FORMALISATION MEMORIES: TOWARDS A PATTERN APPROACH TO LEGAL DESIGN

Leon Qiu / Yiwei Lu / Burkhard Schafer

Leon Qiu, LLB Student, The University of Edinburgh, School of Law; Old College, Edinburgh EH8 9BJ tin.qiu.23@ucl.ac.uk

Yiwei Lu, PhD Student, The University of Edinburgh, School of Law; Old College, Edinburgh EH8 9BJ Y.Lu-104@sms.ed.ac.uk

Burkhard Schafer, Professor of Computational Legal Theory, SCRIPT Centre, University of Edinburgh, b.schafer@ed.ac.uk

Keywords:	**Legal logic, formalisation, design patterns, law as code, computational law, translation patterns**

Abstract: *The paper brings together ideas from translation studies, software design, architecture and legal theory to propose a new approach to the way in which computational models of the law can be deployed for AI safety. With the proliferation of AI-based autonomous systems, ensuring their law compliance has become a challenge for lawyers and developers alike. One approach to overcoming the "black box" problem are neurosymbolic systems, a combination of machine learning with "Good old fashioned' AI". While highly intuitive, this approach faces a number of problems. Formalisation memories, a combination of translation memories with pattern design, could be a way to address some of the resulting issues.*

1. Introduction

This paper brings together ideas from translation studies, software design, architecture and legal theory to propose a new approach to the way in which computational models of the law can be deployed for AI safety. With the proliferation of AI-based autonomous systems in more and more aspects of our lives, ensuring their compliance with relevant legal provisions has become a challenge for lawyers and developers alike.

One approach to overcoming the "black box" problem that challenges traditional attributions of legal liability are neurosymbolic systems, a combination of machine learning with "Good Old Fashioned' AI (GOFAI). The symbolic reasoning part then acts as "guardrails" for the machine learning component and its outputs. Typical applications are attempts to use formal representations of road traffic laws to ensure law-compliant behaviour of autonomous vehicles. While highly intuitive, this approach faces a number of problems, from the lack of agreed standards for formal representations of the law to the need to adapt the car to different sets of rules when jurisdictional borders are crossed. We will argue that "Formalisation memories", a combination of translation memories with pattern design, could be a way to address some of the resulting issues. In the first section, we will introduce autonomous vehicles (AVs) as a case study for neurosymbolic AI compliance. Drawing on work carried out as part of the AISEC project[1] and the Trustworthy Autonomous Systems node for AI governance,[2] we identify a number of obstacles to the idea of a logic-based, computational model of road traffic legislation that enforces adherence of AVs to road traffic law in real-time and while in driving mode.

In particular, we focus on problems that will inevitably occur when different developers choose different formal representations of the same legal provision, all of them "equally correct" – something that results from the open-textured nature of legal language. The second related issue occurs when an AV travels across jurisdictional boundaries and now has to adjust to often subtle differences between otherwise internationally harmonised laws.

---

[1] https://www.macs.hw.ac.uk/aisec/.
[2] https://web.inf.ed.ac.uk/tas/node-people.

What is required to solve these two coordination problems are something like standards – industry standards for the first scenario, and international legal harmonisation that standardises road traffic law even more for the second. But as we will see, neither approach will likely solve the problem.

We suggest that pattern libraries could be a partial answer to this quandary – using and adopting ideas that were developed independently in translation studies ("translation memories") and software design (e.g. "privacy patterns"). Further value could be added when legislators mirror the same "patterned" approach for technology regulation. This connects our discussion to the emerging "law as code" debate – if machines rather than humans become a new audience for laws, what do we need to change in the legislative process to protect democratic accountability and the rule of law?

2. Neurosymbolic approaches to lawful AI: AVs passing driving tests

In 2015, the United Kingdom (UK) Department of Transport published its report, *The Pathway to Driverless Cars*. It contained among a set of recommendations also the following sentence: "*real-world testing of automated technologies is possible in the UK today, providing [...]that the vehicle can be used compatibly with road traffic law*[3]".

In 2021, the Law Commission of England and Wales and the Scottish Law Commission followed up on this statement with a joint report on the regulation of autonomous vehicles beyond the testing stage. In this proposal, some of the duties to "drive compatibly with road traffic law" would in the future be allocated to the car developers (and through them, figuratively, "the car"), others are assigned to a new category, the "non-driver in charge" (a human passenger with special duties) and others still to the "driver" (typically the passenger-in-charge, once they take full control of the vehicle and disable the self-driving mode).

Taken together, these requirements create new challenges for AV designers: how can they demonstrate that their vehicle fulfils the condition set by the DoT and is concretised by the Commission and that their cars adhere (at least) to those road traffic laws that the new regulatory framework assigns to them?

We also face this problem with human drivers, of course, and for some time, a "mixed methods" approach has been used. We label the two methods as "Cartesian" and "Baconian". The Cartesian approach tries to predict lawful behaviour by ensuring the driver-to-be has explicit knowledge of the rules – the theoretical part of the driving test. On its own, explicit knowledge would be insufficient though, and therefore testing new drivers also involves an experience-based, empirical or "Baconian" element: new drivers acquire experience under observation and instruction, first while taking driving lessons, then by demonstrating "in the wild" the acquired skills in a test, and finally by indicating through a "learner" sign that as far as their reliability is concerned, there is as yet insufficient data to make a reliable inductive inference, so that other traffic participants can adjust accordingly.

We find the same duality of "prediction on grounds of first principles" versus "testing in the wild" in the emerging regulatory framework for AI systems. The EU AI Act, e.g. envisages both a "Cartesian" element, where the developers demonstrate certain formal properties of their system for the purpose of certification (e.g. that proper debiasing methods were used), and a Baconian element that obligates them to first test the system where appropriate under realistic conditions, and then to report any malfunction encountered while the system is in use. Similarly, in the US, an executive order by the President creates new obligations for safety and security testing for AIs, coupled with reporting duties when these, or subsequent deployment, experience critical failures.[4]

---

[3] https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/401562/pathway-driverless-cars-summary.pdf Executive Summary, Findings, Point 9.

[4] https://www.whitehouse.gov/briefing-room/presidential-actions/2023/10/30/executive-order-on-the-safe-secure-and-trustworthy-development-and-use-of-artificial-intelligence/.

This paper is concerned mainly with AI regulation's "Cartesian" element and its implications for developers: to what extent can they formally verify that their system will be law-compliant? The UK legal position had required these assurances as a condition for road testing – and that means that trivially, realistic testing had not yet been done before they were first allowed on the roads.

The AISEC project is one of several projects currently underway globally to address the issues of AV regulation, and this paper also benefits from some of the feedback from the industry that we received during our research. The overarching question of AISEC is to what extent the compliance of an AV with applicable law can be it is possible to verify rigorously. Three very different approaches to this question emerged during this research:

1. Formal verification of the neural machine learning algorithms.
2. Symbolic and computational representation of relevant road traffic laws as "guardrails" car-side, controlling and interpreting the neural systems in real-time during driving.
3. Symbolic and computational representation of relevant road traffic laws as part of a smart design environment, that helps developers in making legally sound and defensible design choices.

All three approaches have advantages and limitations. No 1, Formal verification of neural networks remains an active and complex research challenge in particular. While the AISEC team was able to develop some new software prototypes that can assist with the task, ultimately, it proved difficult to develop realistic models that replicated the complexity of driving. A proof-of-concept example, for instance, was able to formally verify that the neural network kept the car consistently between the street markings when exposed to side winds of variable strength and direction. Even such a simple example, however, had to make numerous assumptions and idealisations (about sensor accuracy, quality of the street markings etc) so that on their own, formal verification of the machine learning algorithms is unlikely to scale to the level that meets the requirement of the DoT, that is show general compliance with *all* applicable traffic laws, under all conditions.[5]

On the other end of the spectrum is No 3. Here, the designer or developer of an AV is assisted through a smart design environment that contains a formal representation of the applicable law. The aim is to prove formally not so much that the car will always behave in a law compliant way, but rather that the developers took all relevant laws into consideration when making their design decision. In this approach, all relevant knowledge is represented symbolically, and the inference rules are those of classical logic or one of its variants. All design decisions are made long before the car is allowed on the road. This also means that there is no need to be concerned about runtime issues, as the designer has all the time that is necessary to reach a decision, unlike a car that needs to decide whether to stop at a junction then and there. In terms of AI typology, it represents a "Good old fashioned" approach to AI, more specifically, a combination of formal ontologies with argumentation theory. We presented an outline of this approach at the Jurix and Jurisin conference, and for details, the reader is directed to the respective proceedings.[6]

Approach no 2 combines the neural approach that informs the first paradigm with the symbolic approach that is at the centre of solution 3. It is at the heart of numerous research projects currently underway, several of which presented their findings at the LN2FR workshop in 2022.[7]

[5] DAGGITT/KOKKE/ATKEY/ARNABOLDI/KOMENDANTSKYA, Vehicle: Interfacing Neural Network Verifiers with Interactive Theorem Provers. In: arXiv preprint arXiv:2202.05207 (2022).

[6] LU Y./LIN Y./SCHAFER, B./IRELAND, A./URQUHART L./YU LU, Z. Handling Inconsistent and Uncertain Legal Reasoning for AI Vehicles Design. In: Proceedings of the International Workshop on Methodologies for Translating Legal Norms into Formal Representations (LN2FR 2022), 76 (2022); LU Y./YU, Z./LIN. Y./SCHAFER, B./IRELAND, A./ URQUHART L. An Argumentation and Ontology Based Legal Support System for AI Vehicle Design. In Francesconi, E./Borges G./ Sorge, C. (eds), Legal Knowledge and Information Systems. Frontiers in Artificial Intelligence and Applications, Vol. 362, IOS Press, Amsterdam, 213-218 (2022).

[7] See BHUIYAN, H./GOVERNATORI, G./RAKOTONIRAINY, A./WENG, M.W./MAHAJAN, A. Traffic Rule Formalization for Autonomous Vehicle In: Borges, E./Satoh, K./ Schweighofer, E. (eds.), Proceedings of the International Workshop on Methodologies for Translating Legal Norms into Formal Representations (LN2FR 2022), 22-36 (2022); WESTHOFEN, L. / STIERAND, I./BECKER, J.S./MÖHLMANN, E./

What these projects share, despite differences in their chosen formalisation, is the combination of neural networks with symbolic reasoning systems, the latter acting as a "guardrail" for the former. The idea of combining neural and symbolic reasoning approaches in the legal domain is not an entirely new idea – one of the first systems of its kind was the SPLITUP system developed by Zeleznikow and Stranieri in the 1990s. It combined a symbolic representation of Australian family law with a neural network that analysed 20000 decisions by Australian courts on property distribution, the neural network element adding specificity and nuances to the legal rules.[8] A similar combination of court decisions and statutory rules has been proposed for the AV system by Borges et al. in the paper cited above.

By 2005, the benefits of combining neural network approaches with symbolic reasoning had become mainstream enough to coin a term for systems like this – "neurosymbolic AI" – and the field was deemed mature enough to categorise several sub-disciplines within this broader family.[9]

For relevance for our analysis is in particular the influential book by Daniel Kahneman, *Thinking Fast and Slow*, which provides a broader cognitive-psychological grounding for this approach. In it, he argues that cognition involves two distinct components: System 1 deals with thinking that is fast, reflexive, intuitive, and unconscious. By contrast, system 2 is slow, conscious, and explicit. System 1 is involved in pattern recognition, while system 2 is involved with planning and logical deduction.[10] We noted above the Cartesian and Baconian approach to AV safety, and broadly speaking, they map unto this distinction: passing the practical driving test requires fast and intuitive thinking, while the theoretical test requires the ability to reason with rules and plan behaviour. Combining both approaches into one autonomous system, so the hope, will allow to combine their separate strength and mitigate their individual weaknesses, in the same way we as humans constantly move between these two modes of cognition. Recent advances in robotics saw a renewed enthusiasm for this approach.[11] Both 2 and 3 require a formal representation of road traffic law however, and in the next section, we will discuss some of the many problems that this inevitably entails.

3. Lost in machine translation

While the idea to hard-bake legal rules into an AV seems intuitively plausible – after all human drivers too have to learn the rules – they also face problems. Some of them are of a more technical nature: As we discussed above, the hope is that by combining "slow" and "fast" modes of thinking, the benefits of both are combined. The risk is that instead, their respective weaknesses reinforce each other. If, in particular, the symbolic reasoner sits "on top" of the neural network, interpreting its outcomes so as to turn them into law-compliant actions, even situations that require fast and reflexive decisions will get slowed down to a degree. This was also the feedback we received from our industry partners in AISEC and one of the reasons why we moved our efforts away from approach 3 to 2.

HAGEMANN, W. Towards a Congruent Interpretation of Traffic Rules for Automated Driving Experiences and Challenges. In: Borges, E./Satoh, K./ Schweighofer, E. (eds.), Proceedings of the International Workshop on Methodologies for Translating Legal Norms into Formal Representations (LN2FR 2022), 8-22 (2022); BORGES, G./ WÜST, C./SASDELLI, D./MARGVELASHVILI, S./ KLIER-RINGLE, S. Making the Implicit Explicit: The Potential of Case Law Analysis for the Formalization of Legal Norms, In: Borges, E./Satoh, K./ Schweighofer, E. (eds.), Proceedings of the International Workshop on Methodologies for Translating Legal Norms into Formal Representations (LN2FR 2022), 81-90 (2022). https://deepai.org/publication/proceedings-of-the-international-workshop-on-methodologies-for-translating-legal-norms-into-formal-representations-ln2fr-2022-in-association-with-35th-international-conferenc.

8   ZELEZNIKOW, J./STRANIERI, A. The split-up system: integrating neural networks and rule-based reasoning in the legal domain. In: Proceedings of the 5th international conf. on AI and law (ICAIL'95), 185-194, (1995).

9   BADER, S./ HITZLER, P. Dimensions of neural-symbolic integration-a structured survey, In: arXiv preprint cs/0511042 (2005).

10   KAHNEMAN, D. Thinking, fast and slow, Farrar, New York (2017).

11   See e.g. *Hitzler, P./Sarker, M. K.* (Eds.). Neuro-symbolic artificial intelligence: The state of the art. IOS, Amsterdam (2022); SHETH, A./ROY, K./GAUR, M. Neurosymbolic Artificial Intelligence (Why, What, and How). In: IEEE Intelligent Systems, Vol. 38, No. 3, 56-62 (2023); for AVs see also MANAS, K./PASCHKE, A. Legal Compliance Checking of Autonomous Driving with Formalized Traffic Rule Exceptions. Workshop on Logic Programming and Legal Reasoning in conjunction with 39th International Conference on Logic Programming (ICLP2023), July 9–15, 2023, https://ceur-ws.org/Vol-3437/paper4LPLR.pdf (2022).

However, other problems affect both these symbolic reasoning-based paradigms, though possibly with different severity. One problem is that knowing or adhering to the laws of road traffic is not sufficient for "lawful driving", despite the way the UK DoT formulated its challenge. Consider this rule from the UK Highway Code, Rule 124, emphasis by the issuing authority:

„You **MUST NOT** exceed the maximum speed limits for the road and for your vehicle. "

This seems clear enough, and in principle, easy to enforce through a combination of hardware and software solutions, such as hardwired speed limiters for the absolute maximum on UK roads, and software-based solutions to pick the right upper speed for the specific context the car finds itself in, such as a locally set lower speed limit near schools.

But now assume that a police officer finds the car parked in front of a school – with a bomb on the back seat whose countdown has started so that the only way to avoid a tragedy is to drive as fast as she can away from the build-up area (and assume further that due to the traffic at the time, this is indeed feasible and safe to do). In this case, the general criminal law rules on necessity and self-defence would trump the seemingly strict rule from road traffic law. Laws form a system, they rarely work in isolation, and their interdependence can create exceptions to lower-ranking rules that cannot be read directly from the text of the law, a reason for the widespread use of non-monotonic logic in legal reasoning systems that aim to replicate judicial reasoning.

The developer now has to make a choice – with little guidance – how wide to cast the net, and what other rules to incorporate. Read the brief literally and prevent any and all exceptions? Allow human override under certain conditions (if so, how to authorise and log this?) Increase the "intelligence" of the car so that it can reason about more of these exceptions autonomously?

Some choices are even more difficult to make. Take, for example, legal norm 152 from the UK Highway Code:[12]

"You should drive slowly and carefully on streets where there are likely to be pedestrians, cyclists and parked cars"

To give a formal account of this sentence, we would first rephrase it as a rule: If there are pedestrians, cyclists and parked cars, then the driver must drive slowly and carefully.

We can then formalise this sentence in First Order Predicate Logic as

$$\forall x \, D(x) \land \exists y \, P(y) \land \exists z C(z) \land \exists w PC(w) \rightarrow SDS(x)$$

with D: drives, P: is a pedestrian; C: is a cyclist; PC: is a parked car; DS: should drive slowly

"Read out" this formula now states roughly: For everyone, it holds that if they are driving, and there are pedestrians, cyclists, and parked vehicles, then they must drive slowly.

There are a number of problems with this formalisation that will not concern us further. The issue that we want to focus on is that even though the proposed formalisation is indeed a literal, word-by-word translation, it is also a translation that does not make sense. The problem is the use of the word "and" in the legal text. Read literally, it would permit a driver to speed close to a school, where children play between parked cars, as long as there are not also at the same time cyclists on the road. Very clearly, what the legislator intended here was an "or": any one of these factors, and any combination of them, should trigger the duty to greater care.

While this seems intuitively obvious, some car developers may hesitate to "correct" the legislator – where would their authority come from? Even more problematically, those who stick with the "and" formalisation could give a good, legal-doctrinal reason: violation of the highway code can result in a fine or loss of driver's license. If an accident happened, it could further lead to a more serious criminal charge for dangerous driving.

---

[12]   https://highwaycode.org.uk/rule-152/.

This means functionally, 152 acts like a rule in criminal law, and for these the canons of interpretation require, in case of ambiguity, to choose the interpretation that is more favourable to the accused.

The problem we now face is that different manufacturers, with good reasons, could opt for either one of the two formalisations. Human drivers of course also face that problem, they too can give mutually inconsistent interpretations to contested legal terms, but our ability to reason also about other driver's reasoning, "having a theory of mind" about them, mitigates the consequences. My interpretation of the requirement to drive with "reasonable care" may make me very cautious when the road is wet, but when I observe another driver continuing at speed, I understand this as their different interpretation of the rule, and I adjust my behaviour accordingly (and drive even more carefully while they are around, to compensate). AVs do not have this ability for meta-level reasoning, so different cars using different formal translations of the same legal norm is likely to lead to accidents.

We face a similar issue once we consider that cars can drive across jurisdictional boundaries. Humans are remarkably good at adjusting to different rules – a German driver who has internalised the rules on right-hand traffic, nonetheless, will be able, after a short period of adjustment, to drive with a German car on UK roads and adhere to all the relevant laws without having to re-learn driving from scratch. This is true even though the German car's physical design, its steering wheel placement, favours right-hand drive as a design affordance. This is unlikely to be possible for AVs of type 3. They are programmed to follow one set of legal rules, not two or more mutually inconsistent ones. This is an old legal AI problem: symbolic logic is sensitive to inconsistencies, and automating consistency maintenance when rules, or contexts, change is a difficult issue.

For both problems, standardisation would be an obvious answer. However, this is unlikely to happen. On an international level, there is a lack of political will to further harmonise, in the necessary detail, not just core road traffic laws but, as we also saw, the other rules that could affect lawful driving. On a national level, standardisation bodies have neither the expertise, nor the resources and processes to prescribe, on such a fine-grained level, how ever-changing laws should be formally represented. In the final part of the paper, we propose a more feasible interim solution that draws on experiences in translation studies and software design.

4. Formalisation memories

In the above discussion, we intentionally used the language of translation to talk about formalisation. Both formalisation and translation of law requires interpretation. "Literal" translations are often as unsuitable as "literal formalisations", and every translation and formalisation involves choices that are normatively laden.

One approach that leveraged technological developments for the practice of translation was developed from the 1990s onwards.[13] Translators realised that they frequently encountered the same translation problems and that similar segments often required the same translation, also to keep a text globally consistent. "Translation memories" (TM) are databases that store "segments" (whole sentences or paragraphs as well as sentence subunits above the level of individual words) that have previously been translated. The TM stores both source text and corresponding translation in language pairs of "translation units".

Once the program has divided the source text into segments, it can then look for matches between segments and the previously translated source-target tuples. The human translators can then accept the match, suggest a proposal for a better translation, or accept it in part and modify it in part. In the case of replacements or modifications, these new translations are then in turn added to the database.

While individual translators originally built their own libraries of TMs, it soon became clear that the greatest benefits were achieved if these databases were combined. A series of interoperability standards emerged, such as TMX, the Translation Memory eXchange that enables the interchange of TMs between translation

---

[13] Beginning with KAY, M. The proper place of men and machines in language translation. In: Machine translation, Vol. 12, 3-23 (1997).

suppliers. TermBase eXchange (TBX) is a similar standard that had been developed by the Swiss Localization Industry Standards Association (LISA) and was later revised as ISO 30042. Today, it combines ISO 12620, ISO 12200, and ISO 16642 and is listed here as an example of how these standards were adopted by bodies such as ISO.

For our problem, this means that rather than standardising a formalisation of our legal rule directly, a more feasible approach would be to collectively create "formalisation memory databases" and adapt standards from TM that allow the exchange between formalisation memory databases. These, in turn would be created and curated by individual developer teams. For our example rule, an entry could then be the tuple: <there are likely to be pedestrians, cyclists and parked cars, > A rival translation with "&" could also be stored, together with an associated reason for the counter-proposal.

5. From memories to patterns

While libraries of approved formalisations of components of legal rules ("bilingual dictionaries") are one way to address the problems we described, on their own, they are unwieldy. They would form substantial datasets that are difficult to search and even more difficult to apply to concrete scenarios. What is needed is a way to systematise and group them.

We contend that all legal systems face recurring regulatory problems, especially when it comes to road traffic. All legal systems have to have, e.g. rules for right of way, appropriate speed, or "control over space rules" that tell us how to drive when obstacles, pedestrians or other drivers come too near, such as rule 152 above. In other words, we can abstract even further from the translation memories and group them into patterns. Such a patterned approach that reuses solutions for similar problems has a long history in programming, for instance, in libraries of privacy patterns.[14] In our example, we would abstract, for instance, from the specific FM tuple < there are likely to be pedestrians, cyclists and parked cars, > to the general regulatory pattern "what to do if certain entities are too close to the car."

5.1. Design Patterns in Architecture

The concept of a "design pattern" has its roots in architectural theory. In his ground-breaking work, *A Pattern Language*, Christopher Alexander introduced the idea of a pattern as a representation of a recurrent issue and its quintessential solution.[15] Common examples of patterns include "Main Entrance", "Room Gardens", "Communal Eatings", and "Street Windows". He posited that these solutions are crafted to allow for their repeated applications, potentially a million times over. Alexander suggests that akin to the basic building blocks or 'atoms' of matter, these vast structures are constructed from a limited set of foundational patterns.[16]

Embedded within these patterns are rules that dictate their formation and their spatial relationship with other patterns. For example, a pattern of "stone houses in the South of Italy" is composed of smaller patterns such as a square main room, a two-step main entrance, small rooms off the main room, and arches between rooms.[17] We could also think of this stone house pattern as part of a larger town pattern that includes "street branching", "connected buildings", etc.

---

[14] See e.g. PAPOUTSAKIS, M./FYSARAKIS, K./SPANOUDAKIS, G./IOANNIDIS, S./KOLOUTSOU, K. Towards a collection of security and privacy patterns. In: Applied Sciences, Vol. 11, No. 4, 1396 (2021); CAIZA, J. C./MARTIN, Y.-S./GUAMAN, D.S./DEL ALAMO, J.M./YELMO, J.C. Reusable elements for the systematic design of privacy-friendly information systems: A mapping study. IEEE Access, Vol. 7, 66512-66535 (2019).

[15] ALEXANDER, C./ISHIKAWA, S./SILVERSTEIN M./JACOBSON, M./FIKSDAHL-KING, I./SHLOMO, A. A Pattern Language: Towns, Buildings, Construction. (Oxford University Press, London 2023 (1977).

[16] ALEXANDER C. The Timeless Way of Building, Oxford University Press, London 99-100 (1979).

[17] ALEXANDER C. The Timeless Way of Building, 188.

5.2.   Contract Patterns

Alexander's idea also resonated with lawyers. Gerding's Contract as Pattern Language discussed how Alexander's pattern theory influenced the design of contracts. He referred to a contract pattern as an "encapsulated solution within a legal agreement (or set of agreements) to a specific legal problem."[18] Gerding was not the first to come up with such an idea. Indeed, standardised drafting had already been part of practitioners' work routines. As Triantis put it, standardisation "has a long tradition in transactional legal practice"[19]. Specifically, practitioners are used to adopting well-structured precedents as starting points.[20] Examples of this include the Encyclopaedia of Forms and Precedents,[21] Greens practice styles,[22] the Law Society of Scotland's Scottish Standard Clauses[23] and the forms and the more recent procedures produced by the Property Standardisation Group.[24] Apart from precedents created by others, practitioners also sometimes reuse their own previous work.[25]

Academics who observed this phenomenon hinted at concepts that are similar to what Gerding would call "contract patterns". Smith described the portable and highly standardised language of contractual provisions as "contractual boilerplates."[26] He observed that boilerplates are usually used in more than one contract and are, to some degree, self-contained. Going even further, Lannerö's CommonTerms Project proposed standardising online terms and conditions, emphasising their categories, ordering, formatting, and terminology.[27]

The practitioners' and academic solutions all aim to prevent "reinventing the wheel" in legal drafting, a goal Alexandar shared in architecture. As Gerding observed, "[m]any practitioner's manuals and model agreements serve a similar function of A Pattern Language."[28]

5.3.   Legislative Patterns

If designers contributed to a formalisation library, participation would be entirely voluntary. Different designers could still choose either of the two formalisations. But now, they would make this choice while being aware of the alternative approach, and under a duty, arguably, to a document why they deviate from it.

A more radical approach would be to mandate certain FMs by incorporating them directly in official legislation. Some legislative drafters have endeavoured to craft an equivalent to *A Pattern Language* within the realm of legislative drafting. The UK Parliamentary Counsel Office ("UKPCO") in London, for instance, has published a collection of "legislative solutions" to tackle "recurring policy issues."[29]

Although the scope and quantity of these patterns are still limited, practitioners and academics have indeed debated and discussed their adoption. Legislative patterns remain controversial, though. Lovric pointed out that a mistake in a design pattern could spread throughout the entire legal system, and an overreliance on these

---

[18]   GERDING, E.F. Contract as Pattern Language. In: Washington Law Review, Vol. 88, No. 4, 1323–56 (2013) 1326.

[19]   TRIANTIS, G.G. Improving Contract Quality: Modularity, Technology, and Innovation in Contract Design. In: Stanford Journal of Law, Business & Finance Vol. 18, No. 2, 186 (2012).

[20]   ROACH, M. Toward a new language of legal drafting. J. High Tech. L. 17.43 (2016).

[21]   *Encyclopaedia of Forms and Precedents*. Encyclopaedia of Forms and Precedents.. Sevenoaks, Butterworths (1985).

[22]   *Cusine, D. J.,* (ed) Greens Practice Styles, Edinburgh, W. Green/Sweet & Maxwell (1995).

[23]   *Law Society of Scotland.* Scottish Standard Clauses, last accessed 12.4.2023. https://www.lawscot.org.uk/members/rules-and-guidance/rules-and-guidance/section-f/division-c/advice-and-information/scottish-standard-clauses/ (2022).

[24]   The Property Standardisation Group (PSG), https://psglegal.co.uk/ last accessed 12.4.2023.

[25]   For example, see TRIANTIS, G.G. Improving Contract Quality: Modularity, Technology, and Innovation in Contract Design, 186.

[26]   SMITH, H.E. Modularity in Contracts: Boilerplate and Information Flow. In: University of Michigan Law Review, Vol. 104, No. 5, 1175 https://doi.org/10.1017/CBO9780511611179.016 (2006).

[27]   LANNERÖ, P. CommonTerms - for Meaningful Consent to Online Terms and Conditions!' https://commonterms.org/. last accessed 2.12.2022.

[28]   GERDING, E.F. Contract as Pattern Language, 1341.

[29]   *Office of the Parliamentary Counsel*, Common Legislative Solutions: A Guide to Tackling Recurring Policy Issues in Legislation (2022).

patterns might discourage drafters from delving into the complexities of individual problems.[30] This could lead them to believe that a one-size-fits-all solution suits every issue. In a similar vein, Sir Stephen Laws, the former First Parliamentary Counsel between 2006 and 2012, warned about the "precedent trap": drafters might initially seek solutions already employed for other problems, forcing them to adjust the current case to fit a solution intended for a distinct context.[31]

Conversely, Blackwell posited that the integration of design patterns into legislation could yield many benefits, including rendering laws more straightforward, transparent, and accessible and ensuring they remain consistent and adaptable during amendments.[32] This sentiment was echoed at the 2015 CALC (Commonwealth Association of Legislative Counsel) conference, where a staggering 95% of the audience (many being drafters themselves) agreed with Blackwell's perspective.[33]

6. Bridging the Divide: Patterns as the Intermediary Between Law and Code

The parallels between the texts of programs and laws inform a useful, if sometimes dangerous, analogy.[34] Lisachenko took this a step further, positing that a legal rule is essentially a specific form of computer code.[35] This concept has now also gained practical momentum in the form of the Rule as Code or Law as Code (RaC) movement, which is gaining international traction in countries like New Zealand, Australia, Canada, Singapore, the UK, France, and others.[36]

These developments reinforce Grimmelmann's dictum that "…there is a crucial similarity between lawyers and programmers: the way they use words. Computer science and law are both linguistic professions. Programmers and lawyers use language to create, manipulate, and interpret complex abstractions. A programmer who uses the right words in the right way makes a computer do something. A lawyer who uses the right words in the right way changes people's rights and obligations."

Consequently, patterns, as a shared language of both law and coding, could act as a crucial bridge between these two domains. For instance, part of the Computer-Readable Legislation Project at the Jersey Legislative Drafting Office was to parse their drafts for certain logical structures like If-Then.[37] This analysis could empower drafters to apply markup or devise alternative versions and may lead to the identification of specific patterns in legal texts, making them more likely to be machine-readable. This exploration paves the way for the concept of computer-readable legislation, essentially embodying the idea of "Rule as Code."

---

[30] LOVRIC, D. Legislative Counsel - Future Roles and Innovation. Loophole—Journal of the Commonwealth Association of Legislative Counsel, No. 2 (2020).

[31] LAWS, S. Giving effect to policy in legislation: how to avoid missing the point. In: Statute Law Review, Vol.32, No. 1, 1-16, https://doi.org/10.1093/slr/hmq017 (2011).

[32] BLACKWELL, T.F. Finally Adding Method to Madness: Applying Principles of Object-Oriented Analysis and Design to Legislative Drafting. In: New York University Journal of Legislation and Public Policy, Vol. 3, No. 2, 289 (1999).

[33] See LOVIC, D 7.

[34] GRIMMELMANN, J. Programming Languages and Law: A Research Agenda. In: Proceedings of the 2022 Symposium on Computer Science and Law, Washington DC USA, ACM, 155 https://doi.org/10.1145/3511265.3550447 (2022); skeptical DIVER, L. Digisprudence: Code as Law Rebooted. Edinburgh University Press. Edinburgh https://doi.org/10.1515/9781474485340 (2021).

[35] LISACHENKO, A.V. Law as a Programming Language. In: Review of Central and East European Law, Vol. 37, No. 1 115–24, https://doi.org/10.1163/092598812X13274154886584 (2012).

[36] MORRIS, J. Blawx: Rules as Code Demonstration. MIT Computational Law Report, August. https://law.mit.edu/pub/blawxrulesascodedemonstration/release/1 (2020).

[37] WADDINGTON, M. Jersey's project on parsing drafts for if-then structures for "Rules as Code", September 2023, No. 2 (2023).

7. Conclusion

Should the process of developing pattern languages be systematised and expanded,[38] it could lead to the creation of a legislative formalisation pattern dictionary. This resource would serve as a comprehensive index, linking and defining legislative patterns drafted in natural language alongside their corresponding expressions in programming languages. This would help to address the legitimacy deficit that we face if individual developers decide how to render relevant laws computational while at the same time ensuring the necessary convergence between different solutions.

8. Acknowledgement

---

[38] GRIMMELMANN, J. Programming Languages and Law: A Research Agenda, 160.