetoric, and the Practice of Theory in Literary and Legal Studies* (pp. 87–102). Clarendon.

Friedman, B., & Borning, A. (2002). Value sensitive design as a pattern: Examples from informed consent in web browsers and from urban simulation. *Proc. of DIAC*, *2*, 109–113. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=f46100d2f1739209e3be39f49fd452c31697dea0

Fuller, L. L. (1977). *The Morality of Law*. Yale University Press.

Gaakeer, J. (2019). *Judging from Experience: Law, Praxis, Humanities*. Edinburgh University Press. https://doi.org/10.3366/edinburgh/9781474442480.001.0001

Gadamer, H.-G. (2013). *Truth and method* (J. Weinsheimer & D. G. Marshall, Trans.). Bloomsbury.

Gamma, E. (Ed.). (1995). *Design patterns: Elements of reusable object-oriented software*. Addison-Wesley.

Hayakawa, S. I. (1941). *Language in Action*. Harcourt, Brace and Company.

Hildebrandt, M. (2015). *Smart Technologies and the End(s) of Law: Novel Entanglements of Law and Technology*. Edward Elgar Publishing.

Hoekstra, R., Breuker, J., Di Bello, M., & Boer, A. (2007). The LKIF Core Ontology of Basic Legal Concepts. *Proceedings of the Second Workshop on Legal Ontologies and Artificial Intelligence Techniques*, 43–63. http://ceur-ws.org/Vol-321/LOAIT07-Proceedings.pdf#page=43

Hussain, W., Mougouei, D., & Whittle, J. (2018). Integrating social values into software design patterns. *Proceedings of the International Workshop on Software Fairness*, 8–14. https://doi.org/10.1145/3194770.3194777

Iba, T., & Isaku, T. (2016). A Pattern Language for Creating Pattern Languages. *Proceedings of the 2016 Conference on Pattern Languages of Programs*.

Koenig, A. (1998). Patterns and antipatterns. In *The patterns handbooks: Techniques, strategies, and applications* (pp. 383–389). Cambridge University Press.

Koulu, R., Peters, A., & Pohle, J. (2021). *Finding Design Patterns in Law: An Exploratory Approach* (SSRN Scholarly Paper 3814234). https://doi.org/10.2139/ssrn.3814234

Koulu, R., & Pohle, J. (forthcoming). Legal Design Patterns: New Tools for Analysis and Translations between Law and Technology. *Digital Society*.

Latour, B. (2013). *An inquiry into modes of existence: An anthropology of the moderns*. Harvard University Press.

Luhmann, N. (2004). *Law As A Social System* (F. Kastner, R. Nobles, D. Schiff, & R. Zeigert, Eds.; K. A. Ziegert, Trans.). Oxford University Press.

MacCormick, N. (2007). *Institutions of law: An essay in legal theory*. Oxford University Press.

Marshall, P. (2020). The harm that judges do – misunderstanding computer evidence: Mr Castleton's story. *Digital Evidence and Electronic Signature Law Review*, *17*, 25–48.

McBride, P., & Diver, L. (2024). *Research Study on Computational Law* (p. 80). COHUBICOL. https://publications.cohubicol.com/research-studies/computational-law/

Mears, R., & Howes, S. (2023). *You Reap What You Code: Universal Credit, Digitalisation and the Rule of Law*. Child Poverty Action Group. https://cpag.org.uk/policy-and-campaigns/report/you-reap-what-you-code

Medvedeva, M., & McBride, P. (2023). Legal Judgment Prediction: If You Are Going to Do It, Do It Right. *Proceedings of the Natural Legal Language Processing Workshop 2023*, 73–84. https://doi.org/10.18653/v1/2023.nllp-1.9

Ong, W. J. (2012). *Orality and literacy: The technologizing of the word* (3rd ed.). Routledge.

Parliamentary Counsel Office. (2018). *Guidance on instructing Counsel: Common legislative solutions*. Scottish Government. http://www.gov.scot/publications/guidance-instructing-counsel-common-legislative-solutions/

Radbruch, G. (2006). Statutory lawlessness and supra-statutory law (1946). *Oxford Journal of Legal Studies*, *26*(1), 1–11.

Raz, J. (1979). The Rule of Law and its Virtue. In *The Authority of Law: Essays on Law and Morality* (pp. 210–229). Oxford University Press.

Thompson, E. P. (1975). iv. The Rule of Law. In *Whigs and Hunters: The Origins of the Black Act*.

van Dijk, N. (2015). The Life and Deaths of a Dispute: An Inquiry into Matters of Law. In K. McGee (Ed.), *Latour and the Passage of Law*. Edinburgh University Press.

Vismann, C. (2008). *Files: Law and media technology* (G. Winthrop-Young, Trans.). Stanford University Press.

Waldron, J. (2011). The rule of law and the importance of procedure. *Nomos*, *50*, 3–31.